



Institute for Information  
and Communication Technologies,  
Electronics and Applied Mathematics

# Open Multi-Agent Systems: Representation, Limitations and Decentralized Optimization

Charles Monnoyer de Galland de Carnières

Thesis submitted in partial fulfillment  
of the requirements for the degree of  
*Docteur en sciences de l'ingénieur*

Dissertation committee:

Prof. Julien M. Hendrickx (UCLouvain, advisor)  
Prof. Raphaël M. Jungers (UCLouvain)  
Prof. Paolo Frasca (Université Grenoble Alpes, France)  
Prof. Ruggero Carli (University of Padova, Italy)  
Prof. César A. Uribe (Rice University ECE, USA)  
Prof. Pierre-Antoine Absil (UCLouvain, chair)

Version of September 29, 2022.



# Abstract

Multi-agent systems have been largely used over decades to model and study groups of interconnected independent entities acting together, most often collaboratively, towards some objective. Praised for their efficiency in solving a wide variety of large and decentralized problems, related analyses usually assume that the composition of the system is fixed. Yet, we could wonder if this assumption is reasonable nowadays, as the size of some systems reach huge magnitudes, and with the emergence of harsh systems naturally subject to ever-changing compositions.

We analyze *open multi-agent systems, i.e.*, subject to arrivals and departures of agents at timescales similar to that of the considered process. Because of their time-varying size and composition, the study of such systems is particularly challenging. Moreover, convergence results typically cannot be obtained anymore, and analyses dedicated to closed systems do not easily extend to open ones. Therefore, new tools and models tailored for open systems seem necessary to perform dedicated analyses.

In the first part of this thesis, we study general challenges related to open systems, in particular regarding their representation and the impossibility for them to achieve convergence. To that end, we analyze specific *consensus* problems in open settings, and consider in particular the Gossip algorithm. We show on the one hand how to represent and study this algorithm in settings subject to variable sizes, and study its performance on the other hand by modelling how information evolves in open systems.

In the second part, we study the convergence of *decentralized optimization* algorithms solving the resource allocation problem in open systems. Using two distinct types of metrics, we characterize the evolution of the error achieved by specific algorithms. In particular, we highlight how the openness impacts their performance and efficiency.



# Remerciements

À la rédaction de cette thèse aura été avant tout l'occasion de rencontrer une multitude de gens aussi intéressants que diversifiés. Chacun d'entre eux m'aura donné ses conseils, et comme l'a dit un grand philosophe à jamais âgé de 11 ans : *"Gotta catch'em all!"*.

Tout d'abord, pour m'avoir aidé et guidé tout au long de cette aventure et même avant ça lors de mon mémoire, il me semble indispensable de remercier mon promoteur : Julien Hendrickx. Sans son inestimable expertise, son intuition infaillible, et surtout sans ses précieux enseignements, cette thèse ne serait certainement pas ce qu'elle est aujourd'hui. Quand il est question de mathématiques il "regarde au-delà de ce qu'il voit", et il n'aura eu de cesse de me faire progresser.

Ensuite, je souhaite remercier mon jury de thèse pour leurs précieux conseils et leur relecture attentive et non dénuée d'intérêt de mon travail : Raphaël Jungers et Paolo Frasca, Ruggero Carli, César Uribe et Pierre-Antoine Absil. Les échanges et discussions que nous avons eus auront été autant de bons conseils pour rendre cette thèse encore meilleure.

Merci également à Renato, mon "mariage arrangé" grâce auquel une bonne moitié de cette thèse a vu le jour. C'est d'ailleurs avec plaisir que je continuerai de le côtoyer quelques années, lui et mes autres collègues, grâce à Luc que je remercie également pour l'opportunité qu'il m'offre.

Bien sûr, cette thèse ne serait rien sans l'environnement dans lequel elle a vu le jour, et que même un certain virus n'aura pas réussi à démanteler. Merci à tous les occupants, permanents ou éphémères, du bâtiment Euler (ou est-ce plutôt Euler ?) : François, Alexander et Anne, mes collègues de bureau qui auront toujours été là pour m'aider, m'aiguiller, et me supporter ; Georgios, Tonio, Seb, Balint, Mingming, Isabella, Yassine, Nizar et Anne, que certains reconnaîtront comme la team JUL ; Clémence

et Philémon, qui font évoluer cette team en équipe de chevaliers ; Laurent, Julien, Rémi, et Adrien, qui contribuent à former les Algebra Boyz ; Émilie, Benjamin, François, Loïc, Cécile, Guillaume O., Guillaume B., Adrien, Etienne, Estelle, Florence, pour qui le cruciverbisme est devenu une seconde nature ; Cécile, Roselien, Céline, Margo, Eléonore, Augustin, Rémi, Julien et Victor, qui auront formé l'équipe la plus performante que l'ASCII ait connue ; Et tant d'autres que je ne saurais lister exhaustivement mais qui ont contribué à faire de la vie en Euler une expérience aussi enrichissante qu'agréable.

J'ai une pensée toute particulière pour Cécile et Løic (dont on n'oublie pas le tréma), qui auront commencé et terminé leur parcours avec moi. Merci aussi à Etienne, Marie-Christine, Pascale, François W. et Astrid, qui rendent la vie en Euler à la fois tellement plus facile et inoubliable. Je retiendrai aussi le groupe de course qui, sous la ~~pression~~ les encouragements d'Anne et de François, aura réussi à me faire courir 20km (deux fois !).

Il me faut également remercier les bêta-testeurs de ma thèse, ces héros de l'ombre qui n'ont pas hésité à se plonger en avant-première dans les méandres de mon écriture pour la rendre meilleure, et ce à différentes échelles : Gwennaëlle bien sûr, qui en aura relu l'entièreté en un temps record, mais aussi Anne, Seb, Gilles, Richard, Virginie et tant d'autres dont les conseils m'auront été précieux.

Comme le soutien que j'ai reçu est loin de se limiter à l'Euler, je tiens à remercier mes amis, notamment Gilles, Richard, Virginie, Aude, Xavier et tant d'autres, mes parents et mes frères Gilles et Raphaël. Je remercie en particulier Gwennaëlle, qui ne pourra bientôt plus se vanter d'avoir plus de chapeaux que moi, et qui m'aura soutenu et rassuré du début à la fin de ce périple.

Finalement, quelques mentions aussi aléatoires que sans contexte : je remercie les tartes, le rosé tiède, le meuble de Philémon, le spikeball, les croquettes... Et merci à tous ceux qui ont supporté l'omniprésence pendant 4 ans de Taupiqueur en Euler, d'ailleurs en voilà un !



# Contents

<b>Remerciements</b> . . . . .	<b>iii</b>
<b>Contents</b> . . . . .	<b>v</b>
<b>List of symbols</b> . . . . .	<b>ix</b>
<b>List of acronyms</b> . . . . .	<b>xvii</b>
<b>1 Introduction</b> . . . . .	<b>1</b>
1.1 Multi-agent systems in general . . . . .	3
1.2 Opening our systems . . . . .	5
1.3 Joining and leaving agents . . . . .	8
1.4 Outline . . . . .	9
<b>2 On the modelling of OMAS</b> . . . . .	<b>13</b>
2.1 Elements of graph theory for modelling MAS . . . . .	13
2.1.1 <i>Basic definitions and concepts</i> . . . . .	14
2.1.2 <i>The Laplacian of a graph and its properties</i> . . . . .	16
2.2 Modelling arrivals and departures . . . . .	18
2.2.1 <i>Finite superset of agents</i> . . . . .	18
2.2.2 <i>Dynamic composition</i> . . . . .	20
2.3 Markov chains and counting processes . . . . .	21
2.3.1 <i>Elements of probability theory</i> . . . . .	22
2.3.2 <i>Continuous-time Markov chains</i> . . . . .	24
2.4 Event-based evolution of open systems . . . . .	29
2.5 Intrinsic and external objectives . . . . .	31

# I Consensus: representation and limitations

<b>3</b>	<b>The consensus problem . . . . .</b>	<b>39</b>
3.1	General definition . . . . .	39
3.2	Consensus in closed systems . . . . .	40
3.3	Consensus in open systems . . . . .	45
<b>4</b>	<b>Representing open systems: the case of gossiping . . . . .</b>	<b>49</b>
4.1	Problem formulation: a descriptor-based approach . . . . .	50
4.1.1	<i>System description . . . . .</i>	51
4.1.2	<i>Choice of the descriptors: scale-independent quantities . . . . .</i>	53
4.1.3	<i>Evolution of the descriptors in expectation . . . . .</i>	55
4.2	Generalization of the descriptor approach . . . . .	57
4.2.1	<i>Generalization of the system description . . . . .</i>	58
4.2.2	<i>Evolution of the general descriptors as an ODE . . . . .</i>	60
4.2.3	<i>Interpretation and illustrations . . . . .</i>	61
4.3	Case studies: gossiping in open systems . . . . .	64
4.3.1	<i>Fixed-size systems with random replacements . . . . .</i>	64
4.3.2	<i>Fixed-size systems with adversarial replacements . . . . .</i>	68
4.3.3	<i>Variable-size systems . . . . .</i>	71
4.4	Discussion and conclusions . . . . .	82
	<b>Appendix of Chapter 4 . . . . .</b>	<b>85</b>
4.A	Proofs of Lemmas 4.1 to 4.6 (Effect of events) . . . . .	85
4.B	Proof of Lemma 4.14 (Upper bound on the descriptors) . . . . .	89
4.C	Proofs of Propositions 4.8 and 4.9 (Generalization) . . . . .	92
4.D	Fixed points in fixed-size systems: interpretation . . . . .	94
<b>5</b>	<b>Performance limitations for average consensus in OMAS . . . . .</b>	<b>97</b>
5.1	Problem formulation: performance limitations . . . . .	98
5.1.1	<i>System description . . . . .</i>	100
5.1.2	<i>Problem reformulation . . . . .</i>	101
5.1.3	<i>Objective and methodology . . . . .</i>	102
5.2	Knowledge sets . . . . .	103
5.2.1	<i>Definitions . . . . .</i>	104
5.2.2	<i>Properties for stochastic event sequences . . . . .</i>	108
5.2.3	<i>Generalization to knowledge set functions . . . . .</i>	111
5.3	Derivation of a general lower bound . . . . .	112
5.3.1	<i>Pseudo-CDF and pseudo-PDF . . . . .</i>	112
5.3.2	<i>General lower bounds . . . . .</i>	114

5.3.3 *Proof of Theorem 5.9* . . . . . 116

5.4 Instantiations for the Gossip algorithm . . . . . 118

5.4.1 *Ping updates* . . . . . 119

5.4.2 *Pairwise interactions* . . . . . 124

5.4.3 *Performance analysis of the Gossip algorithm* . . . . . 130

5.5 Discussion and conclusions . . . . . 133

**Appendix of Chapter 5 . . . . . 137**

5.A Proofs of Propositions 5.11 to 5.13 (Main result) . . . . . 137

5.B Proof of Proposition 5.17 (Ping' model) . . . . . 142

5.C Proof of Proposition 5.20 (Gossip model) . . . . . 143

5.D Proof of Corollary 5.22 (XYZ approximation) . . . . . 145

**II Decentralized optimization: algorithms for resource allocation**

**6 Decentralized optimization . . . . . 155**

6.1 General definition . . . . . 155

6.2 Decentralized optimization in closed systems . . . . . 156

6.3 Decentralized optimization in open systems . . . . . 161

6.4 Online optimization . . . . . 162

**7 Random Coordinate Descent in OMAS . . . . . 167**

7.1 Problem formulation: open RA with fixed budget . . . . . 168

7.1.1 *Resource allocation problem* . . . . . 170

7.1.2 *System description* . . . . . 172

7.1.3 *Discretization of the time and reformulation* . . . . . 173

7.1.4 *Random Coordinate Descent (RCD) algorithm* . . . . . 175

7.2 Effect of replacements . . . . . 176

7.2.1 *Interpretation and illustration* . . . . . 178

7.2.2 *PEP analysis using the PESTO toolbox* . . . . . 180

7.3 Effect of RCD steps . . . . . 182

7.3.1 *The  $L_p^+$ -seminorm* . . . . . 182

7.3.2 *Contraction rate for the RCD algorithm in closed system* . . . . . 184

7.3.3 *Homogeneous agents with uniform probabilities* . . . . . 186

7.4 Convergence of the error in open system . . . . . 188

7.4.1 *Interpretation* . . . . . 192

7.4.2 *Illustration of the results* . . . . . 193

7.5 Discussion and conclusions . . . . . 195

**Appendix of Chapter 7 . . . . . 199**

- 7.A Proofs of Lemmas 7.5 and 7.6 (Minimizer's variation) . . . . . 199
- 7.B Proof of Proposition 7.7 (Effect of replacement) . . . . . 201
- 7.C PESTO computations for the effect of replacements . . . . . 204
- 7.D Proof of Proposition 7.13 (Effect of an RCD iteration) . . . . . 206
- 7.E Proof of Proposition 7.15 (Complete homogeneous case) . . . . . 208
- 7.F Proof of Corollary 7.19 (Minimal upper bound) . . . . . 209

**8 An Online Approach for Resource Allocation in OMAS . . . . . 211**

- 8.1 Problem formulation: open RA with varying budget . . . . . 212
  - 8.1.1 *General formulation* . . . . . 213
  - 8.1.2 *Simplifying assumptions and reformulation* . . . . . 214
  - 8.1.3 *Update rules and well-posedness* . . . . . 216
- 8.2 Performance metrics . . . . . 217
- 8.3 Upper bounds on the metrics . . . . . 221
  - 8.3.1 *Upper bound for the potential benefit* . . . . . 222
  - 8.3.2 *Upper bound for the dynamic regret* . . . . . 222
- 8.4 Illustrations and interpretation . . . . . 226
  - 8.4.1 *Setting for the simulations* . . . . . 226
  - 8.4.2 *Potential benefit* . . . . . 227
  - 8.4.3 *Dynamic regret* . . . . . 227
- 8.5 Discussion and conclusions . . . . . 229

**Appendix of Chapter 8 . . . . . 231**

- 8.A Proof of Proposition 8.7 (Well-posedness) . . . . . 231
- 8.B Proof of Proposition 8.10 (Function variation) . . . . . 232
- 8.C Dynamical regret: the case of quadratic functions . . . . . 234

### III Conclusion

**9 Conclusions and perspectives . . . . . 239**

- List of publications . . . . . 245**
- Bibliography . . . . . 247**
- Index . . . . . 269**

# List of symbols

This section summarizes the different notations introduced all along this thesis with a reference to the page where they first appear. They are sorted in alphabetical order, starting with the typographical ones, then the latin ones, and finally greek symbols.

In this thesis, we will use bold lower case letters to denote vectors, and  $v_i$  denotes the  $i$ -th element of the vector  $v$ . Similarly, we use upper case letters to denote matrices, and  $[M]_{i,j}$  denotes the element at the  $i$ -th row and  $j$ -th column of a matrix  $M$ .

$ \cdot $	_____	Absolute value / Cardinality of a set (p. 15)
$\emptyset$	_____	Empty set (p. 22)
$\langle \cdot, \cdot \rangle$	_____	Usual Euclidean scalar product (p. 172)
$\ \cdot\ _{\mathbf{L}_p^\dagger}$	_____	$\mathbf{L}_p^\dagger$ -seminorm from Definition 7.11 (p. 183)
$\ \cdot\ _1$	_____	L1-norm (p. 177)
$\ \cdot\ $	_____	Usual Euclidean norm (p. 17)
$\otimes$	_____	Kronecker product (p. 170)
$\perp$	_____	Orthogonality (p. 17)
$\subseteq$	_____	Set inclusion (p. 14)
$\cup$	_____	Set union (p. 59)
$\cap$	_____	Set intersection (p. 22)

★ | List of symbols

- \* \_\_\_\_\_ Interaction model (p. 112)
- $\mathbf{0}_n$  \_\_\_\_\_ Vector constituted of  $n$  zeroes (p. 16)
- $\mathbf{1}_n$  \_\_\_\_\_ Vector constituted of  $n$  ones (p. 16)
- $\mathbf{a}$  \_\_\_\_\_ Weight distribution of the RA problem (p. 170)
- $a(\epsilon)$  \_\_\_\_\_ Value of  $n(t)$  after an event of  $\epsilon$  (p. 58)
- $\bar{A}$  \_\_\_\_\_ Complementary event of  $A$  (p. 22)
- $A_\eta$  \_\_\_\_\_ Contraction rate for the RCD in open system (p. 188)
- $\text{Arr}_n$  \_\_\_\_\_ Arrival event among  $n$  agents (p. 52)
- $\mathbf{b}$  \_\_\_\_\_ Budget of the RA problem (p. 170)
- $\mathcal{B}(c, r)$  \_\_\_\_\_ Ball centered at  $c$  with radius  $r$  (p. 174)
- $\text{Ben}_T$  \_\_\_\_\_ Benefit (p. 219)
- $\mathcal{C}_t$  \_\_\_\_\_ Instantaneous error of the RCD algorithm (p. 223)
- $\text{Dep}_n$  \_\_\_\_\_ Departure event (random) among  $n$  agents (p. 52)
- $\text{Dep}_n^*$  \_\_\_\_\_ Departure event (adversarial) among  $n$  agents (p. 52)
- $d_i$  \_\_\_\_\_ Individual demand of agent  $i$  (p. 212)
- $\text{diag}(\cdot)$  \_\_\_\_\_ Diagonal matrix whose diagonal is the input vector (p. 17)
- $\mathcal{E}$  \_\_\_\_\_ Set of edges of a graph (p. 14)
- $E$  \_\_\_\_\_ State space of  $\mathcal{S}(t)$  in Chapter 4 (p. 60)
- $\mathbb{E}[\cdot]$  \_\_\_\_\_ Expectation/expected value (p. 23)
- $\mathbb{E}[\cdot|\cdot]$  \_\_\_\_\_ Conditional expectation (p. 24)
- $\mathbf{e}_i$  \_\_\_\_\_ Unit norm vector filled with zeroes except for the entry  $i$  (p. 18)
- $\text{Exp}(\lambda)$  \_\_\_\_\_ Exponential distribution with parameter  $\lambda$  (p. 27)
- $\mathcal{F}_{\alpha,\beta}^d$  - Set of  $\alpha$ -strongly convex and  $\beta$ -smooth functions  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  (p. 170)
- $\hat{\mathcal{F}}_{\alpha,\beta,c}^d$  - Set of functions satisfying the conditions of Assumption 7.4 (p. 174)

$\mathcal{F}_b$	Set of bounded measurable functions (p. 26)
$F_{j \rightarrow i}^{t,*}(s)$	Pseudo-CDF of $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$ (p. 113)
$f_{j \rightarrow i}^{t,*}(s)$	Pseudo-PDF of $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$ (p. 113)
$F_X(\cdot)$	Cumulative distribution function of $X$ (p. 23)
$f_X(\cdot)$	Probability density function of $X$ (p. 23)
$\mathcal{G}$	Graph (p. 14)
$\text{Gos}_n$	Gossip interaction event among $n$ agents (p. 52)
$h$	Step size for the RCD algorithm (p. 175)
$\mathcal{H}$	Deterministic event sequence (p. 104)
$\mathcal{H}^*$	Stochastic event sequence with model $*$ (p. 108)
$\mathcal{I}$	Information exchange event in general (p. 104)
$\mathcal{I}_{i \rightarrow j}$	Information exchange event from agent $i$ to $j$ (p. 100)
$I_n$	Identity matrix of size $n$ (p. 41)
$\mathcal{L}$	Infinitesimal generator operator (p. 26)
$L$	Laplacian matrix of a graph $\mathcal{G}$ (p. 16)
$L^\dagger$	Moore-penrose pseudoinverse of $L$ (p. 17)
$\mathbf{L}_p$	Scaled Laplacian corresponding to the RCD algorithm (p. 176)
$\mathcal{M}(t)$	Mean Squared Error (p. 102)
$\mathcal{M}^*(t)$	MSE of the optimal algorithm for the model $*$ (p. 117)
$\mathcal{M}_{j \rightarrow i}^*(t)$	Optimal MSE related to a single agent for the model $*$ (p. 117)
$\bar{M}_{n,\kappa}^2$	Upper bound for the effect of replacement (p. 177)
$\mathbb{N}$	Natural numbers (p. 14)
$n(t)$	System size (p. 51)
$\bar{n}$	Arrival-to-departure rate ratio (p. 75)

★ | List of symbols

$\mathcal{N}_i$	Set of neighbours of a node $i \in \mathcal{V}$ in a graph $\mathcal{G}$ (p. 15)
$N_j^i(t)$	Event that no information about $j$ is in $\omega_i^{\mathcal{H}^*}(t)$ (p. 113)
$\mathcal{O}(\cdot)$	Big-O notation (p. 179)
$o(\cdot)$	Little-O notation (p. 25)
$\mathbb{P}[\cdot]$	Probability (p. 22)
$\mathbb{P}[\cdot \cdot]$	Conditional probability (p. 22)
$p_{ij}$	Probability that agents $i$ and $j$ interact in the event $\mathcal{U}$ (p. 172)
$\text{Ping}_i$	Ping update event of agent $i$ (p. 119)
$\text{Po}(\lambda)$	Poisson clock with parameter $\lambda$ (p. 26)
$\text{Pot}_T$	Potential benefit (p. 219)
$\mathcal{P}(\cdot)$	Power set (p. 60)
$p_U$	Probability that an event is a pairwise update (p. 174)
$Q$	Transition rate matrix (p. 25)
$\mathbf{Q}^{ij}$	Update matrix corresponding to an RCD iteration (p. 175)
$\mathbb{R}$	Real numbers (p. 23)
$\mathbb{R}_+$	Nonnegative real numbers (p. 24)
$\mathbb{R}_{>0}$	Positive real numbers (p. 158)
$r_{ij}^{\text{eff}}$	Effective resistance between two nodes $i, j \in \mathcal{V}$ (p. 18)
$\text{Reg}_T$	Dynamic regret (in Chapter 8) (p. 219)
$\mathcal{R}$	Replacement event in general (p. 104)
$\mathcal{R}_i$	Replacement event of the agent $i$ (p. 100)
$\text{Rep}_n$	Replacement event (random) among $n$ agents (p. 52)
$\text{Rep}_n^*$	Replacement event (adversarial) among $n$ agents (p. 52)
$R_j[s, t]$	Event that agent $j$ gets replaced between times $s$ and $t$ (p. 108)
$s(\epsilon)$	Value of $n(t)$ before an event of $\epsilon$ (p. 58)

- $\mathcal{S}(t)$  \_\_\_\_\_ State of the Markov jump process (p. 58)  
 $\mathcal{S}_{a,b}$  \_\_\_\_\_ Feasible set of the RA problem with fixed budget (p. 170)  
 $\mathcal{S}_n$  \_\_\_\_\_ Feasible set for the RA problem with varying budget (p. 214)  
 $T_{j \rightarrow i}^{\mathcal{H}}(t)$  \_\_\_\_\_ Age of the most recent information (p. 107)  
 $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  \_\_\_\_\_ Age of the most recent information for the model \* (p. 108)  
 $\mathcal{U}$  \_\_\_\_\_ Pairwise update event in general (p. 172)  
 $\mathcal{U}_{i,j}$  \_\_\_\_\_ Pairwise update event between agents  $i$  and  $j$  (p. 172)  
 $\mathcal{V}$  \_\_\_\_\_ Set of agents in a MAS / of nodes in a graph (p. 14)  
 $\text{Var}(\mathbf{x}(t))$  \_\_\_\_\_ Empirical variance of  $\mathbf{x}$  (p. 54)  
 $\mathbf{X}(t)$  \_\_\_\_\_ Descriptor vector (p. 55)  
 $\mathbf{x}(t)$  \_\_\_\_\_ System state (p. 51)  
 $\mathbf{x}^{*,k}$  \_\_\_\_\_ Solution of the open RA problem at time step  $k$  (p. 174)  
 $\mathbf{x}^{*,t}$  \_\_\_\_\_ Optimal strategy (p. 218)  
 $\overline{\mathbf{x}^2}(t)$  \_\_\_\_\_ Empirical mean of squares of  $\mathbf{x}$  (p. 54)  
 $\bar{\mathbf{x}}(t)$  \_\_\_\_\_ Empirical mean of  $\mathbf{x}$  (p. 53)  
 $\bar{\mathbf{x}}^2(t)$  \_\_\_\_\_ Empirical squared mean of  $\mathbf{x}$  (p. 54)  
 $\mathbf{x}^{s,t}$  \_\_\_\_\_ Selfish strategy (p. 218)  
 $\mathbf{y}_i^*(t)$  \_\_\_\_\_ Optimal estimate (p. 116)  
 $\mathcal{Z}_{\alpha,\beta}^p$  \_\_\_\_\_ Set of functions satisfying the conditions of Assumption 8.1 (p. 213)
- $\alpha$  \_\_\_\_\_ Strong convexity parameter (p. 170)  
 $\beta$  \_\_\_\_\_ Smoothness parameter (p. 170)  
 $\gamma$  \_\_\_\_\_ Gossip-to-departure rate ratio (p. 76)  
 $\Gamma_\eta$  \_\_\_\_\_ Asymptotic error for the RCD in open system (p. 188)

★ | List of symbols

- $\Delta f_t$  \_\_\_\_\_ Instantaneous total cost variation (p. 223)
- $\Delta f_t^*$  \_\_\_\_\_ Instantaneous optimal cost variation (p. 223)
- $\delta_i$  \_\_\_\_\_ Degree of a node  $i \in \mathcal{V}$  in a graph  $\mathcal{G}$  (p. 15)
- $\theta_{n,\kappa}$  \_\_\_\_\_ Sub-result for  $\bar{M}_{n,\kappa}^2$  (p. 177)
- $\kappa$  \_\_\_\_\_ Condition number (p. 171)
- $\lambda^*$  \_\_\_\_\_ Optimality condition parameter for the RA with fixed budget (p. 172)
- $\lambda_2$  \_\_\_\_\_ Algebraic connectivity of a graph  $\mathcal{G}$  (p. 17)
- $\lambda_\epsilon$  \_\_\_\_\_ Rate at which events of  $\epsilon$  take place (p. 58)
- $\lambda_a$  \_\_\_\_\_ Arrival rate (p. 71)
- $\lambda_c$  \_\_\_\_\_ Individual pairwise communication rate (p. 124)
- $\lambda_d$  \_\_\_\_\_ Individual departure rate (p. 71)
- $\lambda_g$  \_\_\_\_\_ Individual gossip rate (p. 64)
- $\lambda_p$  \_\_\_\_\_ Individual Ping update rate (p. 119)
- $\lambda_r$  \_\_\_\_\_ Individual replacement rate (p. 64)
- $\pi_j(t)$  \_\_\_\_\_ Probability that the system size is  $j$  at time  $t$  (p. 60)
- $\rho$  \_\_\_\_\_ Interaction-to-replacement rate ratio (p. 130)
- $\rho_c$  \_\_\_\_\_ Pairwise communication-to-replacement rate ratio (p. 127)
- $\rho_g$  \_\_\_\_\_ Gossip-to-replacement rate ratio (p. 64)
- $\rho_p$  \_\_\_\_\_ Ping-to-replacement rate ratio (p. 121)
- $\rho_R$  \_\_\_\_\_ Replacement-to-update rate ratio (p. 190)
- $\sigma^2$  \_\_\_\_\_ Variance of the values held by the agents (p. 51)
- $\chi_B$  \_\_\_\_\_ Indicator function for the set  $B$  (p. 28)
- $\chi_{n,\kappa}$  \_\_\_\_\_ Sub-result for  $\bar{M}_{n,\kappa}^2$  (p. 177)
- $\psi_{n,\kappa}$  \_\_\_\_\_ Sub-result for  $\bar{M}_{n,\kappa}^2$  (p. 177)
- $\Omega^*$  \_\_\_\_\_ Knowledge set function for the model  $*$  (p. 111)

$\omega_i^{\text{Gossip}}(t)$  \_\_\_\_\_ Knowledge set of agent  $i$  with the Gossip model (p. 124)

$\omega_i^{\mathcal{H}}(t)$  \_\_\_\_\_ Knowledge set of agent  $i$  for the event sequence  $\mathcal{H}$  (p. 105)

$\omega_i^{\text{Ping}'}(t)$  \_\_\_\_\_ Knowledge set of agent  $i$  with the Ping' model (p. 123)

$\omega_i^{\text{Ping}}(t)$  \_\_\_\_\_ Knowledge set of agent  $i$  with the Ping model (p. 119)

$\omega_i^{\text{SI}}(t)$  \_\_\_\_\_ Knowledge set of agent  $i$  with the SI model (p. 128)



# List of acronyms

<b>CDF</b>	Cumulative Distribution Function
<b>CTMC</b>	Continuous-Time Markov Chain
<b>DC</b>	Disadvantageous Choice
<b>DGD</b>	Decentralized Gradient Descent
<b>i.i.d.</b>	independent and identically distributed
<b>MAS</b>	Multi-Agent System
<b>MSE</b>	Mean Squared Error
<b>ODE</b>	Ordinary Differential Equation
<b>OMAS</b>	Open Multi-Agent System
<b>PDF</b>	Probability Density Function
<b>PESTO</b>	Performance Estimation Toolbox
<b>RA</b>	Resource Allocation
<b>RC</b>	Random Choice
<b>RCD</b>	Random Coordinate Descent
<b>SI</b>	Susceptible-Infected
<b>SIS</b>	Susceptible-Infected-Susceptible



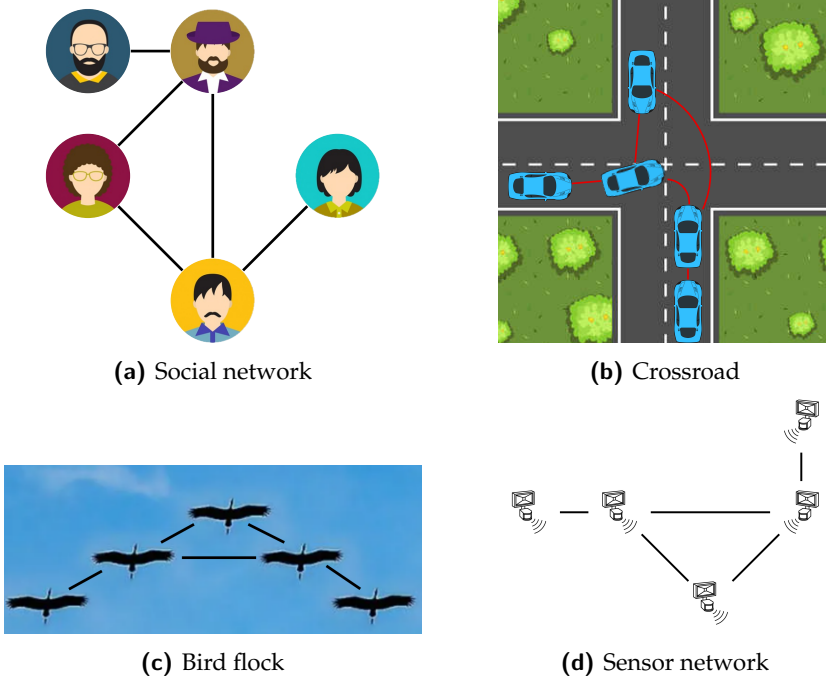
# 1

## Introduction

WHAT do social networks, bird flocks, groups of sensors and cars on the road have in common? Although these situations arise in different fields of expertise, they all share a major property: They consist of individual entities interacting with each other in a common environment to achieve some objective, sometimes collaborative. Typically, users in social networks exchange ideas over a shared topic, birds aim at collaboratively optimizing their flight through specific formations, sensors aggregate single measurements from different locations to improve their accuracy, and cars avoid collisions while evolving on the same road.

A rather straightforward and easy approach to handle and work in such situations would be to rely on some central master unit having access to information about the whole network and taking decisions for the entire community. However such strategies, called *centralized*, become extremely expensive both in terms of cost and complexity when applied to a large number of individuals with limited communication and sensing capabilities, such as sensor networks. More generally, in the situations above, it is rather natural to expect the individuals to be independent, and not to rely on such central entity. Consequently, a shift has been observed over the last decades from *centralized* to *decentralized* approaches, where each individual takes its own decisions based on local information.

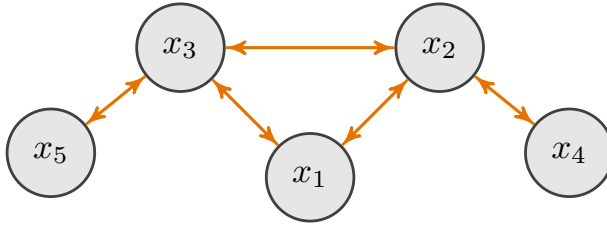
Subsequently, the *Multi-Agent Systems* (MAS) paradigm has been developed to describe, model and study problems of this type: A *system* is con-



**Fig. 1.1** Examples of situations that can be represented as multi-agent systems: (a) is a social network where the agents are users exchanging information with each other when they are in contact, (b) is a crossroad where the agents are cars avoiding collisions while following their respective trajectories, (c) is a bird flock where the agents are birds adopting a V formation by paying attention to their neighbours, and (d) is a group of sensors spread over some area and exchanging local measurements with their neighbours to estimate some measured quantity more accurately.

stituted of individuals called *agents*, interacting with each other through a network with some specific topology. Examples corresponding to the situations previously described are shown in Fig. 1.1, where each system is constituted of five interconnected agents. Not only can each of these examples thus be studied using the MAS paradigm, but in the specific case of Fig. 1.1 they can also be represented using the same network, depicted in Fig. 1.2. In fact, only the nature of the states of the agents and of the interactions changes from an example to another in that case.

In this thesis, we analyze a particular class of multi-agent systems called *Open Multi-Agent Systems (OMAS)*, where agents join and leave during the process, resulting in various additional challenges. By analyzing specific



**Fig. 1.2** Underlying network corresponding to the multi-agent systems presented in Fig. 1.1: Each node is an agent which holds a state  $x_i$ , and each edge is a bidirectional interaction channel. The nature of the states depends on the application (an opinion, a measurement, a position...), and the existence of an interaction channel depends on the physics of the application (proximity, acquaintance...).

standard MAS problems in open settings, we intend to highlight and study general properties of OMAS applicable to a wide range of problems in this framework, and to develop new techniques dedicated to their analysis in general. The remainder of this chapter aims at (i) introducing multi-agent systems in general as well as their use in various fields of applications, and (ii) presenting the impact of considering open multi-agent systems, and the new challenges arising from them.

## 1.1 Multi-agent systems in general

The *Multi-Agent Systems* paradigm is originated from *Distributed Artificial Intelligence*, a subfield of Artificial Intelligence which studies the collaboration between independent intelligent entities [Des02]. Starting from the mid-70's, it rapidly evolved and diversified to become an entire field of research [Wei99], following the promising perspective of its use in several application fields. One of the main examples of Distributed Artificial Intelligence are multi-agent systems (MAS) [Woo02]. A MAS is constituted on independent entities called *agents*, interacting through a network in order to solve, usually collaboratively, a global problem beyond the capabilities and knowledge of the individual entities [Gla06]. Hence, the study of MAS amounts to solve problems at the scale of the network by only acting on the agents individually. In that sense, it differs from game-theoretical frameworks where each player (*i.e.*, agent) makes decisions based solely on its own outcome, regardless of that of the other players [PW02].

Collective behaviors such as those studied with MAS were initially observed in the context of biology [KR02], where groups of animals must take collective decisions while moving together, *e.g.*, in bird flocks, fish schools or insect swarms for instance [LSN<sup>+</sup>11, Rey87, DGFG06]. In particular, animals in such settings have access to only a limited amount of information through interactions with close neighbours, so that decision making is done at the agent level [CKFL05, KRK09]. Those ideas actually transpose to statistical physics [VCBJ<sup>+</sup>95, LRC01, VZ12], in particular in [TT98] where flock models are used to describe the motion of self-propelled particles.

More generally, the ideas of collaborative agents, and hence the MAS paradigm, have been applied in a wide range of domains. Opinion dynamics is of them [BMHT09, HK02, FJ99, VMH18, YTL<sup>+</sup>18], as individuals influence each other's opinions through interactions. The focus of such studies is generally more on the evolution and behavior of the system itself rather than on the goal of the agents, which is sometimes not even clearly defined. It naturally extends to sociology and sociophysics [DeG74, YQG<sup>+</sup>18], typically with applications to social networks [PT17]. Vehicle coordination also relies on MAS to model vehicles in a shared environment [CHW11, CYRC12], *e.g.*, with applications to formation control [MR19, NZJY21], and quickly extends to robotics [FM04a, JLM03, Ota06, DBB14]. Consensus problems, where agents have to collaboratively agree on some decision to make, appear in various fields, including coordination [RBA05], decentralized optimization [NOR18], *rendez-vous* problems [AB06], or for decentralized estimation, for instance over wireless sensor networks [RN04, PKP06, OSFS08]. A survey covering several applications with their respective challenges can be found in [DKJ18].

Working with MAS comes with additional complexity as compared to centralized settings. In particular, because the agents are independent their respective evolution directly depends on their environment and is most of the time subject to randomness. Consequently, predictions and analyses at the system level become challenging, especially since they often require the use of probabilistic tools [Hew91].

In return, using MAS (and hence discarding the use of a central unit) brings a lot of new advantages. Firstly, the absence of a central unit implies that each agent works based solely on local information: This means for instance that they do not need to know the structure of the whole network, which can be crucial for large-scale networks (*e.g.*, with the emergence of the "Internet of Things" (IoT) [Sta18].) More generally, decentralization allows handling large-scale exchanges of information at lower costs through

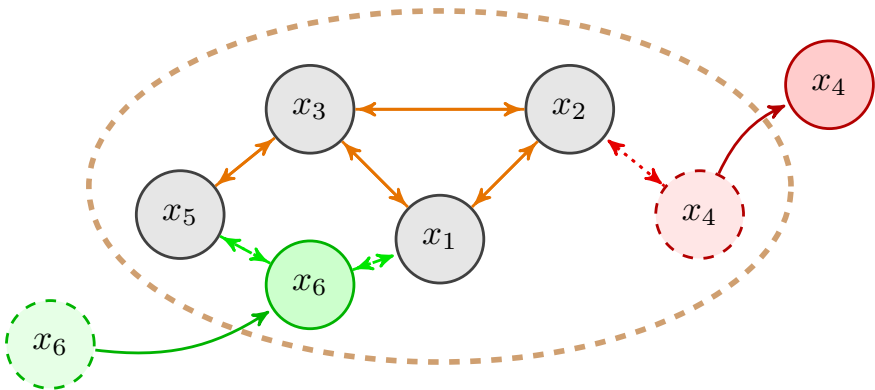
successive local interactions and dealing with huge or private data, as it is spread over the network and not stored in a single unit. Moreover, the failure of a single agent has less impact in decentralized settings, where it is diluted over the agents, than in the presence of a central unit whose single failure would imply that of the whole system. This follows from the fact that each agent in a MAS holds at all times an individual estimate of the solution of the problem it aims at solving, without needing to wait for a central unit to provide it. This usually constitutes an important bottleneck in centralized settings, as the central unit has to deal with numerous computations and communications with all the agents. Hence, in a MAS, each agent has access to a real-time solution, which can be decisive in real-time decision making processes. Finally, MAS allow for scalability, as agents cooperate the same way no matter the system size, which often constitutes a bottleneck for centralized approaches applied to large-scale systems.

## 1.2 Opening our systems

A significant paradox appears when considering multi-agent systems. On the one hand, scalability and flexibility are some of the most cited and desired features of MAS. Yet, on the other hand, most results stated around MAS stand for asymptotic properties under the assumption that the composition of the system remains unchanged: Namely, even if the communications channels between the agents constituting the system can change over time, the agents themselves *remain the same during the entire process*. Looking back at the introductory examples of Fig. 1.1, it is clear that this assumption is rather restrictive: Users are expected to join and leave social networks either through (dis)connections or (un)subscriptions; Birds, and living systems in general, evolve with births and deaths of some individuals or through the fusion or separation of groups; Sensors in a network can experience failures, or new connections might be added; Finally, the cars sharing a stretch of road keep changing all the time as not all drivers ride the same way nor to the same destination.

The apparent contradiction between the desired scalability and flexibility features and the fixed composition of the systems is actually justified when arrivals and departures of agents are rare enough as compared to the timescale at which the studied process takes place: The system is then expected to incorporate the effect of an arrival or departure before the next

one occurs. Nevertheless, this assumption is now getting increasingly challenged. Nowadays, multi-agent systems have to deal with extremely large systems, *e.g.*, with the emergence of the “Internet of Things” (IoT) [Sta18], or with the development of wireless networks of devices in nanomedicine [HB14] that bring system sizes to magnitudes of the order of trillions. At such scales, arrivals and departures become highly significant, as the probability of an arrival or departure and the characteristic length of a process both grow with the system size. This phenomenon typically appears in the study of living systems where the birth and death rates of individuals are usually proportional to their number [DGFGR06]. The rates of arrival, departure or failures can also naturally compare to the slow convergence rates observed in extreme environments, where communications can be difficult or infrequent, *e.g.*, because of losses in transmission. Finally, some systems are inherently subject to frequent arrivals and departures, such as cars that keep joining and leaving a stretch of road [CHW11] or social networks where users keep connecting and disconnecting [PT17].



**Fig. 1.3** Extension of the multi-agent system represented in Fig. 1.2 to an open multi-agent system subject to arrivals (in green) and departures (in red), and whose size and composition thus evolve with time. Each arrival amounts to the introduction of a new agent in the system, connecting with some already existing agents, and each departure amounts to an agent already present leaving the system forever.

In this thesis, we consider a particular class of MAS, which recently started receiving attention, called *open multi-agent systems* (OMAS), which are subject to permanent arrivals and departures of agents during the process that is studied. We present in Fig. 1.3 an illustration of an OMAS which

extends the MAS of Fig. 1.2 with possible arrivals and departures. As discussed above, such configurations appear in many fields of applications, including *e.g.*, dynamic network routing [OYM<sup>+</sup>01] or robotics subject to failures [Ota06].

Typically, OMAS arise in the context of Federated Learning [KMA<sup>+</sup>21, CNDP<sup>+</sup>21] where clients come and go to collaboratively train a model in various contexts, including smart cities, data management or UAVs. By nature, the clients involved in these applications can undergo failures or unavailabilities for instance due to their mobility [Fö07], or simply join or permanently drop out [ITW<sup>+</sup>22]. More generally, with the emergence of the Internet of Things, deploying models over all potential clients at once becomes infeasible, even though standard algorithms such as the well-known *FedAvg* typically assume it to be the case [CNDP<sup>+</sup>21]. Although some recent algorithms start taking joining and leaving clients into account, such as in [MA22], understanding the actual impact of arrivals and departures on such procedures, including consensus and decentralized optimization, and how to analyze them remains unclear so far.

Considering arrivals and departures has a significant impact on the analysis and design of algorithms around MAS, resulting in new challenges. Firstly, every arrival and departure results in an instantaneous change in both the system size and state, making the evolution of the system challenging to analyze on the one hand, and preventing classical asymptotic convergence on the other hand, so that algorithms in open systems cannot be expected to provide “exact” results. Moreover, these events amount to perturbations for the system, which can sometimes even modify the objective pursued by the agents of the system. In particular, at each departure and depending on the nature of the problem, it may be necessary for algorithms to deal with either (i) outdated information related to agents that left the system and that is thus not relevant anymore and should be erased, or (ii) losses of information about agents that left and that should be remembered. Furthermore, preliminary studies have shown that algorithms designed for closed systems do not easily extend to open systems. Moreover, (naive) correction mechanisms for handling few arrivals and departures, or guaranteeing convergence if they were to eventually stop, can be counterproductive if these events keep regularly occurring, see *e.g.* [AHH17, HM16, Bra18].

Hence, algorithms designed for OMAS must be robust to repeated arrivals and departures of agents, and able to cope with potentially variable objectives. Moreover, they generally cannot be simply extended from al-

gorithms designed in closed systems. Designing algorithms tailored for open systems is thus becoming highly relevant nowadays. As an example, being able to handle failures of connected basic devices such as servers for instance is more interesting economically than investing in sophisticated expensive devices that (almost) never fail, as failures are expected to ultimately happen nonetheless. Similarly, tailored representations and analysis techniques are necessary as well to model and study OMAS in general.

### 1.3 Joining and leaving agents

The possibility for agents to join and leave systems during a process has been recognized in computer science, with *e.g.*, specific architectures that have been proposed to deploy large-scale open multi-agent systems, such as the THOMAS project [CGJ<sup>+</sup>09]. In the same line of work, mechanisms allowing distributed computation processes to cope with the failures of nodes or to take advantage of the arrival of new nodes also exist, such as algorithms designed to maintain network connectivity into P2P networks subject to arrivals and departures of nodes [GJR<sup>+</sup>10].

More generally, frameworks similar to open multi-agent systems have been considered, *e.g.*, with Virtual Traffic Lights (VTL) proposed in [Ton18] for autonomous cars to deal with crossroads, which can be assimilated to open systems by nature. In the context of trust and reputation computation, the model FIRE was for instance developed in [HJS06], motivated by the need to determine the reliability of agents joining the system. The study of the behavior of such algorithm however usually remains mostly empirical. In that sense, opinion dynamics models subject to arrivals and departures have been empirically studied in [TIY<sup>+</sup>13, ITY<sup>+</sup>14]. In particular simulation-based analyses were conducted in [SC13] for social phenomena involving arrivals and departures of individuals and in [GJ12] for opinion networks subject to replacements.

Coping with arrivals and departures has also been considered in the more general context of function computation, *e.g.*, with self-stabilization protocols [DGFG06, AAFJ06] where agents can undergo temporary or permanent failures, which can be assimilated to departures. The objective of such protocols is then to ensure asymptotic stability if the systems were to eventually close, *i.e.*, if arrivals and departures were to stop at some point. There is however no guarantee about the transient performance of

such protocols, for which they can even create important disruptions when arrivals and departures become frequent (as discussed *e.g.*, in [Bra18]).

Finally, the Plug and Play implementation, which aims at designing structures on systems where subsystems can be plugged or unplugged has received some attention over the last decade, for instance in the context of model predictive control in [RFF12, RFFt15] or Kalman filtering in [FC18].

## 1.4 Outline

Before jumping to the main course, let us briefly overview the content of this thesis, which is articulated around the exploration and analysis of open multi-agent systems and of their properties.

While each chapter is thought as being self-contained, they are undeniably organized as being part of a whole. The results presented in this thesis fall within the setting described in Chapter 2, which establishes their basis. Our main contributions are then divided in two parts, which present results respectively obtained in the frameworks of consensus and decentralized optimization. Nevertheless, all chapters deal with their own challenges and problems, all related to different aspects of OMAS.

More details on the content of each part and chapter are given below and summarized in Fig. 1.4.

### *Preliminaries*

**Chapter 2** immediately follows this introduction: It sets the scene for the subsequent analyses by introducing definitions, tools and design choices with which we will model OMAS. These design choices will regularly appear throughout this thesis. More specifically, its contribution is twofold. On the one hand, we synthesize and give some mathematical background on *graph and Markov theories* and on the way they are used to model MAS in general. On the other hand, we present how we extend these tools to model OMAS and the corresponding design choices we make in the scope of this thesis. Those have a significant impact on the way we model *arrivals and departures* of agents and the *stochastic evolution* of our systems, but also on the way we *define problems in OMAS*, which strongly influences the challenges to be handled in analyses.

## *Part I: Consensus*

In the first part we investigate the rather basic *consensus* problem in open systems, which allows studying some general properties of OMAS.

- **Chapter 3** presents and defines the consensus problem and several of its variations as they can be found in the literature. It provides an overview of the existing results related to such problems in closed systems, and introduces their extension to open systems or similar settings as they have been considered so far. In particular we define in this chapter the Gossip algorithm, which will be the main case study considered in this part.
- In **Chapter 4** we investigate the *representation* of OMAS of *variable size*. Following preliminary results obtained in [HM17] by J. M. Hendrickx and S. Martin, we show that OMAS can be analyzed via size-independent quantities called *descriptors* whose evolution is characterized by a finite-dimensional dynamical system. We then apply this approach to analyze the behavior of pairwise gossip interactions for the consensus problem. These results rely on tools taken from Markov theory to analyze the evolution of the system state with respect to that of the system size.

[MMH20] Charles Monnoyer de Galland, Samuel Martin and Julien M. Hendrickx, “**Modelling Gossip Interactions in Open Multi-Agent Systems**”, in IEEE Transactions on Automatic Control (*submitted*).

- **Chapter 5** focuses on the derivation of *performance limitations* as a performance criterion for consensus algorithms, as usual convergence is never achieved anyway. The results require the introduction of “*knowledge sets*” to model the *variable information* available to agents, and consist in lower bounds on the performance of any algorithm for some interaction scheme, which we then apply to study the performance of the Gossip algorithm.

[MH19] Charles Monnoyer de Galland and Julien M. Hendrickx, “**Lower bound performances for average consensus in open multi-agent systems**”, in Proceedings of the 58th IEEE Conference on Decision and Control (CDC 2019), Nice (France), pages 7429-7434, December 2019 (*published*).

[MH20] Charles Monnoyer de Galland and Julien M. Hendrickx, “**Fundamental Performance Limitations for Average Consensus in Open Multi-Agent Systems**”, in IEEE Transactions on Automatic Control (*to be published*).

*Part II: Decentralized optimization*

In the second part we focus on the analysis of decentralized optimization algorithms, and in particular dedicated to the *resource allocation* problem. All the results of Part II were obtained in collaboration with Renato Vizuete from L2S CentraleSupélec (Paris) and GIPSA-Lab (Grenoble).

- **Chapter 6** reviews the literature related to decentralized optimization. It notably presents the resource allocation problem, which is at the core of this part, and the *Random Coordinate Descent* algorithm (RCD) which is the main case study that we consider. It also reviews existing results related to decentralized optimization in open systems, and presents the so-called *online optimization* framework which deals with problems sharing similar properties.
- In **Chapter 7** we study the convergence of the RCD algorithm in systems subject to replacements. We characterize the evolution in expectation of the *instantaneous error*, defined as the distance between the estimator and the minimizer, by studying the effects of replacements and of the algorithm separately. We show that the expected asymptotic error is bounded, with some conservatism following the over-approximation of the instantaneous impact of replacements on the error, which accumulates over the iterations.

[MVH<sup>+</sup>21] Charles Monnoyer de Galland, Renato Vizuete, Julien M. Hendrickx, Paolo Frasca and Elena Panteley, “**Random coordinate descent algorithm for open multi-agent systems with complete topology and homogeneous agents**”, in Proceedings of the 60th IEEE Conference on Decision and Control (CDC 2021), Austin (Texas, USA), December 2021 (*to be published*). This paper will receive the *2022 Outstanding Student Paper Prize* awarded by the Networks and Communication Systems TC during CDC2022.

[MVH<sup>+</sup>22] Charles Monnoyer de Galland, Renato Vizuete, Julien M. Hendrickx, Elena Panteley and Paolo Frasca, “**Random Coordinate Descent for Resource Allocation in Open Multi-Agent Systems**”, in IEEE Transactions on Automatic Control (*submitted*).

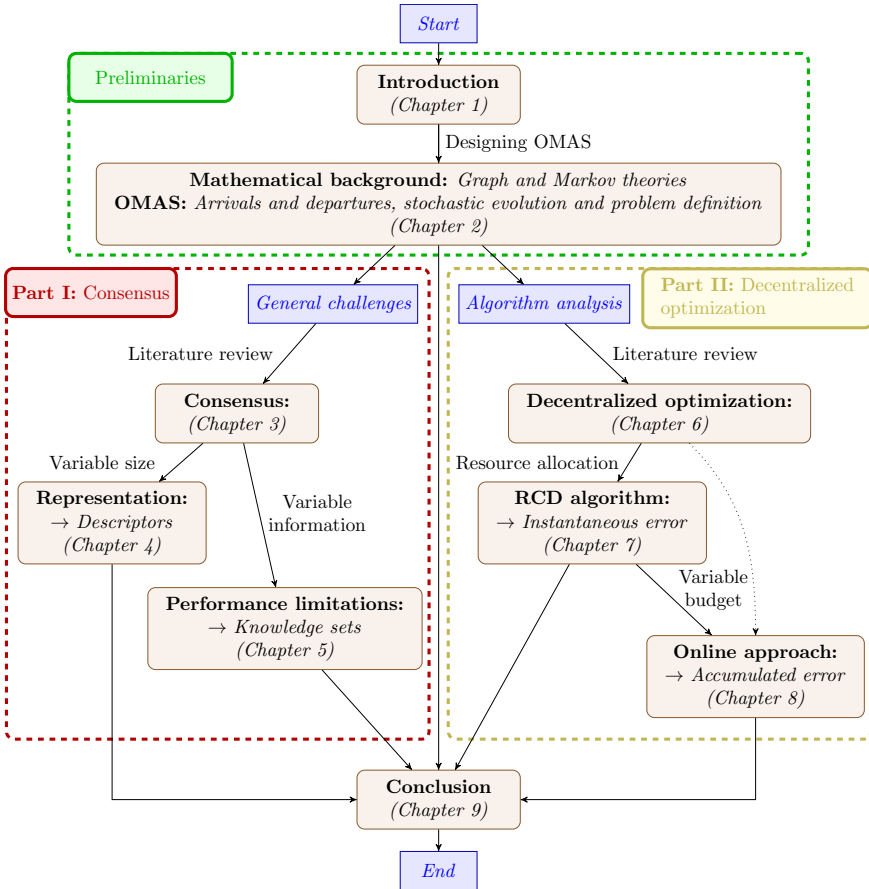
- In **Chapter 8** we consider a variation of the resource allocation problem with a *variable budget* to allocate. Inspired by *online optimization*, we analyze it with new metrics based on the *error accumulated* over the iterations. This approach allows directly evaluating the accumulated effects of replacements rather than the instantaneous one, so

that reasonable bounds can be obtained for well-designed algorithms in a decent setting.

[VMH<sup>+</sup>22] Renato Vizuete, Charles Monnoyer de Galland, Julien M. Hendrickx, Paolo Frasca, Elena Panteley, “**Resource allocation in open multi-agent systems: an online optimization analysis**”, in Proceedings of the 61st IEEE Conference on Decision and Control (CDC 2022), to be held in Cancun (Mexico), December 2022 (*to be published*).

*Part III: Conclusions*

Finally, we give some conclusions and perspectives in **Chapter 9**.



**Fig. 1.4** Outline of this thesis.

# 2

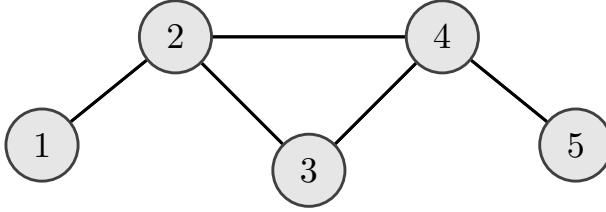
## On the modelling of OMAS

**T**HIS chapter introduces general concepts related to the study of open multi-agent systems, and is organized as follows. We provide background on graph theory in Section 2.1, and present in Section 2.2 the graph-based model we use throughout the thesis to represent OMAS, in particular regarding arrivals and departures. In Section 2.3, we give some elements of probability theory and on Markov chains, as those will be used to model the event-based (stochastic) evolution of OMAS, introduced in Section 2.4. Finally, in Section 2.5, we present the different types of problems that can be defined over OMAS, and the corresponding challenges they imply. Sections 2.1 and 2.3 thus aim at giving some **mathematical background** for the concepts we will use throughout this thesis, while Sections 2.2, 2.4 and 2.5) introduce **design choices** we make all along the thesis in order to model OMAS, and thus already amount as contributions.

### 2.1 Elements of graph theory for modelling MAS

The interactions between the agents constituting a multi-agent system are commonly characterized with a *graph*. Graphs can be graphically represented as circles linked with each other. In the context of MAS, these circles correspond to the different agents, possibly interacting with each other

though the links, depending on the physics of the system. For instance, Fig. 2.1 represents the underlying graph corresponding to the examples of MAS from Fig. 1.1 in Chapter 1. In the case of *e.g.*, a social network, each circle corresponds to a user and a link between two users to a connection between them, meaning that they are able to communicate with each other.



**Fig. 2.1** Graph constituted of 5 interconnected nodes. This graph corresponds to the examples presented in Fig. 1.1, where each node is an agent, and each edge translates the possibility for the two connected agents to interact with each other.

In this section, we provide basic definitions and properties issued from classical graph theory, which are commonly used to represent multi-agent systems. Most of this material is taken and adapted from the book “*Lectures on Network Systems*” of Francesco Bullo [Bul22].

### 2.1.1 Basic definitions and concepts

**Definition 2.1** (Graph). A graph is a pair  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V}$  denotes the set of nodes of the graph, and  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  denotes the set of edges. The existence of an edge  $(i, j) \in \mathcal{E}$  means that the nodes  $i, j \in \mathcal{V}$  are connected.

As an example, the graph represented in Fig. 2.1 is constituted of 5 nodes  $\mathcal{V} = \{1, 2, 3, 4, 5\} \subseteq \mathbb{N}$ , and the edges are given by

$$\mathcal{E} = \{(1, 2), (2, 3), (2, 4), (3, 4), (4, 5)\}.$$

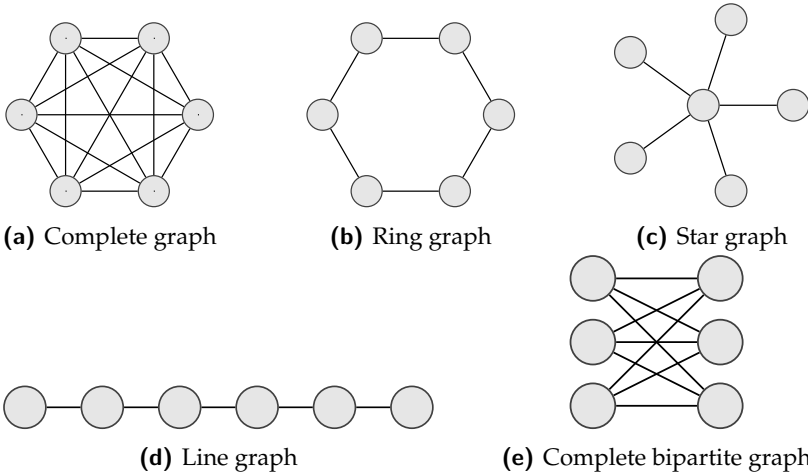
Throughout this thesis we will equivalently use  $\mathcal{V}$  to denote *the set of agents constituting a MAS defined over a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ .*

The graph from the example above is *undirected*. This means that the edges have no orientation, so that two connected nodes interact in both directions. In opposition, edges in *directed* graphs are oriented, and only allow for interactions to take place from one specific agent to the other one. Directed edges are commonly represented with an arrow. In this thesis,

we will however only focus on undirected graphs. We give below a few examples of basic graphs that will be of use throughout this thesis.

**Example 2.2.** We give several basic graphs of  $n$  nodes, illustrated in Fig. 2.2:

- (a) Complete graph: every pair of nodes is connected by an edge;
- (b) Ring graph: every node has two neighbours, so that the nodes and edges can be arranged as those of a regular polygon;
- (c) Star graph: all edges are connected to a specific node, called center;
- (d) Line graph: the nodes can be ordered as a sequence so that the  $n - 1$  edges connect the subsequent nodes of the sequence;
- (e) Complete bipartite graph: the nodes are divided into two groups, and each node is connected to and only to all those of the other group.



**Fig. 2.2** Examples of the specific graphs defined in Example 2.2 with 6 nodes.

Two nodes  $i, j \in \mathcal{V}$  are called *neighbours* if there exists an edge between them, i.e., if  $(i, j) \in \mathcal{E}$ . We denote the set of neighbours of  $i \in \mathcal{V}$  by

$$\mathcal{N}_i = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}, \quad (2.1)$$

and call *degree* of  $i \in \mathcal{V}$  the number of its neighbours:  $\delta_i = |\mathcal{N}_i|$ .

A *path* is an ordered sequence of nodes, where each pair of consecutive nodes of the sequence are connected by an edge. A graph is said to be *connected* if there exists a path between any two nodes  $i, j \in \mathcal{V}$ .

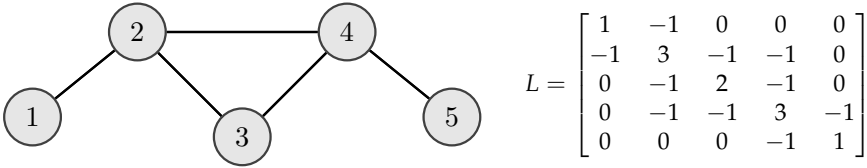
## 2.1.2 The Laplacian of a graph and its properties

One of the most classical ways to algebraically represent graphs is the so-called *Laplacian matrix*, defined below.

**Definition 2.3** (Laplacian matrix). For a (undirected) graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  with  $\mathcal{V} = \{1, \dots, n\}$ , the Laplacian matrix of  $\mathcal{G}$  is the matrix  $L \in \mathbb{R}^{n \times n}$  such that

$$[L]_{i,j} = \begin{cases} \delta_i & \text{if } i = j; \\ -1 & \text{if } (i, j) \in \mathcal{E}; \\ 0 & \text{otherwise} \end{cases} \quad (2.2)$$

More general definitions for the Laplacian matrix can be found in the literature, e.g., for directed graphs with possibly weighted edges. These are omitted here as they are not considered in this thesis. Observe that in the case of undirected graphs, the Laplacian matrix is always symmetric. An example of Laplacian matrix for the graph of Fig. 2.1 is given in Fig. 2.3.



**Fig. 2.3** Laplacian matrix (on the right) corresponding to the given graph (on the left): Each diagonal element is the degree of the corresponding node, and each off-diagonal element is either  $-1$  or  $0$  depending on whether an edge exists between the corresponding nodes.

Observe the rows and columns of the Laplacian matrix sum up to zero by definition. This means that  $\mathbf{1}_n$  is an eigenvector of any Laplacian matrix  $L$  with corresponding eigenvalue  $0$ , as stated in the following property.

**Property 2.4.** ([Bul22, Lemmas 6.2 and 6.4]) Let  $L$  be the Laplacian matrix of an undirected graph, then

$$L\mathbf{1}_n = \mathbf{0}_n \quad \text{and} \quad \mathbf{1}_n^\top L = \mathbf{0}_n^\top. \quad (2.3)$$

More generally, any (symmetric) matrix  $L$  with nonnegative diagonal elements and nonpositive off-diagonal elements which satisfies the property above is called *Laplacian*. One can then show that all the eigenvalues of such matrices are all nonnegative, resulting in the following property.

**Property 2.5.** ([Bul22, Lemmas 6.2 and 6.5]) Let  $L$  be the Laplacian matrix of an undirected graph, then

- (a) all eigenvalues of  $L$  are real and nonnegative;
- (b) at least one eigenvalue of  $L$  is zero.

Therefore, the eigenvalues of  $L$  are commonly written as

$$0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n. \quad (2.4)$$

The second smallest eigenvalue  $\lambda_2$  of  $L$  is called *algebraic connectivity* of the corresponding graph  $\mathcal{G}$ , and satisfies several useful properties. We present some of them in the following proposition.

**Proposition 2.6.** ([Bul22, Corollary 6.8 and Lemma 6.9]) Let  $L$  be the Laplacian matrix of an undirected graph  $\mathcal{G}$ , then

- (a)  $\mathcal{G}$  is connected if and only if  $\lambda_2 > 0$ ;
- (b) the algebraic multiplicity of the eigenvalue 0 of  $L$  corresponds to the number of connected components of  $\mathcal{G}$ , i.e., of connected subgraphs of  $\mathcal{G}$ ;
- (c)  $\lambda_2$  satisfies

$$\lambda_2 = \min_{z \in \mathbb{R}^n: z \perp \mathbf{1}_n} \frac{z^\top L z}{\|z\|^2}. \quad (2.5)$$

Finally, we introduce the following definitions building on Laplacian matrices, and which are used in Chapter 7.

**Definition 2.7** (Pseudoinverse Laplacian). ([Bul22, Lemma 6.12]) Let  $L$  be a Laplacian matrix with decomposition  $L = U \text{diag}(0, \lambda_2, \dots, \lambda_n) U^\top$  where  $U$  is orthogonal, then the Moore-penrose pseudoinverse of  $L$  is

$$L^\dagger := U \begin{bmatrix} 0 & 0 & \dots & 0 \\ 0 & 1/\lambda_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/\lambda_n \end{bmatrix} U^\top, \quad (2.6)$$

and satisfies

$$L^\dagger = (L^\dagger)^\top \succeq 0; \quad L^\dagger \mathbf{1}_n = (L^\dagger)^\top \mathbf{1}_n = \mathbf{0}; \quad LL^\dagger = L^\dagger L = I_n - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^\top.$$

**Definition 2.8** (Effective resistance). ([Bul22, Example 6.13]) Let  $L$  be the Laplacian matrix of an undirected graph  $\mathcal{G}$ , then the effective resistance between two nodes  $i, j \in \mathcal{V}$  is defined as

$$r_{i,j}^{\text{eff}} = (\mathbf{e}_i - \mathbf{e}_j)^\top L^\dagger (\mathbf{e}_i - \mathbf{e}_j). \quad (2.7)$$

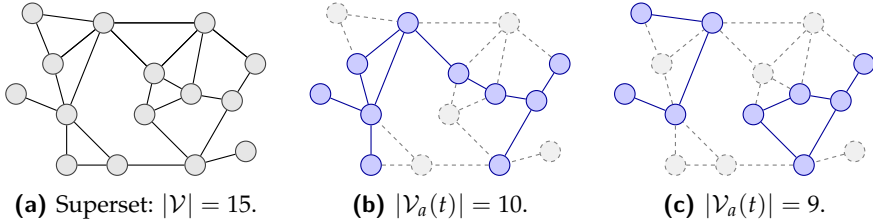
## 2.2 Modelling arrivals and departures

By definition, OMAS are subject to arrivals and departures of agents, and therefore, the way we model them is crucial to properly define and analyze problems. Typically, the way agents are added and removed from the system, their knowledge about it and how they are (dis)connected play an important role in most of the problems that can be studied. In this section, we discuss two models for arrivals and departures whose main difference lies in the ability of agents to potentially come back in the system after leaving. In the same time, we properly define the model we will consider in this thesis, and highlight some of the main challenges and advantages coming alongside.

### 2.2.1 Finite superset of agents

In this first model, we consider that the agents leaving the system eventually come back, in such way that joining agents all end up having already been part of the system at least once in the past. This can equivalently be formulated as the existence of a time-invariant finite “superset”  $\mathcal{V}$  containing all the agents that are or will be part of the system at some time. In this case, only a portion of the agents is *active* at a given time instant  $t$  (denote it  $\mathcal{V}_a(t) \subseteq \mathcal{V}$ ), which constitutes the system at that time. The dimension of the superset thus remains constant, and interactions are only possible between two connected active agents. See for instance Fig. 2.4 which shows the underlying graph of a superset of 15 agents, and two instances of activations of agents in that superset.

Let  $n$  be the (constant) size of the superset in such modelling. In this formulation, one typically defines a vector  $\mathbf{a}(t) \in \{0, 1\}^n$ , which characterizes the activation state of the agents. This allows directly applying standard tools from classical (closed) multi-agent systems to perform analyses,



**Fig. 2.4** Superset formulation in a system of potentially 15 agents. The pre-existing fixed topology of the superset is shown in (a), and (b)-(c) present two possible subsystems (in blue) generated by different activation vectors.

where the topology of the network and the behavior of the agents are dynamical and depend on  $\mathbf{a}(t)$ .

This approach thus allows relying on standard tools applied over a finite time-invariant network with pre-existing fixed topology and on the stochastic properties of the interactions in the network to derive results. It was used *e.g.*, in [VMH18, VFP21] for consensus problems with this type of open systems, and relates with existing analyses in the context of decentralized optimization with time-varying topologies [NO13, OPS18].

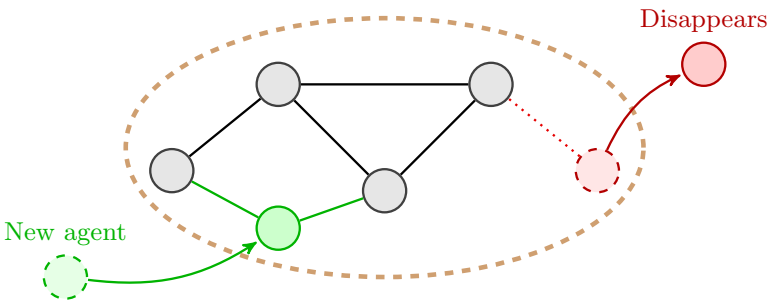
It typically makes sense for applications where a fixed set of agents is frequently subject to (de)activations, such as smart grids [VFP21] or sensor networks [PKP06] which might be subject to disconnections of devices. It also appears in the context of federated learning, where clients may be unavailable at some times, or join the network asynchronously due to their increasing number which prevents them to be active all at once [CNDP<sup>+</sup>21].

Yet, this also means this model requires working on (potentially huge) graph topologies. Moreover, it is limited by the fixed size of the superset, as it does not allow for new agents to permanently join the system nor efficiently handles permanent departures of agents; These tend to become common with the emergence of *e.g.*, mobile clients in the context of federated learning that may permanently leave or join a network during the model training [ITW<sup>+</sup>22]. In the same idea, such model requires taking into account the preliminary information than an agent that reconnects might hold about the system before it left.

In general, this formulation does not fit with the way we define open systems in Chapter 1, and in that sense, we will prefer the second formulation presented below.

## 2.2.2 Dynamic composition

In this second model, we consider that every joining agent is new, *i.e.*, they were never in the system before, and that every departure is permanent, so that leaving agents never come back. This means that the information held by a leaving agent is poised to never come back and might be lost or irrelevant forever, and that an agent joining the system supposedly has no information about it at its arrival. Moreover, each joining agent must be connected to the network through the creation of new edges in the graph, and the edges connecting a leaving agent disappear as it leaves. An illustration is given in Fig. 2.5.



**Fig. 2.5** OMAS representation in the sense of Section 2.2.2: Each joining agent (in green) is completely new, *i.e.*, it was never in the system before and must be connected to the network; each leaving agent (in red) disappears forever, and the edges connecting it disappear as well.

This model thus significantly differs from the previous one, and more generally from time-varying networks with fixed agents, as it is not restricted to a finite pool of agents. Rather, it potentially allows infinitely many agents to go through the system, and forbids agents from coming back once they left. Observe that this model can be interpreted as assuming that agents coming back in the system have been away during enough time to behave as if they were never there before. In that sense, this formulation is more suited for settings such as those described in Chapter 1 in the context of federated learning, social networks or autonomous cars, where agents permanently join and leave, or reconnect as if they were new.

This approach is thus the one we consider all along this thesis. In parallel, it was also used *e.g.*, in [FF21, DFG20] for consensus, where the authors distinguished three subgroups of agents at each time step: those leaving the system, those joining, and those remaining. It was also applied in

[HR20] to study the decentralized gradient descent algorithm in open systems of this type subject to replacements of agents.

However, this model also has some limitations, as it prevents the use of standard tools from classical multi-agent systems, and requires the definition of new ones to handle the variable size and information in the system (more details about this *e.g.*, in Section 3.3). Moreover, it requires defining how arriving agents connect to the network. One can do it by properly defining some connection rule at arrivals, *e.g.*, to maintain connectivity. In this thesis, we will often use “all-to-all” pairwise interactions, meaning that the graph is complete and that interactions happen asynchronously upon activation of an edge with a given probability. This has two advantages: (i) complete graphs are easy to model and analyze, *e.g.*, at each arrival the joining agent simply connects with everyone; (ii) by properly defining the activation probabilities on each edge, one can virtually define any specific graph topology. Another approach used *e.g.*, in Chapter 7 consists in fixing the topology in advance and considering replacements which maintain it.

**Remark 2.1.** *A third model, which extends the first one, consists in the definition of an infinite superset, such that the time-varying system is always “included” in that superset. Interestingly, this approach would theoretically compensate the drawbacks of both approaches presented above, as it allows at the same time agents to come back and infinitely many agents to go through the network. It however requires the use of different tools such as infinite graphs and networks [PMS93, Soa94, HMQS16] to represent the superset. One could also rely on the concept of graphons, which correspond to the limits of convergent sequences of dense graphs and can be used to study families of networks with common patterns [Lov12, VFG20]. Such tools are however out of the scope of this thesis.*

*More generally, one could consider “hybrid” models, where new agents might be added to the network, and departures would amount to (temporary) disconnections from the network. This approach has however not been explored so far to the best of our knowledge.*

## 2.3 Markov chains and counting processes

We present some elements taken from probability and Markov chains theories, as those will be used to characterize the stochastic evolution of OMAS. For the sake of concision, we will only cover the material that will be directly used in this thesis, without detailing proofs or intermediate results.

## 2.3.1 Elements of probability theory

The content of this section is taken and adapted from the book “*Understanding Markov Chains: Examples and Applications*” of Nicolas Privault [Pri18].

*Basic notions*

Probability theory builds on the notion of *Probability space*  $(\Omega, \mathcal{F}, \mathbb{P})$ , which is constituted of three components:

1. A *set of outcomes*  $\Omega$ , containing the possible outcomes of a given experiment;
2. A *set of events*  $\mathcal{F}$ , constituted of subsets of  $\Omega$  called *events*;
3. A *probability measure*  $\mathbb{P} : \mathcal{F} \rightarrow [0, 1]$  that assigns a probability  $\mathbb{P}[A]$  to any event  $A \in \mathcal{F}$ .

The components described above satisfy several properties, which we do not detail exhaustively here, and we refer to [Pri18, Chap. 1] for details. In particular,  $\mathcal{F}$  defines a  $\sigma$ -algebra of  $\Omega$ , and hence must contain  $\emptyset$  and  $\Omega$  itself, such that  $\mathbb{P}[\Omega] = 1$ .

**Example 2.9** (Rolling a die). *Let us consider a simple six-sided die. The set of outcomes for making it roll once is  $\Omega = \{1, 2, 3, 4, 5, 6\}$ . One could then arbitrarily define the (valid) set of events  $\mathcal{F} = \{\emptyset, \Omega, \{1, 3, 5\}, \{2, 4, 6\}\}$ , which corresponds to the information relative to the parity of the result from rolling the die. The event  $A = \{2, 4, 6\}$  thus corresponds to “the result of rolling the die is an even number”, and if the die is regular there holds  $\mathbb{P}[A] = \frac{1}{2}$ .*

For an event  $A \in \mathcal{F}$ , we use  $\bar{A}$  to denote its *complementary event* such that  $\mathbb{P}[\bar{A}] = 1 - \mathbb{P}[A]$  (by definition one has  $\bar{A} \in \mathcal{F}$ ). In Example 2.9,  $\bar{A}$  thus corresponds to the event “the result of rolling the die is an odd number”. For two events  $A, B \in \mathcal{F}$ , we define the *conditional probability* as

$$\mathbb{P}[A|B] = \frac{\mathbb{P}[A \cap B]}{\mathbb{P}[B]}, \quad (2.8)$$

called “probability of  $A$  given  $B$ ”, and which stands for the probability of the event  $A$  given the knowledge that the event  $B$  happened. This definition allows defining the notion of *independence*.

**Definition 2.10** (Independence). *Two events  $A$  and  $B$  are said independent if*

$$\mathbb{P}[A|B] = \mathbb{P}[A], \quad (2.9)$$

or equivalently if

$$\mathbb{P}[A \cap B] = \mathbb{P}[A] \mathbb{P}[B]. \quad (2.10)$$

### Random variables

Definition 2.11 and those that follow stand for real-valued random variables, *i.e.*, where the value they take is drawn from the state space  $\mathbb{R}$ . One could nevertheless define random variables on a more general state space, and adapt the definitions that follow accordingly.

**Definition 2.11.** *A real-valued random variable  $X$  is a random quantity that depends on the outcome of an experiment, and is defined as the mapping*

$$X : \Omega \rightarrow \mathbb{R} : \omega \mapsto X(\omega). \quad (2.11)$$

Random variables are generally simply written as  $X$ , by dropping the dependency to  $\omega$ . In the remainder of this section, we focus on *continuous* random variables.

**Definition 2.12 (CDF and PDF).** *Let  $X$  be a continuous random variable, then the Cumulative Distribution Function (CDF) of  $X$  is the function  $F_X$  satisfying*

$$F_X(x) = \mathbb{P}[X \leq x]. \quad (2.12)$$

*If  $X$  is absolutely continuous (i.e., if  $F_X$  is absolutely continuous), then the Probability Density Function (PDF) of  $X$  is the function  $f_X(x) = \frac{d}{dx}F_X(x)$ , and in particular*

$$F_X(x) = \mathbb{P}[X \leq x] = \int_{-\infty}^x f_X(x) dx. \quad (2.13)$$

By definition, one has  $\lim_{x \rightarrow \infty} F_X(x) = 1$ . The CDF and PDF of a random variable allow defining its *expected value* as follows.

**Definition 2.13 (Expectation).** *Let  $X$  be a real-valued continuous random variable, then the expected value  $X$  is defined as*

$$\mathbb{E}[X] = \int_{\mathbb{R}} x f_X(x) dx. \quad (2.14)$$

More generally, let  $\phi : \mathbb{R} \rightarrow \mathbb{R}$  be a measurable function of  $X$ , then

$$\mathbb{E}[\phi(X)] = \int_{\mathbb{R}} \phi(x) f_X(x) dx. \quad (2.15)$$

## 2 | On the modelling of OMAS

The expected value  $\mathbb{E}[\phi(X)]$  thus characterizes the “averaged” behavior of a given function  $\phi$  of the random variable  $X$ . For two random variables  $X$  and  $Y$ , we also define the *conditional expectation*  $\mathbb{E}[X|Y]$  as the expected value of  $X$  evaluated given the knowledge of the variable  $Y$ , and in particular one has

$$\mathbb{E}[\mathbb{E}[X|Y]] = \int_{\mathbb{R}} \mathbb{E}[X|Y = y] f_Y(y) dy = \mathbb{E}[X]. \quad (2.16)$$

### 2.3.2 Continuous-time Markov chains

The content of this section is taken and adapted from “*Understanding Markov Chains: Examples and Applications*” of Nicolas Privault [Pri18, Chapter 9], “*Applied Stochastic Processes*” of Grigorios A. Pavliotis [Pav09, Chapter 4], and “*Markov Processes*” of Andreas Eberle and [Ebe20, Chapter 5].

#### *Basic definitions*

Extending the concepts previously introduced, we now focus on stochastic processes, which can be interpreted as time-varying random variables.

**Definition 2.14** (Stochastic process). *A continuous-time stochastic process is a collection of random variables  $(X_t)_{t \in \mathbb{R}_+}$  indexed by  $\mathbb{R}_+$ .*

A stochastic process is said to be a *Markov process* if it satisfies the Markov property, which essentially means that the evolution of the process given its past is entirely characterized by its most recent state.

**Definition 2.15** (CTMC). *A continuous-time stochastic process  $(X_t)_{t \in \mathbb{R}_+}$  with state space  $\mathbb{S}$  is said to be a continuous-time Markov chain (CTMC) if for all  $j \in \mathbb{S}$  and all  $0 < s_1 < s_2 < \dots < s_n < t$  there holds*

$$\mathbb{P}[X_t = j | X_{s_1}, X_{s_2}, \dots, X_{s_n}] = \mathbb{P}[X_t = j | X_{s_n}]. \quad (2.17)$$

The processes defined above are also called *continuous-time Markov processes* or *Markov jump processes*, as their state only evolves with instantaneous jumps at random times, and remains constant in between. We focus in particular on *time-homogeneous* CTMC, namely where transitions do not depend on the time: For all  $t \in \mathbb{R}_+$ ,  $\tau \geq 0$  and  $i, j \in \mathbb{S}$  there holds

$$\mathbb{P}[X_{t+\tau} = j | X_t = i] = \mathbb{P}[X_\tau = j | X_0 = i]. \quad (2.18)$$

In that setting, one can define *transition rates* between the different states of  $S$  as the  $\lambda_{i,j} \in \mathbb{R}_+$  such that for all  $t \in \mathbb{R}_+$  and  $i, j \in S$ :

$$\mathbb{P}[X_{t+\tau} = j | X_t = i] = \lambda_{i,j}\tau + o(\tau). \quad (2.19)$$

There is an extremely dense theory existing around CTMC and Markov chains in general, and we refer to [Pri18, Ebe20, Pav09] for additional details. In this thesis, we will represent CTMC using the so-called *Infinitesimal generator* of  $(X_t)_{t \in \mathbb{R}_+}$ , which determines the infinitesimal transitions of  $X_t$  between the states of  $S$ .

#### Transition rate matrix

When the state space  $S$  of the process is countable, then the infinitesimal generator reduces to the so-called *Transition rate matrix*  $Q$  defined below.

**Definition 2.16** (Transition rate matrix). *Let  $(X_t)_{t \in \mathbb{R}_+}$  be a CTMC with state space  $S = \{1, \dots, N\}$ , and let  $\lambda_{i,j}$  be the transition rates from (2.19), then the Transition rate matrix is the matrix  $Q \in \mathbb{R}^{N \times N}$  satisfying*

$$[Q]_{i,j} = \begin{cases} \lambda_{i,j} & \text{if } i \neq j \\ -\sum_{\ell \neq i} \lambda_{i,\ell} & \text{otherwise} \end{cases}. \quad (2.20)$$

The matrix  $Q$  has several uses when analysing CTMC. In particular, let  $\pi(t) \in \mathbb{R}^N$  be the vector such that  $\pi_i(t) = \mathbb{P}[X_t = i]$ , then one shows that

$$\frac{d}{dt} \pi(t) = Q^\top \pi(t), \quad (2.21)$$

so that

$$\pi(t) = e^{tQ^\top} \pi(0). \quad (2.22)$$

The definition of  $Q$  above is limited to countable state spaces. It is thus not sufficient to handle state spaces involving real values for instance, which typically arise with OMAS and lead to the next definitions.

#### Infinitesimal generator operator

A more general formulation than the transition rate matrix applicable for more elaborate CTMC is the so-called *Infinitesimal generator operator*. We first define the related notions of *transition kernel* and *instantaneous jump rate*, which generalize the transition probabilities and rates of CTMC.

**Definition 2.17** (Transition kernel). *Let  $(\mathbb{S}, \mathcal{S})$  be a measurable space (i.e., where  $\mathcal{S}$  is a set of subsets of  $\mathbb{S}$ ), and let  $X_t$  be a homogeneous Markov process with state space  $\mathbb{S}$ , then the transition kernel of  $X_t$  is the probability measure  $\kappa_\tau : \mathbb{S} \times \mathcal{S} \rightarrow [0, 1]$  which satisfies for all  $x \in \mathbb{S}$ ,  $B \in \mathcal{S}$  and all  $t$*

$$\kappa_\tau(x, B) = \mathbb{P}[X_{t+\tau} \in B | X_t = x]. \quad (2.23)$$

**Definition 2.18** (Instantaneous jump rate). *Let  $(\mathbb{S}, \mathcal{S})$  be a measurable space, and let  $X_t$  be a homogeneous CTMC with state space  $\mathbb{S}$ , then the instantaneous jump rate of  $X_t$  is the nonnegative measure  $q : \mathbb{S} \times \mathcal{S} \rightarrow \mathbb{R}_+$  which satisfies for all  $x \in \mathbb{S}$ ,  $B \in \mathcal{S}$  and all  $t$*

$$\kappa_\tau(x, B) = q(x, B)\tau + o(\tau). \quad (2.24)$$

The instantaneous jump rate  $q$  thus generalizes the transition rates of (2.19), as it measures the rate at which  $X_t$  jumps from a given state  $x \in \mathbb{S}$  to one of those contained in  $B$ . We now give the following definition of the infinitesimal generator operator.

**Definition 2.19** (Infinitesimal generator operator). *([Pav09, Def. 4.4.2]) Let  $(\mathbb{S}, \mathcal{S})$  be a measurable space, and let  $X_t$  be a homogeneous CTMC with state space  $\mathbb{S}$ . Let  $\mathcal{F}_b(\mathbb{S})$  be the set of bounded measurable functions  $f : \mathbb{S} \rightarrow \mathbb{R}^d$ , then the infinitesimal generator operator of  $X_t$  is the operator  $\mathcal{L} : \mathcal{F}_b(\mathbb{S}) \rightarrow \mathcal{F}_b(\mathbb{S})$  such that for all  $f \in \mathcal{F}_b(\mathbb{S})$  there holds*

$$\mathcal{L}f(x) = \lim_{\tau \rightarrow 0} \frac{\mathbb{E}[f(X_{t+\tau}) | X_t = x] - f(x)}{\tau}. \quad (2.25)$$

One shows that there holds

$$\mathcal{L}f(x) = \int (f(y) - f(x))q(x, dy), \quad (2.26)$$

where the measure  $q$  from Definition 2.18 is used with the infinitesimal element  $dy$  as an argument to define the integral. We refer to the example *Pure jump process* of [Ebe20, Section 5.1] for a similar example.

*Examples of CTMC of interest*

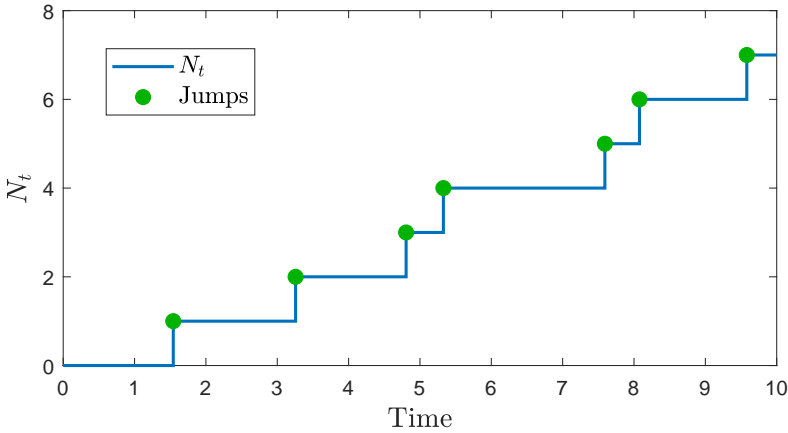
We now give examples of CTMC which will appear in this manuscript, and their corresponding representations.

**Example 2.20** (Poisson process). *The Poisson process is a continuous-time counting process, i.e., where each jump increases  $N_t$  by 1. The occurrence of jumps*

is governed by a Poisson clock, denoted  $\text{Po}(\lambda)$ . This means that the time  $T$  separating two consecutive jumps is a random variable drawn from an exponential distribution of fixed rate  $\lambda$ , i.e.,  $T \sim \text{Exp}(\lambda)$  and

$$\mathbb{P}[T \leq t] = 1 - e^{-\lambda t}. \tag{2.27}$$

See Fig. 2.6 for an illustration of a realization of a Poisson process.

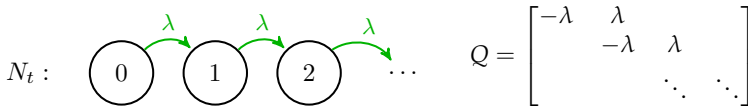


**Fig. 2.6** Single realization of a Poisson process  $N_t$  with rate  $\lambda = 2$ .

Poisson processes as counting processes satisfy several properties. Clearly  $N_t \in \mathbb{N}$  for all  $t$ , and there holds  $N_0 = 0$ . Moreover

$$\mathbb{P}[N_t = n] = \frac{(\lambda t)^n}{n!} e^{-\lambda t}; \quad \mathbb{E}[N_t] = \lambda t. \tag{2.28}$$

We present in Fig. 2.7 an illustration of the CTMC defined by  $N_t$  and the shape of the corresponding Transition rate matrix  $Q$ .



**Fig. 2.7** Representation of a Poisson process (left) and corresponding Transition rate matrix  $Q$  (right).

## 2 | On the modelling of OMAS

Finally, observe that the instantaneous jump rate of the process is given by

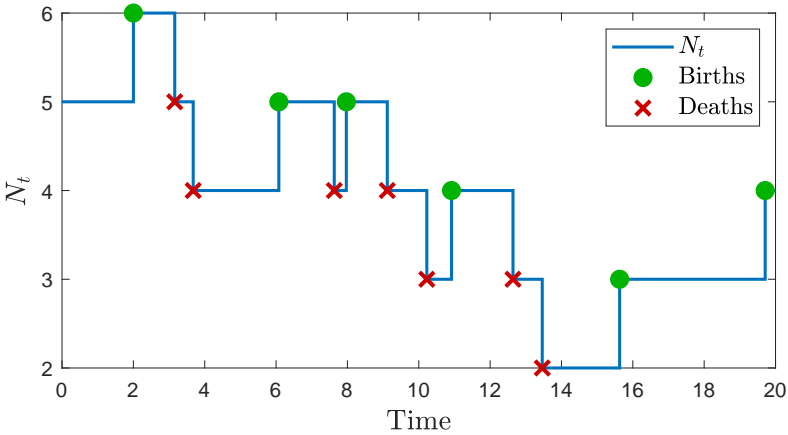
$$q(n, B) = \lambda \chi_B(n + 1), \quad (2.29)$$

where  $\chi_B$  is the indicator function such that  $\chi_B(n) = 1$  if  $n \in B$ , and  $\chi_B(n) = 0$  otherwise. Consider the function  $f(N_t) = N_t$ , then (2.26) becomes

$$\mathcal{L}f(n) = \sum_{y \in \mathbb{N}} (f(y) - f(n)) \lambda \chi_y(n + 1) = \lambda(n + 1 - n) = \lambda, \quad (2.30)$$

which corresponds to the transition rate of the process when  $N_t = n$ .

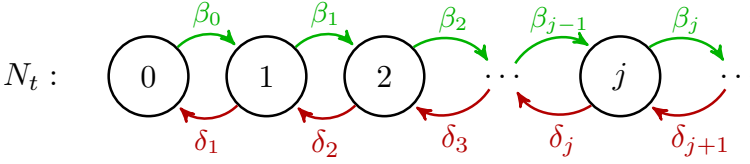
**Example 2.21** (Birth-death process). The Birth-death process is a CTMC subject to two types of jumps: “births”, which increase  $N_t$  by 1, and “deaths”, which decrease it by 1. These jumps follow Poisson clocks with rates that depend on the state of  $N_t$ , so that if  $N_t = j$ , then births happen with rate  $\beta_j \geq 0$  and deaths with rate  $\delta_j \geq 0$ . An illustration of the behavior of that process is given in Fig. 2.8. We illustrate in Fig. 2.9 the CTMC defined by  $N_t$  for the birth-death process.



**Fig. 2.8** Single realization of a Birth-death process  $N_t$  with rates  $\beta_j = \delta_j = \frac{j}{10}$ .

The Transition rate matrix  $Q$  of that process is given by

$$Q = \begin{bmatrix} -\beta_0 & \beta_0 & & & \\ \delta_1 & -(\beta_1 + \delta_1) & \beta_1 & & \\ & \delta_2 & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots \end{bmatrix}. \quad (2.31)$$



**Fig. 2.9** Representation of a Birth-death process: When  $N_t = j$ , then births happen with rate  $\beta_j$  and deaths with rate  $\delta_j$ .

Remember that  $\pi_j(t) = \mathbb{P}[N_t = j]$ , then (2.31) combined with (2.21) yields  $\frac{d}{dt}\pi_0(t) = \delta_1\pi_1(t) - \beta_0\pi_0(t)$  and

$$\frac{d}{dt}\pi_j(t) = \beta_{j-1}\pi_{j-1}(t) + \delta_{j+1}\pi_{j+1}(t) - (\beta_j + \delta_j)\pi_j(t) \quad (2.32)$$

for all  $j \geq 1$ . In particular, if  $N_t$  is ergodic, then there exists  $\pi_j^* = \lim_{t \rightarrow \infty} \pi_j(t)$  for all  $j \in \mathbb{N}$ , and solving the equations above yields

$$\pi_0^* = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=1}^k \frac{\beta_{i-1}}{\delta_i}}; \quad \pi_j^* = \pi_0^* \prod_{i=1}^j \frac{\beta_{i-1}}{\delta_i}. \quad (2.33)$$

for all  $j \geq 1$ . Finally, the instantaneous jump rate of the process is given by

$$q(n, B) = \beta_n \chi_B(n+1) + \delta_n \chi_B(n-1). \quad (2.34)$$

and considering the function  $f(N_t) = N_t$ , then (2.26) becomes

$$\begin{aligned} \mathcal{L}f(n) &= \sum_{y \in \mathbb{N}} (f(y) - f(n)) (\beta_n \chi_y(n+1) + \delta_n \chi_y(n-1)) \\ &= (n+1-n)\beta_n + (n-1-n)\delta_n = \beta_n - \delta_n, \end{aligned} \quad (2.35)$$

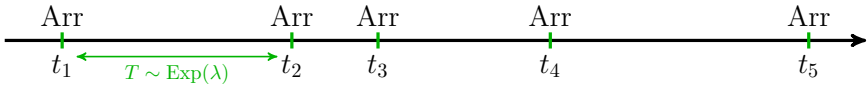
which corresponds to the transition rate of the process when  $N_t = n$ .

## 2.4 Event-based evolution of open systems

The OMAS we consider in this thesis are assumed to evolve solely through instantaneous modifications triggered *e.g.*, by arrivals, departures, or interactions in the network, each of these happening punctually at random

times. The system thus remains constant in between.

Consider as an example the specific case of only-growing systems, *i.e.*, where only arrivals of single agents happen at random times determined with a Poisson clock of fixed rate  $\lambda$ . No departure nor any other type of interaction happens in that system. Its evolution is solely defined by the activation at specific random time-instants  $t_k$  of the clock characterizing the arrivals, which we denote Arr (see Fig. 2.10 for an illustration). This system is thus a CTMC (see Section 2.3.2) whose jumps are triggered by arrivals, which are themselves characterized by a Poisson process of parameter  $\lambda$ .



**Fig. 2.10** Timeline presenting the time-instants at which modifications happen in the specific only-growing system described above. The time  $T$  between any two consecutive arrivals is randomly drawn from an exponential distribution with parameter  $\lambda$ , consistently with the Poisson clock determining their occurrence.

In general settings, several different processes can be involved in parallel in the evolution of an OMAS (typically arrivals, departures, replacements or interactions in the system). Each process is defined with its own clock, which can potentially depend on other clocks or on the system state. In that sense, the state of the system is a CTMC whose jumps are triggered by the activation of these different clocks.

Given a time-varying quantity  $y$  being modified upon activation of a given clock  $\epsilon$  at some time  $t$ , we denote by

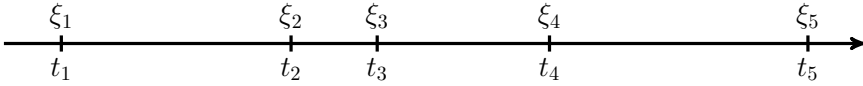
- $y$  or  $y(t^-)$ : the value of  $y$  immediately *before* the activation of  $\epsilon$ ;
- $y^+$  or  $y(t^+)$ : the value of  $y$  immediately *after* the activation of  $\epsilon$ .

Hence, for instance,  $\mathbb{E}[y^+|y, \epsilon]$  stands for the expected value of  $y$  immediately after the activation of the clock  $\epsilon$  given its state immediately before (independently of the time at which it happens). For the simplicity of the interpretation, and with a slight abuse of language, we will refer to the activation of a given clock  $\epsilon$  as an “*event of  $\epsilon$* ”. This terminology can actually be assimilated to the notion of “probabilistic event” from probability theory, as  $\epsilon$  defines the type of modification experienced by the system when a clock ticks, which can be seen as the outcome of an experiment. To avoid ambiguity, we will sometimes define the (probabilistic) event  $\epsilon_t$  that  $\epsilon$  activates at time  $t$ , so that the expression above is equivalent to  $\mathbb{E}[y(t^+)|y(t^-), \epsilon_t]$ .

With this *event-based* formulation, we can define the *event set*  $\Xi$  of the OMAS, as the set of all the existing clocks triggering modifications of the system. From this definition, we can also build the *history* of the process up to some time  $t$  as the sequence of all the events that happened in the system and the time-instant at which they happened until that time:

$$\mathcal{H}(t) = \{(t_1, \xi_1), \dots, (t_k, \xi_k)\}, \quad (2.36)$$

where  $t_i \leq t$  and  $\xi_i \in \Xi$  for all  $i = 1, \dots, k$ . See Fig. 2.11 for an illustration.



**Fig. 2.11** Timeline presenting the time-instants at which a modification of an OMAS happens. Each  $t_i$  corresponds to the (random) time-instant at which an event of  $\xi_i \in \Xi$  happens, triggering a modification of the system state.

Observe that the formulation above typically allows discretizing the time at the time instants where events take place (up to the assumption that two distinct events never happen simultaneously). In that case, one can then alternatively define the history of the process up to the  $k$ -th time step as

$$\mathcal{H}^k := \{(1, \xi_1), \dots, (k, \xi_k)\}, \quad (2.37)$$

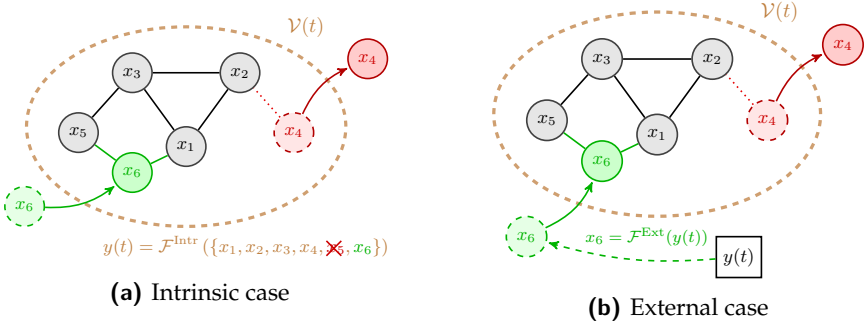
where  $\xi_i \in \Xi$  for all  $i = 1, \dots, k$ . Under the additional assumption that the different processes (and hence the clocks) are independent of each other and of the system states (past or present), one can also define the probabilities of each clock to tick at a given time step, so that the effects of the different clocks can be decoupled in the subsequent analyses. Such approaches will be used *e.g.*, in Chapters 7 and 8.

## 2.5 Intrinsic and external objectives

We distinguish two main types of problems in OMAS, both illustrated in Fig. 2.12, depending on the nature of the objective pursued by the agents in the system: *intrinsic* and *external*. In the former case, the agents aim at solving a problem which directly depends on the composition of the system, and whose solution thus varies as agents join and leave the system. In

## 2 | On the modelling of OMAS

the latter, this solution is defined outside and independently of the system; the initial state of the agents joining the system then typically depends on this solution and is used to solve the corresponding problem.



**Fig. 2.12** Representation of an intrinsic (left) and external (right) problem in OMAS: The objective  $y(t)$  changes as agents join and leave in the intrinsic case, whereas it does not in the external case and rather influences the value taken by agents joining the system.

More formally, let  $y(t)$  be the (potentially time-varying) objective that the system aims at reaching at time  $t$ , and let  $\mathcal{V}(t)$  denote the set of agents in the system at time  $t$ . Let moreover  $x_{i,0}$  denote the initial value held by an agent  $i$  at its arrival in the system. In intrinsic problems, the objective is built upon  $\mathcal{V}(t)$ , and for some function  $\mathcal{F}^{\text{Intr}}$  there holds

$$y(t) = \mathcal{F}^{\text{Intr}}(\{x_{i,0} | i \in \mathcal{V}(t)\}). \quad (2.38)$$

By contrast, in external problems, the objective  $y(t)$  exists independently of the system and contributes to define the initial state of any agent, such that for some function  $\mathcal{F}^{\text{Ext}}$  and for  $t_{i,0}$  the arrival time of agent  $i$  there holds

$$x_{i,0} = \mathcal{F}^{\text{Ext}}(y(t_{i,0})). \quad (2.39)$$

We illustrate this distinction in the following example for the *max consensus problem*: Each agent  $i \in \mathcal{V}$  holds a real value  $x_i \in \mathbb{R}$ , and their objective is to collaboratively estimate the maximal value in the system, *i.e.*,  $\max\{x_i | i \in \mathcal{V}\}$ . A simple algorithm, called *Flooding*, consists in defining pairwise interactions of the type  $x_i^+ = x_j^+ = \max\{x_i, x_j\}$ . In closed system, one can then show that this algorithm solves the problem if the underlying graph is connected, as it floods the system with the maximal

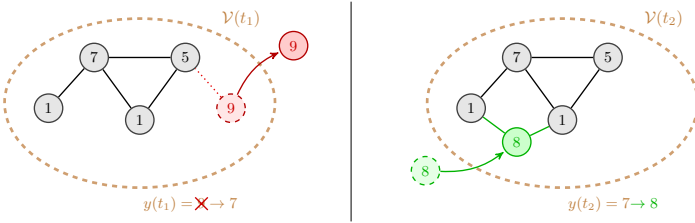
value (more details about all this in Section 3.2).

**Example 2.22.** Let us consider two variations of an open version of the max consensus problem defined above, one intrinsic and one external. Let us assume that each agent  $i \in \mathcal{V}(t)$  holds a nonnegative bounded real quantity  $x_i \leq M \in \mathbb{R}_+$ , for all  $t$ . The initial value  $x_i$  held by agent  $i$  at its arrival in the system is randomly drawn from some distribution while guaranteeing  $x_i \leq M$ .

1. **Intrinsic max consensus:** In this problem, the agents aim at computing the maximal value among those held by the agents present in the system:

$$y(t) = \max \{x_i | i \in \mathcal{V}(t)\}. \quad (2.40)$$

The objective  $y(t)$  thus changes as agents get in and out, and agents must adapt consequently: (i) if the agent holding the largest value leaves, then the other agents must potentially “forget” its value, as the current maximum has changed; (ii) if a joining agent holds the new largest value, then the other agents must learn it. See Fig. 2.13 for an illustration.



**Fig. 2.13** Intrinsic max consensus: effect on the objective  $y(t)$  of the departure at time  $t_1$  (left) and arrival at time  $t_2$  (right) of the agent holding the largest value.

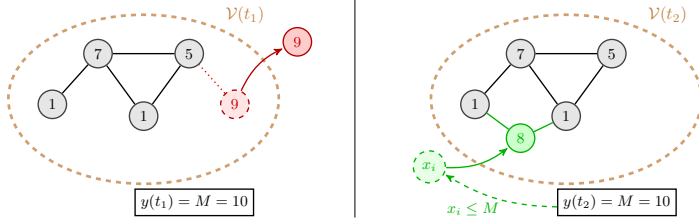
2. **External max estimation:** In this version of the problem, the objective of the agents is to estimate  $M$ , the maximal value that agents potentially hold at their arrivals:

$$y(t) = M. \quad (2.41)$$

In that case the objective  $y(t)$  is not impacted by the agents in the system. Actually, it contributes to defining their initial values, and to estimate it the agents aim at collecting as many information about each other as possible during the process. See Fig. 2.14 for an illustration.

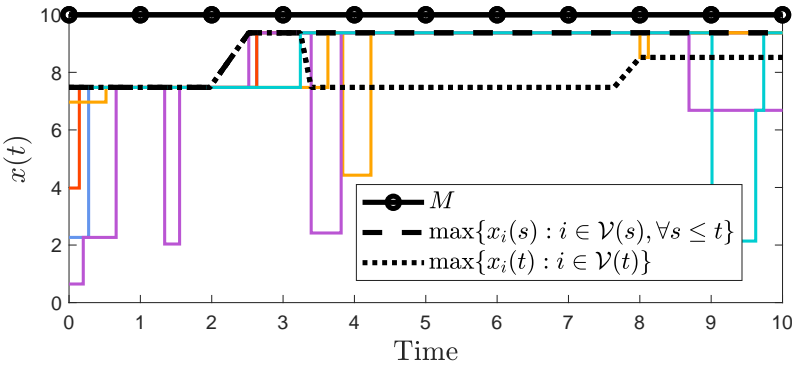
In this setting agents will typically try to estimate an approximation of  $M$ , given by

$$\tilde{y}(t) = \max \{x_i | i \in \mathcal{V}(s), \forall s \leq t\}. \quad (2.42)$$



**Fig. 2.14** External max estimation: effect on the system of the departure at time  $t_1$  (left) and arrival at time  $t_2$  (right) of the agent holding the largest value.

An illustration of the behavior of the Flooding algorithm described above (and detailed in Section 3.2) with respect to all the variations of the objective described above is shown in Fig. 2.15. It shows that in the considered setting the Flooding algorithm is not efficient for handling the intrinsic objective (2.40), as it is not designed to forget outdated information. In opposition, it behaves rather good for approximating the external objective (i.e., for computing  $\tilde{y}(t)$  from (2.42)), for the exact same reason. This highlights how the design of algorithms can fundamentally change depending on the nature of the objective.



**Fig. 2.15** Behavior of the Flooding algorithm for solving the different variations of the open max consensus problem defined in Example 2.22 in a system of 5 agents subject to replacements once every 5 interactions on average. Each plain line is the estimate held by an agent, and the black thick lines are the different objectives.

The intrinsic and external formulations show some sort of interpretation trough the scope of the superset model presented in Section 2.2.1. Let us consider the superset  $\mathcal{V}$ , and the (time-varying) set of active nodes  $\mathcal{V}_a(t)$ ,

then one can see intrinsic objectives as defined over  $\mathcal{V}_a(t)$ , *i.e.*,

$$y(t) = \mathcal{F}^{Intr} (\{x_i | i \in \mathcal{V}_a(t)\}).$$

In that case, external objectives can be approximately seen as the same function defined over the entire superset  $\mathcal{V}$ :

$$y(t) = \mathcal{F}^{Intr} (\{x_i | i \in \mathcal{V}\}).$$

This interpretation can then be pushed further to hybrid models (*i.e.*, subject to activations and deactivations, and where new agents join the system): External objectives are then evaluated over the entire past of the process, which actually amounts to  $\tilde{y}(t)$  defined in (2.42) from Example 2.22.

More generally, the distinction between *intrinsic* and *external* problems already exists in the context of closed systems, but its implication is then rather limited. By contrast, in open settings it has a significant impact on the challenges that must be handled both in analyses and algorithm design.

In particular, since the objective of intrinsic problems depends on the composition of the system, the information related to agents having left the system is most of the time irrelevant and must consequently be eliminated. This can be a serious challenge for algorithm design, as it somehow requires keeping track of the origin and validity of the information held by the agents. This is rather clear from Example 2.22, and strategies for this problem can be found *e.g.*, in [AHH17] where the max consensus in open systems is studied, and where information is given a life span and must be eliminated once it is exceeded.

By contrast, in external problems the objective does not depend on the agents in the system. Instead, the states of the agents are related to the objective, so that any information corresponding to any agent having been in the system at any time is potentially relevant, and must be remembered. The challenge is then for algorithms to gather all the information without losing any, and for analyses to somehow quantify the information having been in the system at any time. The design of algorithm solving external consensus problems of this type in open systems was explored in a master thesis that was conducted in parallel of this thesis [Col20].



**PART I**  
**Consensus: representation**  
**and limitations**



# 3

## The consensus problem

THE first part of this thesis is dedicated to the study of *consensus* problems. Through the scope of this basic problem, we highlight and analyze some general properties of OMAS. Therefore, in this chapter we properly define this problem as it is commonly formulated, and review some standard results about it in closed systems as well as their possible extension to open systems.

### 3.1 General definition

Consensus is probably one of the most basic and yet essential problems arising from multi-agent systems, as it plays a major role in many protocols designed for advanced applications. It consists in making all the agents constituting a MAS agree on a given state or quantity based on local interactions, as stated in the following definition.

**Definition 3.1.** Consider a multi-agent system whose set of agents is  $\mathcal{V}$ , and where each agent  $i$  holds a time-varying quantity  $x_i(t) \in \mathbb{R}^d$ . We say that consensus is reached on some constant state  $\mathbf{x}^* \in \mathbb{R}^d$  if for all  $i \in \mathcal{V}$

$$\lim_{t \rightarrow \infty} \|x_i(t) - \mathbf{x}^*\| = 0. \quad (3.1)$$

The value  $x^*$  to which the state of the agents converge is called the *agreement state*. It sometimes consists in a specific function of the initial states of the agents such as the average, maximal, minimal or median value (typically if those values are real numbers). Such problems play a crucial role in applications such as decentralized optimization (defined in Chapter 6) [NOR18] where consensus and optimization steps typically alternate, or in vehicle coordination [OSFM07, MR19, RBA05] where *e.g.*, consensus must be reached on the distance separating the vehicles to achieve some formation. It also appears in sociology, such as in [DeG74, TIY<sup>+</sup>13]. Consensus was more generally used as a case study in a wide range of analyses, including [HOT11, MLB20].

**Remark 3.1.** *A more general formulation for Definition 3.1 is to require the state of the agents to satisfy for all  $i, j \in \mathcal{V}$*

$$\lim_{t \rightarrow \infty} \|x_i(t) - x_j(t)\| = 0, \quad (3.2)$$

*or equivalently  $\lim_{t \rightarrow \infty} \max_{(i,j) \in \mathcal{V}^2} \|x_i(t) - x_j(t)\| = 0$ . This formulation actually allows agents to agree on some time-varying trajectory instead of simply a fixed point. This generalization, called synchronization, will however not be covered in this thesis, and we will restrict to the particular case of consensus.*

### 3.2 Consensus in closed systems

From now on, and in the scope of this thesis, we will restrict to consensus problems where the states of the agents are real numbers. Moreover, we consider a discrete evolution of the time, and use  $x(k) \in \mathbb{R}^n$  to denote the vector containing the states of all  $n$  agents in the system at the time step  $k$ .

Most consensus algorithms can be written under the compact form

$$x(k+1) = P_k x(k), \quad (3.3)$$

where the update matrix  $P_k$  is a row-stochastic matrix (namely, such that each row sums up to one:  $P_k \mathbf{1}_n = \mathbf{1}_n$ ). The matrix  $P_k$  encapsulates the properties of the interactions happening in the system, and is thus strongly related to the underlying graph  $\mathcal{G}$  of the system. Moreover, depending on the consensus protocol, it potentially depends on the time step  $k$  as well. Observe we omit here a possible dependency of  $P_k$  on the system state  $x$ .

### 3.2.1 Synchronous and asynchronous algorithms

We distinguish two types of algorithms: *synchronous* algorithms where all agents communicate and update their state simultaneously, and *asynchronous* algorithms where only a portion of the agents perform the communication and update.

Synchronous algorithms are however not suited to study OMAS in general, as the arrivals and departures of agents make the definition of update rules relying on the whole system difficult to design and exploit (even though some studies relying on synchronous update rules can be found in specific settings [HR20]). More generally, synchronous algorithms may be very expensive or even impossible to use in practice with large systems. Therefore, we will consider asynchronous algorithms in this thesis. In particular, we will restrict to a specific type of asynchronous updates, called *symmetric pairwise updates*, where only pairs of agents exchanging information in both directions are involved.

Nevertheless, because they represent a significant amount of results in the field of consensus, we give a few insights on synchronous consensus algorithms below (and we refer *e.g.*, to [Bul22] for additional details).

**Synchronous consensus algorithms** When interactions are synchronous, a standard choice for an agent  $i \in \mathcal{V}$  is to aggregate the information it received from its neighbours  $\mathcal{N}_i$ . One of the most popular protocols of this type is provided in [OSM04] and reads

$$x_i^+ = x_i + \epsilon \sum_{j \in \mathcal{N}_i} (x_j - x_i), \quad (3.4)$$

for some parameter  $\epsilon \in \mathbb{R}$ . The update rule (3.4) can then be extended to the whole system using the Laplacian matrix  $L$  of the underlying graph  $\mathcal{G}$  of the system, so that it can be written under the form (3.3) as

$$\mathbf{x}^+ = (I_n - \epsilon L)\mathbf{x}. \quad (3.5)$$

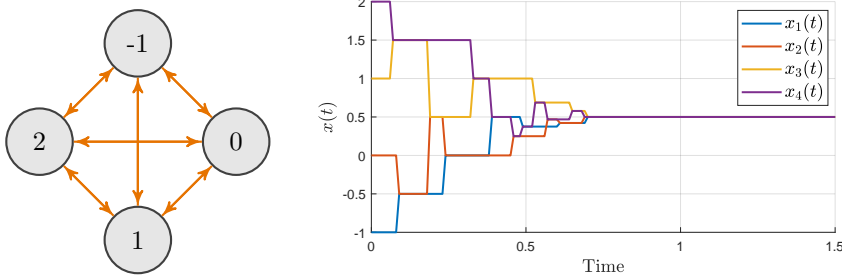
The update rule (3.5) thus relies on the properties of the network, and convergence conditions can then be derived for achieving consensus in that setting [Bul22, Theorem 5.1].

Many variations of the protocol given in (3.4) and (3.5) have been considered for several applications, including vehicle coordination [FM04b, JLM03], flocking behaviors [VCBJ<sup>+</sup>95, VZ12] or decentralized optimization [TBA86, NOP10].



When the composition of the system does not change with time (in closed system), it is known that pairwise gossip interactions achieve consensus as long as the graph defining the system is connected, *i.e.*, that there is no pair of agents isolated from each other in the system [BGPS06, FZ07].

We illustrate the behavior of the pairwise Gossip algorithm in Fig. 3.1 for a system of 4 agents subject to all-to-all pairwise gossip interactions happening at random times. The figure shows that in this particular connected closed system, convergence to the average of the values initially held by the agents is achieved in finite time.



**Fig. 3.1** Realization of the Gossip algorithm in a fully connected network constituted of four agents. The corresponding graph is represented on the left, and the evolution of the values held by the agents on the right. Consensus on the average of the values initially held by the agents is achieved in finite time.

Gossip pairwise interactions will be used in the two following chapters as the main case study to illustrate our results. The analyses we propose in this part nonetheless intend to be general, and their application to different algorithms can be considered in the future, such as the Push-sum algorithm which we present below.

**Push-sum algorithm** The *push-sum algorithm*, introduced in [KDG03], is more elaborate than the Gossip algorithm: Each agent holds two variables,  $x_i(t)$  and  $w_i(t)$ , such that  $x_i(0)$  is its initial value and  $w_i(0) = 1$ . Whenever agent  $i$  sends information to agent  $j$ , there holds

$$x_i^+ = \frac{x_i}{2}; \quad x_j^+ = \frac{x_i + x_j}{2}; \quad (3.9)$$

$$w_i^+ = \frac{w_i}{2}; \quad w_j^+ = \frac{w_i + w_j}{2}. \quad (3.10)$$

### 3 | The consensus problem

Agent  $i$  then builds its estimate of the average at time  $t$  as  $y_i(t) = \frac{x_i(t)}{w_i(t)}$ , which thus amounts to a linear combination of the initial values of the agents weighted by its importance at the system level. The push-sum improves on the Gossip algorithm, notably as it allows asymmetric interactions (which make gossip interactions fail). It was shown in [KDG03] that exponential convergence to the average is guaranteed with probability 1 in closed systems. Moreover, [GH18] showed that convergence is still guaranteed under message losses, yet not to the exact average.

The push-sum algorithm will not be directly analyzed in this thesis. Nevertheless, as it improves on the Gossip algorithm, it represents an interesting extension for the analyses that will be conducted on it, and will be discussed as perspectives for future work.

#### 3.2.3 The max consensus problem

Another well-known consensus problem is the so-called *max consensus*, where agents aim at agreeing on the maximal value in the system.

**Definition 3.3** (Max consensus). *Consider a multi-agent system whose set of agents is  $\mathcal{V}$ , where each agent  $i$  holds a time-varying quantity  $x_i(t)$ , and let  $x_i(0)$  denote the value held by agent  $i$  at the initialization of the process. We say that max consensus is reached if for all  $i \in \mathcal{V}$  we have*

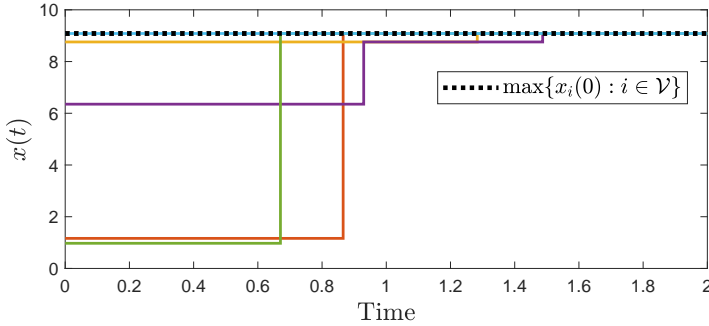
$$\lim_{t \rightarrow \infty} x_i(t) = \max \{x_i(0) \mid i \in \mathcal{V}\}. \quad (3.11)$$

Problems of this type appear *e.g.*, in the context of leader election or self-organization, typically where the slowest entity of a group defines its speed, or in the context of maximum likelihood estimation, for instance for estimating the maximal cardinality of a network [LVDH15]. They can also typically appear as an intermediate step for more advanced tasks, see *e.g.*, [DFG20], where max consensus is used to estimate the system size. A simple algorithm for solving it is the *Flooding* algorithm.

**Flooding** As its name indicates, the *Flooding* algorithm simply consists in flooding the system with the maximal initial value. This can be achieved with pairwise interactions by applying the following update rule when two agents  $i, j \in \mathcal{V}$  interact:

$$x_i^+ = x_j^+ = \max \{x_i, x_j\}. \quad (3.12)$$

One can show that as long as the system is connected, the Flooding algorithm makes the agents eventually converge to the maximal value in the system (see Fig. 3.2 for an illustration).



**Fig. 3.2** Realization of the Flooding algorithm to solve the max consensus problem with 5 agents: Each plain line is the estimate  $x_i(t)$  of an agent.

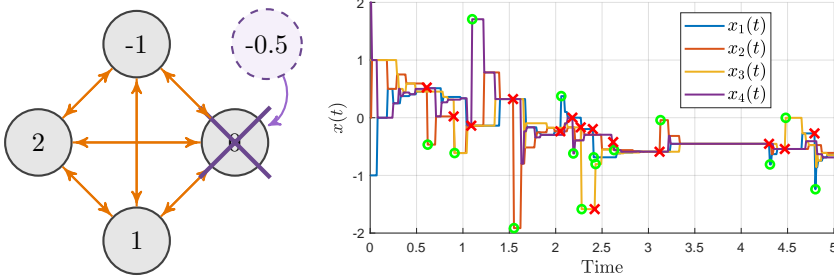
Whereas the max consensus problem and the Flooding algorithm will not be directly analyzed in the context of open systems, they allow highlighting typical challenges arising in open systems (see *e.g.*, Section 2.5).

### 3.3 Consensus in open systems

Systems subject to varying topologies have already been studied in the context of consensus, *e.g.*, in [OSM04, NAR10] for average and max consensus respectively, or more recently in [WBH<sup>+</sup>21] for opinion dynamics. However, in these studies the topology of the graph defining the interactions is assumed to be time-varying, but the set of nodes remains the same during the whole process, so that tools from closed system can be applied. By contrast, actually allowing agents to join and leave the system has a significant impact on the way consensus problems are solved, analyzed, and even formulated.

In open settings, both the system size and state are regularly subject to instantaneous variations. As a consequence, tracking the system state becomes challenging, and so does the analysis of consensus algorithms. Moreover, in such permanently disturbed setting, the notions of stability and even of consensus as in Definition 3.1 do not make sense anymore, as

each arrival or departure potentially drives the system away from agreement, even if it was almost achieved. This last observation is illustrated in Fig. 3.3, which depicts the same setting as in Fig. 3.1 (*i.e.*, in closed system), but where on average one replacement happens every ten gossip interactions. The figure shows that even if sometimes intermediate consensus is reached between replacements, they prevent it from being maintained, and potentially drive the system state far away when they happen.



**Fig. 3.3** Realization of the Gossip algorithm in a fully connected network constituted of four agents subject to replacements of agents (on average one every ten gossip interactions). The corresponding graph is represented on the left, and the evolution of the values held by the agents on the right (where red crosses and green circles highlight replacements, by denoting the value of the replaced agent respectively before and after its replacement). Consensus cannot be achieved anymore.

Consensus in OMAS started receiving attention in the recent years. We list below several works considering consensus in OMAS and relying different approaches and models that were conducted in parallel of this thesis.

**Alternative criterion for consensus with the finite superset model:** Since the usual notion of consensus cannot be applied in open systems, the authors in [VMN18] propose to study an alternative criterion instead, called *Majority action preservation*. Their analysis stands for systems where the states of the agents are binary under the finite superset model described in Section 2.2.1. In that setting, they provide conditions under which a large fraction of the agents that are active in the system maintain a common state almost surely, which can be seen as a weaker formulation of consensus for such binary states. Relying on the same model to define openness, the authors in [VFP21] study the sensitivity of consensus algorithms to noise by characterizing the deviation of the states of the agents it induces with respect to the consensus point. The methodologies used in these analyses

might however be difficult to adapt to our settings as they strongly depend on the model they rely on to define the openness, which differs from ours.

**Dynamic consensus:** Consensus has also been considered under the control theory perspective, *e.g.*, with *dynamical average consensus* [KVC<sup>+</sup>19], where consensus is achieved over a set of (potentially time-varying) reference signals spread over the agents. Such formulation was considered in the context of OMAS in [FF18, FF21] where the notion of stability was extended to that of *open stability*: This requires the trajectory of the system state to remain close to a *trajectory of interest*, which generalizes the notion of equilibrium. The authors then apply this criterion to analyze the stability of algorithms for the Proportional Dynamic Consensus in open system. Dynamic consensus was also applied in a similar way for tracking the median of the values held by the agents in open systems in [SSF19], and for the max/min consensus problems as well as size estimation in [DFG20], which was recently summarized in a thesis [Dep21].

**Appropriate representation:** In the preliminary studies [HM16, HM17], the authors propose an appropriate representation of open multi-agent systems which allows performing analyses while handling their variable size through the use of size-independent quantities. Using a finite size dynamical system to characterize their evolution, they analyze the behavior of gossip interactions and the evolution of the agreement between the agents for specific open systems. In Chapter 4, we formalize and generalize these ideas using tools taken from Markov theory, and we extend their application to general systems of varying size. Those results are presented in [MMH20]. A complementary analysis which also extends ideas from [HM16, HM17] to incomplete graph topologies can be found in [Oli18].

**Algorithms:** Algorithms solving the max consensus problem in open systems have been studied *e.g.*, in [AHH17], where the use of additional variables provides guarantees on the probability for the system to converge to the correct value if arrivals and departures were to stop. More generally, variations of algorithms, and in particular gossip interactions, were considered in a master thesis [Bra18] under the perspective of asymptotic performance if the arrivals and departures stop. It typically allowed to show that (naive) correcting mechanisms guaranteeing asymptotic correctness if the system were to close tend to make transient performance even worse than if no mechanism were used at all (see *e.g.*, [Bra18, Chap. 3] for details).

### 3 | The consensus problem

**Performance measure:** In the absence of usual convergence of consensus algorithms, measuring performance is a challenge as well. This question was investigated in my master thesis [Mon18], and then formalized in [MH19, MH20], which constitute the content of Chapter 5. The objective is then to derive lower bounds on the expected performance of average consensus algorithms, which thus constitute an alternative quality criterion for them.

**External consensus:** As discussed in Section 2.5, the challenges to be handled by algorithms solving a consensus problem in open systems strongly depend on its formulation. Typically, average consensus in open system is usually defined based solely on the agents *present in the system at that time*, thus requiring the elimination of the information about agent that left. By contrast, consensus on a quantity defined *outside* of the system requires aggregating as much information as possible, and hence remembering information about agents having left the system. The development of algorithms solving this version of consensus in open system was investigated in a recent master thesis [Col20].

# 4

## Representing open systems: the case of gossiping

*The results of this chapter were submitted to IEEE Transactions on Automatic Control (TAC) in [MMH20]; They build on some preliminary research initiated by J.M. Hendrickx and S. Martin in [HM16, HM17], whose results are formalized in a new framework and extended to different settings.*

IN this chapter, we focus on the question of the *representation* of OMAS. By nature, the size of such systems potentially varies with time, and the state of the agents they are constituted of may suffer abrupt instantaneous changes. These features make OMAS challenging to analyze and algorithms difficult to develop. A methodology suited for handling these challenges proposed in [HM16, HM17] consists in analyzing specific quantities, called *descriptors*, which do not depend on the size of the system but still characterize its state. In this chapter, we formalize and generalize this approach by putting it in the framework of Markov processes, and apply it to extend the results of [HM16, HM17] to a wider range of settings.

For that purpose, we consider the specific problem of consensus. In particular we analyze the behavior of all-to-all pairwise gossip interactions in a system subject to arrivals and departures of agents. We show in that situation that even though arrivals and departures induce non-negligible variations in the size and composition of the system, and hence prevent

convergence, the two first moments of the system (*i.e.*, the descriptors) evolve following a fixed-size affine dynamical system. We can thus derive convergence results for these specific quantities. Moreover, they allow monitoring the variance of the system, which characterizes the disagreement between the agents.

We formulate the problem in Section 4.1, where we properly describe our system and introduce the *descriptors* we will use to study its evolution. Our approach is based on the analysis of the instantaneous effect on the descriptors of arrivals, departures, replacements and interactions, which are the asynchronous events driving the evolution of the system (in the sense of Section 2.4). We show in Section 4.2 that our system is actually a particular case of a more general formulation, so that our approach can be generalized to any set of events and descriptors, resulting in a general expression for their evolution in expectation as an ordinary differential equation. Our approach can thus be generalized to other problems by defining the corresponding events and their effect accordingly. Finally, we apply our general result from Section 4.2 to analyze the specific system of Section 4.1 in Section 4.3, where we consider systems of fixed-size under two different types of replacements and systems of variable size.

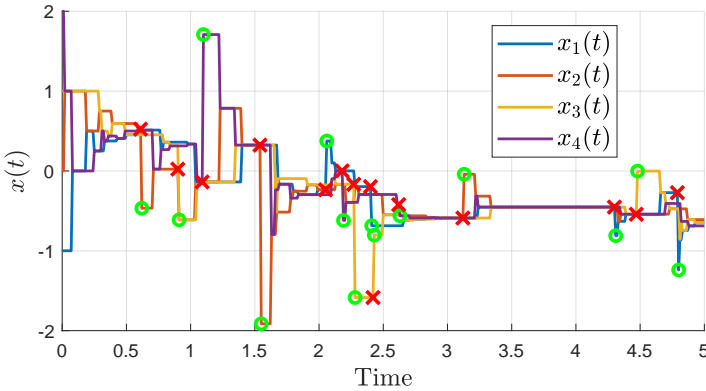
## 4.1 Problem formulation: a descriptor-based approach

In this chapter, we consider an all-to-all version of the Gossip algorithm (3.7) defined in Section 3.2 [BGPS06], where the states  $x_i$  of the agents are real values. This means that the interaction graph is complete, so that any pair of agents  $i, j \in \mathcal{V}$  can potentially interact, resulting in

$$x_i^+ = x_j^+ = \frac{x_i + x_j}{2}. \quad (4.1)$$

In closed system, it is known that (4.1) solves the consensus problem as defined in (3.1) for that setting, so that  $\lim_{t \rightarrow \infty} \|x_i(t) - x^*\| = 0$  for some  $x^*$ . This is however not the case in open systems, as discussed in Section 3.3 and illustrated in Fig. 4.1, where it appears that consensus cannot be maintained because of the everlasting replacements.

More generally, consensus as defined in (3.1) simply cannot be achieved in open system, as the permanent changes in the system composition keep perturbing the system. Instead, the general behavior of the system is ex-



**Fig. 4.1** Realization of the Gossip algorithm in a fully connected system of four agents subject to replacements (one every ten gossip steps on average). Each line is the value held by an agent, while the red crosses and green circles highlight replacements, denoting the value of a replaced agent respectively before and after being replaced.

pected to reach some form of steady-state, which can be described and analyzed with relevant quantities. Of course, the choice of these quantities is not neutral, and different quantities might behave in very different ways. This section focuses on the selection of relevant quantities, which we call *descriptors* of the system, in order to analyze the behavior in expectation of all-to-all pairwise gossip interactions (4.1) in open systems.

#### 4.1.1 System description

We consider a multi-agent system whose composition changes with time. We use integers to label the agents, and denote the set of agents in the system at time  $t$  by  $\mathcal{V}(t) \subset \mathbb{N}$ . As a consequence, the dimension of the system, which is denoted by  $n(t) = |\mathcal{V}(t)|$ , changes with time as well. Each agent  $i$  holds a value  $x_i(t) \in \mathbb{R}$ ; We assume that the value initially held by the agents at the beginning of the process (at time  $t = 0$ ) is a bounded value randomly and independently drawn from some distribution  $\mathcal{D}$  with mean 0 and variance  $\sigma^2$ . Our results can immediately be adapted to distributions  $\mathcal{D}$  with arbitrary constant mean  $\mu$ . Finally, we define the state of the system at time  $t$  as the vector  $\mathbf{x}(t) \in \mathbb{R}^{n(t)}$  containing the values  $x_i(t)$  of all the agents  $i \in \mathcal{V}(t)$  present in the system at that time.

The evolution of the system is event-based in the sense of Section 2.4,

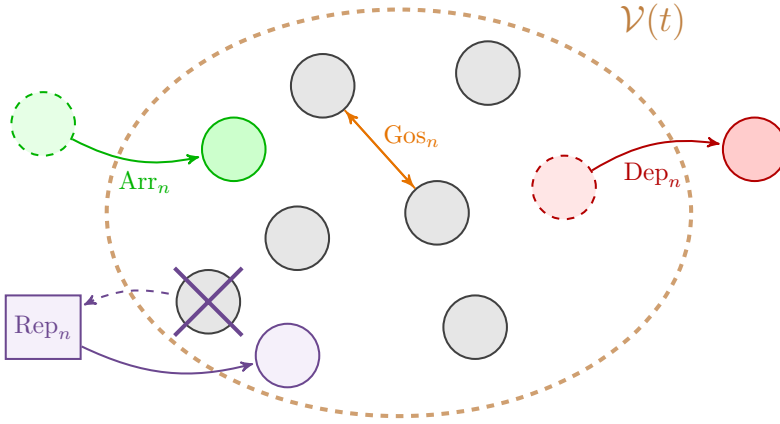
meaning it is characterized by instantaneous modifications triggered by *events*, which correspond to the activation of several clocks. We consider four types of events that modify both the system size  $n(t)$  and state  $x(t)$ : *gossip interactions*, *arrivals*, *departures* and *replacements*. Each of these events is associated with a Poisson clock with respective size-dependent rates  $\lambda_{\text{Gos}_n}$ ,  $\lambda_{\text{Arr}_n}$ ,  $\lambda_{\text{Dep}_n}$  and  $\lambda_{\text{Rep}_n}$ . We detail below the effect of each type of event, using the notations of Section 2.4. Moreover, we represent the system described above and the different types of events in Fig. 4.2.

1. A *gossip interaction* among  $n$  agents, denoted  $\text{Gos}_n$ , consists in selecting uniformly randomly and independently  $i, j \in \mathcal{V}$  (with possibly  $i = j$ ) to update their values  $x_i$  and  $x_j$  by performing a pairwise average  $x_i^+ = x_j^+ = \frac{x_i + x_j}{2}$ . Hence we have  $\mathcal{V}^+ = \mathcal{V}$  and  $n^+ = n$ .
2. An *arrival* among  $n$  agents, denoted  $\text{Arr}_n$ , consists in one “new” agent  $i \notin \mathcal{V}(s)$ ,  $\forall s \leq t$  joining the system, so that  $\mathcal{V}^+ = \mathcal{V} \cup \{i\}$ , and thus  $n^+ = n + 1$ . The initial value of the arriving agent is then independently and randomly drawn from the distribution  $\mathcal{D}$  used to initialize the system.
3. A *departure* among  $n > 0$  agents, denoted  $\text{Dep}_n$ , consists in selecting an agent  $i \in \mathcal{V}$  to leave the system, so that  $\mathcal{V}^+ = \mathcal{V} \setminus \{i\}$ , and thus  $n^+ = n - 1$ .
4. A *replacement* among  $n > 0$  agents, denoted  $\text{Rep}_n$ , consists in the simultaneous occurrence of both a departure and an arrival. There thus holds  $n^+ = n$  (but  $\mathcal{V}^+ \neq \mathcal{V}$ ).

Observe that we did not specify how the agent leaving the system is selected at departures. In this chapter, we will consider two possible ways of making that choice:

1. *Random departures*, denoted  $\text{Dep}_n$ , where the leaving agent is uniformly randomly selected among those in the system;
2. *Adversarial departures*, denoted  $\text{Dep}_n^*$ , where no assumption can be made on that choice (e.g., as if some opponent was doing that choice based on its own criterion).

These two types of choice also apply for replacements, and we will refer later to *random replacements* and *adversarial replacements* accordingly (respectively  $\text{Rep}_n$  and  $\text{Rep}_n^*$ ). Other types of departures and replacements



**Fig. 4.2** Representation of the OMAS of this chapter subject to four types of events for  $|\mathcal{V}(t)| = n$ : pairwise gossips  $\text{Gos}_n$  (in orange), arrivals  $\text{Arr}_n$  (in green), departures  $\text{Dep}_n$  (in red) and replacements  $\text{Rep}_n$  (in purple).

could of course be considered the same way. Moreover, as it will be discussed later, different clocks, *i.e.*, other types of events, could be considered to characterize the evolution of the system, *e.g.*, if other algorithms relying on different types of interactions were to be analyzed.

#### 4.1.2 Choice of the descriptors: scale-independent quantities

In the setting of the previous section the system size changes with time, and as a consequence tracking the system state  $x(t)$  is a challenging task. Instead, in this chapter, we show that we can study the evolution of some specific size-independent quantities, *i.e.*, that do not scale with the system size, with a finite-dimensional affine dynamical system. A possible choice for such quantity is the *mean* of the system state defined as

$$\bar{x}(t) := \frac{1}{n(t)} \sum_{i \in \mathcal{V}(t)} x_i(t). \quad (4.2)$$

This quantity was considered *e.g.*, in [HM16]. Its analysis is however omitted here, since as we will see it is not crucial to study the behavior of the system under the scope of the consensus problem. Rather, we will consider the empirical squared mean  $\bar{x}^2(t)$  and mean of squares  $\overline{x^2}(t)$  as *descriptors* of the system. These correspond to the two first moments of the

system state respectively defined as

$$\bar{x}^2(t) := \left( \frac{1}{n(t)} \sum_{i \in \mathcal{V}(t)} x_i(t) \right)^2 ; \quad \overline{x^2}(t) := \frac{1}{n(t)} \sum_{i \in \mathcal{V}(t)} x_i^2(t). \quad (4.3)$$

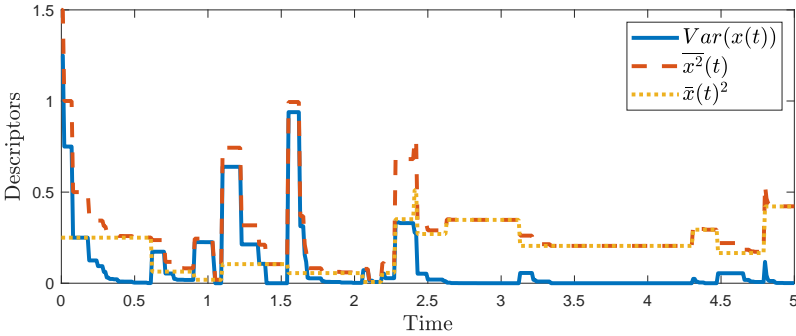
Using these specific quantities, we can also track the variance of the system  $\text{Var}(x(t))$ , defined as

$$\text{Var}(x(t)) := \frac{1}{n(t)} \sum_{i \in \mathcal{V}(t)} (x_i(t) - \bar{x}(t))^2 = \overline{x^2}(t) - \bar{x}^2(t). \quad (4.4)$$

The variance measures the average distance between the states of the agents and the average of their values  $\bar{x}(t)$ . Since the Gossip algorithm is sum-preserving (that is, it maintains  $\sum_{i \in \mathcal{V}} x_i(t)$  constant through interactions), consensus is expected to be achieved on  $\bar{x}(t)$ , which is thus its agreement state [BGPS06, FZ07]. In that sense, the variance measures the level of disagreement between the agents, *i.e.*, “how far” the system is from consensus. It is thus a valid criterion to study consensus, and the criterion (3.1) can be equivalently defined as

$$\lim_{t \rightarrow \infty} \text{Var}(x(t)) = 0. \quad (4.5)$$

We illustrate in Fig. 4.3 the evolution of these quantities (namely  $\bar{x}^2(t)$ ,  $\overline{x^2}(t)$ , and  $\text{Var}(x(t))$ ) for the specific realization of Fig. 4.1.



**Fig. 4.3** Evolution of the mean of squares  $\overline{x^2}(t)$ , the squared mean  $\bar{x}^2(t)$ , and the variance  $\text{Var}(x(t)) = \overline{x^2}(t) - \bar{x}^2(t)^2$  corresponding to the realization of the system presented in Fig. 4.1.

As discussed previously, it is known that in closed system gossip interactions allow achieving consensus [FZ07], and hence guarantee (4.5). This is no more the case in open system as the value held by any joining agent will be different from the value of the agents already present in the system with high probability (see *e.g.*, Fig. 4.1). Hence, by studying the evolution of the descriptors, we can analyze the variance as well, and therefore the level of disagreement between the agents, as shown in Fig. 4.3. In particular, we will be interested in the evolution of  $\mathbb{E}\text{Var}(\mathbf{x}(t))$ , which will thus require monitoring  $\mathbb{E}\bar{x}^2(t)$  and  $\mathbb{E}\bar{x}^2(t)$ .

### 4.1.3 Evolution of the descriptors in expectation

We now show that the evolution in expectation of the descriptors with respect to any of the events we consider is governed by a 2-dimensional affine dynamical system. Let us define the *descriptor vector* as

$$\mathbf{X}(t) := \begin{pmatrix} \bar{x}^2(t) & \overline{x^2}(t) \end{pmatrix}^\top, \quad (4.6)$$

so that there holds

$$\text{Var}(\mathbf{x}(t)) = \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{X}(t). \quad (4.7)$$

In the following lemmas, whose proofs are in Appendix 4.A, we give the evolution of  $\mathbb{E}\mathbf{X}$  with respect to the different types of events using the notations of Section 2.4, from which we derive that of  $\mathbb{E}\text{Var}(\mathbf{x})$ .

**Lemma 4.1** (Gossip step). *At an event of  $\text{Gos}_n$  there holds*

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Gos}_n] = \underbrace{\begin{pmatrix} 1 & 0 \\ \frac{1}{n} & 1 - \frac{1}{n} \end{pmatrix}}_{=A_{\text{Gos}_n}} \mathbf{X}. \quad (4.8)$$

Since  $\text{Var}(\mathbf{x}(t)) = \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{X}(t)$ , one then has

$$\mathbb{E} [\text{Var}(\mathbf{x}^+) | \text{Var}(\mathbf{x}), \text{Gos}_n] = \left(1 - \frac{1}{n}\right) \text{Var}(\mathbf{x}). \quad (4.9)$$

**Lemma 4.2** (Arrival). *At an event of  $\text{Arr}_n$  there holds*

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Arr}_n] = \frac{n}{n+1} \underbrace{\begin{pmatrix} \frac{n}{n+1} & 0 \\ 0 & 1 \end{pmatrix}}_{=A_{\text{Arr}_n}} \mathbf{X} + \underbrace{\begin{pmatrix} \frac{1}{(n+1)^2} \\ \frac{1}{n+1} \end{pmatrix}}_{=b_{\text{Arr}_n}} \sigma^2. \quad (4.10)$$

Remember that we consider two different types of departures: *random* and *adversarial*, both introduced in Section 4.1.1. They differ in how the leaving agent is chosen at departures, and consequently their impact on the evolution of the descriptors in expectation differs as well. Hence, we study their respective evolution separately. This of course also holds true for replacements, which we remind consist in the occurrence of a departure immediately followed by an arrival, the latter applied to a system of size  $n - 1$  joined by a  $n^{\text{th}}$  agent.

*Random departures and replacements*

As a quick reminder, *random* departures consist in the uniform random choice of the leaving agents among those in the system, resulting in the following lemmas.

**Lemma 4.3** (Random departure). *At an event of  $\text{Dep}_n$  there holds*

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Dep}_n] = \underbrace{\begin{pmatrix} 1 - \frac{1}{(n-1)^2} & \frac{1}{(n-1)^2} \\ 0 & 1 \end{pmatrix}}_{=A_{\text{Dep}_n}} \mathbf{X}. \quad (4.11)$$

Since  $\text{Var}(\mathbf{x}(t)) = \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{X}(t)$ , one then has

$$\mathbb{E} [\text{Var}(\mathbf{x}^+) | \text{Var}(\mathbf{x}), \text{Dep}_n] = \left( 1 - \frac{1}{(n-1)^2} \right) \text{Var}(\mathbf{x}). \quad (4.12)$$

**Lemma 4.4** (Random replacement). *At an event of  $\text{Rep}_n$  there holds*

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Rep}_n] = \underbrace{\begin{pmatrix} \frac{n-2}{n} & \frac{1}{n^2} \\ 0 & \frac{n-1}{n} \end{pmatrix}}_{=A_{\text{Rep}_n}} \mathbf{X} + \underbrace{\begin{pmatrix} \frac{1}{n^2} \\ \frac{1}{n} \end{pmatrix}}_{=b_{\text{Rep}_n}} \sigma^2. \quad (4.13)$$

### *Adversarial departures and replacements*

As a quick reminder, adversarial departures amount to letting the choice of the leaving agent to an opponent, so that no assumption can be made on how that choice is done. Hence, obtaining the exact evolution of the descriptors in expectation is not achievable, and instead we obtain upper bounds on that of the expected variance. In this case the expectation on the choice of the leaving agent thus corresponds to some sort of “worst case” on that choice. This results in the following lemmas.

**Lemma 4.5** (Adversarial departure). *At an event of  $\text{Dep}_n^*$  there holds*

$$\mathbb{E} [\text{Var}(\mathbf{x}^+) | \text{Var}(\mathbf{x}), \text{Dep}_n^*] \leq \frac{n}{n-1} \text{Var}(\mathbf{x}). \quad (4.14)$$

**Lemma 4.6** (Adversarial replacement). *At an event of  $\text{Rep}_n^*$ , and under the conditions of Lemma 4.14, there holds*

$$\mathbb{E} [\text{Var}(\mathbf{x}^+) | \text{Var}(\mathbf{x}), \text{Rep}_n^*] \leq \text{Var}(\mathbf{x}) + \frac{\sigma^2}{n}. \quad (4.15)$$

**Remark 4.1.** *Lemmas 4.1 to 4.6 already give insights on the behavior of the system with respect to each type of event. Typically, one can observe that the eigenvalues of the matrix  $A_\epsilon$  describing the evolution of  $\mathbf{X}$  with respect to any event of  $\epsilon$  are all upper bounded by 1, so that the system defining the evolution of the descriptors is marginally stable (a similar conclusion can be obtained for the variance). In the particular case of arrivals (and thus replacements), which involve an additive term, it can be noted that the additive term scales with  $\frac{1}{n}$  or  $\frac{1}{n^2}$ , so that its effect becomes negligible as the system size increases.*

## 4.2 Generalization of the descriptor approach

The system we described in the previous section is actually a particular case of a more general formulation, which we describe and analyze in this section. We show that with that formulation the descriptors can be modelled as a function of the state of a specific Markov jump process, and derive a general expression for their evolution in expectation as a flow equation, which is related to Kolmogorov equations. This generalization is one important contribution of this chapter with respect to [HM16, HM17].

## 4.2.1 Generalization of the system description

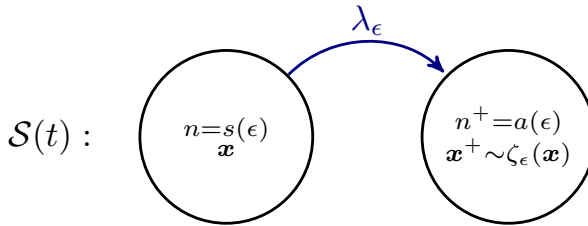
The system described in Section 4.1.1 is a particular case of the following general formulation. Let us define the *state* of the system (in the sense of stochastic processes) as

$$\mathcal{S}(t) = (n(t), \mathbf{x}(t)), \quad (4.16)$$

which is constituted of the system size  $n(t) \in \mathbb{N}$  and of the vector containing the values held by the agents  $\mathbf{x}(t) \in \mathbb{R}^{n(t)}$  at that time. The evolution of the system is event-based in the sense of Section 2.4, so that  $\mathcal{S}(t)$  evolves following a Markov jump process (see Definition 2.15). For each value of  $n(t)$  we associate a finite set of independent Poisson clocks which correspond to the events driving the evolution of the system. The activation of a clock  $\epsilon$ , called *event of  $\epsilon$* , happens with fixed rate  $\lambda_\epsilon$  and therefore triggers the jump of  $\mathcal{S}(t)$  that modifies both  $n(t)$  and  $\mathbf{x}(t)$  as follows.

- The modification of  $n(t)$  is deterministic, so that for an event  $\epsilon$  we can define  $s(\epsilon) \in \mathbb{N}$  and  $a(\epsilon) \in \mathbb{N}$  in such way that it changes  $n(t)$  from  $s(\epsilon)$  to  $a(\epsilon)$ .
- The modification of  $\mathbf{x}(t)$  is stochastic, so that an event of  $\epsilon$  modifies it according to some distribution which depends on both  $\mathbf{x}(t)$  and on the nature of  $\epsilon$  (the latter thus implies that  $n(t) = s(\epsilon)$ ).

The transition of the system upon the activation of a clock  $\epsilon$  is illustrated in Fig. 4.4. We use  $\Xi$  to denote the set of all the clocks of the system (*i.e.*, for all values of  $n$ ), which we call *event set*.

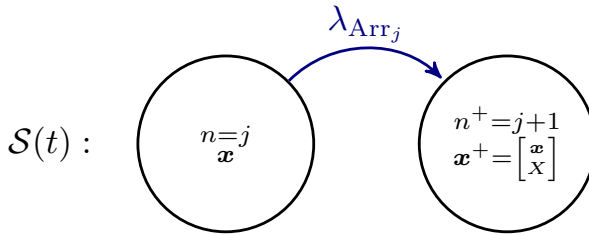


**Fig. 4.4** Illustration of the transition of the system state  $\mathcal{S}(t) = (n, \mathbf{x})$  through the activation of the clock  $\epsilon$ . The clock  $\epsilon$  activates with  $\lambda_\epsilon$ , triggering the modification of  $\mathcal{S}(t)$ . It modifies  $n = s(\epsilon)$  in a deterministic way into  $n^+ = a(\epsilon)$ , and randomly modifies  $\mathbf{x}$  according to a given distribution written  $\zeta_\epsilon$ , which depends on both the nature of  $\epsilon$  and on  $\mathbf{x}$ . By definition, this transition does not happen if  $n \neq s(\epsilon)$ .

**Example 4.7.** Let us now apply this formulation for the specific system described in Section 4.1.1. For the given system size  $n$ , the clocks driving the evolution of the systems correspond to the gossip interactions, arrivals, departures and replacements (we omit adversarial departures and replacements for simplicity, but they could be included exactly the same way). Hence, in that case, we have

$$\Xi = \bigcup_{n \in \mathbb{N}} \{\text{Gos}_n, \text{Arr}_n, \text{Dep}_n, \text{Rep}_n\}. \quad (4.17)$$

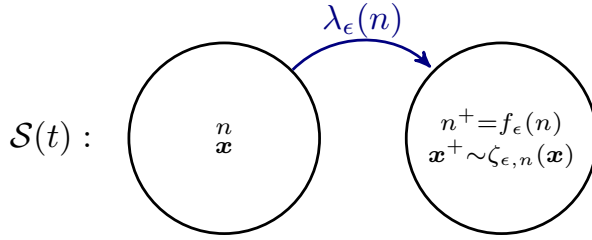
For instance, an event of  $\text{Arr}_j$  happens with rate  $\lambda_{\text{Arr}_j}$  when the system size is  $n(t) = s(\text{Arr}_j) = j$ , triggering a jump in the process, that modifies the system size into  $a(\text{Arr}_j) = j + 1$  and the system state  $\mathbf{x}(t)$  by adding a new value  $X$  randomly drawn from the distribution  $\mathcal{D}$ , as illustrated in Fig. 4.5.



**Fig. 4.5** Effect of an event of  $\text{Arr}_j$ , i.e., of an arrival among  $j$  agents, on the state  $S(t)$  in the system described in Section 4.1.1. The clock activates with rate  $\lambda_{\text{Arr}_j}$  if  $n(t) = s(\text{Arr}_j) = j$ : It modifies  $n$  into  $a(\text{Arr}_j) = j + 1$  and  $\mathbf{x}$  by adding to it a new random value  $X$  drawn from  $\mathcal{D}$ , following the specifications of arrivals.

**Remark 4.2.** The formulation for the evolution of the system presented in this section relies on the definition of a set of clocks for each values of  $n(t)$ , so that, for instance, a replacement among 3 and 4 agents are characterized by two different clocks, namely  $\text{Rep}_3$  and  $\text{Rep}_4$  respectively.

An alternative formulation illustrated in Fig. 4.6 consists in defining the clocks at the scale of the whole system (e.g.,  $\text{Rep}$  for replacements at the system level). In that case, the activation of a given clock is not conditioned by the system size anymore. Instead, the rate at which a clock  $\epsilon$  ticks becomes size-dependent:  $\lambda_\epsilon(n)$ . The modification of the system size  $n(t)$  is then characterized by some deterministic size-dependent function  $f_\epsilon(n)$ , and the distribution defining the evolution of  $\mathbf{x}$  becomes size-dependent as well (i.e.,  $\zeta_{\epsilon,n}(\mathbf{x})$ ). One can show that both representations allow representing the evolution of the same system, yet the first one is preferred in this chapter, as it is easier to exploit in order to formulate our results.



**Fig. 4.6** Alternative modelling for the evolution of the state  $\mathcal{S}(t)$ : Each clock  $\epsilon$  ticks with a size-dependent rate  $\lambda_\epsilon(n)$ , resulting in the deterministic modification of  $n$  according to some function  $f_\epsilon(n)$ , and the random modification of  $\mathbf{x}$  according to some distribution  $\zeta_{\epsilon,n}(\mathbf{x})$ .

### 4.2.2 Evolution of the general descriptors as an ODE

Let  $E$  denote the state space over which the system state  $\mathcal{S}(t)$  is defined, namely, the set of objects  $(n(t), \mathbf{x}(t))$  where  $n(t) \in \mathbb{N}$  and  $\mathbf{x}(t) \in \mathbb{R}^{n(t)}$ . In the same line as the reformulation of the system, the descriptors can be more generally defined as the result of a function  $f : E \rightarrow \mathbb{R}^d$  of the system state. For instance, the descriptors  $\bar{x}^2(t)$  and  $\overline{x^2}(t)$  from Section 4.1.2 can be formulated with the function  $f : E \rightarrow \mathbb{R}^2$  defined as follows, where we omit the dependence to the time to lighten the notations:

$$f(\mathcal{S}) = \begin{bmatrix} \left( \frac{1}{n} \mathbf{1}_n^\top \mathbf{x} \right)^2 \\ \frac{1}{n} \mathbf{x}^\top \mathbf{x} \end{bmatrix} = \begin{bmatrix} \bar{x}^2 \\ \overline{x^2} \end{bmatrix} = \mathbf{X}. \quad (4.18)$$

Hence, our goal is more generally to analyze the evolution in expectation of a given bounded function  $f : E \rightarrow \mathbb{R}^d$ . For that purpose, we provide variations of standard results on Markov processes in the following propositions, which are connected with Kolmogorov equations [Ebe20, Pav09]. These results, whose proofs are given in Appendix 4.C, will serve as a basis for establishing the evolution in expectation of the descriptors of our system conditioned by its size.

**Proposition 4.8.** *Let  $(E, \mathcal{P}(E))$  be a measurable space, and  $\mathcal{S}(t) = (n(t), \mathbf{x}(t))$  a Markov jump process as defined in Section 4.2.1 with state-space  $E$ . Moreover, let  $f : E \rightarrow \mathbb{R}^d$  for some  $d \in \mathbb{N}$  be a measurable bounded function. Finally, for  $j \in \mathbb{N}$ , let  $F_j(t) = \mathbb{E}[f(\mathcal{S}(t)) \mid n(t) = j]$  and  $\pi_j(t) = P[n(t) = j]$ . If for all*

$\epsilon \in \Xi$  there exist  $A_\epsilon \in \mathbb{R}^{d \times d}$  and  $b_\epsilon \in \mathbb{R}^d$  such that

$$\mathbb{E} [f(\mathcal{S}^+) | f(\mathcal{S}), \epsilon] = A_\epsilon f(\mathcal{S}) + b_\epsilon, \quad (4.19)$$

where we use the notations of Section 2.4, then there holds

$$\frac{d}{dt}(F_j \pi_j) = \sum_{\epsilon: a(\epsilon)=j} \lambda_\epsilon \pi_{s(\epsilon)} (A_\epsilon F_{s(\epsilon)} + b_\epsilon) - \sum_{\epsilon: s(\epsilon)=j} \lambda_\epsilon \pi_j F_j, \quad (4.20)$$

where we omit the dependence to the time to lighten the notations.

The following proposition is then a variation of Proposition 4.8 from above, where the inequality (4.21) holds instead of the equality (4.19).

**Proposition 4.9.** *In the same setting as that of Proposition 4.8, if*

$$\mathbb{E} [f(\mathcal{S}^+) | f(\mathcal{S}), \epsilon] \leq A_\epsilon f(\mathcal{S}) + b_\epsilon, \quad (4.21)$$

holds instead of (4.19) for all  $\epsilon \in \Xi$ , then there holds

$$\frac{d}{dt}(F_j \pi_j) \leq \sum_{\epsilon: a(\epsilon)=j} \lambda_\epsilon \pi_{s(\epsilon)} (A_\epsilon F_{s(\epsilon)} + b_\epsilon) - \sum_{\epsilon: s(\epsilon)=j} \lambda_\epsilon \pi_j F_j, \quad (4.22)$$

where we omit the dependence to the time to lighten the notations.

### 4.2.3 Interpretation and illustrations

Proposition 4.8 describes, as a flow equation, how the evolution of a given state of  $n(t)$  contributes to that of  $\mathbb{E} [f(\mathcal{S}(t))] = \sum_{j \in \mathbb{N}} F_j(t) \pi_j(t)$ . Consider for instance the simple case where  $f(\mathcal{S}(t)) = 1$ . In that case (4.20) reduces to

$$\frac{d}{dt} \pi_j(t) = \sum_{\epsilon: a(\epsilon)=j} \lambda_\epsilon \pi_{s(\epsilon)}(t) - \sum_{\epsilon: s(\epsilon)=j} \lambda_\epsilon \pi_j(t), \quad (4.23)$$

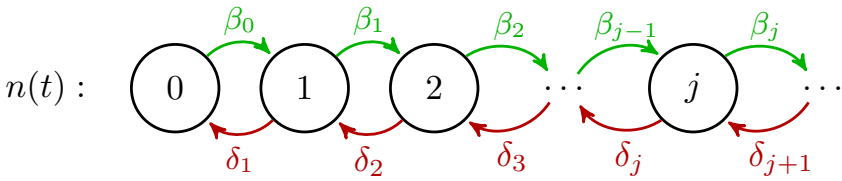
which corresponds to the simple Markov transitions between the different values of  $n(t)$ , driven by the different clocks  $\epsilon \in \Xi$ . Hence,  $f(\mathcal{S}(t))$  can be seen as some weight distribution on the different values of  $n(t)$ , and whose evolution is conditioned by that of  $n(t)$ . In the following example, we propose to emphasize this interpretation by applying the result to simple functions  $f$  in a simple setting, where the evolution of  $n(t)$  is conducted by a birth-death process.

#### 4 | Representing open systems: the case of gossiping

**Example 4.10.** Let us consider two functions  $f_1$  and  $f_2$  of the system state  $\mathcal{S}(t)$ , respectively defined as

$$f_1(\mathcal{S}(t)) := 1 \quad \text{and} \quad f_2(\mathcal{S}(t)) := n(t). \quad (4.24)$$

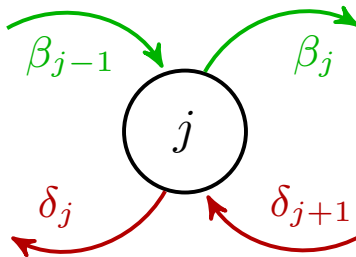
Neither  $f_1$  nor  $f_2$  depends on  $\mathbf{x}(t)$ , and hence we only need to characterize the evolution of  $n(t)$  to apply Proposition 4.8. For simplicity, we assume that  $n(t)$  evolves according to a birth-death process, as presented in Example 2.21 from Section 2.3, and represented in Fig. 4.7.



**Fig. 4.7** Usual representation of a birth-death process, where births happen with rate  $\beta_i$  and deaths with rate  $\delta_i$  for  $n(t) = i$ .

The birth-death process above is a well-known continuous-time Markov chain subject to two types of transitions: “births” which increase  $n(t)$  by 1, and “deaths” which decrease it by 1. In particular, for a given system size  $n(t) = j$ , births happen with rate  $\beta_j$  and deaths with rate  $\delta_j$ . Observe that this formulation for the evolution of  $n(t)$  can be assimilated to that of the system described in Section 4.1.1, with  $\lambda_{\text{Gos}_n} = \lambda_{\text{Rep}_n} = 0$ ,  $\lambda_{\text{Arr}_n} = \beta_n$ , and  $\lambda_{\text{Dep}_n} = \delta_n$ .

In order to apply Proposition 4.8, we focus on the transitions for a specific system size  $n(t)$ , represented in Fig. 4.8.



**Fig. 4.8** Transitions related to a specific system size  $j$  for the birth-death process.

1. First consider  $f_1(\mathcal{S}(t)) = 1$ . For all  $j \in \mathbb{N}$ , there holds

$$F_j(t) = \mathbb{E} [f_1(\mathcal{S}(t)) | n(t) = j] = 1$$

and

$$\mathbb{E} [f_1(\mathcal{S}^+) | f_1(\mathcal{S}), \beta_j] = \mathbb{E} [f_1(\mathcal{S}^+) | f_1(\mathcal{S}), \delta_j] = 1.$$

Then, focusing on the transitions corresponding to  $n(t) = j$  presented in Fig. 4.8, (4.20) becomes

$$\frac{d}{dt} \pi_j(t) = \beta_{j-1} \pi_{j-1}(t) + \delta_{j+1} \pi_{j+1}(t) - (\beta_j + \delta_j) \pi_j(t), \quad (4.25)$$

which exactly corresponds to the Markov transitions between the different values of  $n(t)$  in standard birth-death processes of (2.32), and which is consistent with (4.23) in that specific setting.

2. Second, consider  $f_2(\mathcal{S}(t)) = n(t)$ . For all  $j \in \mathbb{N}$ , there holds

$$F_j(t) = \mathbb{E} [f_2(\mathcal{S}(t)) | n(t) = j] = n(t)$$

and

$$\mathbb{E} [f_2(\mathcal{S}^+) | f_2(\mathcal{S}), \beta_j] = j + 1; \quad \mathbb{E} [f_2(\mathcal{S}^+) | f_2(\mathcal{S}), \delta_j] = j - 1.$$

Then, focusing on the transitions corresponding to  $n(t) = j$  presented in Fig. 4.8, (4.20) becomes

$$\frac{d}{dt} j \pi_j(t) = \beta_{j-1} j \pi_{j-1}(t) + \delta_{j+1} j \pi_{j+1}(t) - (\beta_j + \delta_j) j \pi_j(t). \quad (4.26)$$

The result above then yields

$$\frac{d}{dt} \mathbb{E} [f_2(\mathcal{S}(t))] = \sum_{j \in \mathbb{N}} \frac{d}{dt} j \pi_j(t) = \sum_{j \in \mathbb{N}} (\beta_j - \delta_j) \pi_j(t). \quad (4.27)$$

Hence, in general  $f(\mathcal{S}(t))$  can be seen as some weight distribution over the different values of  $n(t)$ . In particular, in (4.27), this weight distribution happens to be the system size itself, so that  $\frac{d}{dt} \mathbb{E} [f_2(\mathcal{S}(t))]$  represents the expected transition rate of the birth-death process.

### 4.3 Case studies: gossiping in open systems

We now apply the general methodology proposed in Section 4.2 to the setting described in Section 4.1, *i.e.*, subject to gossip interactions, arrivals, departures and replacements. We will restrict to two particular characterizations of that setting: (i) a fixed-size system where  $n(t) = n$ , which is thus only subject to gossip interactions and replacements, and (ii) a more general setting where arrivals and departures are independent processes. This latter characterization typically allows the analysis of specific settings, such as only growing systems. By default, we will consider *random* departures and replacements (see Section 4.1.1), so that

$$\Xi := \bigcup_{n \in \mathbb{N}} \{\text{Gos}_n, \text{Arr}_n, \text{Dep}_n, \text{Rep}_n\}. \quad (4.28)$$

However, we will also sometimes consider *adversarial* departures and replacements instead, then resulting in

$$\Xi^* := \bigcup_{n \in \mathbb{N}} \{\text{Gos}_n, \text{Arr}_n, \text{Dep}_n^*, \text{Rep}_n^*\}. \quad (4.29)$$

#### 4.3.1 Fixed-size systems with random replacements

We first consider the simple setting where only gossip interactions and replacements take place, so that the number of agents remains constant:  $n(t) = n$ . In a first time, we will consider the case of random replacements, and hence work with  $\Xi$  as defined in (4.28). Gossip interactions and replacements take place with rates  $\lambda_{\text{Gos}_n} = n\lambda_g$  and  $\lambda_{\text{Rep}_n} = n\lambda_r$  respectively for some  $\lambda_g, \lambda_r \geq 0$ , and as no arrival nor departure ever happens,  $\lambda_{\text{Arr}_n} = \lambda_{\text{Dep}_n} = 0$ . This means that on average  $n\lambda_g$  gossip interactions and  $n\lambda_r$  replacements happen in the whole system per unit of time, and thus the rate of any event for a given individual agent is independent of  $n$ . Moreover, the expected number of gossip interactions taking place between two replacements defined as

$$\rho_g = \frac{\lambda_g}{\lambda_r} \quad (4.30)$$

and the probability of a random event being a replacement

$$p = \frac{\lambda_r}{\lambda_r + \lambda_g} \quad (4.31)$$

both remain constant as  $n$  grows (observe that  $p = \frac{1}{1+\rho_g}$ ).

#### *Evolution of the descriptors*

We can apply Proposition 4.8 combined with Lemmas 4.1 and 4.4 to derive the expected evolution of the descriptors for the system described above, *i.e.*, subject to gossips and random replacements.

**Theorem 4.11.** *In the system of Section 4.1 with the setting described above, *i.e.*, subject to gossip interactions and random replacements, there holds*

$$\frac{d}{dt} \mathbb{E} \mathbf{X}(t) = \begin{pmatrix} -2\lambda_r & \frac{\lambda_r}{n} \\ \lambda_g & -(\lambda_r + \lambda_g) \end{pmatrix} \mathbb{E} \mathbf{X}(t) + \lambda_r \begin{pmatrix} \frac{1}{n} \\ 1 \end{pmatrix} \sigma^2. \quad (4.32)$$

*Proof.* We consider the function  $f(\mathcal{S}(t)) = \mathbf{X}(t)$  defined in (4.18), which computes the descriptors. Since the system size is fixed  $n(t) = n$ , there holds for any  $t$ :

$$\begin{aligned} \pi_n(t) &= \mathbb{P}[n(t) = n] = 1; \\ F_n(t) &= \mathbb{E}[\mathbf{X}(t) | n(t) = n] = \mathbb{E} \mathbf{X}(t). \end{aligned}$$

Hence, reminding that  $\lambda_{\text{Gos}_n} = n\lambda_g$  and  $\lambda_{\text{Rep}_n} = n\lambda_r$ , applying Proposition 4.8 combined with Lemmas 4.1 and 4.4 yields

$$\begin{aligned} \frac{d}{dt} \mathbb{E} \mathbf{X}(t) &= n\lambda_r \left( A_{\text{Rep}_n} \mathbb{E} \mathbf{X}(t) + b_{\text{Rep}_n} \right) + n\lambda_g A_{\text{Gos}_n} \mathbb{E} \mathbf{X}(t) \\ &\quad - n(\lambda_r + \lambda_g) \mathbb{E} \mathbf{X}(t), \end{aligned}$$

where  $A_{\text{Rep}_n}$  and  $b_{\text{Rep}_n}$  come from Lemma 4.6 and  $A_{\text{Gos}_n}$  from Lemma 4.1. A few algebraic steps then conclude the proof.  $\square$

#### *Fixed points*

We now analyze the fixed point of (4.32) given by

$$\mathbb{E} \bar{x}^2|_{eq} = \frac{2 + \rho_g}{2n(1 + \rho_g) - \rho_g} \sigma^2 \quad ; \quad \mathbb{E} \bar{x}^2|_{eq} = \frac{2n + \rho_g}{2n(1 + \rho_g) - \rho_g} \sigma^2, \quad (4.33)$$

## 4 | Representing open systems: the case of gossiping

leading to the asymptotic variance

$$\mathbb{E}\text{Var}(\mathbf{x})|_{eq} = \frac{1 - \frac{1}{n}}{1 + \rho_g \left(1 - \frac{1}{2n}\right)} \sigma^2 \underset{n \rightarrow \infty}{\sim} \frac{\sigma^2}{\rho_g + 1}, \quad (4.34)$$

where we remind  $\rho_g = \lambda_g / \lambda_r$  is the ratio between gossip and replacement rates. These asymptotic values show some interpretation, detailed below.

**Infinitely many replacements:** As gossips become less frequent ( $\rho_g \rightarrow 0$ ) the system eventually consists of  $n$  agents holding i.i.d. values with mean 0 and variance  $\sigma^2$ , so that  $\mathbb{E}\bar{x}^2|_{eq} \rightarrow \frac{\sigma^2}{n}$  and  $\mathbb{E}\text{Var}(\mathbf{x})|_{eq} \rightarrow \frac{n-1}{n}\sigma^2$ . This actually corresponds to the fixed point of (4.13) from Lemma 4.4.

**Vanishingly few replacements:** As replacements become less frequent ( $\rho_g \rightarrow \infty$ ) the system achieves consensus between any two replacements, so that  $\mathbb{E}\bar{x}^2|_{eq} \rightarrow \frac{1}{2n-1}\sigma^2$  and  $\mathbb{E}\text{Var}(\mathbf{x})|_{eq} \rightarrow 0$ . The latter corresponds to the behavior of a closed system, which asymptotically achieves convergence. By contrast, the former results from the weighted average of the values of all the agents that have been part of the system at some past or present time (more details about this phenomenon are given in Appendix 4.D).

**Large system:** As the system size grows, one shows that  $\mathbb{E}\bar{x}^2|_{eq} \rightarrow 0$  and  $\mathbb{E}\text{Var}(\mathbf{x})|_{eq} \rightarrow \frac{\sigma^2}{1+\rho_g} = p\sigma^2$  (where  $p$  was defined in (4.31)). This last expression is consistent with [HM17], where an equivalent result was derived in discrete time. More generally, increasing  $\lambda_g$  makes the expected asymptotic variance decay, as the system gets closer to consensus; conversely, increasing  $\lambda_r$  makes it increase.

### Convergence rate

Theorem 4.11 also allows studying the rate at which the expected descriptors will converge to the fixed points of (4.33) and (4.34). In particular, one shows that the eigenvalues of the matrix in (4.32) are  $r_{1,2} = \frac{-\lambda_g - 3\lambda_r \pm \sqrt{\Delta}}{2}$ , where  $\Delta = (\lambda_g - \lambda_r)^2 + 4\frac{\lambda_g\lambda_r}{n}$ . Hence,  $\sqrt{\Delta} = |\lambda_g - \lambda_r| + o\left(\frac{1}{n}\right)$ , and

$$r_1 = -(\lambda_g + \lambda_r) + o\left(\frac{1}{n}\right) \quad \text{and} \quad r_2 = -2\lambda_r + o\left(\frac{1}{n}\right). \quad (4.35)$$

These values actually correspond to the diagonal elements of the matrix in (4.32). The eigenvectors are then given by  $v_{1,2} = \left( \frac{\rho_g - 1 \pm \sqrt{(\rho_g - 1)^2 + 4 \frac{\rho_g}{n}}}{2\rho_g}, 1 \right)^\top$ , yielding the pairs  $(r_1, v_1)$  and  $(r_2, v_2)$ , and using the same reasoning as for the eigenvalues one gets

$$v_1 = (o(1) \quad 1)^\top \quad \text{and} \quad v_2 = \left( \frac{\rho_g - 1}{\rho_g} + o(1) \quad 1 \right)^\top. \quad (4.36)$$

Observe that the convergence of the expected descriptors towards their respective fixed points is guaranteed, as both eigenvalues are always non-positive (since  $\lambda_g, \lambda_r \geq 0$  by definition). We provide below some additional interpretation of the convergence rates.

**Replacements in majority:** When  $\lambda_g \ll \lambda_r$ , then the impact of  $v_2$  vanishes faster than that of  $v_1$ , so that convergence is driven by  $v_1 = (o(1), 1)^\top$ , whose component  $\mathbb{E}\bar{x}^2$  is vanishingly small. Hence, as  $\mathbb{E}\bar{x}^2$  quickly decays to 0,  $\mathbb{E}\text{Var}(x)$  becomes equivalent to  $\mathbb{E}\bar{x}^2$ , and their convergence rate is then characterized by  $r_2 \approx -(\lambda_g + \lambda_r)$ .

**Gossips in majority:** When  $\lambda_g \gg \lambda_r$ , then  $|r_1| \gg |r_2|$ . Hence, as the effect of  $v_1$  vanishes quickly as compared to that of  $v_2$ , the convergence is conducted by  $v_2 \approx (1, 1)^\top$ , so that both  $\mathbb{E}\bar{x}^2$  and  $\mathbb{E}x^2$  converge at the same rate characterized by  $r_2 \approx -2\lambda_r$ . In particular, it means that the expected variance quickly gets closer to 0 (as in that case almost-consensus is achieved between each replacement), and the convergence of the descriptors themselves is driven by the replacements which bring in new information.

**Remark 4.3.** For the specific case where  $\lambda_r \rightarrow 0$ , i.e., when the system behaves as a closed system, then  $r_2 \rightarrow 0$  and no external contribution is brought to the system anymore. Hence,  $\mathbb{E}\bar{x}^2$  remains unchanged, and  $\mathbb{E}x^2$  ends up converging towards  $\mathbb{E}\bar{x}^2$ . As a consequence  $\mathbb{E}\text{Var}(x) = \mathbb{E}x^2 - \mathbb{E}\bar{x}^2$  asymptotically decays to 0, following the convergence rate of  $\mathbb{E}\bar{x}^2$ , which is conditioned by the gossip steps.

#### Illustration

To illustrate Theorem 4.11 and the interpretation that followed, we consider a system of 50 agents subject to random replacements and gossips in such a way that on average one event in twenty is a replacement (i.e.,  $\rho_g = 19$ ). The initial values of the agents are randomly drawn from a normal distribution with zero mean and constant variance  $\sigma^2 = 1$ . The

simulated evolution of the system, and in particular of its descriptors, is displayed and compared with the theoretical results in Fig. 4.9.

Fig. 4.9(a) shows a single realization of the system, and in particular how convergence is prevented by replacements even if the agents occasionally get close to consensus. Figures 4.9(b)-(d) compare the evolution of the descriptors in expectation with the theoretical expectations of Theorem 4.11 and the fixed-points computed in (4.33) and (4.34), and show that they accurately match each other.

We can also explicitly compute the eigenvalues from (4.35) and eigenvectors from (4.36) in this configuration, yielding the pairs

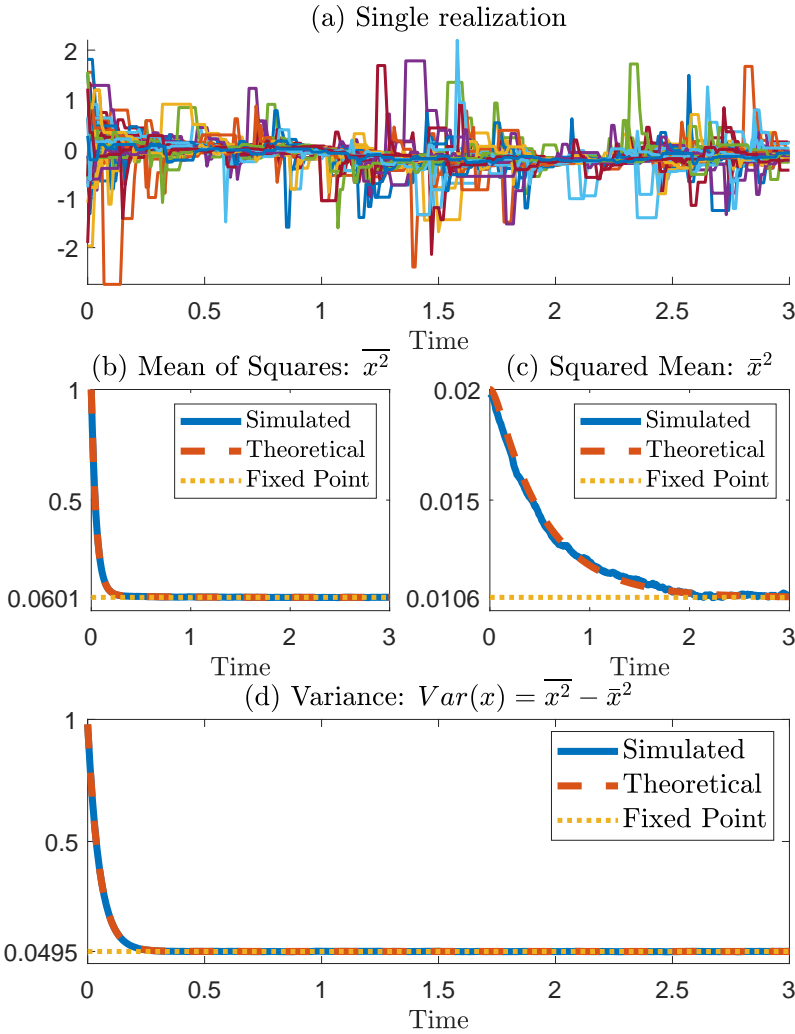
$$\begin{aligned} r_1 &\approx -20; & v_1 &\approx (0.001 \ 1)^\top; \\ r_2 &\approx -2; & v_2 &\approx (0.95 \ 1)^\top. \end{aligned}$$

These values, consistently with Fig. 4.9 and with the convergence rate analysis performed above, suggest a faster convergence of  $\mathbb{E}\bar{x}^2$  as compared to that of  $\mathbb{E}\bar{x}^2$ , and the convergence of the variance ends up being driven by that of  $\mathbb{E}\bar{x}^2$ . Nevertheless, since in that case  $\mathbb{E}\bar{x}^2$  significantly dominates  $\mathbb{E}\bar{x}^2$  in magnitude,  $\mathbb{E}\text{Var}(x)$  gets very close to its fixed-point as fast as  $\mathbb{E}\bar{x}^2$  converges.

### 4.3.2 Fixed-size systems with adversarial replacements

Let us now consider a variation of the setting of the previous section: We assume that departures and replacements are *adversarial* instead of *random*, and hence we work with  $\Xi^*$  as defined in (4.29). As a quick reminder, whereas with random departures the leaving agent is uniformly randomly chosen, with adversarial departures no assumption can be made on that choice, so that only upper bounds can be derived. Observe that this formulation of departures and replacements actually encapsulates the random one, and can thus be seen as a generalization.

Therefore, in this section, we consider a system of fixed size just as previously, but where departures and replacements are adversarial, *i.e.*, the considered clocks are  $\text{Dep}_n^*$  and  $\text{Rep}_n^*$  with equivalent rates, namely  $\lambda_{\text{Gos}_n} = n\lambda_g$ ,  $\lambda_{\text{Arr}_n} = \lambda_{\text{Dep}_n^*} = 0$  and  $\lambda_{\text{Rep}_n^*} = n\lambda_r$ . We can then apply Proposition 4.9, combined with Lemmas 4.1 and 4.6 to derive the following upper bound on the evolution of the variance in expectation for fixed-size systems subject to adversarial replacements.



**Fig. 4.9** Expected behavior of a system of 50 agents subject to random replacements and gossips with rates  $\lambda_r = 1$  and  $\lambda_g = 19$ , so that  $\lambda_r + \lambda_g = 20$ : (a) shows the simulated evolution of one realization of the process, where each line is the estimate of an agent; (b)-(d) respectively show the evolutions of the expected mean of squares  $\overline{x^2}$ , squared mean  $\bar{x}^2$  and variance  $Var(x)$  simulated over 10000 realizations (in plain blue line), and compare them with the theoretical results of Theorem 4.11 where the variance is deduced from  $Var(x) = \overline{x^2} - \bar{x}^2$  (in dashed red line), and with the theoretical fixed-points computed in (4.33) and (4.34) (in yellow dotted line).

**Theorem 4.12.** *In the system of Section 4.1 with the setting described above, i.e., subject to gossip interactions and adversarial replacements, there holds*

$$\frac{d}{dt} \mathbb{E} \text{Var}(\mathbf{x}((t))) \leq -\lambda_g \mathbb{E} \text{Var}(\mathbf{x}((t))) + \lambda_r \sigma^2. \quad (4.37)$$

*Proof.* We consider the function  $f(\mathcal{S}(t)) = \text{Var}(\mathbf{x}((t)))$ , which computes the variance. Since the system size is fixed  $n(t) = n$ , there holds for any  $t$ :

$$\begin{aligned} \pi_n(t) &= \mathbb{P}[n(t) = n] = 1; \\ F_n(t) &= \mathbb{E}[\text{Var}(\mathbf{x}((t)) | n(t) = n] = \mathbb{E} \text{Var}(\mathbf{x}((t))). \end{aligned}$$

Hence, reminding that  $\lambda_{\text{Gos}_n} = n\lambda_g$  and  $\lambda_{\text{Rep}_n} = n\lambda_r$ , applying Proposition 4.9 combined with Lemmas 4.1 and 4.6 yields

$$\begin{aligned} \frac{d}{dt} \mathbb{E} \text{Var}(\mathbf{x}((t))) &\leq n\lambda_g \frac{n-1}{n} \mathbb{E} \text{Var}(\mathbf{x}(t)) + n\lambda_r \left( \mathbb{E} \text{Var}(\mathbf{x}(t)) + \frac{\sigma^2}{n} \right) \\ &\quad - (n\lambda_g + n\lambda_r) \mathbb{E} \text{Var}(\mathbf{x}(t)), \end{aligned}$$

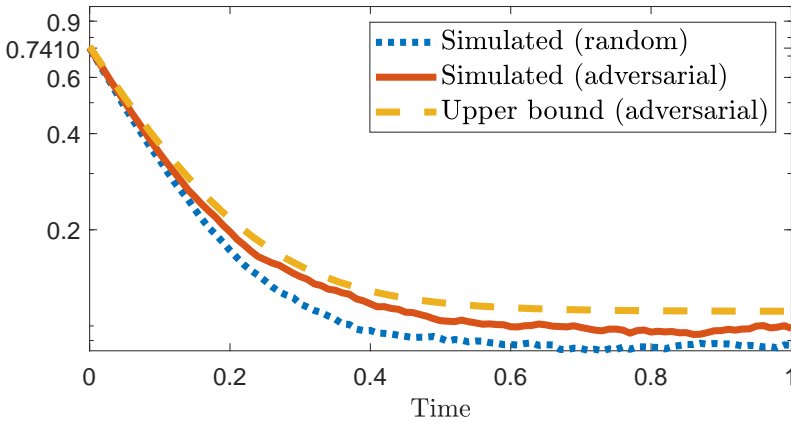
and a few algebraic steps then conclude the proof.  $\square$

By definition adversarial replacements allow arbitrarily choosing the leaving agent at replacements, including the “worst case” with respect to the evolution of the variance. Therefore, they do not allow deriving its exact evolution but only an upper bound. Observe we can determine the fixed-point from (4.37), yielding

$$\lim_{t \rightarrow \infty} \mathbb{E} \text{Var}(\mathbf{x}(t)) \leq \frac{\lambda_r}{\lambda_g} \sigma^2 = \frac{\sigma^2}{\rho_g}. \quad (4.38)$$

Consistently with the interpretation above, this expression actually corresponds to an overapproximation of (4.34), the fixed-point for the expected variance with random replacements, especially for  $n \rightarrow \infty$  where it becomes equivalent up to a constant 1.

We show in Fig. 4.10 the evolution of the expected variance simulated over a system of 4 agents where one every ten events on average is an adversarial replacement. Adversarial replacements are here arbitrarily defined as choosing the agent  $j$  with minimal  $|x_j|$  at the time of the replacement to leave the system. The figure compares the evolution of the variance in this setting with the upper bound of Theorem 4.12 and with the



**Fig. 4.10** Evolution (in logarithmic scale) of the expected variance of an open system of fixed size with 4 agents subject to gossip steps (with rate  $\lambda_g = 9$ ) and to replacements (with  $\lambda_r = 1$ ): The blue dotted line corresponds to the simulated variance subject to adversarial replacements as defined in Section 4.3.2, the orange plain line to the simulated variance subject to random replacements, and the yellow dashed line to the bound (4.37) from Theorem 4.12.

evolution of the variance under random replacements. Whereas the arbitrary definition of adversarial replacements we consider might not be the “worst case” with respect to the variance, it appears that it induces a larger variance than with random replacements. More generally, the bound we present in the figure is valid for any way of defining the choice of the leaving agent at replacements, including the worst ones with respect to any criterion.

### 4.3.3 Variable-size systems

We now consider a more general case where arrivals and (random) departures are decoupled, so that the system size changes with time. We consider that the rate of arrivals is independent of the system size, whereas both the gossip and departure rates scale with it, so that when the system size is  $n$ , one has  $\lambda_{\text{Arr}_n} = \lambda_a$ ,  $\lambda_{\text{Dep}_n} = n\lambda_d$ , and  $\lambda_{\text{Gos}_n} = n\lambda_g$  for some  $\lambda_a, \lambda_d, \lambda_g \geq 0$ . We also assume that no replacement happens, so that  $\lambda_{\text{Rep}_n} = 0$ . This characterization of the different rates is arbitrary, and one could have defined them in many different ways. Nevertheless, this choice is rather natural, as it means that (i) the rate at which a single agent expe-

riences gossip interactions and departures does not depend on the system size; (ii) the arrival rate is defined at the system level, independently of the number of agents in the system, so that the probability for an agent to join the system is always the same.

*Evolution of the descriptors*

We can now follow the same approach as in Section 4.3.1 to characterize the exact evolution of the descriptors in expectation. Applying Proposition 4.8 combined with Lemmas 4.1 to 4.3 thus yields the following result.

**Theorem 4.13.** *Consider the system of Section 4.1, and let  $\pi_j(t) := \mathbb{P}[n(t) = j]$  and  $\mathbf{X}_j(t) := \mathbb{E}[\mathbf{X}(t)|n(t) = j]$ . With the setting of Section 4.3.3, i.e., subject to gossip interactions, arrivals and random departures, there holds*

$$\begin{aligned} \frac{d}{dt} \mathbb{E} \mathbf{X}(t) &= \sum_{j=1}^{\infty} \pi_j(t) \begin{pmatrix} -\frac{2j+1}{(j+1)^2} \lambda_a - \frac{1}{(j-1)^2} \lambda_d & \frac{j}{(j-1)^2} \lambda_d \\ \lambda_g & -\lambda_g - \frac{1}{j+1} \lambda_a \end{pmatrix} \mathbf{X}_j(t) \\ &+ \sum_{j=0}^{\infty} \lambda_a \pi_j(t) \begin{pmatrix} \frac{1}{(j+1)^2} \\ \frac{1}{j+1} \end{pmatrix} \sigma^2 - \lambda_a \pi_0(t) \mathbf{X}_0(t). \end{aligned} \quad (4.39)$$

*Proof.* We consider the function  $f(\mathcal{S}(t)) = \mathbf{X}(t)$  defined in (4.18), which computes the descriptors. There holds by definition for all  $j \in \mathbb{N}$ :

$$\begin{aligned} \pi_j(t) &= \mathbb{P}[n(t) = j]; \\ F_j(t) &= \mathbb{E}[\mathbf{X}(t)|n(t) = j] = \mathbf{X}_j(t). \end{aligned}$$

Hence, reminding that  $\lambda_{\text{Gos}_j} = j\lambda_g$ ,  $\lambda_{\text{Dep}_j} = n\lambda_r$  and  $\lambda_{\text{Arr}_j} = \lambda_a$  for all  $j \in \mathbb{N}$ , applying Proposition 4.8 combined with Lemmas 4.1 to 4.3 yields

$$\begin{aligned} \frac{d}{dt} \pi_j(t) \mathbf{X}_j(t) &= j\lambda_g A_{\text{Gos}_j} \pi_j(t) \mathbf{X}_j(t) \\ &+ \lambda_a \left( A_{\text{Arr}_{j-1}} \mathbf{X}_{j-1}(t) + b_{\text{Arr}_{j-1}} \right) \pi_{j-1}(t) \\ &+ (j+1)\lambda_d A_{\text{Dep}_{j+1}} \pi_{j+1}(t) \mathbf{X}_{j+1} \\ &- (j\lambda_g + \lambda_a + j\lambda_d) \pi_j(t) \mathbf{X}_j(t), \end{aligned}$$

where  $A_{\text{Gos}_j}$ ,  $A_{\text{Dep}_{j+1}}$ ,  $A_{\text{Arr}_{j-1}}$  and  $b_{\text{Arr}_{j-1}}$  come from Lemmas 4.1 to 4.3. Summing over all values of  $j$  then concludes the proof.  $\square$

By using (4.39), one can track the exact evolution of the descriptors, and

hence of the variance as well. Yet, obtaining an exact description happens to be far more complex in that case than for the fixed-size setting, it is thus omitted in this thesis. Instead, in the remainder of this section, we obtain an upper bound on the evolution of the expected variance starting from (4.39) by bounding some intermediate terms.

*Upper bound for the variance*

In order to obtain the next theorem, we first give the following lemma which shows that the descriptors are bounded in expectation given the system size, and which is proved in Appendix 4.B.

**Lemma 4.14.** *In the setting described in Section 4.1.1, and for any fixed time  $t \geq 0$ , there holds*

$$\mathbb{E} \left[ \bar{x}^2(t) | n(t) = j \right] \leq \frac{1}{j} \sigma^2; \quad \mathbb{E} \left[ \overline{x^2}(t) | n(t) = j \right] \leq \sigma^2, \quad (4.40)$$

where we remind  $n(t)$  is the system size at time  $t$ .

Observe that from the initialization of the system, there holds

$$\mathbb{E} \left[ \bar{x}^2(0) | n(0) = j \right] = \frac{1}{j} \sigma^2 \quad \text{and} \quad \mathbb{E} \left[ \overline{x^2}(0) | n(0) = j \right] = \sigma^2. \quad (4.41)$$

Hence, Lemma 4.14 actually states if the values held by the agents are randomly initialized as in Section 4.1.1, then the expected values of the descriptors obtained from a sequence of arrivals, departures, gossips and replacements at any time  $t$  (conditioned by the system size at that time) never exceed their initial conditional expected values. We now give the following theorem.

**Theorem 4.15.** *In the system of Section 4.1 with the setting described above, i.e., subject to gossip interactions, arrivals and random departures, there holds*

$$\frac{d}{dt} \mathbb{E} \text{Var}(x(t)) \leq -\lambda_g \mathbb{E} \text{Var}(x(t)) + \mathbb{E} \left[ \frac{\lambda_a}{n(t) + 1} \right] \sigma^2. \quad (4.42)$$

*Proof.* Remember that  $\mathbb{E} \text{Var}(x(t)) = (-1 \ 1) \mathbb{E} X(t)$ . Hence, starting from

#### 4 | Representing open systems: the case of gossiping

(4.39), one finds that

$$\begin{aligned} \frac{d}{dt} \mathbb{E} \text{Var}(x(t)) &= \begin{pmatrix} -1 & 1 \end{pmatrix} \frac{d}{dt} \mathbb{E} \mathbf{X}(t) \\ &= \sum_{j=1}^{\infty} \pi_j(t) \left( \lambda_g + \frac{2j+1}{(j+1)^2} \lambda_a + \frac{1}{(j-1)^2} \lambda_d - \lambda_g - \frac{1}{j+1} \lambda_a - \frac{1}{(j-1)^2} \lambda_d \right) \mathbf{X}_j(t) \\ &\quad + \sum_{j=0}^{\infty} \lambda_a \pi_j(t) \left( \frac{1}{j+1} - \frac{1}{(j+1)^2} \right) \sigma^2 - \lambda_a \pi_0(t) V_0(t). \end{aligned}$$

Let  $V_j(t) := \mathbb{E} [\text{Var}(x(t)) | n(t) = j]$ . Since  $V_j(t) = \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{X}_j(t)$  we have

$$\begin{aligned} \frac{d}{dt} \mathbb{E} \text{Var}(x(t)) &= - \sum_{j=1}^{\infty} \pi_j(t) \left( \lambda_g + \frac{1}{(j-1)^2} \lambda_d - \frac{1}{j+1} \lambda_a \right) V_j(t) \\ &\quad + \sum_{j=1}^{\infty} \pi_j(t) \frac{j}{(j+1)^2} \lambda_a \mathbb{E} [\bar{x}^2(t) | n(t) = j] \\ &\quad + \sum_{j=0}^{\infty} \pi_j(t) \frac{j}{(j+1)^2} \lambda_a \sigma^2 - \lambda_a \pi_0(t) V_0(t). \end{aligned}$$

Observe that the first term of the expression above satisfies

$$- \sum_{j=1}^{\infty} \pi_j(t) \left( \lambda_g + \frac{1}{(j-1)^2} \lambda_d - \frac{1}{j+1} \lambda_a \right) V_j(t) \leq -\lambda_g \mathbb{E} \text{Var}(x(t)).$$

Moreover, from Lemma 4.14, we have  $\mathbb{E} [\bar{x}^2(t) | n(t) = j] \leq \frac{\sigma^2}{j}$ . Finally, observe that  $-\lambda_a \pi_0(t) V_0(t) \leq 0$ , so that

$$\frac{d}{dt} \mathbb{E} \text{Var}(x(t)) \leq -\lambda_g \mathbb{E} \text{Var}(x(t)) + \sum_{j=0}^{\infty} \pi_j(t) \left( \frac{1}{(j+1)^2} + \frac{j}{(j+1)^2} \right) \lambda_a \sigma^2.$$

The conclusion then follows from  $\sum_{j=0}^{\infty} \lambda_a \frac{\pi_j(t)}{j+1} = \mathbb{E} \left[ \frac{\lambda_a}{n(t)+1} \right]$ .  $\square$

The result of Theorem 4.15 gives an upper bound on the evolution of the expected variance. In that sense, it is parallel to (4.37) obtained for systems of fixed size with adversarial replacements. Interestingly, it overestimates the effect of arrivals on the variance, while neglecting the slight favourable impact of departures, thus introducing conservatism in the bound.

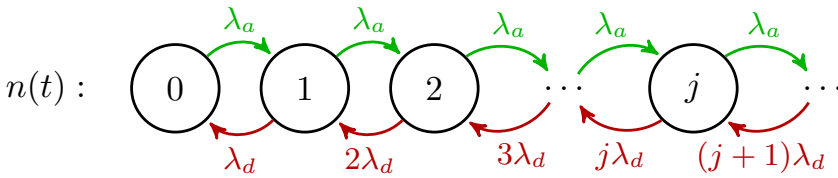
**Remark 4.4.** The same approach as that used in Theorem 4.15 can be applied for the fixed-size case starting from Theorem 4.11, yielding the following upper bound on the evolution of the variance in expectation:

$$\frac{d}{dt} \mathbb{E} \text{Var}(\mathbf{x}(t)) \leq - \left( \lambda_g + \frac{n+1}{n} \lambda_r \right) \mathbb{E} \text{Var}(\mathbf{x}(t)) + \frac{n^2 - 1}{n^2} \lambda_r \sigma^2. \quad (4.43)$$

This result is however weaker than that we obtain by directly tracking the descriptors using Theorem 4.11, as it allows the exact characterization of  $\frac{d}{dt} \mathbb{E} \text{Var}(\mathbf{x}(t))$  using the fact that  $\mathbb{E} \text{Var}(\mathbf{x}(t)) = (-1 \ 1) \mathbb{E} X(t)$ . This highlights how our descriptor-based approach allows for a more detailed analysis of the evolution of the variance, even if one were not interested in the descriptors themselves but only in the variance.

*Upper bound for the asymptotic variance*

Observe that the system size  $n(t)$  evolves following a birth-death process as depicted in Figure 4.11 where birth and death rates respectively follow the arrival and departure rates.



**Fig. 4.11** Birth-death process governing the evolution of the system size  $n(t)$  in the setting of this section, *i.e.*, where arrivals correspond to births and departures to deaths.

Hence, we have the following lemma.

**Lemma 4.16.** Let  $\pi_i(t) := \mathbb{P} [n(t) = i]$  for all  $i \in \mathbb{N}$ . Assume  $\bar{n} := \frac{\lambda_a}{\lambda_d} < \infty$ , then for all  $i \in \mathbb{N}$  there holds

$$\pi_i^* = \lim_{t \rightarrow \infty} \pi_i(t) = \frac{\bar{n}^i}{i!} e^{-\bar{n}}. \quad (4.44)$$

Moreover,  $\bar{n}$  is the characteristic size of the system, *i.e.*,

$$\lim_{t \rightarrow \infty} \mathbb{E} n(t) = \bar{n}. \quad (4.45)$$

*Proof.* Standard results on birth-death processes given in (2.33) show that

#### 4 | Representing open systems: the case of gossiping

if  $n(t)$  is ergodic, then  $\pi_i^* = \lim_{t \rightarrow \infty} \pi_i(t)$  exists for all  $i$  and satisfies

$$\pi_i^* = \pi_0^* \prod_{j=1}^i \frac{\lambda_a}{j\lambda_d}$$

with

$$\pi_0^* = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{j=1}^k \frac{\lambda_a}{j\lambda_d}} = \frac{1}{1 + \sum_{k=1}^{\infty} \frac{\bar{n}^k}{k!}} = e^{-\bar{n}},$$

yielding  $\pi_i^* = \frac{\bar{n}^i}{i!} e^{-\bar{n}}$ . The fact that  $n(t)$  is ergodic if  $\lambda_a/\lambda_d < \infty$  then yields (4.44). Finally, there holds

$$\lim_{t \rightarrow \infty} \mathbb{E}n(t) = \sum_{j=0}^{\infty} j\pi_j^* = \sum_{j=1}^{\infty} j \frac{\bar{n}^j}{j!} e^{-\bar{n}} = \bar{n} e^{-\bar{n}} \sum_{j=0}^{\infty} \frac{\bar{n}^j}{j!} = \bar{n} e^{-\bar{n}} e^{\bar{n}} = \bar{n},$$

which yields (4.45). □

Using Lemma 4.16, we derive the following result from Theorem 4.15, which bounds the asymptotic expected variance.

**Theorem 4.17.** *Let  $\bar{n} = \lambda_a/\lambda_d$  and  $\gamma = \lambda_g/\lambda_d$ , then in the same setting as that of Theorem 4.15 there holds*

$$\lim_{t \rightarrow \infty} \mathbb{E}\text{Var}(\mathbf{x}(t)) \leq \frac{1 - e^{-\bar{n}}}{\gamma} \sigma^2. \quad (4.46)$$

*Proof.* Let us define  $h(t) := \mathbb{E} \left[ \frac{\lambda_a \sigma^2}{n(t)+1} \right] = \lambda_a \sum_{j=0}^{\infty} \frac{\pi_j(t)}{j+1} \sigma^2$ . Then, using Lemma 4.16, for  $\lambda_d \neq 0$  there holds

$$\lim_{t \rightarrow \infty} h(t) = \lambda_a e^{-\bar{n}} \sum_{j=0}^{\infty} \frac{\bar{n}^j}{(j+1)!} \sigma^2 = \lambda_a \frac{1 - e^{-\bar{n}}}{\bar{n}} \sigma^2.$$

Grönwall's lemma [Pac98] then gives

$$\mathbb{E}\text{Var}(\mathbf{x}(t)) \leq e^{-\lambda_g t} \mathbb{E}\text{Var}(\mathbf{x}(0)) + \int_0^t h(s) e^{-\lambda_g(t-s)} ds.$$

Standard results from dynamical systems then yield

$$\lim_{t \rightarrow \infty} \int_0^t h(s) e^{-\lambda_g(t-s)} ds = \lim_{t \rightarrow \infty} \frac{h(t)}{\lambda_g} = \frac{\lambda_a}{\lambda_g} \frac{1 - e^{-\bar{n}}}{\bar{n}} \sigma^2,$$

which concludes the proof.  $\square$

Theorem 4.17 allows interpreting the behavior of the variance in expectation in our setting. As soon as  $\bar{n}$  is not very small, namely, if arrivals are not too infrequent with respect to departures, (4.46) is approximately  $\frac{\sigma^2}{\gamma}$ , and hence the expected variance decays to zero when  $\gamma$  becomes large (for gossip steps more frequent than departures). Interestingly, this bound is inversely proportional to the expected number of gossip interactions involving a given agent before it leaves the system  $\gamma$ , similarly to (4.43) obtained in Section 4.3.1 for the fixed-size case (up to a constant 1). More generally, when  $\lambda_d \rightarrow 0$ , then (4.46) becomes  $\lim_{t \rightarrow \infty} \mathbb{E}\text{Var}(\mathbf{x}(t)) \rightarrow 0$ .

**Remark 4.5.** *In the specific case where  $\lambda_d = 0$  (i.e., for an only growing system), one can actually show that*

$$\lim_{t \rightarrow \infty} \mathbb{E}\text{Var}(\mathbf{x}(t)) = 0 \quad (4.47)$$

The proof of that result uses arguments very similar to those used to prove Theorem 4.17. In particular, in ever growing systems, there holds

$$\lim_{t \rightarrow \infty} h(t) = \lim_{t \rightarrow \infty} \lambda_a e^{-\bar{n}} \sum_{j=0}^{\infty} \frac{\bar{n}^j}{(j+1)!} \sigma^2 = 0,$$

so that Grönwall's lemma gives  $\lim_{t \rightarrow \infty} \int_0^t h(s) e^{-\lambda_g(t-s)} ds = \lim_{t \rightarrow \infty} \frac{h(t)}{\lambda_g} = 0$  which yields the result.

*Refining the asymptotic bound*

Let us define  $V_j(t) := \mathbb{E} [\text{Var}(\mathbf{x}(t)) | n(t) = j]$ . Following the same reasoning as in the proof of Theorem 4.15 with  $V_j(t) = (-1 \ 1) \mathbf{X}_j(t)$ , and using the fact that  $\mathbb{E} [\bar{x}^2(t) | n(t) = j] \leq \frac{\sigma^2}{j}$  from lemma 4.40, we obtain

$$\begin{aligned} \frac{d}{dt} V_j(t) \pi_j(t) &\leq (j+1) \lambda_d \left(1 - \frac{1}{j}\right) V_{j+1}(t) \pi_{j+1}(t) + \lambda_a \frac{j-1}{j} V_{j-1}(t) \pi_{j-1}(t) \\ &\quad - (\lambda_a + \lambda_g + j \lambda_d) V_j(t) \pi_j(t) + \lambda_a \pi_{j-1}(t) \frac{\sigma^2}{j}, \end{aligned} \quad (4.48)$$

#### 4 | Representing open systems: the case of gossiping

where we remind  $\pi_j(t) := \mathbb{P}[n(t) = j]$ . We now show that, starting from this expression instead of the result of Theorem 4.15, we can refine the upper bound on the asymptotic expected variance obtained in Theorem 4.17. We first provide the following intermediate result.

**Proposition 4.18.** *Let  $(z_j)_{j \in \mathbb{N}}$  be a sequence of positive numbers with  $z_0 = 0$  and satisfying*

$$(\bar{n} + j + \gamma)z_j \geq jz_{j+1} + \left(1 - \frac{1}{(j-1)^2}\right) \bar{n}z_{j-1} + \frac{\bar{n}^j}{j!} e^{-\bar{n}} \quad (4.49)$$

for all  $j \geq 1$ , then there holds

$$\lim_{t \rightarrow \infty} \mathbb{E} \text{Var}(\mathbf{x}(t)) \leq \sum_{j=0}^{\infty} z_j \sigma^2. \quad (4.50)$$

*Proof.* Let  $V_j^* := \lim_{t \rightarrow \infty} F_j(t)$ , and let  $\pi_j^* = \lim_{t \rightarrow \infty} \mathbb{P}[n(t) = j] = \frac{\bar{n}^j}{j!} e^{-\bar{n}}$  from Lemma 4.16, such that  $\lim_{t \rightarrow \infty} \mathbb{E} \text{Var}(\mathbf{x}(t)) = \sum_{j=0}^{\infty} \pi_j^* V_j^*$ .

First, observe that asymptotically, (4.48) becomes

$$\begin{aligned} & (\lambda_a + j\lambda_g + \lambda_d) V_j^* \frac{\bar{n}^j}{j!} e^{-\bar{n}} \\ & \leq \left( \frac{j-1}{j} + \frac{\sigma^2}{j} \right) \lambda_a \frac{\bar{n}^{j-1}}{(j-1)!} e^{-\bar{n}} + \left( 1 - \frac{1}{j^2} \right) \lambda_d \frac{\bar{n}^{j+1}}{(j+1)!} e^{-\bar{n}}. \end{aligned}$$

Reminding that  $\bar{n} = \lambda_a / \lambda_d$  and  $\gamma = \lambda_g / \lambda_d$ , and dividing both sides of the inequality above by  $\lambda_d$ , one then has

$$(\bar{n} + j + \gamma) V_j^* \leq (j-1) V_{j-1}^* + \left( 1 - \frac{1}{j^2} \right) \bar{n} V_{j+1}^* + \sigma^2. \quad (4.51)$$

Now observe that the last term of (4.49) corresponds to  $\pi_j^*$ , and hence

$$\begin{aligned} \lim_{t \rightarrow \infty} \mathbb{E} \text{Var}(\mathbf{x}(t)) &= \sum_{j=0}^{\infty} \pi_j^* V_j^* \\ &\leq \sum_{j=0}^{\infty} \left( (\bar{n} + j + \gamma) z_j - j z_{j+1} - \left( 1 - \frac{1}{(j-1)^2} \right) \bar{n} z_{j-1} \right) V_j^* \\ &= \sum_{j=0}^{\infty} \left( (\bar{n} + j + \gamma) V_j^* - (j-1) V_{j-1}^* - \left( 1 - \frac{1}{j^2} \right) \bar{n} V_{j+1}^* \right) z_j, \end{aligned}$$

where the last equality follows from reordering the different terms. The factor multiplying  $z_j$  in the last expression actually corresponds to (4.51), and in particular

$$\left( (\bar{n} + j + \gamma)V_j^* - (j - 1)V_{j-1}^* - \left(1 - \frac{1}{j^2}\right) \bar{n}V_{j+1}^* \right) z_j \leq \sigma^2 z_j,$$

hold for all  $j \in \mathbb{N}$ , which concludes the proof.  $\square$

Based on the result above, we provide in the following theorem an improved upper bound on the asymptotic expected variance.

**Theorem 4.19.** *Let  $\bar{n} = \lambda_a / \lambda_d$  and  $\gamma = \lambda_g / \lambda_d$ , then in the same setting as that of Theorem 4.15 there holds*

$$\lim_{t \rightarrow \infty} \mathbb{E} \text{Var}(\mathbf{x}(t)) \leq \frac{1 - e^{-\bar{n}}}{1 + \gamma} \sigma^2. \quad (4.52)$$

*Proof.* One can verify that the sequence  $(z_j)_{j \in \mathbb{N}}$  such that  $z_0 = 0$  and  $z_j = \frac{\bar{n}^j}{(1+\gamma)j!} e^{-\bar{n}}$  for  $j \geq 1$  satisfies (4.49). Hence, it follows from Proposition 4.18 that

$$\lim_{t \rightarrow \infty} \mathbb{E} \text{Var}(\mathbf{x}(t)) \leq \frac{\sigma^2}{1 + \gamma} e^{-\bar{n}} \sum_{j=1}^{\infty} \frac{\bar{n}^j}{j!} = \frac{\sigma^2}{1 + \gamma} e^{-\bar{n}} (e^{\bar{n}} - 1),$$

which concludes the proof.  $\square$

The bound of the above theorem reinforces the interpretation of the previous one, as it is inversely proportional to  $\gamma + 1$ . In that sense, it is extremely similar to (4.34) obtained in Section 4.3.1 for the fixed-size case: This bound was inversely proportional to  $\rho_g + 1$ , where  $\rho_g$  characterizes the expected number of interactions experienced by an agent before leaving the system, exactly like  $\gamma$  in our varying-size setting.

The result of Theorem 4.19 was obtained by choosing an appropriate sequence  $(z_j)$  in the sense of Proposition 4.18. More generally, any such sequence  $(z_j)$  satisfying these conditions yields a valid bound. In particular

#### 4 | Representing open systems: the case of gossiping

one could compute or numerically approximate

$$\min_{(z_j)_{j \in \mathbb{N}}, z_j \in \mathbb{R}_+} \sum_{j=0}^{\infty} z_j \sigma^2 \quad (4.53)$$

$$\text{s.t. } (\bar{n} + j + \gamma)z_j \geq jz_{j+1} + \left(1 - \frac{1}{(j-1)^2}\right) \bar{n}z_{j-1} + \frac{\bar{n}^j}{j!} e^{-\bar{n}} \quad \forall j,$$

which can be interpreted as the “best possible bound” that can be obtained in the sense of Proposition 4.18. In fact, this result strongly relates with the concept of duality in optimization. Indeed, the proof of Proposition 4.18, and in particular expression (4.51) gives conditions on the quantities  $V_j(t)$ , so that an upper bound for the asymptotic variance can be obtained by solving

$$\max_{(x_j)_{j \in \mathbb{N}}, x_j \in \mathbb{R}_+} \sum_{j=0}^{\infty} \pi_j^* x_j \quad (4.54)$$

$$\text{s.t. } (\bar{n} + j + \gamma)x_j \leq (j-1)x_{j-1} + \left(1 - \frac{1}{j^2}\right) \bar{n}x_{j+1} + \sigma^2 \quad \forall j,$$

where the constraint exactly corresponds to (4.51). Any feasible sequence  $(z_j)$  in the sense of Proposition 4.49 thus corresponds to a feasible dual solution for (4.54), and in particular the solution of (4.53) corresponds, up to the duality gap, to that of (4.54).

#### Summary

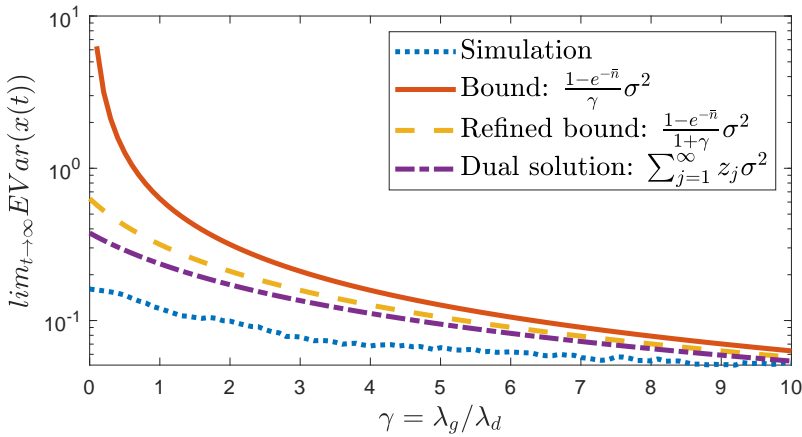
We briefly summarize the different bounds that were obtained for the asymptotic expected variance in the following table.

	Reference	Bound
Simple bound	Theorem 4.17	$\frac{1-e^{-\bar{n}}}{\gamma} \sigma^2$
Refined bound	Theorem 4.19	$\frac{1-e^{-\bar{n}}}{1+\gamma} \sigma^2$
Dual solution	Expression (4.53)	$\min_{(z_j)} \sum_{j=0}^{\infty} z_j \sigma^2$ s.t. (4.49)
Primal solution	Expression (4.54)	$\max_{(x_j)} \sum_{j=0}^{\infty} \pi_j^* x_j$ s.t. (4.51)

**Table 4.1** Summary of the different upper bounds obtained throughout this section for  $\lim_{t \rightarrow \infty} \mathbb{E}\text{Var}(x(t))$ .

*Illustration*

We illustrate the different results previously obtained in Fig. 4.12. The figure shows the asymptotic expected variance of a system of variable size of initially 5 agents for different values of  $\gamma = \lambda_g/\lambda_d$ , namely the expected number of interactions involving an agent before its departure. It compares it with the bounds respectively obtained in Theorems 4.17 and 4.19 for the asymptotic variance, and with the bound obtained from solving an finite-dimensional approximation the dual optimization problem (4.53).



**Fig. 4.12** Asymptotic expected variance of an open system of initially 5 agents subject to arrivals, departures and gossip interactions, with  $\lambda_a = \lambda_d = 1$ , for several values of  $\lambda_g$ , with respect to the expected number of gossip interactions experienced by an agent before leaving the system  $\gamma = \lambda_g/\lambda_d$ . The blue dotted line shows simulated results, whereas the red plain line shows the bound (4.46) from Theorem 4.17, the yellow dashed line shows the bound (4.52) from Theorem 4.19, and the purple dash-dotted line shows the bound obtained from solving a finite-dimensional approximation of the dual optimization problem (4.53).

The figure shows that the different bounds are rather conservative for small values of  $\gamma$  (especially (4.46) which diverges as  $\gamma \rightarrow 0$ ). This is however consistent with the very chaotic setting it describes, as agents are then expected to perform very few interactions before leaving the system. More generally, some conservatism is due to Lemma 4.2, from which only an upper bound on the impact of arrivals on the variance can be obtained. This conservatism is thus expected to be eliminated by considering the descriptors directly, similarly to what was done in Section 4.3.1, as an equality is then available. Observe that it is particularly strong when the system size

$n(t)$  is small.

Moreover, since (4.52), the result of Theorem 4.19, follows from using a sequence  $(z_j)$  which is suboptimal in the sense of the optimization problem (4.53), it also inherently generates additional conservatism with respect to the optimal sequence, numerically approximated in the figure. Nonetheless, it remains close to the “optimal bound” in that sense, especially as soon as  $\gamma \approx 5$ , and they both accurately approximate the actual asymptotic variance as soon as  $\gamma \approx 10$ . Finally, the bound (4.46) from Theorem 4.17 also adds conservatism with respect to the other ones, essentially following from the fact that its derivation is rather straightforward. Yet, its derivation builds on Theorem 4.15, which gives intuition on the behavior of the bound in transient regime as well.

## 4.4 Discussion and conclusions

In this chapter, we analyzed how arrivals and departures impact the representation of open multi-agent systems, and thus makes their analysis challenging. In particular, such systems are subject to permanent variations of their dimension, and are poised to never converge. We have shown that despite these challenges, such systems can be studied in terms of specific scale-independent quantities – the descriptors of the system – whose evolution is governed by a fixed-size dynamical system. We have then applied this methodology to analyze the behavior of all-to-all pairwise gossip interactions in open systems of both fixed and variable sizes. In particular, we have shown that the expected evolution of the two first moments of the system can be characterized exactly, thus allowing monitoring the variance as well. The application of our methodology to directly track the variance was also exposed as an alternative, typically for more complex settings, resulting in the derivation of upper bounds on its evolution

Our main contribution relies on the use of continuous-time tools taken from Markov theory to formalize the descriptor-based approach initially introduced in [HM16, HM17]. To the best of our knowledge, their use was never considered so far to handle open systems, yet they appear to be particularly appropriate regarding the challenges related to the varying size of open systems. In fact, the methodology we present in this chapter seems to strongly relate with the notion of JETS [Son08], and a natural extension might be to transpose it to that framework as well.

Moreover, our methodology can be adapted to more general settings, with different definitions of arrivals and departures, or different types of interactions between the agents. In that sense, it aims at being a useful tool to represent and analyze open multi-agent systems in general, provided that it is adapted accordingly to suit the problem formulation.

Typically, a natural continuity of this work is its application to interactions restricted to a graph. The challenge is then to find an appropriate set of descriptors to exploit the graph topology. As an example, the case of complete bipartite graphs illustrated in Fig. 4.13 has been considered in [Oli18]: The set of agents  $\mathcal{V}$  is divided into two disjoint partitions  $A$  and  $B$ , such that  $\mathcal{V} = A \cup B$  with  $A \cap B = \emptyset$ . In that case, it was shown that an appropriate choice of descriptors for monitoring the variance consists in the two first moments of each partition and the covariance between them:

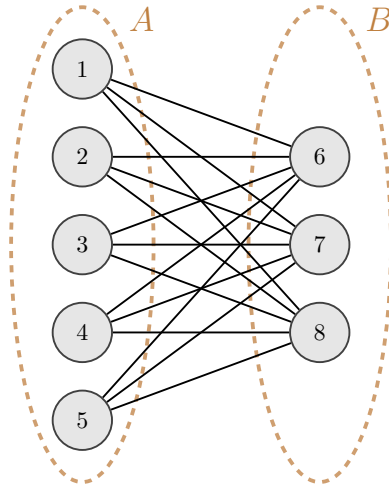
$$\mathbf{X} = \left[ \overline{x_A^2}(t) \quad \overline{x_B^2}(t) \quad \overline{x_A^2}(t) \quad \overline{x_B^2}(t) \quad \overline{x_A x_B}(t) \right]^\top. \quad (4.55)$$

In that specific setting, following only the two first moments of the whole system, such as in this chapter, cannot be expected to yield tight results. This observation more generally applies to general network topologies corresponding to a specific Laplacian matrix  $L$  (see Definition 2.3); One possible choice for the descriptors could directly involve it by considering, *e.g.*,  $x^\top Lx$ . Hence, a shift in the choice of the descriptors is necessary, and consequently the effect of each type of transition must be redefined accordingly. Still, the global methodology remains the same starting from the appropriate definition of the descriptors and their respective evolution.

Similarly, another natural extension of the results of this chapter is to consider other interaction rules, such as the Push-Sum algorithm introduced in Section 3.2. As a reminder, this algorithm relies on two state variables for each agent  $i$ ,  $x_i$  and  $w_i$ , whose dynamics is defined over directed exchanges such that when  $i$  sends information to  $j$  there holds

$$\begin{aligned} x_i^+ &= \frac{x_i}{2}; & x_j^+ &= \frac{x_i + x_j}{2}; \\ w_i^+ &= \frac{w_i}{2}; & w_j^+ &= \frac{w_i + w_j}{2}. \end{aligned}$$

In that case, one would probably need to follow all  $x_i$  and  $w_i$  as a state vector. Moreover, one would need to redefine the events to model directed exchanges. Those change would then also imply redefining proper de-



**Fig. 4.13** Complete bipartite graphs of 8 agents, where  $A = \{1,2,3,4,5\}$  and  $B = \{6,7,8\}$  constitute the partitions of the system. The agents of  $A$  only interact with those of  $B$  and conversely.

scriptors in terms of the new state, and their evolution with respect to the events. The methodology would then remain conceptually the same, yet the adaptation might be rather challenging to perform in practice.

The observations above more generally apply to a wide variety of variations of the problem studied in this chapter, and even to a larger range of problems. In that sense, the methodology proposed here constitutes a first step towards the efficient representation of open multi-agent systems in general. Its extension to complex problems or settings might however not be easy in practice. Typically, more complex settings require the addition of relevant descriptors, and hence of complexity, to obtain tight results, as illustrated throughout this chapter. Nevertheless, as we saw with the analysis of the expected variance for systems of varying size in Section 4.3.3, upper bounds can always be obtained in general. Finally, some additional challenges are left untackled in this chapter, and deserve further investigation in the future, such as establishing the probability distribution of our descriptors instead of only their expected value, or the introduction of noise and delay in the analysis

# Appendix of Chapter 4

## 4.A Proofs of Lemmas 4.1 to 4.6 (Effect of events)

We now provide the proofs of Lemmas 4.1 to 4.6 which characterize the evolution of the descriptors in expectation with each type of event. We provide the proof of each lemma after reminding its statement.

### 4.A.1 Proof of Lemma 4.1 (Gossip step)

The results of Lemma 4.1 read:

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Gos}_n] = \begin{pmatrix} 1 & 0 \\ \frac{1}{n} & 1 - \frac{1}{n} \end{pmatrix} \mathbf{X}; \quad (4.56)$$

$$\mathbb{E} [\text{Var}(\mathbf{x}^+) | \text{Var}(\mathbf{x}), \text{Gos}_n] = \left(1 - \frac{1}{n}\right) \text{Var}(\mathbf{x}). \quad (4.57)$$

*Proof.* Let us first fix the nodes  $i, j$  involved in the gossip step. Firstly, observe that

$$x_i^+ + x_j^+ = 2 \frac{x_i + x_j}{2} = x_i + x_j,$$

and that  $x_k^+ = x_k$  for all  $k \neq i, j$ . Hence  $\bar{x}^+ = \bar{x}$ , which establishes the first line of (4.56). Secondly, since  $x_k = x_k^+$  for every  $k \neq i, j$ , there holds

$$\begin{aligned} \overline{x^2}^+ &= \frac{1}{n} \sum_{k=1}^n (x_k^+)^2 = \overline{x^2} + \frac{1}{n} \left( 2 \left( \frac{x_i + x_j}{2} \right)^2 - x_i^2 - x_j^2 \right) \\ &= \overline{x^2} + \frac{1}{n} \left( x_i x_j - \frac{1}{2} x_i^2 - \frac{1}{2} x_j^2 \right). \end{aligned} \quad (4.58)$$

## 4 | Representing open systems: the case of gossiping

Observe that

$$\mathbb{E} [x_i^2 | x] = \mathbb{E} [x_j^2 | x] = \bar{x}^2 \quad \text{and} \quad \mathbb{E} [x_i x_j | x] = \bar{x}^2$$

Taking the expectation with respect to  $i$  and  $j$  in (4.58) thus yields

$$\mathbb{E} [\bar{x}^{2+} | x] = \left(1 - \frac{1}{n}\right) \bar{x}^2 + \frac{1}{n} \bar{x}^2,$$

from which the second line of (4.56) follows. The result on the variance follows from computing

$$\begin{aligned} \mathbb{E} [\text{Var}(x^+) | \mathbf{X}, \text{Gos}_n] &= (-1 \quad 1) \mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Gos}_n] \\ &= \left(1 - \frac{1}{n}\right) (-1 \quad 1) \mathbf{X} = \left(1 - \frac{1}{n}\right) \text{Var}(x), \end{aligned}$$

which yields (4.57), and concludes the proof.  $\square$

### 4.A.2 Proof of Lemma 4.2 (Arrival)

The result of Lemma 4.2 reads:

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Arr}_n] = \frac{n}{n+1} \begin{pmatrix} \frac{n}{n+1} & 0 \\ 0 & 1 \end{pmatrix} \mathbf{X} + \begin{pmatrix} \frac{1}{(n+1)^2} \\ \frac{1}{n+1} \end{pmatrix} \sigma^2. \quad (4.59)$$

*Proof.* We label  $n+1$  the arriving agent for simplicity, so that  $x_k^+ = x_k$  for all  $k \leq n$ . We begin by computing the new average :

$$\bar{x}^+ = \frac{1}{n+1} \left( x_{n+1}^+ + \sum_{k=1}^n x_k \right) = \frac{n}{n+1} \bar{x} + \frac{1}{n+1} x_{n+1}^+. \quad (4.60)$$

Since  $\mathbb{E} x_{n+1}^+ = 0$ , we have

$$\mathbb{E} [\bar{x}^+ | x] = \frac{n}{n+1} \bar{x}.$$

By exactly the same reasoning but using  $\mathbb{E} [(x_{n+1}^+)^2] = \sigma^2$  we also obtain

$$\mathbb{E} [\bar{x}^{2+} | x] = \frac{n}{n+1} \bar{x}^2 + \frac{1}{n+1} \sigma^2, \quad (4.61)$$

from which the second line of (4.59) follows. Turning to the first line, we

obtain from (4.60)

$$\begin{aligned}\mathbb{E} \left[ (\bar{x}^2)^+ | x \right] &= \frac{1}{(n+1)^2} \left( n^2 \bar{x}^2 + n \bar{x} \mathbb{E} x_{n+1}^+ + \mathbb{E} \left[ (x_{n+1}^+)^2 \right] \right) \\ &= \frac{n^2}{(n+1)^2} \bar{x}^2 + 0 + \frac{1}{(n+1)^2} \sigma^2.\end{aligned}$$

□

### 4.A.3 Proof of Lemma 4.3 (Random departure)

The results of Lemma 4.3 read:

$$\mathbb{E} [X^+ | X, \text{Dep}_n] = \begin{pmatrix} 1 - \frac{1}{(n-1)^2} & \frac{1}{(n-1)^2} \\ 0 & 1 \end{pmatrix} X; \quad (4.62)$$

$$\mathbb{E} [\text{Var}(x^+) | \text{Var}(x), \text{Dep}_n] = \left( 1 - \frac{1}{(n-1)^2} \right) \text{Var}(x). \quad (4.63)$$

*Proof.* Let  $j$  be the randomly selected agent that leaves the system. It follows

$$\bar{x}^+ = \frac{1}{n-1} \left( \left( \sum_{k=1}^n x_k \right) - x_j \right) = \frac{1}{n-1} (n\bar{x} - x_j). \quad (4.64)$$

By exactly the same reasoning, there holds

$$\bar{x}^{2+} = \frac{1}{n-1} (n\bar{x}^2 - x_j^2).$$

Since  $j$  is randomly selected,  $\mathbb{E}(x_j^2 | x) = \bar{x}^2$ . Hence,

$$\mathbb{E} [\bar{x}^{2+} | x] = \frac{1}{n-1} (n\mathbb{E} \bar{x}^2 - \mathbb{E} \bar{x}^2) = \mathbb{E} \bar{x}^2,$$

which implies the second line of (4.62). For the first line, taking into account  $\mathbb{E} [x_j | x] = \bar{x}$ , it follows from (4.64) that

$$\begin{aligned}\mathbb{E} ((\bar{x}^2)^+ | x) &= \frac{1}{(n-1)^2} \left( n^2 \bar{x}^2 - 2n\bar{x} \mathbb{E} [x_j | x] + \mathbb{E} [x_j^2 | x] \right) \\ &= \frac{n^2 - 2n}{(n-1)^2} \bar{x}^2 + \frac{1}{(n-1)^2} \bar{x}^2.\end{aligned}$$

## 4 | Representing open systems: the case of gossiping

The result on the variance follows from computing

$$\begin{aligned} \mathbb{E} [\text{Var}(\mathbf{x}^+) | \mathbf{X}, \text{Dep}_n] &= \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Dep}_n] \\ &= \left(1 - \frac{1}{(n-1)^2}\right) \begin{pmatrix} -1 & 1 \end{pmatrix} \mathbf{X} = \left(1 - \frac{1}{(n-1)^2}\right) \text{Var}(\mathbf{x}), \end{aligned}$$

which yields (4.63), and concludes the proof.  $\square$

### 4.A.4 Proof of Lemma 4.4 (Random replacement)

The result of Lemma 4.4 reads:

$$\mathbb{E} [\mathbf{X}^+ | \mathbf{X}, \text{Rep}_n] = \begin{pmatrix} \frac{n-2}{n} & \frac{1}{n^2} \\ 0 & \frac{n-1}{n} \end{pmatrix} \mathbf{X} + \begin{pmatrix} \frac{1}{n^2} \\ \frac{1}{n} \end{pmatrix} \sigma^2. \quad (4.65)$$

*Proof.* The result follows from a combination of Lemmas 4.3 and 4.2, the latter applied to a system of size  $n - 1$  joined by an  $n^{\text{th}}$  agent.  $\square$

### 4.A.5 Proof of Lemma 4.5 (Adversarial departure)

The result of Lemma 4.5 reads:

$$\mathbb{E} [\text{Var}(\mathbf{x}^+) | \text{Var}(\mathbf{x}), \text{Dep}_n^*] \leq \frac{n}{n-1} \text{Var}(\mathbf{x}). \quad (4.66)$$

*Proof.* For  $n$  values  $x_j \in \mathbb{R}$ , let us define the function  $g(y) := \sum_{j=1}^n (x_j - y)^2$ . One can then show that

$$\arg \min_y g(y) = \bar{x} = \frac{1}{n} \sum_{j=1}^n x_j.$$

Hence, an alternative formulation for the variance is given by

$$\text{Var}(\mathbf{x}) = \frac{1}{n} \sum_{j=1}^n (x_j - \bar{x})^2 = \frac{1}{n} g(\bar{x}) = \frac{1}{n} \min_y \sum_{j=1}^n (x_j - y)^2.$$

Without loss of generality, and for the sake of simplicity, let us assume that the agents in the system are labelled from 1 to  $n$ , and that the agent labelled

$n$  is the one leaving at a departure. Then there holds after the departure

$$\begin{aligned} \text{Var}(x^+) &= \frac{1}{n-1} \min_y \sum_{j=1}^{n-1} (x_j - y)^2 \leq \frac{1}{n-1} \sum_{j=1}^{n-1} (x_j - \bar{x})^2 \\ &\leq \frac{1}{n-1} \sum_{j=1}^n (x_j - \bar{x})^2 = \frac{n}{n-1} \text{Var}(x). \end{aligned}$$

The last inequality follows from the nonnegativity of  $(x_n - \bar{x})^2$ , and taking the expectation of the above result concludes the proof.  $\square$

#### 4.A.6 Proof of Lemma 4.6 (Adversarial replacement)

The result of Lemma 4.6 reads:

$$\mathbb{E} [\text{Var}(x^+) | \text{Var}(x), \text{Rep}_n^*] \leq \text{Var}(x) + \frac{\sigma^2}{n}. \quad (4.67)$$

*Proof.* An adversarial replacement  $\text{Rep}_n^*$  consists in the occurrence of an adversarial departure  $\text{Dep}_n^*$  immediately followed by an arrival  $\text{Arr}_n$ , the latter applied on a system of  $n-1$  agents. Let  $\tilde{x}$  denote the state of the system after the adversarial departure. By applying Lemma 4.2 on  $\tilde{x}$  (of dimension  $n-1$ ), which is obtained using 4.5, we have

$$\begin{aligned} \mathbb{E} [\text{Var}(x^+) | \text{Var}(x), \text{Rep}_n^*] &\leq \mathbb{E} \left[ \frac{n-1}{n} \text{Var}(\tilde{x}) + \frac{\sigma^2}{n} | \text{Var}(x), \text{Dep}_n^* \right] \\ &\leq \frac{n-1}{n} \frac{n}{n-1} \text{Var}(x) + \frac{\sigma^2}{n}, \end{aligned}$$

which concludes the proof.  $\square$

## 4.B Proof of Lemma 4.14 (Upper bound on the descriptors)

We now provide the proof of Lemma 4.14, which we remind states:

$$\mathbb{E} [\bar{x}^2(t) | n(t) = j] \leq \frac{1}{j} \sigma^2; \quad \mathbb{E} [\overline{x^2}(t) | n(t) = j] \leq \sigma^2. \quad (4.68)$$

*Proof.* Let  $S$  denote the set of agents in the system at time  $t$ . We first prove

#### 4 | Representing open systems: the case of gossiping

that  $\mathbb{E} [\sum_{i \in S} x_i(t) \mid |S|]^2 \leq |S|\sigma^2$  and  $\mathbb{E} [\sum_{i \in S} x_i(t)^2 \mid |S|] \leq |S|\sigma^2$  conditional to a particular sequence of events  $\omega$ , which is equivalent to considering a fixed sequence. We then obtain the final result by taking its expectation over all possible sequences  $\omega$ .

Let us fix some time  $t \geq 0$  and some sequence of events  $\omega$  starting at time 0 such that the set of agents at time  $t$  is  $\mathcal{V}(t) = S$ , and let  $\mathcal{T}$  be the set of the labels of all the agents that have been in the system at some time between times 0 and  $t$  (hence  $S \subseteq \mathcal{T}$ ). Those are determined by  $\omega$ .

Moreover, let  $t_i^a$  and  $t_i^d$  respectively denote the arrival and departure times of agent  $i$  in the system, and we define the vector  $\xi(s) \in \mathbb{R}^{|\mathcal{T}|}$  for  $0 \leq s \leq t$  such that for all  $i \in \mathcal{T}$ :

$$\xi_i(s) := \begin{cases} x_i(t_i^a) & \text{if } s \leq t_i^a \\ x_i(s) & \text{if } t_i^a \leq s \leq t_i^d \\ x_i(t_i^d) & \text{if } t_i^d \leq s \end{cases}.$$

Arrivals and departures have no direct effect on  $\xi(s)$ , whereas gossip interactions result in its multiplication by a doubly-stochastic matrix. Let  $A$  be the product of all those matrices since time 0, then  $A$  is doubly-stochastic, and there holds  $\xi(t) = A\xi(0)$ . Let  $w_j := \sum_{i \in S} [A]_{i,j}$ , then one has

$$\sum_{i \in S} x_i(t) = \sum_{i \in S, j \in \mathcal{T}} [A]_{i,j} \xi_j(0) = \sum_{j \in \mathcal{T}} w_j \xi_j(0).$$

Since  $A$  is doubly-stochastic, and since  $S \subseteq \mathcal{T}$ , one has  $w_j \in [0, 1]$ , and thus  $w_j^2 \leq w_j$ . Moreover, there holds

$$\sum_{i \in S} [A]_{i,j}^2 = w_j^2 - \sum_{i, k \in S, i \neq k} [A]_{i,j} [A]_{k,j} \leq w_j^2,$$

and hence

$$\begin{aligned} \mathbb{E} \left[ \sum_{i \in S} x_i(t) \right]^2 &= \mathbb{E} \left[ \sum_{j \in \mathcal{T}} w_j \xi_j(0) \right]^2 \\ &= \sum_{j \in \mathcal{T}} w_j^2 \mathbb{E} \left[ \xi_j(0)^2 \right] + \sum_{\substack{j, k \in \mathcal{T} \\ j \neq k}} w_j w_k \mathbb{E} \left[ \xi_j(0) \xi_k(0) \right] \\ &= \sigma^2 \sum_{j \in \mathcal{T}} w_j^2 \leq \sigma^2 \sum_{j \in \mathcal{T}} w_j, \end{aligned} \tag{4.69}$$

$$\begin{aligned} \mathbb{E} \left[ \sum_{i \in S} x_i(t)^2 \right] &= \mathbb{E} \left[ \sum_{i \in S} \left( \sum_{j \in \mathcal{T}} [A]_{i,j} \zeta_j(0) \right)^2 \right] \\ &= \sum_{j \in \mathcal{T}} \sum_{i \in S} [A]_{i,j}^2 \mathbb{E} \left[ \zeta_j(0)^2 \right] \leq \sigma^2 \sum_{j \in \mathcal{T}} w_j. \end{aligned} \quad (4.70)$$

where the absence of correlation between the initial values  $\zeta_j(0)$  were used to nullify the crossed products. Finally, since

$$\sum_{j \in \mathcal{T}} w_j = \sum_{i \in S} \sum_{j \in \mathcal{T}} [A]_{i,j} = |S|,$$

(4.69) and (4.70) become

$$\mathbb{E} \left( \sum_{i \in S} x_i(t) \right)^2 \leq |S| \sigma^2; \quad \mathbb{E} \left( \sum_{i \in S} x_i(t)^2 \right) \leq |S| \sigma^2. \quad (4.71)$$

Assume now that  $\omega$  is a stochastic sequence of events such that  $n(t) = |S|$ , then

$$\mathbb{E} \left[ \left( \sum_{i \in S} x_i(t) \right)^2 \mid \omega, |S| \right] \leq |S| \sigma^2; \quad \mathbb{E} \left[ \left( \sum_{i \in S} x_i(t)^2 \right) \mid \omega, |S| \right] \leq |S| \sigma^2$$

hold from (4.71). Since  $\omega$  is independent of the agents initial values, taking the expectation over all possible  $\omega$  yields

$$\begin{aligned} \mathbb{E} \left[ \left( \sum_{i \in S} x_i(t) \right)^2 \mid |S| \right] &\leq \mathbb{E} \left[ |S| \sigma^2 \mid |S| \right] = |S| \sigma^2; \\ \mathbb{E} \left[ \left( \sum_{i \in S} x_i(t)^2 \right) \mid |S| \right] &\leq \mathbb{E} \left[ |S| \sigma^2 \mid |S| \right] = |S| \sigma^2, \end{aligned}$$

and it directly follows that

$$\begin{aligned} \mathbb{E} \left[ \bar{x}^2 \mid n(t) = j \right] &= \mathbb{E} \left[ \left( \frac{1}{|S|} \sum_{i \in S} x_i(t) \right)^2 \mid |S| = j \right] \leq \frac{1}{j^2} j \sigma^2; \\ \mathbb{E} \left[ \overline{x^2} \mid n(t) = j \right] &= \mathbb{E} \left[ \frac{1}{|S|} \sum_{i \in S} x_i^2(t) \mid |S| = j \right] \leq \frac{1}{j} j \sigma^2, \end{aligned}$$

which concludes the proof.  $\square$

## 4.C Proofs of Propositions 4.8 and 4.9 (Generalization)

We provide the proof of Proposition 4.8, which we remind reads that if

$$\mathbb{E} [f(\mathcal{S}^+) | f(\mathcal{S}), \epsilon] = A_\epsilon f(\mathcal{S}) + b_\epsilon, \quad (4.72)$$

then

$$\frac{d}{dt}(F_j \pi_j) = \sum_{\epsilon: a(\epsilon)=j} \lambda_\epsilon \pi_{s(\epsilon)} (A_\epsilon F_{s(\epsilon)} + b_\epsilon) - \sum_{\epsilon: s(\epsilon)=j} \lambda_\epsilon \pi_j F_j, \quad (4.73)$$

with  $F_j(t) = \mathbb{E} [f(\mathcal{S}(t)) | n(t) = j]$  and  $\pi_j(t) = P[n(t) = j]$ .

Before stating the proof, we briefly remind that the *instantaneous jump rate* of a CTMC  $\mathcal{S}(t)$  defined on the state space  $E$  (Definition 2.18) is the nonnegative measure  $q : E \times \mathcal{E} \rightarrow [0, +\infty)$  which satisfies

$$\mathbb{P}[\mathcal{S}(t + \tau) \in B | \mathcal{S}(t) = z] = q(z, B)\tau + o(\tau). \quad (4.74)$$

Moreover, we remind that the *Infinitesimal Generator Operator* of  $\mathcal{S}(t)$  (Definition 2.19) is the operator  $\mathcal{L}$  defined for any bounded measurable  $f$  as

$$\mathcal{L}f(x) = \lim_{\tau \rightarrow 0} \frac{\mathbb{E} [f(\mathcal{S}(t + \tau)) | \mathcal{S}(t) = x] - f(x)}{\tau}. \quad (4.75)$$

Finally, we remind from Definition 2.19 that

$$\mathcal{L}f(x) = \int (f(y) - f(x))q(x, dy). \quad (4.76)$$

*Proof of Proposition 4.8.* Let  $\chi_a$  be the indicator function such that  $\chi_a(n) = 1$  if  $n = a$  and  $\chi_a(n) = 0$  otherwise, and let  $f_j(\mathcal{S}(t)) = f(\mathcal{S}(t))\chi_j(n(t))$  for some fixed  $j \in \mathbb{N}$ , so that

$$\mathbb{E}[f_j(\mathcal{S}(t))] = \sum_{k \in \mathbb{N}} \mathbb{E} [f_j(\mathcal{S}(t)) | n(t) = k] \pi_k(t) = F_j(t)\pi_j(t).$$

Hence,  $\frac{d}{dt}(F_j(t)\pi_j(t)) = \frac{d}{dt}\mathbb{E}[f_j(\mathcal{S}(t))]$ , and we compute

$$\begin{aligned} \frac{d}{dt}\mathbb{E}[f_j(\mathcal{S}(t))] &= \lim_{\tau \rightarrow 0} \frac{\mathbb{E}[f_j(\mathcal{S}(t+\tau)) - f_j(\mathcal{S}(t))]}{\tau} \\ &= \lim_{\tau \rightarrow 0} \frac{\mathbb{E}[\mathbb{E}[f_j(\mathcal{S}(t+\tau)|\mathcal{S}(t))] - \mathbb{E}[f_j(\mathcal{S}(t))]]}{\tau} \\ &= \mathbb{E}\left[\lim_{\tau \rightarrow 0} \frac{\mathbb{E}[f_j(\mathcal{S}(t+\tau))|\mathcal{S}(t)] - f_j(\mathcal{S}(t))}{\tau}\right] \\ &= \mathbb{E}[\mathcal{L}f_j(\mathcal{S}(t))], \end{aligned}$$

where the two last equalities respectively follow from linearity and from (4.75), with  $\mathcal{L}$  the infinitesimal generator operator associated with  $\mathcal{S}(t)$ . Let  $g_{z,\epsilon}(y)$  denote the probability density function related to the state taken by  $\mathcal{S}(t)$  after an event of  $\epsilon$  given that its state right before was  $z \in E$ , where  $z = (n_z, x_z)$  with  $n_z \in \mathbb{N}$  and  $x_z \in \mathbb{R}^{n_z}$ . There holds

$$q(z, B) = \sum_{\epsilon \in \Xi} \lambda_\epsilon \chi_{s(\epsilon)}(n_z) \int_B g_{z,\epsilon}(y) dy, \quad (4.77)$$

where we remind  $s(\epsilon)$  is the state of  $n(t)$  before the event of  $\epsilon$ . In the sequel we omit the dependence to the time to lighten the notation. Using (4.76) with (4.77) yields

$$\begin{aligned} \mathcal{L}f_j(\mathcal{S}) &= \sum_{\epsilon \in \Xi} \lambda_\epsilon \chi_{s(\epsilon)}(n) \int_E (f_j(y) - f_j(\mathcal{S})) g_{\mathcal{S},\epsilon}(y) dy \\ &= \sum_{\epsilon \in \Xi} \lambda_\epsilon \chi_{s(\epsilon)}(n) (\mathbb{E}[f_j(\mathcal{S}^+) | \mathcal{S}, \epsilon] - f_j(\mathcal{S})). \end{aligned}$$

By taking the expectation over  $\mathcal{S}$ , and after some algebraic manipulations omitted here, we obtain

$$\begin{aligned} \mathbb{E}[\mathcal{L}f_j(\mathcal{S})] &= \sum_{\epsilon \in \Xi} \lambda_\epsilon \pi_{s(\epsilon)}(n) \mathbb{E}[f_j(\mathcal{S}^+) | \epsilon, n = s(\epsilon), n^+ = a(\epsilon)] \\ &\quad - \sum_{\epsilon \in \Xi} \lambda_\epsilon \pi_{s(\epsilon)}(n) \mathbb{E}[f_j(\mathcal{S}) | n = s(\epsilon)]. \end{aligned}$$

Finally, using the definition of  $f_j(\mathcal{S})$  and (4.72), one has that

$$\mathbb{E}[f_j(\mathcal{S}) | n = s(\epsilon)] = \begin{cases} F_{s(\epsilon)} & \text{if } s(\epsilon) = j \\ 0 & \text{otherwise} \end{cases},$$

## 4 | Representing open systems: the case of gossiping

and

$$\mathbb{E} [f_j(\mathcal{S}^+) | \epsilon, n^+ = a(\epsilon), n = s(\epsilon)] = \begin{cases} A_\epsilon F_{s(\epsilon)} + b_\epsilon & \text{if } a(\epsilon) = j \\ 0 & \text{otherwise} \end{cases},$$

which yields the conclusion.  $\square$

We can then prove Proposition 4.9, which reads that if

$$\mathbb{E} [f(\mathcal{S}^+) | f(\mathcal{S}), \epsilon] \leq A_\epsilon f(\mathcal{S}) + b_\epsilon, \quad (4.78)$$

then

$$\frac{d}{dt}(F_j \pi_j) \leq \sum_{\epsilon: a(\epsilon)=j} \lambda_\epsilon \pi_{s(\epsilon)} (A_\epsilon F_{s(\epsilon)} + b_\epsilon) - \sum_{\epsilon: s(\epsilon)=j} \lambda_\epsilon \pi_j F_j, \quad (4.79)$$

*Proof of Proposition 4.9.* The proof follows the exact same development as that of Proposition 4.8, where the inequality (4.78) is used instead of (4.72) in the last step of the proof.  $\square$

### 4.D Fixed points in fixed-size systems: interpretation

In this section, we provide additional details on the interpretation behind the behavior of the fixed point reached by the expected squared mean  $\bar{x}^2$  in open systems of fixed-size described in Section 4.3.1. In particular, we highlight the intuition behind the difference between the cases where extremely many and extremely few replacements take place in the system.

As a reminder, let  $\rho_g = \lambda_g / \lambda_r$  denote the expected number of gossip interactions happening between two replacements, then there respectively holds with infinitely many replacements (*i.e.*,  $\rho_g \rightarrow 0$ ) and vanishingly few replacements (*i.e.*,  $\rho_g \rightarrow \infty$ ):

$$\begin{aligned} \rho_g \rightarrow 0 : & \quad \mathbb{E} \bar{x}^2 |_{eq} \rightarrow \frac{1}{n} \sigma^2; \\ \rho_g \rightarrow \infty : & \quad \mathbb{E} \bar{x}^2 |_{eq} \rightarrow \frac{1}{2n-1} \sigma^2. \end{aligned}$$

**Infinitely many replacements** ( $\rho_g \rightarrow 0$ ): In this setting, interactions between the agents are so infrequent that the information about the agents that left the system disappears forever. Consequently, the system is essentially the same as if  $n$  i.i.d. values were chosen with zero mean and variance  $\sigma^2$ , and it directly follows that

$$\mathbb{E}\bar{x}^2|_{eq} = \mathbb{E} \left[ \left( \frac{\sum_{i=1}^n x_i}{n} \right)^2 \right] = \frac{1}{n^2} \mathbb{E} \left[ \sum_{i=1}^n x_i^2 + \sum_{\substack{i,j=1 \\ i \neq j}}^n x_i x_j \right] = \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n},$$

where we nullify  $\mathbb{E}x_i x_j$  by using the absence of correlation between the agents. Hence, this situation exactly corresponds to a system where no interaction ever happens.

**Infinitesimally few replacements** ( $\rho_g \rightarrow \infty$ ): This case is slightly more elaborate. Let us start with a small example: Consider a system of  $n = 5$  agents, and let  $\bar{x}_0$  be the average of the values of the agents in the system at the initialization of the process. Since infinitely many interactions are expected to happen between two replacements, it is expected that all agents have converged to the value  $\bar{x}_0$  before the first replacement occurs. Hence, when the first replacement occurs, one agent leaves the system, and a new agent with value  $x'$  gets in, so that the new average is

$$\bar{x}_1 = \frac{4}{5}\bar{x}_0 + \frac{1}{5}x'.$$

Once more, since infinitely many interactions happen, all agents are expected to converge to the value  $\bar{x}_1 = \frac{1}{5}x' + \frac{4}{5}\bar{x}_0$  before the second replacement (which introduces the new value  $x''$  in the system), and the new average to which all agents will converge before the third replacement is

$$\bar{x}_2 = \frac{4}{5}\bar{x}_1 + \frac{1}{5}x'' = \frac{1}{5}x'' + \frac{4}{5} \left( \frac{1}{5}x' + \frac{4}{5}\bar{x}_0 \right) = \frac{1}{5}x'' + \frac{4}{25}x' + \frac{16}{25}\bar{x}_0.$$

One can then keep going as long as desired, yielding

$$\bar{x}_k = \frac{1}{5} \left( \left( \frac{4}{5} \right)^0 x^{(k)} + \left( \frac{4}{5} \right)^1 x^{(k-1)} + \left( \frac{4}{5} \right)^2 x^{(k-2)} + \left( \frac{4}{5} \right)^3 x^{(k-3)} + \dots \right),$$

where  $x^{(k)}$  is the  $k$ -th value that joined the system after its initialization.

#### 4 | Representing open systems: the case of gossiping

We now consider the general case of a system of  $n$  agents. Let us denote by  $\hat{x}_s$  the initial value of the  $s^{\text{th}}$  agent that entered the system,  $t_s$  its arrival time, and  $w_s(t)$  its weight in the time-varying average at time  $t$  denoted  $\bar{x}(t)$ , and there holds

$$\bar{x}(t_s) = \sum_{k=1}^s w_k(t_s) \hat{x}_k.$$

At time  $t_s$ , the weight of  $\hat{x}_s$  is exactly  $\frac{1}{n}$ , since it did not impact nor was impacted by any other agent yet. However, since a perfect averaging is expected to happen between two replacements, the weight of  $\hat{x}_{s-1}$  is smaller:  $\hat{x}_{s-1}$  was evenly distributed among the  $n$  agents presently in the system between the arrivals of the  $(s-1)^{\text{th}}$  and  $s^{\text{th}}$  agents, and one of these left the system, so that one computes that  $w_{s-1}(t_s) = \frac{1}{n} \left(1 - \frac{1}{n}\right)$ . Similarly, one verifies that  $w_{s-k}(t_s) = \frac{1}{n} \left(1 - \frac{1}{n}\right)^k$  for any  $k$  such that at least  $k$  agents entered the system since  $t = 0$ . One can then compute

$$\lim_{s \rightarrow \infty} \bar{x}(t_s) = \lim_{s \rightarrow \infty} \sum_{k=0}^s \frac{1}{n} \left(1 - \frac{1}{n}\right)^k \hat{x}_{s-k},$$

and using the fact that  $\mathbb{E}\hat{x}_k = 0$  and  $\mathbb{E}\hat{x}_k^2 = \sigma^2$  for all  $k$ , and using the absence of correlation between the different  $\hat{x}_k$ , it follows that

$$\begin{aligned} \mathbb{E}\bar{x}^2|_{eq} &= \sum_{k=0}^{\infty} \left(\frac{1}{n} \left(1 - \frac{1}{n}\right)^k\right)^2 \mathbb{E}\hat{x}_{s-k}^2 = \frac{\sigma^2}{n^2} \sum_{k=0}^{\infty} \left(\left(1 - \frac{1}{n}\right)^k\right)^2 \\ &= \frac{\sigma^2}{n^2} \frac{1}{1 - \left(1 - \frac{1}{n}\right)^2} = \frac{\sigma^2}{2n - 1}. \end{aligned}$$

This means that in that specific case, information about the agents having left the system remains even after their departure, yet with decreasing importance. As a consequence, the asymptotic expected squared mean is built upon more values than if no interaction ever happens, and hence is smaller.

This development explains the difference between both cases, which follows from the information that is either lost forever with infinitely many replacements, or that remains in the system with a decreasing importance with vanishingly few replacements.

# 5

## Performance limitations for average consensus in OMAS

*The results and analyses of this chapter take their origin in my Master thesis [Mon18]; They were accepted and will appear in IEEE Transactions on Automatic Control (TAC) [MH20], and a preliminary version was presented at the IEEE Conference on Decision and Control (CDC) 2019 [MH19].*

**B**ECAUSE of the ever-changing nature of OMAS, especially in terms of composition and size, algorithms in open systems are poised to never converge. This is even more clear when considering intrinsic objectives (see Section 2.5 for details), as they rely on, and thus vary with, the composition of the system. This has two main implications: On the one hand it means that there always remains some error that algorithms cannot get rid of, and on the other hand it means that convergence is not a suited criterion for evaluating the performance of algorithms. This chapter focuses on these two issues by defining *fundamental performance limitations* for given problems in OMAS. Those take the form of lower bounds on some arbitrarily chosen performance metric, which correspond to the “best possible performance” that can be achieved by any algorithm for a given problem. These bounds thus constitute a relevant quality criterion to measure the performance of algorithms solving that problem, and highlight at the same time the main bottlenecks arising from their analysis.

To this end, we consider the specific *intrinsic average consensus problem* in OMAS of fixed size, *i.e.*, where no arrival nor departure takes place, but only replacements. We then derive lower bounds on the Mean Squared Error (MSE) achievable by any averaging algorithm in that setting. More specifically, we obtain a general lower bound that depends on the properties of the model defining the interactions happening in the system. This bound can then be instantiated for a specific model of interactions, which thus yields a valid performance criterion for any algorithm that can be implemented with that model.

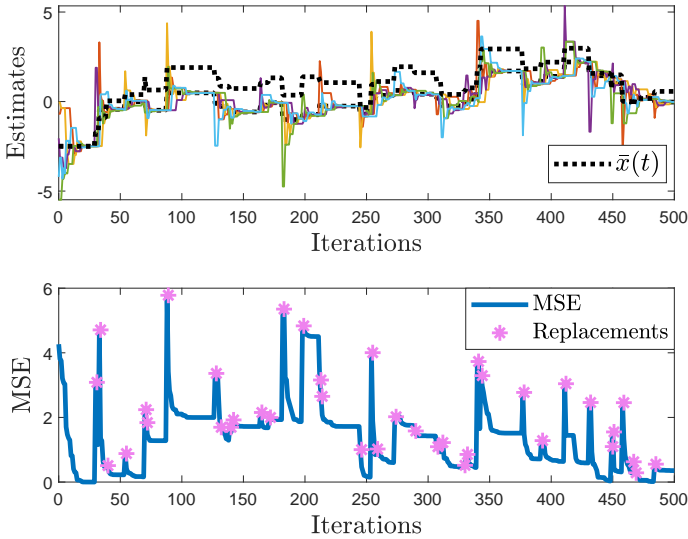
We formulate the problem in Section 5.1, where we describe the system and introduce several relaxations that will make the derivation of lower bounds possible. It also requires the introduction in Section 5.2 of the concept of *knowledge sets*, which quantify the information agents may have access to at some time given some model defining the interactions in the system. From these, we can define a conceptual algorithm whose performance is provably optimal in that model; it thus constitutes a general lower bound on that of any algorithm in that model, which we obtain in Section 5.3. We finally particularize it for two specific interaction models which allow the implementation of the Gossip algorithm in Section 5.4, where we compare the obtained bounds with its actual expected performance.

## 5.1 Problem formulation: performance limitations

The average consensus problem was defined in Section 3.2 (Definition 3.2) for closed systems where the set of agents  $\mathcal{V}$  remains the same all along the process. In that case, the average of the initial values  $x_i$  held by the agents is constant and defined as  $\bar{x} := \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} x_i$ , and consensus to the average is achieved if for all  $i \in \mathcal{V}$

$$\lim_{t \rightarrow \infty} \|x_i(t) - \bar{x}\| = 0. \quad (5.1)$$

This problem can be extended to open systems, *i.e.*, where the set of agents  $\mathcal{V}(t)$  is time-varying. In its intrinsic version, it amounts for the agents  $i \in \mathcal{V}(t)$  to collaboratively estimate the time-varying average of the values initially held by the agents *presently in the system at that time*, namely  $\bar{x}(t) = \frac{1}{|\mathcal{V}(t)|} \sum_{i \in \mathcal{V}(t)} x_i$ . As already mentioned, convergence cannot be achieved for such problem, and hence (5.1) does not make sense anymore



**Fig. 5.1.1** Realization of the Gossip algorithm in a system of 5 agents subject to random replacements, once every 5 gossip steps: (top) shows the evolution of the estimates of the agents (plain line) and of the objective  $\bar{x}(t)$  (dotted line); (bottom) shows the evolution of the MSE, where each purple star highlights a replacement.

to define intrinsic average consensus in open system. In particular, since each arrival and departure of agents changes the system composition and the objective, and since the corresponding information takes time to propagate in the system, and error always remains.

We illustrate this fact in Fig. 5.1.1 for a realization of the Gossip algorithm defined in Section 3.2 in a system subject to replacements. Clearly, convergence is prevented, as each replacement generates a perturbation. This actually goes even further, as even when consensus is temporarily reached, it is generally not on the desired value  $\bar{x}(t)$ . This is due to the intrinsic nature of the problem, which requires outdated information about agents that left the system to be eliminated. This is not supported by the Gossip algorithm, and thus introduces some bias in the agreement state, generating additional error, and thus a bias on the MSE.

Since an error always remains, one can theoretically compute the “best possible error” that could potentially be achieved by an algorithm for that problem. In this section, we detail the *intrinsic average consensus* problem in OMAS of fixed size we consider, and define a criterion (the MSE) to study

the performance of algorithms solving it. We then describe the methodology that will allow us to evaluate the performance of a provably optimal algorithm in our setting, defining the fundamental performance limitations for that problem.

### 5.1.1 System description

We consider a multi-agent system subject to arrivals and departures of agents. The system is initialized at time  $t = 0$  with  $n$  agents labelled from 1 to  $n$ . This composition evolves with arrivals and departures so that the set of agents in the system at time  $t$ , denoted  $\mathcal{V}(t)$ , is time-varying and satisfies  $\mathcal{V}(0) = \{1, \dots, n\}$ . In particular,

- (i) for an arrival happening at time  $t$ , there holds  $\mathcal{V}(t^+) = \mathcal{V}(t^-) \cup \{i'\}$ , with  $i' \notin \mathcal{V}(s)$  for all  $s \leq t$ ;
- (ii) for a departure happening at time  $t$ , then  $\mathcal{V}(t^+) = \mathcal{V}(t^-) \setminus \{i\}$  for some  $i \in \mathcal{V}(t^-)$ .

Each agent  $i$  receives an i.i.d. constant random value  $x_i$  when it joins the system (either at an arrival or at initialization). This value is randomly drawn from some distribution whose mean is assumed to be a known constant, set to zero without loss of generality, and with variance  $\mathbb{E}x_i^2 = \sigma^2$ .

We consider that the evolution of the system is event-based in the sense of Section 2.4, *i.e.*, that it is characterized only by instantaneous modifications triggered by so-called *events*, whose nature defines the occurrence and effect on the system. In this chapter we consider two types of events: *replacements* and *information exchanges between any pair of agents*, as defined below.

1. A *replacement* of an agent  $i$ , denoted  $\mathcal{R}_i$  (for any  $i \in \mathcal{V}(t)$ ), consists in the simultaneous occurrence of the departure of agent  $i$  and the arrival of a new agent  $i'$  that replaces it, so that the system size remains constant. The joining agent  $i'$  receives a new value  $x'_i$  drawn from the same distribution as that of agent  $i$ .
2. An *information exchange* denoted  $\mathcal{I}_{i \rightarrow j}$  (for  $i, j \in \mathcal{V}(t)$ , with  $i \neq j$ ) consists in agent  $i$  sending information to agent  $j$ ; there is no limit to what agent  $i$  can send, and it can possibly send everything it knows. More complex exchanges, *e.g.*, pairwise or all-to-one, can be defined based on  $\mathcal{I}_{i \rightarrow j}$ , depending of the model defining their occurrence.

Since the system solely evolves with the events described above, *its size  $n$  remains constant* all along the process. The goal of the agents in the system is to collaboratively estimate the average of the values held by the agents *presently in the system at time  $t$* , namely

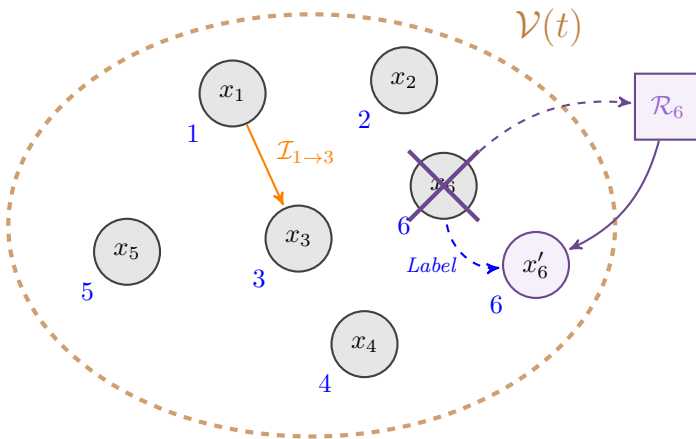
$$\bar{x}(t) = \frac{1}{|\mathcal{V}(t)|} \sum_{i \in \mathcal{V}(t)} x_i = \frac{1}{n} \sum_{i \in \mathcal{V}(t)} x_i. \quad (5.2)$$

### 5.1.2 Problem reformulation

The system description from the previous section can actually be simplified by introducing a relaxation, building on the fixed-size property of the system. We assume that an agent joining the system at a replacement knows the label of the one it replaces, or equivalently that it inherits it. With this new assumption, the replacement of an agent  $i$  can equivalently be reformulated as both:

- (i) the attribution of a new (random) value  $x'_i$  to that agent;
- (ii) the erasure of all other variables it holds.

The system reformulated with that relaxation is illustrated in Fig. 5.1.2.



**Fig. 5.1.2** Representation of the OMAS we consider in this chapter, subject to two types of events: **information exchanges**  $\mathcal{I}_{i \rightarrow j}$  (in orange) and **replacements**  $\mathcal{R}_i$  (in purple), where the joining agent inherits the **label** of the replaced one (in blue).

This reformulation means that the labels of the agents in the system always remain  $\mathcal{V} = \{1, \dots, n\}$ , and the value held by an agent  $i$  can be seen as a piecewise constant time-varying quantity  $x_i(t)$ , that remains constant between replacements. Moreover, the objective (5.2) then becomes

$$\bar{x}(t) := \frac{1}{n} \sum_{i=1}^n x_i(t). \quad (5.3)$$

The results we derive in the remainder of this chapter, which rely on this relaxation, are also valid for any algorithm that would not make use of it. In that sense, this relaxation somehow slightly widens the scope of algorithms for which our results apply.

**Remark 5.1.** *Observe that there is a major difference between this formulation and (closed) systems where each agent would “simply” hold a time-varying signal  $x_i(t)$ : from (ii) above, each variation of  $x_i(t)$  is coupled with the erasure of the memory of agent  $i$ . This means that each time the value of agent  $i$  changes, it cannot rely on any preliminary knowledge it might have gathered previously (consistently with a replacement). An equivalent comparison can be conducted with open systems defined over a fixed superset of agents (see Section 2.2.1 for a description). This difference will have a major impact in the subsequent analysis. More generally, observe that the reformulation we propose here is not directly feasible for systems of variable size*

### 5.1.3 Objective and methodology

Let us use  $y_i(t)$  to denote the estimate of  $\bar{x}(t)$  held by an agent  $i \in \mathcal{V}(t)$  at time  $t$ . As previously illustrated, the usual convergence of  $y_i(t)$  to  $\bar{x}(t)$  cannot be achieved because of the time-varying nature of  $\bar{x}(t)$  and of the time it takes for the agents to react to perturbations. Hence, in order to measure the performance of algorithms computing such estimate in our setting, we make the arbitrary but natural choice to study the MSE in the system as a quality criterion, defined as

$$\mathcal{M}(t) := \frac{1}{n} \sum_{i=1}^n (\bar{x}(t) - y_i(t))^2. \quad (5.4)$$

Our goal is to derive lower bounds on the expected value of the criterion above, which are valid for any algorithm implementable in our setting.

For that purpose, we consider a setting more favorable than what is

typically allowed for such problems, in the sense that agents have access to more information they usually do. In particular, we assume that agents know:

- the system size  $n$  and the labels of the agents  $\mathcal{V} = \{1, \dots, n\}$ ;
- the distribution defining the values  $x_i$  held by the agents;
- the dynamics of the system, *i.e.*, the stochastic properties of the replacements and information exchanges taking place in the system.

Moreover, we assume the agents have access to a universal time, and that they have unlimited memory, so that there is no limit to both what agents can send and receive during information exchanges.

We analyze the MSE, denoted  $\mathcal{M}^*(t)$ , achieved by an estimate which is provably optimal in the setting described above, *i.e.*, such that the MSE  $\mathcal{M}(t)$  achieved by any other estimate that can be computed in the same setting satisfies

$$\mathbb{E}[\mathcal{M}(t)] \geq \mathbb{E}[\mathcal{M}^*(t)]. \quad (5.5)$$

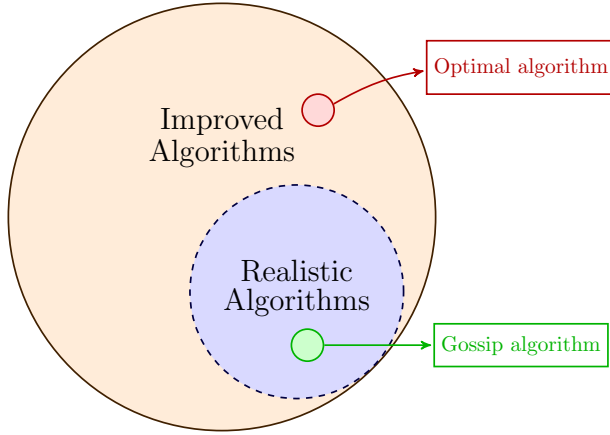
With a slight abuse of language, we call “optimal algorithm” the conceptual (and possibly unrealistic) algorithm which produces this specific optimal estimate. The performance achieved by this optimal algorithm is thus by definition a lower bound on that of any other algorithm that can be implemented in the improved setting, including all (more realistic) algorithms that may not make use of the additional assumptions of Sections 5.1.2 and 5.1.3. More generally, the bounds we derive that way are also valid for any setting allowing less information to be accessible to the agents or subject to more constraints, as illustrated in Fig. 5.1.3.

## 5.2 Knowledge sets

Let us define

$$\mathcal{R} = \bigcup_{i \in \mathcal{V}} \mathcal{R}_i \quad \text{and} \quad \mathcal{I} = \bigcup_{\substack{i, j \in \mathcal{V} \\ i \neq j}} \mathcal{I}_{i \rightarrow j}, \quad (5.6)$$

whose definition builds on that of  $\mathcal{R}_i$  and  $\mathcal{I}_{i \rightarrow j}$  from Section 5.1.1: They respectively correspond to the event of a replacement and an information exchange at the whole system level, independently of the involved



**Fig. 5.1.3** Abstract representation of the set of “realistic” algorithms (in blue) solving the problem as described in Section 5.1.1, such as the Gossip algorithm. This set is included into that of the “improved” algorithms (in orange), which rely on the additional relaxations and assumptions of Sections 5.1.2 and 5.1.3. The performance of the optimal algorithm in that setting is a lower bound with respect to those in that set, thus including the algorithms in the blue set as well.

agent(s). One can then define the *event set* of the system described above as  $\Xi = \mathcal{R} \cup \mathcal{I}$ . From this, one can define an *event sequence* constituted of replacements and information exchanges as follows:

$$\mathcal{H} := \{(t_1, \zeta_1), (t_2, \zeta_2), \dots\}, \quad (5.7)$$

with  $t_1 \leq t_2 \leq \dots$ , and with  $\zeta_k \in \Xi$  for all  $k$  (it is actually related to the *history of the process* in the sense of (2.36) from Section 2.4). We provide in this section a general representation for the information an agent *may have access to* at some time  $t$  given some event sequence as defined above and we call it *knowledge set*.

### 5.2.1 Definitions

We first provide definitions and properties of knowledge sets for deterministic event sequences  $\mathcal{H}$ . A knowledge set can be seen as the result of a function that maps an event sequence to a set containing all the information about the system a given agent  $i$  might have access to at some given time  $t$  based on that sequence. In our setting, the information contained of

that set can be of two natures:

- (i) related to the value  $x$  held by an agent  $j \in \mathcal{V}$  at some time  $s$ , denoted  $(x, j, s)$ ;
- (ii) related to the occurrence of an information exchange  $\mathcal{I}_{j \rightarrow k}$  at some time  $s$  for  $j, k \in \mathcal{V}$ , denoted  $(\mathcal{I}_{j \rightarrow k}, s)$ .

For the sake of simplicity, we define the notion of knowledge set by construction over the events constituting an event sequence as follows. It could however be equivalently defined by characterizing the function mapping the event sequence to the knowledge set instead.

**Definition 5.1** (Knowledge set). *Let  $\mathcal{H}$  be an event sequence in the sense of (5.7). The knowledge set of agent  $i$  at time  $t$  is the set  $\omega_i^{\mathcal{H}}(t)$  initialized as  $\omega_i^{\mathcal{H}}(0) = \{(x_i(0), i, 0)\}$ , and which is updated when events occur as follows:*

- In the event of a replacement  $\mathcal{R}_i$  at time  $t_R$ , then

$$\omega_i^{\mathcal{H}}(t_R^+) = \{(x_i(t_R), i, t_R)\}; \quad (5.8)$$

- In the event of an information exchange  $\mathcal{I}_{i \rightarrow j}$  at time  $t_I$  for  $j \in \mathcal{V}$ , then

$$\omega_i^{\mathcal{H}}(t_I^+) = \omega_i^{\mathcal{H}}(t_I^-) \cup \omega_j^{\mathcal{H}}(t_I^-) \cup \{(x_j(t_I), j, t_I)\} \cup \{(\mathcal{I}_{j \rightarrow i}, t_I)\}. \quad (5.9)$$

The dependence of  $\omega_i^{\mathcal{H}}(t)$  to the values held by the agents during the process is omitted to lighten the notations. The knowledge set as defined above thus represents the information that could be made available to an agent  $i$  throughout the different events that happened according to  $\mathcal{H}$ . Namely, at its arrival in the system, and each time it is replaced, agent  $i$  only knows itself. Then, each time it receives information from an agent  $j$  through an event  $\mathcal{I}_{j \rightarrow i}$ , its knowledge set records everything  $j$  knows at that time (*i.e.*, it assimilates  $j$ 's knowledge set), and keeps track of its value and of the occurrence of  $\mathcal{I}_{j \rightarrow i}$  at that time (those respectively correspond to the last two terms of the union in (5.9)). Observe that information exchanges could be given a label to be recorded during information exchanges as well; this is however omitted since it is not used in the derivation of our results.

Standard results in computer science guarantee that the result of any deterministic algorithm can be obtained based solely on knowledge sets as defined above [YK96a, YK96b, Hen14], and can thus be defined as the result of a deterministic function  $\Phi$  applied on  $\omega_i^{\mathcal{H}}(t)$ :  $y_i(t) = \Phi(\omega_i^{\mathcal{H}}(t))$ .

## 5 | Performance limitations for average consensus in OMAS

We illustrate the way knowledge sets evolve and behave through events in the following example.

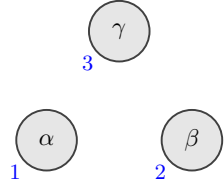
**Example 5.2.** Let us consider a simple system constituted of three agents  $\mathcal{V} = \{1, 2, 3\}$ , initialized at time  $t = 0$  with  $x_1(0) = \alpha$ ,  $x_2(0) = \beta$  and  $x_3(0) = \gamma$ . Up to time  $t$ , the system is subject to 3 events: an information exchange  $\mathcal{I}_{1 \rightarrow 2}$  at time  $t_1$ , then another information exchange  $\mathcal{I}_{2 \rightarrow 3}$  at time  $t_2 > t_1$ , and finally a replacement of agent 2 at time  $t_3 > t_2$ , where it receives a new value  $x_2(t_3^+) = \delta$ . The knowledge sets of the agents evolve through those events as follows.

**Time  $t_0$ : initialization:**

$$\omega_1^{\mathcal{H}}(t_0^+) = \{(\alpha, 1, t_0)\};$$

$$\omega_2^{\mathcal{H}}(t_0^+) = \{(\beta, 2, t_0)\};$$

$$\omega_3^{\mathcal{H}}(t_0^+) = \{(\gamma, 3, t_0)\}.$$

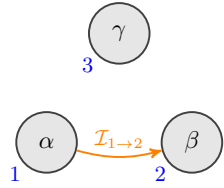


**Time  $t_1$ : information exchange  $\mathcal{I}_{1 \rightarrow 2}$ :**

$$\omega_1^{\mathcal{H}}(t_1^+) = \{(\alpha, 1, t_0)\};$$

$$\omega_2^{\mathcal{H}}(t_1^+) = \{(\beta, 2, t_0), (\alpha, 1, t_0), (\alpha, 1, t_1), (\mathcal{I}_{1 \rightarrow 2}, t_1)\};$$

$$\omega_3^{\mathcal{H}}(t_1^+) = \{(\gamma, 3, t_0)\}.$$

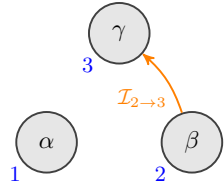


**Time  $t_2$ : information exchange  $\mathcal{I}_{2 \rightarrow 3}$ :**

$$\omega_1^{\mathcal{H}}(t_2^+) = \{(\alpha, 1, t_0)\};$$

$$\omega_2^{\mathcal{H}}(t_2^+) = \{(\beta, 2, t_0), (\alpha, 1, t_0), (\alpha, 1, t_1), (\mathcal{I}_{1 \rightarrow 2}, t_1)\};$$

$$\omega_3^{\mathcal{H}}(t_2^+) = \{(\gamma, 3, t_0), (\beta, 2, t_0), (\alpha, 1, t_0), (\alpha, 1, t_1), (\mathcal{I}_{1 \rightarrow 2}, t_1), (\beta, 2, t_2), (\mathcal{I}_{2 \rightarrow 3}, t_2)\}.$$

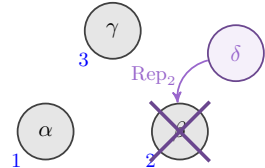


**Time  $t_3$ : replacement  $\text{Rep}_2$**

$$\omega_1^{\mathcal{H}}(t_3^+) = \{(\alpha, 1, t_0)\};$$

$$\omega_2^{\mathcal{H}}(t_3^+) = \{(\delta, 2, t_3)\};$$

$$\omega_3^{\mathcal{H}}(t_3^+) = \{(\gamma, 3, t_0), (\beta, 2, t_0), (\alpha, 1, t_0), (\alpha, 1, t_1), (\mathcal{I}_{1 \rightarrow 2}, t_1), (\beta, 2, t_2), (\mathcal{I}_{2 \rightarrow 3}, t_2)\}.$$



The following lemma directly follows Definition 5.1, and states an important and rather intuitive property of knowledge sets. It means that, at any time  $t$ , the information contained in the knowledge set obtained from a given event sequence would be also gathered if more information exchanges happened. As a consequence, any deterministic function that can be computed based on the information gathered from an event sequence can also be computed if more information exchanges happen.

**Lemma 5.3.** *Let  $\mathcal{H}$  and  $\mathcal{H}'$  be two event sequences in the sense of (5.7) such that they exactly share the same replacement events and satisfy  $\mathcal{H} \subseteq \mathcal{H}'$  (i.e.,  $\mathcal{H}'$  is  $\mathcal{H}$  improved with information exchanges), then for all  $i \in \mathcal{V}$  and all time  $t$  one has*

$$\omega_i^{\mathcal{H}}(t) \subseteq \omega_i^{\mathcal{H}'}(t). \quad (5.10)$$

Consequently, the result of any deterministic function  $\Phi$  applied on  $\omega_i^{\mathcal{H}}(t)$  can also be computed based on  $\omega_i^{\mathcal{H}'}(t)$ .

*Proof.* The proof directly follows the definition by construction of  $\omega_i^{\mathcal{H}}(t)$ :

- At initialization and replacements, then  $\omega_i^{\mathcal{H}}(t) = \omega_i^{\mathcal{H}'}(t)$ ;
- Since all information exchanges in  $\mathcal{H}$  are also in  $\mathcal{H}'$ , then by construction at each exchange of  $\mathcal{H}'$  there holds  $\omega_i^{\mathcal{H}}(t) \subseteq \omega_i^{\mathcal{H}'}(t)$ ,

which concludes the proof. □

Using the definition of knowledge sets, we can now define the *age of the most recent information about an agent  $j$  potentially available to agent  $i$  at time  $t$* , denoted  $T_{j \rightarrow i}^{\mathcal{H}}(t)$  for a given sequence  $\mathcal{H}$ . This quantity will be crucial later to quantitatively assess the relevance of an information in  $\omega_i^{\mathcal{H}}(t)$ .

**Definition 5.4** (Age of the most recent information). *Given a knowledge set  $\omega_i^{\mathcal{H}}(t)$ , the age of the most recent information about agent  $j$  potentially available to agent  $i$  at time  $t$  is defined (if it exists) as follows:*

$$T_{j \rightarrow i}^{\mathcal{H}}(t) := \min \left\{ s : (x_j(t-s), j, t-s) \in \omega_i^{\mathcal{H}}(t) \right\}. \quad (5.11)$$

Observe that  $T_{j \rightarrow i}^{\mathcal{H}}(t)$  as defined above corresponds to the elapsed time since the time instant corresponding to the most recent information about the value held by  $j$  available to  $i$  at time  $t$ . This means that it exists only if there is such information in  $\omega_i^{\mathcal{H}}(t)$ , which might not be the case, e.g., when agent  $i$  gets replaced.

### 5.2.2 Properties for stochastic event sequences

From now on, we will consider stochastic event sequences  $\mathcal{H}^*$ , satisfying the following assumption.

**Assumption 5.5.** *Let  $\mathcal{H}^*$  be a stochastic event sequence constituted of replacements and information exchanges. Then*

1. *the individual replacements of a given agent follow a Poisson clock of rate  $\lambda_r$ ; they are independent of each other, of information exchanges and of the values held by the agents;*
2. *information exchanges happen according to a stochastic interaction model denoted  $*$ ; they are independent of replacements and of the values held by the agents (but not necessarily of each other).*

Assumption 5.5 defines the stochastic evolution of the system. It implies that each agent is replaced  $\lambda_r$  times on average per unit of time, so that  $n\lambda_r$  replacements on average are expected to take place in the whole system per unit of time. We use Poisson clocks because they are standard for modelling memoryless events such as replacements, or more generally arrivals and departures. Moreover, we will see later that this choice implies that the relevance of an information held by an agent  $i$  about another agent  $j$  (i.e., the probability that  $j$  is still present in the system based on agent  $i$ 's knowledge) is then entirely determined by the age of the most recent information it holds about it.

The two following lemmas state useful properties of knowledge sets for stochastic event sequences satisfying Assumption 5.5, that we will use to derive our results.

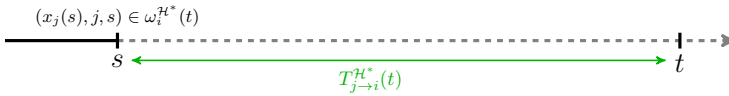
**Lemma 5.6.** *Consider a stochastic event sequence  $\mathcal{H}^*$  satisfying Assumption 5.5. Let  $R_j[s, t]$  denote the (probabilistic) event that agent  $j$  was replaced at least once between times  $s$  and  $t$ . For any time  $s$  such that  $(x_j(s), j, s) \in \omega_i^{\mathcal{H}^*}(t)$ , then*

$$\mathbb{P} \left[ R_j[s, t] | \omega_i^{\mathcal{H}^*}(t) \right] = \begin{cases} 1 & \text{if } R_j[s, t - T_{j \rightarrow i}^{\mathcal{H}^*}(t)] \\ 1 - e^{-\lambda_r T_{j \rightarrow i}^{\mathcal{H}^*}(t)} & \text{otherwise} \end{cases}. \quad (5.12)$$

Lemma 5.6 means that the probability that an agent  $j$  was replaced since the time instant corresponding to an information held by agent  $i$  about it is entirely determined by the age of the most recent information held by  $i$  about  $j$  (which might not be the same information). Let us give some

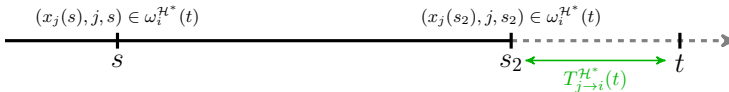
interpretation for (5.12): Let  $(x, j, s)$  be some information about the value held by agent  $j$  at time  $s$ , lying in  $\omega_i^{\mathcal{H}^*}(t)$ , *i.e.*, known by agent  $i$  at time  $t$ . We distinguish two cases.

1. If  $s$  corresponds to the most recent information held by agent  $i$  about  $j$ , as illustrated in Fig. 5.2.5, then agent  $i$  has no information about whether  $j$  was replaced since then. This corresponds to the second case of (5.12), and the probability of  $R_j[s, t]$  given the knowledge of agent  $i$  thus directly depends on  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  following (2.27) from the properties of the Poisson process governing replacements.



**Fig. 5.2.5** Timeline corresponding to the case 1.

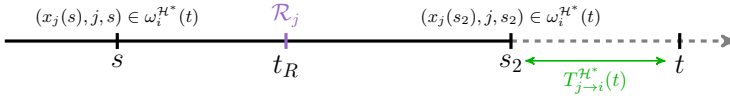
2. If  $s$  is not the most recent information about agent  $j$  held by agent  $i$ , then there exists some time instant  $s_2 > s$  corresponding to that information, and we distinguish again two cases:
  - (a) If there was no replacement of  $j$  between  $s$  and  $s_2$  as illustrated in Fig. 5.2.6, then agent  $i$  knows it, but does not know anything about after time  $s_2$ ; we are once more in the second case of (5.12), and the probability directly depends on  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  (which is not the time spent since  $s$  but since  $s_2$ ).



**Fig. 5.2.6** Timeline corresponding to the case 2(a).

- (b) If there was a replacement of  $j$  between  $s$  and  $s_2$ , as illustrated in Fig. 5.2.7, then agent  $i$  knows it happened for sure because of the information it holds from time  $s_2$ ; this corresponds to the first case of (5.12), and the probability of  $R_j[s, t]$  given the knowledge of agent  $i$  is thus 1.

The probability in (5.12) measures the relevance of the information held by agent  $i$  about agent  $j$  at time  $s$  that is considered. Lemma 5.6 shows that this



**Fig. 5.2.7** Timeline corresponding to the case 2(b).

relevance is entirely characterized by  $T_{j \rightarrow i}^{t^*}(t)$ , and does not depend on the labels of the agents nor the values they hold. It thus prevents situations where the replacement probability can be assessed from something else than the age (e.g., if agents were to inform everyone at their arrival), or where information can be neglected or deleted because of the value or label of the agents. Moreover, it provides a proper expression for the probability that an agent was replaced according to the knowledge of another agent, which will reveal very useful later.

**Lemma 5.7.** *Let  $t_0$  be an arrival time of agent  $i$  in the system (at initialization or through a replacement), then for all times  $s \leq t_0$ , and for any agent  $j$ ,  $x_i(t_0^+)$  and  $\omega_j^{t^*}(s)$  are independent.*

Lemma 5.7 means that at the moment agent  $i$  is replaced, its new value is independent of any prior information held by any agent in the system. It is thus impossible for other agents to estimate that value better than if they had no information at all as long as they do not learn it afterwards. In particular, no information about that value can be deduced from previous values, interaction times, or even from the absence of such information.

**Remark 5.2.** *Lemmas 5.6 and 5.7 are direct consequences of Assumption 5.5 and of the system description in Section 5.1. Hence, we can define alternative situations where some of the assumptions do not hold, and then so do the lemmas. We propose below two examples of such situations, but many more could be defined.*

1. *Assume the time separating the replacements of an agent corresponds to its value. This setting violates Assumption 5.5-1, which states that the values of the agents are independent of their replacement times. In that case, agents can learn each other's replacement time based on their respective values for sure whenever they interact, which contradicts Lemma 5.6.*
2. *Assume the value held by an agent depends on how often it interacted before being replaced. Such setting violates Assumption 5.5-2, which states that the values of the agents are independent of the model defining their interactions. In that case, agents can infer on each other's value based on how often they interacted in the past, which contradicts Lemma 5.7.*

### 5.2.3 Generalization to knowledge set functions

We mentioned in the beginning of this section that knowledge sets can be interpreted as the result of a function that maps a (potentially stochastic) event sequence  $\mathcal{H}^*$  to a set of information  $\omega_i^{\mathcal{H}^*}(t)$ . In that sense, an alternative definition consists in directly defining that function and its properties rather than its result. This alternative definition, which we give below, generalizes Definition 5.1, and depends on an event sequence that can be constituted of other types of events than those considered in this work.

**Definition 5.8.** *Let  $\mathcal{H}^*$  be a (stochastic) event sequence in the sense of (5.7) constituted of events drawn from an event set  $\Xi$  according to a stochastic model denoted  $*$ . A knowledge set function is a function  $\Omega^*$  that maps  $\mathcal{H}^*$  to a set of information  $\omega_i^{\mathcal{H}^*}(t)$  for a given agent  $i \in \mathcal{V}$  and a given time  $t \in \mathbb{R}_+$ . The way the mapping is performed depends on the model  $*$ . Moreover, the set  $\omega_i^{\mathcal{H}^*}(t)$  contains information either of the type  $(x, j, s)$  for the value  $x$  held by agent  $j$  at time  $s$  or  $(\zeta, s)$  with  $\zeta \in \Xi$  for the occurrence of the event  $\zeta$  a time  $s$ , and satisfies both Lemmas 5.6 and 5.7.*

This generalization allows generating the knowledge set as in Definition 5.1, but also other types of similar alternative definitions for that set corresponding to other types of events or other definitions of replacements. We will nevertheless not directly use it in this thesis, as the formulation of Definition 5.1 is sufficient to derive our results and easier to interpret. Still, it is worth describing it since it contributes to generalize our analysis to different settings, and at some extent to different problems, as the analysis itself then remains mostly the same. We describe possible extensions and variations below.

In the remainder of this chapter, we will consider different interaction models defining random interactions as the simultaneous occurrence of specific information exchanges  $\mathcal{I}_{i \rightarrow j}$ . As an example, one could consider a *broadcast* model, where agents send their information to all the other agents at the same time. This can be modelled as the simultaneous occurrence of the effect of  $\mathcal{I}_{i \rightarrow j}$  described in Definition 5.1 in (5.9) for all  $j \in \mathcal{V}$ . Alternatively, one could define a *broadcast* event as

$$\text{Broad}_i := \bigcup_{j \in \mathcal{V}} \mathcal{I}_{i \rightarrow j}, \quad (5.13)$$

and build a new knowledge set by properly defining the corresponding knowledge set function, e.g.,  $\Omega^{\text{Broad}}$ , to handle its effect.

Similarly, we could consider different types of replacement models, *e.g.*, by considering some memory inheritance. This would result in the effect of replacements of agent  $i$  at time  $t_R$ , previously defined in (5.8) from Definition 5.1, becoming

$$\omega_i^{\mathcal{H}}(t_R^+) = \omega_i^{\mathcal{H}}(t_R^-) \cup \{(x_i(t_R^+), i, t_R)\}. \quad (5.14)$$

This actually amounts to differently defining the way replacement events are taken into account when building to the knowledge set. Equivalently, it consists in defining differently the underlying knowledge set function. Alternatively, one can also define new types of events (*e.g.*, “replacements with memory inheritance”) to be handled by a new knowledge set function.

At the end of the day, the notion of *knowledge set function* amounts to defining a mapping rule based on the different types of events happening in the system, which depend themselves on some stochastic model. In that sense, the knowledge set function conceptually allows defining any model with any type of events. Observe however that in order to obtain our results it needs to satisfy the properties of Lemmas 5.6 and 5.7.

### 5.3 Derivation of a general lower bound

Using the notion of knowledge set as described in Definition 5.1, we can now derive a generic lower bound on the expected MSE given by  $\mathbb{E}[\mathcal{M}(t)]$  (where we remind  $\mathcal{M}(t)$  is defined in (5.4)). The lower bound we derive here is valid for any algorithm that can be implemented with a given interaction model, denoted  $*$ , that governs the way information exchanges take place in the system. For that purpose, we still need to introduce one last concept regarding the stochastic properties of knowledge sets  $\omega_i^{\mathcal{H}^*}(t)$  obtained from a given model  $*$ .

#### 5.3.1 Pseudo-CDF and pseudo-PDF

Consider a stochastic event sequence  $\mathcal{H}^*$  satisfying Assumption 5.5, where the information exchanges in  $\mathcal{H}^*$  are obtained using the model  $*$ . Then the knowledge set of agent  $i$  at time  $t$  obtained from that sequence, namely  $\omega_i^{\mathcal{H}^*}(t)$ , is stochastic as well. Consequently, the age of the most recent information about an agent  $j$  held by agent  $i$  at time  $t$  from Definition 5.4, *i.e.*,

$T_{j \rightarrow i}^{\mathcal{H}^*}(t)$ , is a random variable. Furthermore, from Definition 5.4,  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  is well-defined only if there exists some information about the value held by agent  $j$  in  $\omega_i^{\mathcal{H}^*}(t)$ , and so is its CDF (see Definition 2.12).

Hence, we generalize the notion of CDF and we call “pseudo-CDF” of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  the function

$$F_{j \rightarrow i}^{t,*}(s) := \mathbb{P} \left[ T_{j \rightarrow i}^{\mathcal{H}^*}(t) \leq s \right]. \quad (5.15)$$

This function differs from the usual CDF in the sense that it allows

$$\lim_{s \rightarrow \infty} F_{j \rightarrow i}^{t,*}(s) < 1. \quad (5.16)$$

This situation indeed happens in our case if there exists no information about the value held by agent  $j$  in  $\omega_i^{\mathcal{H}^*}(t)$ . Subsequently, we similarly call “pseudo-PDF” of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  the function  $f_{j \rightarrow i}^{t,*}(s)$  satisfying

$$f_{j \rightarrow i}^{t,*}(s) := \frac{d}{ds} F_{j \rightarrow i}^{t,*}(s). \quad (5.17)$$

The definition of the pseudo-PDF is actually linked with that of conditional probability distributions [Par18]. We provide below some interpretation in that sense, which allows discussing the existence of  $f_{j \rightarrow i}^{t,*}(s)$  as well.

Let  $N_j^i(t)$  denote the (probabilistic) event that there is no information about the value held by agent  $j$  in  $\omega_i^{\mathcal{H}^*}(t)$ , and let  $\bar{N}_j^i(t)$  denote its complementary event. We can then define the following conditional CDF:

$$F_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s) := \mathbb{P} \left[ T_{j \rightarrow i}^{\mathcal{H}^*}(t) \leq s | \bar{N}_j^i(t) \right]. \quad (5.18)$$

This function corresponds to the probability that  $T_{j \rightarrow i}^{\mathcal{H}^*}(t) \leq s$  given  $\bar{N}_j^i(t)$ , i.e., given that  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  is well-defined. It is thus a usual CDF satisfying the corresponding properties. Moreover, by definition of the conditional probability in (2.8), we have

$$F_{j \rightarrow i}^{t,*}(s) = \mathbb{P} \left[ \bar{N}_j^i(t) \right] F_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s). \quad (5.19)$$

Therefore, the pseudo-PDF of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  satisfies

$$\begin{aligned} f_{j \rightarrow i}^{t,*}(s) &= \frac{d}{ds} F_{j \rightarrow i}^{t,*}(s) = \mathbb{P} \left[ \bar{N}_j^i(t) \right] \frac{d}{ds} F_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s) \\ &= \mathbb{P} \left[ \bar{N}_j^i(t) \right] f_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s), \end{aligned} \quad (5.20)$$

where  $f_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s) = \frac{d}{ds} F_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s)$  is the PDF corresponding to the conditional CDF  $F_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s)$ . Following the properties of PDFs, the existence of  $f_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s)$  is guaranteed if  $F_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s)$  is absolutely continuous. Hence, since  $f_{j \rightarrow i}^{t,*}(s)$  can be defined based on  $f_{T_{j \rightarrow i}^{\mathcal{H}^*}(t) | \bar{N}_j^i(t)}(s)$ , its existence is guaranteed following the same condition (or equivalently if  $F_{j \rightarrow i}^{t,*}(s)$  is absolutely continuous), which we assume to be the case.

**Remark 5.3.** *An alternative to deal with the issue discussed above would be to assume that there exists a “default” information about each agent in  $\omega_i^{\mathcal{H}^*}(t)$ , each aged of  $+\infty$ . Then a variation of the usual CDF of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  can be defined with a jump at  $+\infty$ . If this definition solves the problem related to the asymptotic value of the CDF, it also requires dealing with partially infinite variables and properly defining that “default” information with respect to our model. Hence, the definition of pseudo-PDFs is preferred for its simplicity of use.*

**Remark 5.4.** *The notion of pseudo-PDF actually captures the way information travels in the system. It thus takes into account any type of restriction on the communications, including the effect of some specific network topology. In fact, in our analysis, the topology exclusively interferes with the interaction model, and is thus completely characterized by the pseudo-PDF. Hence, even though the system description in Section 5.1 technically allows for information exchanges between any pair of agents, the results we obtain in the remainder of this chapter hold for any type of network topology by defining the corresponding probabilities accordingly.*

### 5.3.2 General lower bounds

We can now provide the generic lower bound on  $\mathbb{E} [\mathcal{M}(t)]$ , *i.e.*, that is valid for any interaction model  $*$ , and which is the main result of this chapter. It is stated in the following theorem, whose proof is in Section 5.3.3.

**Theorem 5.9.** *Let  $\omega_i^{\mathcal{H}^*}(t)$  be the stochastic knowledge set of agent  $i$  at time  $t$ , obtained from an interaction model  $*$  under Assumption 5.5. Then, for all  $i \in \mathcal{V}$ ,*

and for any deterministic function  $\Phi_i$  such that the estimate of agent  $i$  at time  $t$  is  $y_i(t) = \Phi_i(\omega_i^{H^*}(t))$  there holds

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{1}{n^3} \sum_{i=1}^n \sum_{j=1}^n \left( 1 - \int_0^t f_{j \rightarrow i}^{t,*}(s) e^{-2\lambda_r s} ds \right) \sigma^2, \quad (5.21)$$

where we remind  $\mathcal{M}(t)$  is the MSE defined in (5.4), and where  $f_{j \rightarrow i}^{t,*}(s)$  denotes the pseudo-PDF defined in (5.17) corresponding to  $T_{j \rightarrow i}^{H^*}(t)$  with the model  $*$ .

This theorem means that (5.21) holds for any deterministic algorithm that can be implemented by any agent  $i$  at time  $t$  based on the information contained in its knowledge set  $\omega_i^{H^*}(t)$ . Observe moreover that the result above is obtained under the assumptions of Sections 5.1.2 and 5.1.3. This means that it holds even if the agents know the system size  $n$ , the replacement rate  $\lambda_r$ , the distribution defining the values of the agents, and  $f_{j \rightarrow i}^{t,*}(s)$  as well as all the parameters it relies on.

By instantiating  $f_{j \rightarrow i}^{t,*}(s)$  according to the considered model, one then obtains a specific lower bound valid for any algorithm implementable with that model. In fact, for a given model  $*$ , the instantiation of the result above with any other pseudo-PDF  $f(s)$  that bounds  $f_{j \rightarrow i}^{t,*}(s)$  (i.e., corresponding to a random variable smaller than  $T_{j \rightarrow i}^{H^*}(t)$  in the usual stochastic order [MS02]) also yields a valid bound for  $*$ . This actually amounts to considering a more favorable probability distribution for the age of the most recent information, and thus essentially consists in an additional relaxation that can be added in the derivation of the bound. See e.g., [MH19] for an example of application of this property.

The following corollary particularizes (5.21) for specific time-invariant interaction models that do not make any distinction between the agents. It moreover specifies that result for settings where we assume the system has been running for a long time.

**Corollary 5.10.** *If there exists a function  $f^*$  such that  $f_{j \rightarrow i}^{t,*}(s) = f^*(s)$  for all  $s \geq 0$ , for any  $i, j \in \mathcal{V}$  and for any time  $t$ , then (5.21) becomes*

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( 1 - \int_0^t f^*(s) e^{-2\lambda_r s} ds \right) \sigma^2. \quad (5.22)$$

In addition, when  $t \rightarrow \infty$ , there holds

$$\liminf_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( 1 - \int_0^\infty f^*(s) e^{-2\lambda_r s} ds \right) \sigma^2. \quad (5.23)$$

Observe that  $\mathcal{M}(t)$  directly depends on the algorithm that is considered. For instance, for a given model  $*$ , one could arbitrarily define an algorithm whose expected performance  $\mathbb{E}[\mathcal{M}(t)]$  is oscillating. In that case  $\lim_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)]$  does not exist, even if a valid lower bound whose limit as  $t \rightarrow \infty$  exists can be obtained with that model. Because of such situations we use  $\liminf$  in Corollary 5.10, and in the results we obtain based on it thereafter.

### 5.3.3 Proof of Theorem 5.9

The remainder of this section is dedicated to proving Theorem 5.9. To this end, we first provide several intermediate results below, whose proofs are given in Appendix 5.A to facilitate their presentation. These results will then allow us to build the proof of Theorem 5.9 step by step.

**Optimal estimate description:** The derivation of our bounds builds on the analysis of a provably optimal estimate, whose performance is a lower bound on that of any other estimate that can be computed with the same information. We first show in the next proposition that this optimal estimate for an agent is obtained by choosing the expected value of  $\bar{x}(t)$  based on the information into its knowledge set. An algorithm computing this estimate would thus achieve optimal performance (with respect to the information accessible to the agents), but might not be implementable with reasonable resources.

**Proposition 5.11.** *Let  $\omega_i^{\mathcal{H}^*}(t)$  be the knowledge set of agent  $i$  at time  $t$  obtained from an interaction model  $*$ , and let us define the estimate*

$$y_i^*(t) := \mathbb{E} \left[ \bar{x}(t) | \omega_i^{\mathcal{H}^*}(t) \right]. \quad (5.24)$$

*For any deterministic function  $\Phi_i$  such that the estimate of agent  $i$  at time  $t$  is  $y_i(t) = \Phi_i(\omega_i^{\mathcal{H}^*}(t))$  for all  $i \in \mathcal{V}$ , there holds*

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \right]. \quad (5.25)$$

*We therefore refer to  $y_i^*(t)$  as the optimal estimate of agent  $i$ .*

*Proof.* The proof is given in Appendix 5.A.1. □

**Decomposing the MSE into independent terms:** The next proposition builds on the description of the optimal estimate from Proposition 5.11 to highlight a convenient property of the system. It shows that, from the definition of  $y_i^*(t)$ , the analysis of  $\mathbb{E}[\mathcal{M}^*(t)]$  reduces to that of the MSE related to the estimation of a single agent's value.

**Proposition 5.12.** *Let  $\omega_i^{\mathcal{H}^*}(t)$  be the knowledge set of agent  $i$  obtained from an interaction model  $*$ . Let  $\mathcal{M}^*(t) := \frac{1}{n} \sum_{i=1}^n (\bar{x}(t) - y_i^*(t))^2$  denote the MSE achieved by the optimal estimate (5.24), then there holds*

$$\mathbb{E}[\mathcal{M}^*(t)] = \frac{1}{n^3} \sum_{i=1}^n \sum_{j=1}^n \mathbb{E} \left[ \left( x_j(t) - \mathbb{E} \left[ x_j(t) | \omega_i^{\mathcal{H}^*}(t) \right] \right)^2 \right]. \quad (5.26)$$

*Proof.* The proof is given in Appendix 5.A.2. □

**Single error analysis:** Proposition 5.12 means that we only need to analyze the MSE related to the estimation of a single agent's value to characterize  $\mathbb{E}[\mathcal{M}^*(t)]$ , and thus to obtain a lower bound on  $\mathbb{E}[\mathcal{M}(t)]$ . Let us define the following intermediate criterion, corresponding to the error made by agent  $i$  when estimating the value of agent  $j$  with the optimal estimate:

$$\mathcal{M}_{j \rightarrow i}^*(t) := \left( x_j(t) - \mathbb{E} \left[ x_j(t) | \omega_i^{\mathcal{H}^*}(t) \right] \right)^2, \quad (5.27)$$

where the dependence of  $\mathcal{M}_{j \rightarrow i}^*(t)$  to  $x_j(t)$  is once more omitted to lighten the notations. We can now state the following proposition, which gives an expression for  $\mathbb{E}[\mathcal{M}_{j \rightarrow i}^*(t)]$ .

**Proposition 5.13.** *Let  $\omega_i^{\mathcal{H}^*}(t)$  be the knowledge set of agent  $i$  obtained from an interaction model  $*$ , then there holds*

$$\mathbb{E}[\mathcal{M}_{j \rightarrow i}^*(t)] = \left( 1 - \int_0^t f_{j \rightarrow i}^{t,*}(s) e^{-2\lambda_r s} ds \right) \sigma^2, \quad (5.28)$$

where we remind  $f_{j \rightarrow i}^{t,*}(s)$  is the pseudo-PDF of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  defined in (5.17).

*Proof.* The proof is given in Appendix 5.A.3. □

**Proof of Theorem 5.9:** We now have all the necessary intermediate results to build the proof of Theorem 5.9, which we give below.

## 5 | Performance limitations for average consensus in OMAS

*Proof of Theorem 5.9.* Firstly, Proposition 5.11 shows that the performance  $\mathcal{M}(t)$  achieved by any algorithm implementable with an interaction model  $*$  is lower bounded in expectation by that achieved by the optimal estimate  $y_i^*(t) = \mathbb{E} \left[ \bar{x}(t) | \omega_i^{\mathcal{H}^*}(t) \right]$ , so that there holds

$$\mathbb{E} [\mathcal{M}(t)] \geq \mathbb{E} [\mathcal{M}^*(t)] = \frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \right]. \quad (5.29)$$

Second, from Proposition 5.12, we have

$$\mathbb{E} [\mathcal{M}^*(t)] = \frac{1}{n^3} \sum_{i=1}^n \sum_{j=1}^n \mathbb{E} \left[ \left( x_j(t) - \mathbb{E} \left[ x_j(t) | \omega_i^{\mathcal{H}^*}(t) \right] \right)^2 \right]. \quad (5.30)$$

Third, Proposition 5.13 shows that

$$\mathbb{E} \left[ \left( x_j(t) - \mathbb{E} \left[ x_j(t) | \omega_i^{\mathcal{H}^*}(t) \right] \right)^2 \right] = \left( 1 - \int_0^\infty f_{j \rightarrow i}^{t,*}(s) e^{-2\lambda_r s} ds \right) \sigma^2. \quad (5.31)$$

Finally, injecting (5.3.3) into (5.30), and combining it with (5.29) concludes the proof.  $\square$

**Remark 5.5.** *Even though the proof of Theorem 5.9 builds on the analysis of a specific algorithm (defined in Proposition 5.24), its result is actually valid for any algorithm that can be implemented based on the same interaction model and under the same assumptions, including unstable or inefficient ones.*

### 5.4 Instantiations for the Gossip algorithm

In this section we instantiate the lower bounds on  $\mathbb{E} [\mathcal{M}(t)]$  obtained in Theorem 5.9 and Corollary 5.10 for two interaction models. The first one serves illustrative purposes, and supposes that the agents receive information from the whole system at once and do not suffer from memory erasures. The second one consists in all-to-all pairwise interactions. We then show that both models allow implementing the Gossip algorithm defined in Section 3.2, so that they provide valid bounds for its performance. We finally give comparisons of these to illustrate how our results can be used.

## 5.4.1 Ping updates

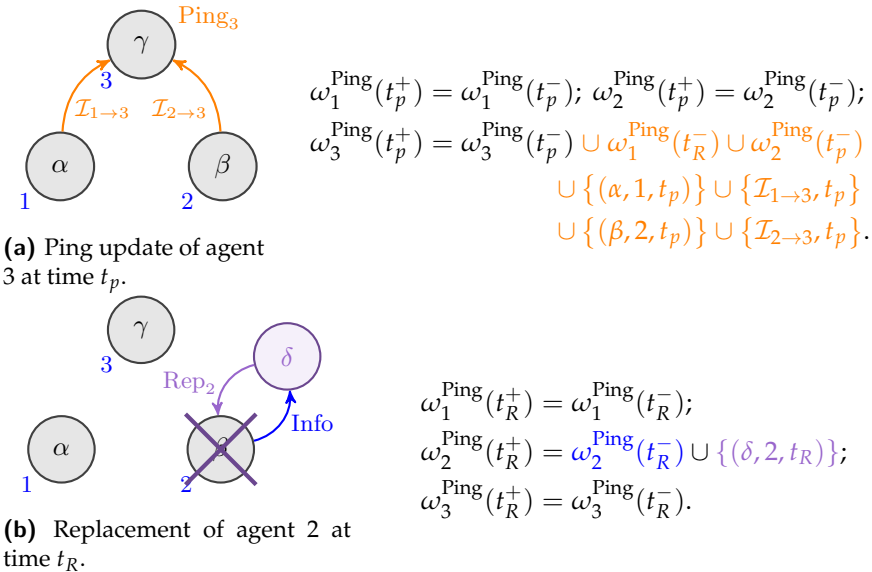
We first consider the following interaction model, which defines the way information exchanges  $\mathcal{I}_{j \rightarrow i}$  take place.

**Definition 5.14** (Ping model). *The Ping interaction model is defined as follows:*

- (1) Each agent  $i \in \mathcal{V}$  receives information about all the other agents in the system at random times, defined by an individual Poisson clock of rate  $\lambda_p$ . This corresponds to the simultaneous occurrence of  $\mathcal{I}_{j \rightarrow i}$  for all  $j \neq i$ , and is called a “Ping update” of agent  $i$ , denoted  $\text{Ping}_i$ ;
- (2) During a replacement, the joining agent inherits all the information held by the agent being replaced.

We use  $\omega_i^{\text{Ping}}(t)$  to denote the knowledge set held by an agent  $i$  at time  $t$ , corresponding to a stochastic event sequence  $\mathcal{H}^{\text{Ping}}$  obtained with the Ping model.

We illustrate in Fig. 5.4.1 a possible evolution upon Ping updates and replacements of the knowledge sets obtained with the Ping model for a system of three agents.



**Fig. 5.4.1** Evolution of the knowledge sets held by the agents subject to (a) a Ping update and (b) a replacement in a system of three agents with the Ping model.

Note that the Ping model requires a slight adaptation of Definition 5.1 to allow memory inheritance during replacements. For a replacement of agent  $i$  taking place at time  $t_R$ , such modification would read

$$\omega_i^{\mathcal{H}^*}(t_R^+) = \omega_i^{\mathcal{H}^*}(t_R^-) \cup \{x_i(t_R^+), i, t_R\}. \quad (5.32)$$

One can verify that our results and arguments from Section 5.3 still hold in that case. More generally, this adaptation amounts to the use of another knowledge set function (in the sense of Section 5.2.3) which handles memory inheritance and Ping updates as events, and which satisfies the necessary properties for our results to hold.

Observe moreover that no assumption is made on the dependence or independence of the Poisson clocks driving the Ping updates of the different agents; the results happen to be independent of such assumptions (in that sense it is actually a class of models rather than a model). Introducing dependencies will reveal useful, as *e.g.*, having pairs of agents performing simultaneous Ping updates provides a relaxation for pairwise interactions, and allows for instance implementing the Gossip algorithm.

*Pseudo-PDF*

In order to apply Theorem 5.9 and obtain a valid bound for the Ping model, we first need to properly characterize the pseudo-PDF  $f_{j \rightarrow i}^{t, \text{Ping}}(s)$ , in the sense of (5.17), corresponding to that model.

**Proposition 5.15.** *With the Ping model, the random variable  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}}}(t)$  admits the following pseudo-PDF for any  $i \neq j$ :*

$$f_{j \rightarrow i}^{t, \text{Ping}}(s) = \lambda_p e^{-\lambda_p s}. \quad (5.33)$$

*Proof.* Remember that  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}}}(t)$  corresponds to the age of the most recent information about the value of agent  $j$  held by agent  $i$  at time  $t$ . With the Ping model, since information is never lost, even at replacements, this age corresponds to the elapsed time since the last Ping update performed by agent  $i$ . Since these are governed by a Poisson clock of rate  $\lambda_p$ , one finds

$$F_{j \rightarrow i}^{t, \text{Ping}}(s) = \mathbb{P} \left[ T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}}}(t) \leq s \right] = 1 - e^{-\lambda_p s}.$$

The conclusion thus follows from  $f_{j \rightarrow i}^{t, \text{Ping}}(s) = \frac{d}{ds} F_{j \rightarrow i}^{t, \text{Ping}}(s)$ . □

### Main bounds for the Ping model

Since the pseudo-PDF  $f_{j \rightarrow i}^{t, \text{Ping}}(s)$  obtained in Proposition 5.15 does not depend on the involved agents  $i$  and  $j$  nor on the time  $t$ , the lower bound on  $\mathbb{E}[\mathcal{M}(t)]$  for that model can be obtained by directly applying Corollary 5.10. This results in the following theorem, which means that the performance of any algorithm that can be implemented based on the Ping interaction model satisfies (5.34).

**Theorem 5.16.** *For all  $i \in \mathcal{V}$ , let  $y_i(t) = \Phi_i(\omega_i^{\text{Ping}}(t))$  be the result of a deterministic function  $\Phi_i$  with the Ping model, then there holds*

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( \frac{1}{1 + \frac{1}{2} \frac{\lambda_p}{\lambda_r}} + \frac{e^{-(\lambda_p + 2\lambda_r)t}}{1 + 2 \frac{\lambda_r}{\lambda_p}} \right) \sigma^2. \quad (5.34)$$

In particular, we have

$$\liminf_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( \frac{1}{1 + \frac{1}{2} \frac{\lambda_p}{\lambda_r}} \right) \sigma^2. \quad (5.35)$$

*Proof.* The instantiation of  $f^*(s)$  from Corollary 5.10 with  $f_{j \rightarrow i}^{t, \text{Ping}}(s)$  from Proposition 5.15 yields

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( 1 - \int_0^t \lambda_p e^{-\lambda_p s} e^{-2\lambda_r s} ds \right) \sigma^2. \quad (5.36)$$

The first result thus follows from some algebraic manipulations after integrating the expression above. Equation (5.35) then follows either from taking the limit as  $t \rightarrow \infty$  of (5.34), or by directly applying the second result of Corollary 5.10 with  $f_{j \rightarrow i}^{t, \text{Ping}}(s)$ .  $\square$

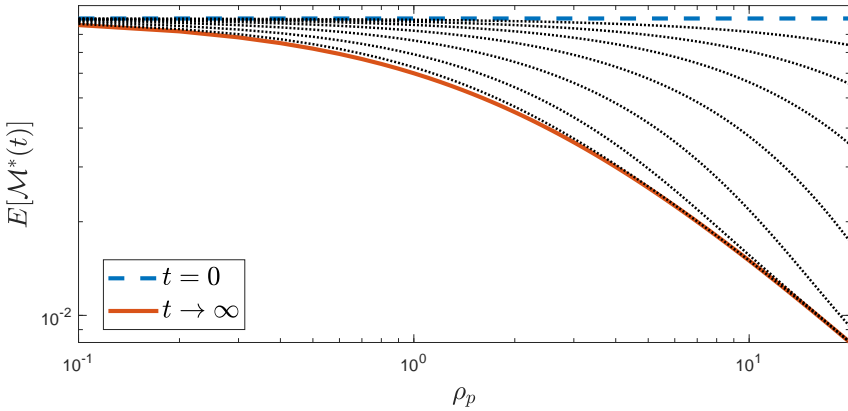
### Illustration and interpretation of the result

Let us define the rate ratio

$$\rho_p = \frac{\lambda_p}{\lambda_r}, \quad (5.37)$$

which corresponds to the average number of Ping updates performed by a given agent before leaving the system at a replacement. Fig. 5.4.2 shows the evolution of the bound (5.34) for 10 agents with respect to  $\rho_p$  at different time instants from the initialization of the system until its convergence to

the time-independent bound (5.35) as  $t \rightarrow \infty$ .



**Fig. 5.4.2** Time-dependent lower bound (5.34) on  $\mathbb{E}[\mathcal{M}(t)]$  for the Ping model with respect to  $\rho_p$  for a system of  $n = 10$  agents with  $\sigma^2 = 1$ : The dashed blue line is the initial MSE (at  $t = 0$ ) and the plain red line is the asymptotic MSE for  $t \rightarrow \infty$  from (5.35); the dotted black lines are bounds obtained at some times in between.

At the initialization of the system (*i.e.*, when  $t = 0$ ), the only information agents have access to is their own value, and the MSE is then exactly  $\frac{n-1}{n^2}\sigma^2$  no matter  $\rho_p$ , consistently with a system constituted of  $n$  i.i.d. values. As time starts passing, the expected MSE starts decaying more or less depending on  $\rho_p$ , until it eventually converges to the time-independent bound as  $t \rightarrow \infty$ . When communications are rare (*i.e.*,  $\rho_p \rightarrow 0$ ), then the MSE remains close to  $\frac{n-1}{n^2}\sigma^2$  as the amount of information available to the agents remains mostly the same all along the process. By contrast, the expected MSE tends to decay as the number of communications increases, and in particular when  $\rho_p \rightarrow \infty$  (*i.e.*, when replacements become extremely rare), then the system progressively behaves as a closed system, so that the expected MSE eventually converges to 0.

Interestingly, the only dependency of the bound (5.34) in  $n$  appears in the factor  $\frac{n-1}{n^2}\sigma^2$ , which corresponds to the initial MSE achieved at the initialization of the system (and for  $\rho_p = 0$ ). This means that the effects of the system size and the different rates are decoupled, and the decrease of the expected MSE is characterized solely by the Ping update and replacement rates. In particular, as  $t \rightarrow \infty$ , it is characterized only by  $\rho_p = \lambda_p/\lambda_r$ .

*Extension: memory erasures*

The model considered here is very simple, and mostly serves illustrative purposes. This simplicity partially follows from the fact that memory erasures are neglected in this model from condition (2) of Definition 5.14. Hence, a rather straightforward extension of it amounts to remove this condition, and thus to consider memory erasures at replacements, while keeping the same interaction scheme with the same parameters. Let us call Ping' this variation of the Ping model.

The model Ping' is less permissive than the standard Ping model, and requires recomputing the corresponding pseudo-PDF  $f_{j \rightarrow i}^{t, \text{Ping}'}(s)$ . The proof of the following proposition is given in Appendix 5.B.

**Proposition 5.17.** *With the Ping' model, the random variable  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}'}}(t)$  admits the following pseudo-PDF for any  $i \neq j$ :*

$$f_{j \rightarrow i}^{t, \text{Ping}'}(s) = \lambda_p e^{-(\lambda_p + \lambda_r)s}. \quad (5.38)$$

This new pseudo-PDF thus yields the following result, whose proof is omitted because it is almost identical to that of Theorem 5.16.

**Theorem 5.18.** *For all  $i \in \mathcal{V}$ , let  $y_i(t) = \Phi_i(\omega_i^{\text{Ping}'}(t))$  be the result of a deterministic function  $\Phi_i$  with the Ping' model, then there holds*

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( \frac{1}{1 + \frac{1}{3} \frac{\lambda_p}{\lambda_r}} + \frac{e^{-(\lambda_p + 3\lambda_r)t}}{1 + 3 \frac{\lambda_r}{\lambda_p}} \right) \sigma^2. \quad (5.39)$$

*In particular, we have*

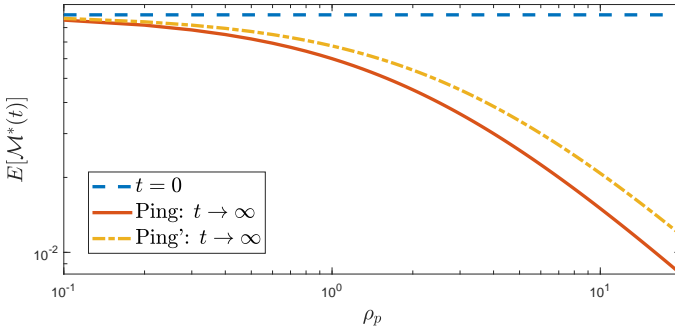
$$\liminf_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( \frac{1}{1 + \frac{1}{3} \frac{\lambda_p}{\lambda_r}} \right) \sigma^2. \quad (5.40)$$

We compare the (asymptotic) bounds (5.35) and (5.40) respectively obtained with the Ping and Ping' models in Fig. 5.4.3. At initialization, both bounds are the same, but with the time passing it appears that the bound obtained with the Ping model is more conservative than that obtained with the Ping' model. This follows the fact that the Ping' model is more restrictive than the Ping one, as information can be lost during replacements in the former but not in the latter. In fact, the Ping model can be seen as a relaxation of the Ping' one, so that  $F_{j \rightarrow i}^{t, \text{Ping}}(s)$  bounds  $F_{j \rightarrow i}^{t, \text{Ping}'}(s)$  (as described

## 5 | Performance limitations for average consensus in OMAS

in Section 5.3.2). Interestingly, the asymptotic bound obtained for the Ping' model with Ping rate  $\lambda'_p$  exactly corresponds to that one would obtain by applying the Ping model with Ping rate  $\lambda_p = \frac{2}{3}\lambda'_p$ . The impact of memory erasures at replacements thus amount to a scaling of the rate at which Ping updates happen in that perspective.

At the end of the day, both bounds are extremely similar, up to that factor, and they thus share most of their interpretation. For that reason, the Ping' bound will not be considered in the upcoming comparisons, but is still presented here for the sake of completeness.



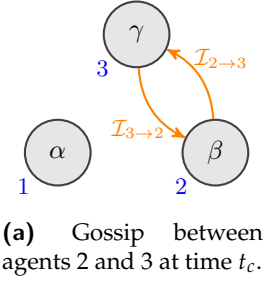
**Fig. 5.4.3** Comparison between the asymptotic bounds (5.35) in plain red line and (5.40) in dotted yellow line, respectively obtained with the Ping and Ping' models in a system of  $n = 10$  agents with  $\sigma^2 = 1$ , with respect to  $\rho_p$ .

### 5.4.2 Pairwise interactions

We now consider a second interaction model to characterize the way information exchanges  $\mathcal{I}_{j \rightarrow i}$  take place, defined as follows.

**Definition 5.19** (Gossip model). *The Gossip interaction model consists in the system being subject to pairwise interactions: Each pair of agents  $(i, j) \in \mathcal{V}^2$  (with  $i \neq j$ ) interacts at random times defined by a Poisson clock of rate  $\frac{1}{n-1}\lambda_c$ , resulting in the simultaneous occurrence of  $\mathcal{I}_{i \rightarrow j}$  and  $\mathcal{I}_{j \rightarrow i}$ . We use  $\omega_i^{\text{Gossip}}(t)$  to denote the knowledge set held by an agent  $i$  at time  $t$ , corresponding to a stochastic event sequence  $\mathcal{H}^{\text{Gossip}}$  obtained with the Gossip model.*

We illustrate in Fig. 5.4.4 a possible evolution of the knowledge sets held by the agents upon a pairwise interaction with the Gossip model for a system of three agents.



$$\begin{aligned}
 \omega_1^{\text{Gossip}}(t_c^+) &= \omega_1^{\text{Gossip}}(t_c^-); \\
 \omega_2^{\text{Gossip}}(t_c^+) &= \omega_2^{\text{Gossip}}(t_c^-) \cup \omega_3^{\text{Gossip}}(t_c^-) \\
 &\quad \cup \{(\gamma, 3, t_c)\} \cup \{I_{3 \rightarrow 2}, t_c\}; \\
 \omega_3^{\text{Gossip}}(t_c^+) &= \omega_3^{\text{Gossip}}(t_c^-) \cup \omega_2^{\text{Gossip}}(t_c^-) \\
 &\quad \cup \{(\beta, 2, t_c)\} \cup \{I_{2 \rightarrow 3}, t_c\}.
 \end{aligned}$$

**Fig. 5.4.4** Evolution of the knowledge sets held by the agents subject to a pairwise interaction in a system of three agents with the Gossip interaction model.

The Gossip model amounts to defining pairwise undirected exchanges of interactions, called *gossips*. The description of the Poisson clocks characterizing the occurrences of gossips means that each agent interacts on average  $\lambda_c$  times per unit of time.

This model actually raises more challenges than the Ping model previously described does. Additionally to the possibly outdated information from unknown replacements already considered with the Ping model, memory erasures as well as the time it takes for information to propagate in the system must be taken into account with the Gossip model.

#### *Pseudo-PDF*

In order to apply Theorem 5.16, we once more need to properly characterize the pseudo-PDF  $f_{j \rightarrow i}^{t, \text{Gossip}}(s)$  corresponding to the Gossip model. This is done in the following proposition, proved in Appendix 5.C.

**Proposition 5.20.** *With the Gossip model, the random variable  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Gossip}}}(t)$  admits the following pseudo-PDF for any  $i \neq j$ :*

$$f_{j \rightarrow i}^{t, \text{Gossip}}(s) = \mathbf{w}^\top A e^{As} \mathbf{e}_1, \quad (5.41)$$

where  $\mathbf{w} \in \mathbb{R}^n$  with  $w_k = \frac{k-1}{n-1}$ , and where  $A \in \mathbb{R}^{n \times n}$  is a tridiagonal matrix with  $[A]_{k,k} = -\left(\frac{k(n-k)}{n-1} \lambda_c + (k-1) \lambda_r\right)$ ,  $[A]_{k,k+1} = k \lambda_r$  and  $[A]_{k+1,k} = \frac{k(n-k)}{n-1} \lambda_c$ .

*Proof.* The proof is given in Appendix 5.C. □

Interestingly, the result of Proposition 5.20 exactly corresponds to the behavior of an SIS infection process (Susceptible-Infected-Susceptible), see e.g., [BCC01]. Namely, the information about the value held by an agent

can be seen as a disease spreading in the system: New infections occur when agents holding that information interact with agents that do not, and healing corresponds to replacements of such agents, whose memory is then erased. In that sense,  $F_{j \rightarrow i}^{t, \text{Gossip}}(s)$  actually corresponds to the probability that an agent  $i$  is infected at time  $t$  by a disease emitted by agent  $j$  since time  $t - s$ .

*Main bounds for the pairwise interactions model*

Since the pseudo-PDF  $f_{j \rightarrow i}^{t, \text{Gossip}}(s)$  does not depend on the agents  $i$  and  $j$  nor on the time  $t$ , the lower bound on  $\mathbb{E}[\mathcal{M}(t)]$  for the Gossip model can directly be obtained by applying Corollary 5.10. This results in the following theorem, which means that the performance of any algorithm that can be implemented based on the Gossip interaction model satisfies (5.42).

**Theorem 5.21.** *For all  $i \in \mathcal{V}$ , let  $y_i(t) = \Phi_i(\omega_i^{\text{Gossip}}(t))$  be the result of a deterministic function  $\Phi_i$  with the Gossip model, then there holds*

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left(1 - \mathbf{w}^\top A (A - 2\lambda_r)^{-1} \left(e^{(A-2\lambda_r)t} - I\right) \mathbf{e}_1\right) \sigma^2, \quad (5.42)$$

where  $\mathbf{w}$  and  $A$  are defined in Proposition 5.20. In particular, there holds

$$\liminf_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left(1 - \mathbf{w}^\top A (2\lambda_r - A)^{-1} \mathbf{e}_1\right) \sigma^2. \quad (5.43)$$

*Proof.* The instantiation of  $f^*(s)$  from Corollary 5.10 with  $f_{j \rightarrow i}^{t, \text{Gossip}}(s)$  from Proposition 5.20 yields

$$\mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left(1 - \int_0^t \mathbf{w}^\top A e^{As} \mathbf{e}_1 e^{-2\lambda_r s} ds\right) \sigma^2. \quad (5.44)$$

Using the commutativity between  $A$  and  $2\lambda_r$ , the integral in the expression above becomes

$$\int_0^t \mathbf{w}^\top A e^{As} \mathbf{e}_1 e^{-2\lambda_r s} ds = \mathbf{w}^\top A \int_0^t e^{(A-2\lambda_r)s} ds \mathbf{e}_1.$$

One then shows that  $A - 2\lambda_r$  is invertible as long as  $\lambda_r \neq 0$ , so that

$$\mathbf{w}^\top A \int_0^t e^{(A-2\lambda_r)s} ds \mathbf{e}_1 = (A - 2\lambda_r)^{-1} \left(e^{(A-2\lambda_r)t} - I\right).$$

Injecting this result into (5.44) then yields (5.42). Equation (5.43) then fol-

lows from either taking the limit as  $t \rightarrow \infty$  of (5.42), or by applying the second result of Corollary 5.10 with  $f_{j \rightarrow i}^{t, \text{Gossip}}(s)$ .  $\square$

Similarly as for the Ping model, let us define the rate ratio

$$\rho_c = \frac{\lambda_c}{\lambda_r}, \quad (5.45)$$

corresponding to the number of pairwise interactions involving a given agent that are expected to happen on average before it gets replaced. In the following corollary, proved in Appendix 5.D, we give an analytic approximation for (5.43) which gets more accurate as  $\rho_c$  grows and as the system size  $n$  decreases. Eventually, it becomes a lower bound on (5.43), and thus a valid lower bound for  $\mathbb{E}[\mathcal{M}(t)]$ .

**Corollary 5.22.** *In the same setting as that of Theorem 5.21, there holds*

$$\liminf_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( \frac{7}{2} + \log\left(\frac{n-2}{2}\right) + h_n \right) \rho_c^{-1} \sigma^2 + \mathcal{O}\left(\frac{n}{\rho_c}\right) \sigma^2, \quad (5.46)$$

where  $h_n$  is some known polynomial in  $\mathcal{O}(n^{-1})$ .

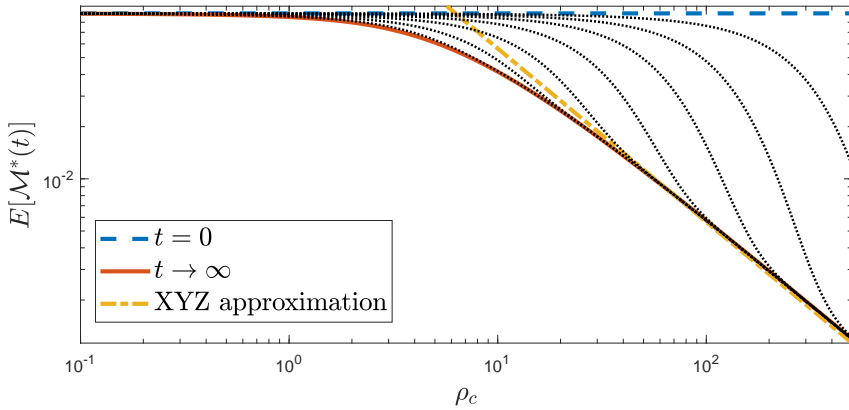
*Proof.* The proof is given in Appendix 5.D.  $\square$

In what follows, we arbitrarily refer to (5.46) as the ‘‘XYZ approximation’’, based on the proof of Corollary 5.22 which relies on the use of three matrices  $X$ ,  $Y$  and  $Z$  to approximate (5.43) and eventually bound it by below.

*Illustration and interpretation of the result*

In Fig. 5.4.5 we show the evolution of the bound (5.42) with respect to  $\rho_c$  in the same setting as in Fig. 5.4.2 (for the Ping model), but for the Gossip model. The figure also shows the result of Corollary 5.22.

The same preliminary observations as for the Ping model can be made regarding the initial MSE being  $\frac{n-1}{n^2} \sigma^2$ , which either remains close to that value as time passes for  $\rho_c \rightarrow 0$  (very few interactions), or decays to zero as  $\rho_c \rightarrow \infty$  (towards a close system). Nevertheless, the bound obtained with the Gossip model differs from the Ping one as the system size  $n$  has here an impact on the decay of the expected MSE as  $\rho_c$  grows. This can be due to the time it takes for information to propagate in the system with the Gossip model (as opposed to the Ping model where the propagation is instantaneous at Ping updates), or at some extent to the memory losses



**Fig. 5.4.5** Time-dependent lower bound (5.42) on  $\mathbb{E}[\mathcal{M}(t)]$  for the Gossip model with respect to  $\rho_c$  for a system of  $n = 10$  agents with  $\sigma^2 = 1$ : The dashed blue line is the initial MSE (at  $t = 0$ ) and the plain red line is the asymptotic MSE for  $t \rightarrow \infty$  from (5.43); the dotted black lines are bounds obtained at some times in between; the yellow dash-dotted line is the XYZ approximation (5.46) from Corollary 5.22.

happening at replacements with the Gossip model only. The latter difference is however likely to be less impactful, regarding the analysis of the Ping' model in the previous section where it was considered alone.

Finally, the XYZ approximation from Corollary 5.22 appears to rather accurately approximate the asymptotic bound on (5.43) as  $\rho_c$  gets large. In particular, it gets satisfyingly accurate starting from  $\rho_c \approx 50$ .

*Relaxation: no memory erasure*

As previously observed while proving Proposition 5.20, the propagation of information in the system with the Gossip model can be assimilated to the propagation of a disease following an SIS infection model, where new infections follow interactions and healing follow replacements. A straightforward variation of this model consists in, similarly to the Ping model, assume that no memory erasure happens at replacements. With the disease analogy, this is directly adapted by assuming it propagates following an SI infection model (Susceptible-Infected) rather than an SIS one. With that alternative model, which we arbitrarily call the "SI" model following its interpretation, both Proposition 5.20 and Theorem 5.21 can be directly adapted by defining the matrix  $A$  according to an SI infection, yielding the following lower bound.

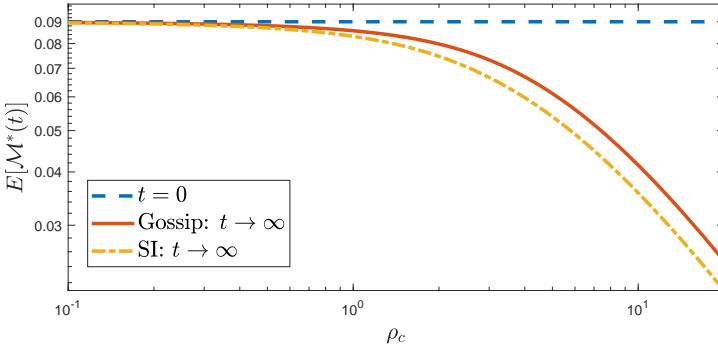
**Theorem 5.23.** For all  $i \in \mathcal{V}$ , let  $y_i(t) = \Phi_i(\omega_i^{\text{SI}}(t))$  be the result of a deterministic function  $\Phi_i$  with the SI model, then there holds

$$\mathbb{E} [\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left(1 - \mathbf{w}^\top \tilde{A} (\tilde{A} - 2\lambda_r)^{-1} \left(e^{(\tilde{A}-2\lambda_r)t} - I\right) \mathbf{e}_1\right) \sigma^2, \quad (5.47)$$

where  $\mathbf{w} \in \mathbb{R}^n$  satisfies  $w_k = \frac{k-1}{n-1}$ , and where  $\tilde{A} \in \mathbb{R}^{n \times n}$  is a bidiagonal matrix satisfying  $[\tilde{A}]_{k,k} = -\frac{k(n-k)}{n-1} \lambda_c$  and  $[\tilde{A}]_{k+1,k} = \frac{k(n-k)}{n-1} \lambda_c$ . In particular, there holds

$$\liminf_{t \rightarrow \infty} \mathbb{E} [\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left(1 - \mathbf{w}^\top \tilde{A} (2\lambda_r - \tilde{A})^{-1} \mathbf{e}_1\right) \sigma^2. \quad (5.48)$$

We omit the proof because it is essentially the same as that of Theorem 5.21 with the alternative matrix  $A$  related to an SI infection process. We compare the (asymptotic) bounds (5.43) and (5.48) in Fig. 5.4.6. The conclusions are similar to those drawn from comparing the Ping and Ping' models in the previous section, as their difference is the same. Typically, the Gossip model is actually included into the SI model, so that it yields a tighter bound as it is more restrictive. Once more, as they both share a similar interpretation, we will focus on the Gossip bound in the upcoming analysis, but show the SI model here for the sake of completeness.



**Fig. 5.4.6** Comparison between the asymptotic bounds (5.43) in plain red line and (5.48) in dotted yellow line, respectively obtained with the Gossip and SI models in a system of  $n = 10$  agents with  $\sigma^2 = 1$ , with respect to  $\rho_c$ .

### 5.4.3 Performance analysis of the Gossip algorithm

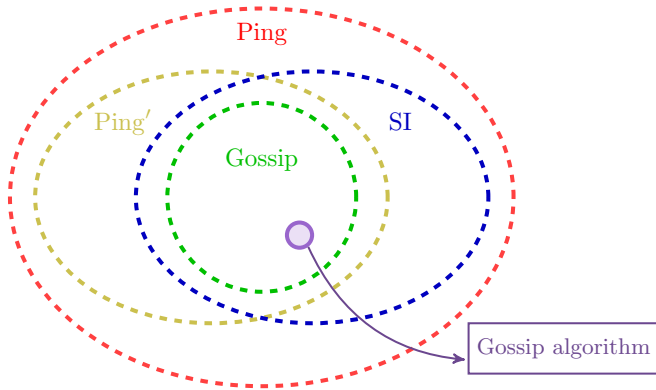
We finally use the bounds obtained so far to analyze the performance of the Gossip algorithm defined in Section 3.2, and which we remind consists in the following:

- when it joins the system, an agent  $i$  chooses its own value  $x_i$  as initial estimate  $y_i(t)$ ;
- when two agents  $i$  and  $j$  interact, they update their respective estimates as  $y_i^+ = y_j^+ = \frac{y_i^- + y_j^-}{2}$ .

The Gossip algorithm is thus defined over pairwise interactions, and can therefore be implemented on  $\omega_i^{\text{Gossip}}(t)$  (and on  $\omega_i^{\text{SI}}(t)$  as well). Actually, since no assumption is made on the dependency between the clocks triggering Ping updates in the Ping model, one could arbitrarily decide that two agents perform a Ping update whenever they interact. This is allowed by Definition 5.14, and hence the Gossip algorithm can be implemented on  $\omega_i^{\text{Ping}}(t)$  (and on  $\omega_i^{\text{Ping}'}$  since it does not consider memory inheritance at replacements). The Ping update rate of an agent would thus correspond to the rate at which it is involved in pairwise interactions:  $\lambda_p = \lambda_c$  (this can be shown using Lemma 5.3). Hence, the same way as previously, we define the common interaction-to-replacement rate ratio as  $\rho = \lambda_c / \lambda_r = \lambda_p / \lambda_r$ .

This means that the bounds obtained with each of these models are valid lower bounds for the performance of the Gossip algorithm. More generally, the Gossip algorithm is directly defined over the Gossip model. Therefore, since all other models are relaxations of the Gossip one, they are all valid for the Gossip algorithm. This is equivalently formulated as observing the any algorithm that can be implemented over the Gossip model can also be implemented over all the other ones, as presented in Fig. 5.4.7.

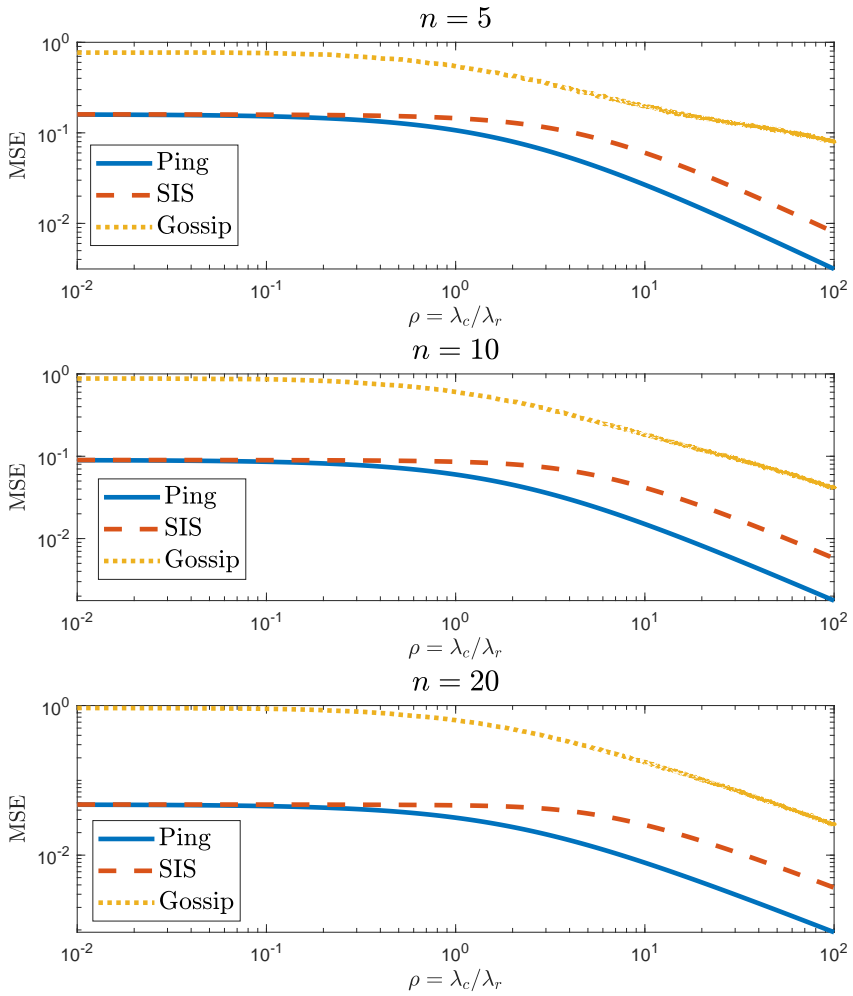
In what follows, we will refer to the bound obtained with the Gossip model as the “SIS bound” to avoid confusion with the performance of the Gossip algorithm. In Fig. 5.4.8, we compare both the asymptotic Ping bound (5.35) from Theorem 5.16 and the asymptotic SIS bound (5.43) from Theorem 5.21 with the expected performance of the Gossip algorithm, respectively with 5, 10 and 20 agents. The figure confirms that both results are valid lower bounds for the Gossip algorithm, and that the SIS bound is tighter than the Ping one (as the Gossip model is more restrictive). Consequently, it is closer to the performance achieved by the Gossip algorithm.



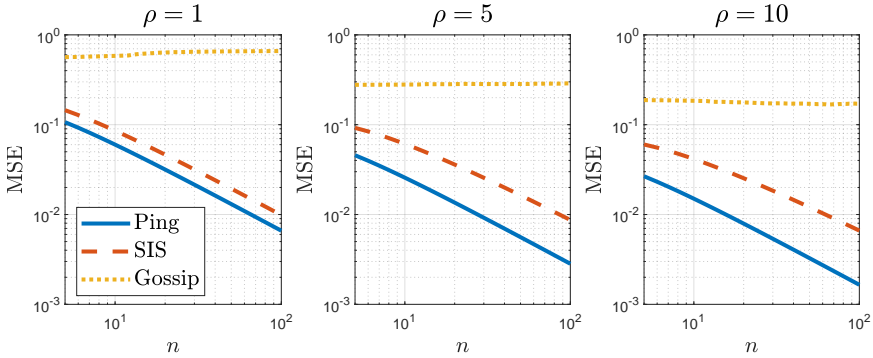
**Fig. 5.4.7** Sets of algorithms that can be implemented with a given model. The Ping model considers memory inheritance and all-to-one interactions, and thus includes the Ping' model that lifts the memory inheritance assumption, and the SI model that restricts to pairwise interactions. Those contain the Gossip model, which does both at the same time, and in with the Gossip algorithm is defined.

Nevertheless, there is still an important quantitative gap between our bounds and the actual performance observed for the simulated Gossip algorithm. Interestingly, the size of the system has barely no impact on the performance of the Gossip algorithm, especially small values of  $\rho$ . In opposition, the bounds tend to decrease as the system size increases. This behavior is shown by Fig. 5.4.9, where the bounds are compared with the performance of the Gossip algorithm with respect to the system size  $n$  for several values of  $\rho$ . This gives insight on how knowing the size of the system (which is assumed to be the case for the bounds, but is not for the Gossip algorithm) can influence the performance of algorithms. Typically, the MSE corresponding to our bounds is scaled based on  $n$ , whereas it has no influence on the MSE of the Gossip algorithm.

Nonetheless, the behavior of the performance of the Gossip algorithm with respect to the rate ratio  $\rho$  is qualitatively well captured by both bounds. This is surprising, especially because the Gossip algorithm is a particularly naive algorithm: It only relies on one variable, and does not make use of any identifier nor any of the information related to the openness of the system or to the distribution of the intrinsic values of the agents, by contrast with our bounds. This thus questions the interest of such parameters in the design of efficient algorithms, and the exact impact they have on their performance.



**Fig. 5.4.8** Expected MSE achieved by the Gossip algorithm (in yellow dotted line) as compared to the asymptotic Ping bound (5.35), in plain blue line, and the asymptotic SIS bound (5.43), in red dashed line, with respect to  $\rho$  for several system sizes  $n$  (5, 10 and 20); the simulation is performed over 200 events and 500 realizations.



**Fig. 5.4.9** Expected MSE achieved by the Gossip algorithm (in yellow dotted line) as compared to the asymptotic Ping bound (5.35), in plain blue line, and the asymptotic SIS bound (5.43), in red dashed line, with respect to  $n$  for several values of  $\rho$ ; the simulation is performed over 500 events and 500 realizations.

## 5.5 Discussion and conclusions

In this chapter, we studied the notion of *performance* of algorithms in open multi-agent systems, which are poised to never converge due to the ever-changing nature of such systems in terms of size, composition and objective. In particular, since convergence cannot be achieved, we investigated the *fundamental performance limitations* of specific problems in open systems, which characterize the “best possible performance” that can be achieved by any algorithm trying to solving them. We focused on the average consensus problem in open systems, and derived lower bounds on the expected MSE that can be achieved by averaging algorithms in open systems of fixed size. We first obtained a general bound that depends on the model defining the interactions in the system, from which we derived specific bounds for all-to-all and all-to-one interactions models. Finally, we compared our bounds with the performance of the Gossip algorithm, which can be implemented on the specific models we considered, and for which our bounds are thus valid.

Along this chapter, different *tools* were introduced in order to define a general methodology. In particular, the derivation of our bounds relied on the analysis of some (virtual) provably *optimal estimate* with respect to the MSE. This required the introduction of *knowledge sets* to quantify the information made potentially available to an agent at some time for

some *interaction model*. Because the system size was fixed in our analysis, the main challenge that was considered in this work was precisely to handle this information, which takes time to propagate and potentially becomes outdated as agents get replaced. This aspect was captured by the notion of *pseudo-PDF*, which allowed the characterization of the validity of information. Hence, another important challenge left untackled here is to handle the potentially variable size of systems subject to decoupled arrivals and departures. This was actually envisioned in Chapter 4, which focused on the *representation* of open systems, and constitutes an important and natural follow-up for the work presented in this chapter.

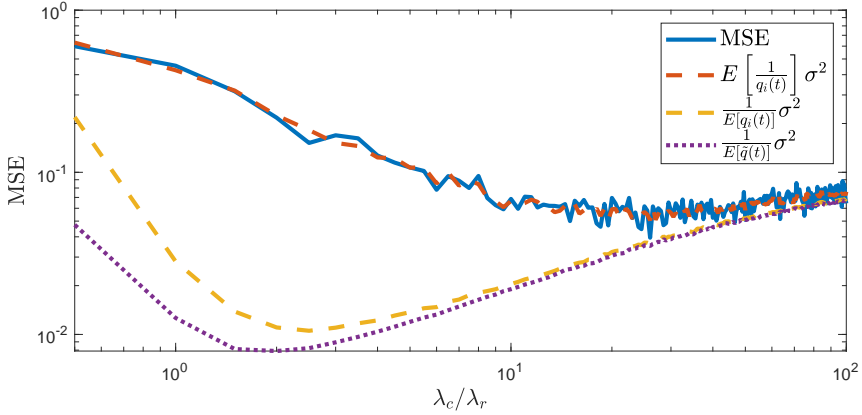
More generally, this chapter aimed at setting up tools and a methodology for studying open systems in general, and which can be applied to other types of problems. Typically, problems building on average consensus, such as decentralized optimization [CDAJW10] or formation control [MR19], could be considered to directly extend the results presented here. Alternatively, variations of this problem, *e.g.*, relying on more structured or constrained interaction models, or other problems for which our methodology can be applied are also a natural continuation for this work.

As an example, a similar analysis was conducted in my Master thesis [Mon18] for *external averaging*, where a (fixed) quantity defined outside of the system must be estimated based on noisy measurements held by the agents. In that problem, the objective of the agents was to estimate the mean of *all the values that have been in the system since the very beginning of the process*. The challenge was thus exactly the opposite of that of this chapter, where outdated information had to be eliminated. This question was also (re)considered in this thesis, but left unfinished and is thus not presented here, we however give a few insights of it below.

Let us use  $q_i(t)$  to denote the *quantity of information held by agent  $i$  at time  $t$* . In that setting, it would correspond to the *number of agents that are or have been in the system agent  $i$  knows the value of*. Namely, let  $\mathcal{Q}(t)$  be the set of all the agents that have been in the system since the beginning of the process (where each agent has a unique label), and let  $\mathcal{Q}_i(t) \subseteq \mathcal{Q}(t)$  denote the subset of  $\mathcal{Q}$  corresponding the agents whose values are known by agent  $i$ , then  $q_i(t) = |\mathcal{Q}_i(t)|$ . Let us also use  $\tilde{q}(t)$  to denote the *total amount of information that has been in the system since the beginning of the process, i.e.*,  $\tilde{q}(t) = |\mathcal{Q}(t)|$ . Then, using  $\mathcal{M}(t)$  to denote the MSE it was shown that

$$\mathbb{E}[\mathcal{M}(t)] \geq \mathbb{E}\left[\frac{1}{q_i(t)}\right] \sigma^2 \geq \frac{1}{\mathbb{E}[q_i(t)]} \sigma^2 \geq \frac{1}{\mathbb{E}[\tilde{q}(t)]} \sigma^2. \quad (5.49)$$

These bounds are empirically illustrated in Fig. 5.5.1.



**Fig. 5.5.1** Asymptotic MSE of the (implemented) optimal algorithm after 200 events (in plain blue line) for 10 agents with respect to  $\lambda_c/\lambda_r$ , corresponding to the expected number of interactions involving an agent before it gets replaced. The other curves represent each of the approximations in (5.49) numerically evaluated over the simulation.

The derivation of closed-form expressions for  $\mathbb{E}\left[\frac{1}{q_i(t)}\right]$  and  $\mathbb{E}[q_i(t)]$  still remains a challenge today, as it relies on complex analyses of the impact of arrivals and departures. Moreover, the approximations from above tend to introduce a non-negligible amount of conservatism, as shows Fig. 5.5.1. An approach for upper bounding  $\mathbb{E}[q_i(t)]$  was proposed nevertheless in [Col20], but it also adds conservatism to the result, and efficiently handling this problem still remains an interesting open question.

Another particularly relevant continuation of this chapter is the investigation of the effect of the different parameters that are assumed to be known by our bounds on the actual performance of algorithms. Typically, whereas our analysis strongly relied on the identifiers held by the agents, their actual impact on algorithms is questionable. This issue is in particular supported by the analysis of the performance of the Gossip algorithm, that does not make use of identifiers, and still had decent performance as compared to our bounds, which do have access to identifiers. Finally, a last interesting follow-up for this chapter is its extension to continuous-time algorithms (as our derivation relies on essentially discrete-time ideas), and which might be particularly challenging.



# Appendix of Chapter 5

## 5.A Proofs of Propositions 5.11 to 5.13 (Main result)

We detail here the proofs of the intermediate results used to build the proof of Theorem 5.9.

### 5.A.1 Proof of Proposition 5.11

*Proof.* Our goal is to show that the estimate that achieves the minimal expected MSE (for agent  $i$  based on its information) is given by

$$y_i^*(t) = \mathbb{E} \left[ \bar{x}(t) | \omega_i^{\mathcal{H}^*}(t) \right].$$

First remember that any estimate  $y_i$  computed by agent  $i$  is deterministic conditional to its knowledge set  $\omega_i^{\mathcal{H}^*}(t)$ , so that

$$\frac{d}{dy_i} \mathbb{E} \left[ (\bar{x}(t) - y_i)^2 | \omega_i^{\mathcal{H}^*}(t) \right] = \mathbb{E} \left[ 2y_i - 2\bar{x}(t) | \omega_i^{\mathcal{H}^*}(t) \right] = 0$$

if and only if  $y_i = y_i^*(t) = \mathbb{E} \left[ \bar{x}(t) | \omega_i^{\mathcal{H}^*}(t) \right]$ . Hence, since  $(\bar{x}(t) - y_i)^2$  is convex with respect to  $y_i$ , it follows that

$$y_i^*(t) = \arg \min_y \left\{ \mathbb{E} \left[ (\bar{x}(t) - y)^2 | \omega_i^{\mathcal{H}^*}(t) \right] \right\}.$$

Consequently, the result  $y_i$  of any other algorithm that can be imple-

mented by agent  $i$  based on  $\omega_i^{\mathcal{H}^*}(t)$  satisfies by definition

$$\mathbb{E} \left[ (\bar{x}(t) - y_i)^2 \mid \omega_i^{\mathcal{H}^*}(t) \right] \geq \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \mid \omega_i^{\mathcal{H}^*}(t) \right].$$

The relation above is true for any realization of  $\omega_i^{\mathcal{H}^*}(t)$  obtained with the model  $*$ , and therefore there holds for any estimate  $y_i$

$$\begin{aligned} \mathbb{E} \left[ (\bar{x}(t) - y_i)^2 \right] &= \mathbb{E} \left[ \mathbb{E} \left[ (\bar{x}(t) - y_i)^2 \mid \omega_i^{\mathcal{H}^*}(t) \right] \right] \\ &\geq \mathbb{E} \left[ \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \mid \omega_i^{\mathcal{H}^*}(t) \right] \right] = \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \right]. \end{aligned}$$

Hence, for any estimates  $y_i$  computed by the agents  $i$  in the system, there holds

$$\mathbb{E} [\mathcal{M}(t)] = \frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ (\bar{x}(t) - y_i)^2 \right] \geq \frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \right],$$

which concludes the proof.  $\square$

### 5.A.2 Proof of Proposition 5.12

*Proof.* Our goal is to study the MSE achieved with  $y_i^*(t)$  from (5.24):

$$\mathbb{E} [\mathcal{M}^*(t)] = \mathbb{E} \left[ \frac{1}{n} \sum_{i=1}^n (\bar{x}(t) - y_i^*(t))^2 \right].$$

First observe that the optimal estimate from (5.24) can be written as

$$y_i^*(t) = \mathbb{E} \left[ \bar{x}(t) \mid \omega_i^{\mathcal{H}^*}(t) \right] = \frac{1}{n} \sum_{j=1}^n \mathbb{E} \left[ x_j(t) \mid \omega_i^{\mathcal{H}^*}(t) \right].$$

Hence, since  $\mathbb{E} [\mathcal{M}^*(t)] = \frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ (\bar{x}(t) - y_i^*(t))^2 \right]$ , it follows that

$$\begin{aligned} \mathbb{E} [\mathcal{M}^*(t)] &= \frac{1}{n} \sum_{i=1}^n \mathbb{E} \left[ \left( \frac{1}{n} \sum_{j=1}^n x_j(t) - \frac{1}{n} \sum_{j=1}^n \mathbb{E} \left[ x_j(t) \mid \omega_i^{\mathcal{H}^*}(t) \right] \right)^2 \right] \\ &= \frac{1}{n^3} \sum_{i=1}^n \mathbb{E} \left[ \left( \sum_{j=1}^n \left( x_j(t) - \mathbb{E} \left[ x_j(t) \mid \omega_i^{\mathcal{H}^*}(t) \right] \right) \right)^2 \right]. \end{aligned}$$

Let us use  $\hat{x}_j^{(i)}(t) = \mathbb{E} \left[ x_j(t) | \omega_i^{\mathcal{H}^*}(t) \right]$  to denote the estimation of agent  $j$ 's value based on the information held by agent  $i$ . Then, because of the absence of correlation between the values of the agents, one has for  $j \neq k$ :

$$\mathbb{E} \left[ \left( x_j(t) - \hat{x}_j^{(i)}(t) \right) \left( x_k(t) - \hat{x}_k^{(i)}(t) \right) \right] = 0.$$

As a consequence there holds for all  $i \in \mathcal{V}$ :

$$\mathbb{E} \left[ \left( \sum_{j=1}^n \left( x_j(t) - \hat{x}_j^{(i)}(t) \right) \right)^2 \right] = \mathbb{E} \left[ \sum_{j=1}^n \left( x_j(t) - \hat{x}_j^{(i)}(t) \right)^2 \right].$$

Injecting that result into the expression of  $\mathbb{E} [\mathcal{M}^*(t)]$  then yields

$$\mathbb{E} [\mathcal{M}^*(t)] = \frac{1}{n^3} \sum_{i=1}^n \sum_{j=1}^n \mathbb{E} \left[ \left( x_j(t) - \hat{x}_j^{(i)}(t) \right)^2 \right],$$

which concludes the proof.  $\square$

### 5.A.3 Proof of Proposition 5.13

In order to prove Proposition 5.13, we first provide the following lemma.

**Lemma 5.24.** *Let  $Y, Z$  be two i.i.d. zero mean random variables with variance  $\sigma^2$ , and let  $X$  be another random variable defined as follows:*

$$X := \begin{cases} Z & \text{with probability } p \\ Y & \text{with probability } 1 - p \end{cases}, \quad (5.50)$$

such that the event related to the probability  $p$  is independent of  $Z$  and  $Y$ . The estimator  $\hat{X}$  that minimizes  $\mathbb{E} \left[ (X - \hat{X})^2 \right]$  given the value of  $Z$  is then given by

$$\hat{X} = \arg \min_x \left\{ \mathbb{E} \left[ (X - x)^2 | Z \right] \right\} = pZ, \quad (5.51)$$

and the corresponding error is given by

$$\mathbb{E} \left[ (X - \hat{X})^2 \right] = (1 - p^2)\sigma^2. \quad (5.52)$$

## 5 | Performance limitations for average consensus in OMAS

*Proof.* By definition of  $X$ , the MSE of an estimator  $\hat{X}$  of  $X$  given  $Z$  is

$$\begin{aligned}\mathbb{E} \left[ (X - \hat{X})^2 | Z \right] &= \mathbb{E} \left[ (Z - \hat{X})^2 | Z \right] p + \mathbb{E} \left[ (Y - \hat{X})^2 | Z \right] (1 - p) \\ &= p(Z - \hat{X})^2 + (1 - p)\hat{X}^2 + (1 - p)\sigma^2 \\ &= pZ^2 - 2pZ\hat{X} + \hat{X}^2 + (1 - p)\sigma^2.\end{aligned}$$

The random variable  $Y$  is eliminated in the development above by using the fact that it is independent of  $Z$ , and that  $\mathbb{E}[Y] = 0$  and  $\mathbb{E}[Y^2] = \sigma^2$  by definition. That estimator is thus optimal if

$$\begin{aligned}\frac{d}{d\hat{X}} \mathbb{E} \left[ (X - \hat{X})^2 | Z \right] &= \frac{d}{d\hat{X}} \left( pZ^2 - 2pZ\hat{X} + \hat{X}^2 + (1 - p)\sigma^2 \right) \\ &= -2pZ + 2\hat{X} = 0,\end{aligned}$$

which is the case if and only if  $\hat{X} = pZ$ . This establishes (5.51). The MSE for that estimator  $\hat{X}$  conditional to  $Z$  is then given by

$$\begin{aligned}\mathbb{E} \left[ (X - \hat{X})^2 | Z \right] &= pZ^2 - 2p^2Z^2 + p^2Z^2 + (1 - p)\sigma^2 \\ &= p(1 - p)Z^2 + (1 - p)\sigma^2.\end{aligned}$$

This finally yields

$$\begin{aligned}\mathbb{E} \left[ (X - \hat{X})^2 \right] &= p(1 - p)\mathbb{E} \left[ Z^2 \right] + (1 - p)\sigma^2 \\ &= (1 - p)(1 + p)\sigma^2 = (1 - p^2)\sigma^2,\end{aligned}$$

which establishes (5.52), and concludes the proof.  $\square$

*Proof of Proposition 5.13.* We aim at characterizing

$$\mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) \right] = \mathbb{E} \left[ \left( x_j(t) - \mathbb{E} \left[ x_j(t) | \omega_i^{\epsilon^*}(t) \right] \right)^2 \right].$$

Let  $N_j^i(t)$  denote the (probabilistic) event that no information about the value of agent  $j$  lies in  $\omega_i^{\mathcal{H}^*}(t)$  (i.e., the event that  $(x_j(s), j, s) \notin \omega_i^{\epsilon^*}(t)$  for all  $s \leq t$ ), and let  $\bar{N}_j^i(t)$  denote its complementary event. By definition,

$T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  is well-defined only in the event  $\bar{N}_j^i(t)$ , so that there holds

$$\begin{aligned} \mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) \right] &= P \left[ N_j^i(t) \right] \mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) | N_j^i(t) \right] \\ &\quad + \int_0^t f_{j \rightarrow i}^{t,*}(s) \mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) | T_{j \rightarrow i}^{\mathcal{H}^*}(t) = s \right] ds, \end{aligned} \quad (5.53)$$

where we remind  $f_{j \rightarrow i}^{t,*}(s)$  is the pseudo-PDF of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  as defined in (5.17). We analyze both terms of the above sum separately below.

**In the event  $N_j^i(t)$ :** Lemma 5.7 guarantees that in the absence of information about an agent's value, then the best possible estimate is the mean of the distribution that defines it. Hence  $\mathbb{E} \left[ x_j(t) | \omega_i^{\mathcal{H}^*}(t) \cap N_j^i(t) \right] = 0$  holds, and it follows that

$$\mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) | N_j^i(t) \right] = \mathbb{E} \left[ (x_j(t) - 0)^2 \right] = \sigma^2. \quad (5.54)$$

**In the event  $\bar{N}_j^i(t)$ :** The value of  $x_j(t)$  given the information in  $\omega_i^{\mathcal{H}^*}(t)$  is

$$x_j(t) = \begin{cases} x_j \left( t - T_{j \rightarrow i}^{\mathcal{H}^*}(t) \right) & \text{w.p. } e^{-\lambda_r T_{j \rightarrow i}^{\mathcal{H}^*}(t)} \\ X & \text{otherwise} \end{cases},$$

where

- $X$  is the (unknown) current random value of agent  $j$  if it was replaced, independent of  $x_j \left( t - T_{j \rightarrow i}^{\mathcal{H}^*}(t) \right)$  but following the same distribution;
- $e^{-\lambda_r T_{j \rightarrow i}^{\mathcal{H}^*}(t)}$  is the probability that no replacement of agent  $j$  occurred since time  $t - T_{j \rightarrow i}^{\mathcal{H}^*}(t)$  given  $\omega_i^{\mathcal{H}^*}(t)$ , obtained from Lemma 5.6.

We can then apply Lemma 5.24, and in particular (5.52), to obtain

$$\mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) | \bar{N}_j^i(t) \right] = \left( 1 - e^{-2\lambda_r T_{j \rightarrow i}^{\mathcal{H}^*}(t)} \right) \sigma^2. \quad (5.55)$$

Moreover, from the definition of  $T_{j \rightarrow i}^{\mathcal{H}^*}(t)$ , we have

$$\int_0^t f_{j \rightarrow i}^{t,*}(s) ds = \mathbb{P} \left[ T_{j \rightarrow i}^{\mathcal{H}^*}(t) \leq t \right] = \mathbb{P} \left[ \bar{N}_j^i(t) \right] = 1 - \mathbb{P} \left[ N_j^i(t) \right]. \quad (5.56)$$

**Back to (5.53):** Injecting (5.54) to (5.56) into (5.53) finally yields

$$\begin{aligned} \mathbb{E} \left[ \mathcal{M}_{j \rightarrow i}^*(t) \right] &= \mathbb{P} \left[ N_j^i(t) \right] \sigma^2 + \int_0^t f_{j \rightarrow i}^{t,*}(s) \left( 1 - e^{-2\lambda_r s} \right) \sigma^2 ds \\ &= \left( \mathbb{P} \left[ N_j^i(t) \right] + 1 - \mathbb{P} \left[ N_j^i(t) \right] - \int_0^t f_{j \rightarrow i}^{t,*}(s) e^{-2\lambda_r s} ds \right) \sigma^2 \\ &= \left( 1 - \int_0^t f_{j \rightarrow i}^{t,*}(s) e^{-2\lambda_r s} ds \right) \sigma^2, \end{aligned}$$

which concludes the proof.  $\square$

## 5.B Proof of Proposition 5.17 (Ping' model)

We remind the statement of Proposition 5.17 reads

*With the Ping' model, the random variable  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}'}}$  (t) admits the following pseudo-PDF for any  $i \neq j$ :*

$$f_{j \rightarrow i}^{t, \text{Ping}'}(s) = \lambda_p e^{-(\lambda_p + \lambda_r)s}. \quad (5.57)$$

*Proof.* Let  $T_{\text{Rep}_i}(t)$  and  $T_{\text{Ping}_i}(t)$  respectively denote the elapsed time since the last replacement of agent  $i$  and the last time agent  $i$  performed a Ping update. Then, by definition of the Ping' model,  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}'}}(t) = T_{\text{Ping}_i}(t)$  if  $T_{\text{Ping}_i}(t) < T_{\text{Rep}_i}(t)$  (i.e., if the last Ping update of agent  $i$  is more recent than its last replacement), and is not defined otherwise. Hence, we have

$$\begin{aligned} F_{j \rightarrow i}^{t, \text{Ping}'}(s) &= \mathbb{P} \left[ T_{j \rightarrow i}^{\mathcal{H}^{\text{Ping}'}}(t) \leq s \right] \\ &= \mathbb{P} \left[ T_{\text{Ping}_i}(t) \leq s \mid T_{\text{Ping}_i}(t) \leq T_{\text{Rep}_i}(t) \right] \mathbb{P} \left[ T_{\text{Ping}_i}(t) \leq T_{\text{Rep}_i}(t) \right]. \end{aligned}$$

Since Ping updates and replacements are characterized by Poisson processes of respective rates  $\lambda_p$  and  $\lambda_r$ , we have  $\mathbb{P} \left[ T_{\text{Ping}_i}(t) \leq T_{\text{Rep}_i}(t) \right] = \frac{\lambda_p}{\lambda_p + \lambda_r}$ . Moreover, using the properties of Poisson processes, there holds

$$\mathbb{P} \left[ T_{\text{Ping}_i}(t) \leq s \mid T_{\text{Ping}_i}(t) \leq T_{\text{Rep}_i}(t) \right] = \begin{cases} 1 & \text{if } T_{\text{Rep}_i}(t) \leq s \\ e^{-\lambda_p s} & \text{otherwise} \end{cases}.$$

Hence, since  $\mathbb{P} \left[ T_{\text{Rep}_i}(t) \leq s \right] = 1 - e^{-\lambda_r s}$ , we have

$$\mathbb{P} \left[ T_{\text{Ping}_i}(t) \leq s \mid T_{\text{Ping}_i}(t) \leq T_{\text{Rep}_i}(t) \right] = 1 - e^{-\lambda_r s} + e^{-\lambda_r s} \left( 1 - e^{-\lambda_p s} \right),$$

and it follows that

$$F_{j \rightarrow i}^{t, \text{Ping}'}(s) = \frac{\lambda_p}{\lambda_p + \lambda_r} \left( 1 - e^{-(\lambda_p + \lambda_r)s} \right).$$

The conclusion finally comes from  $f_{j \rightarrow i}^{t, \text{Ping}'}(s) = \frac{d}{ds} F_{j \rightarrow i}^{t, \text{Ping}'}(s)$ .  $\square$

## 5.C Proof of Proposition 5.20 (Gossip model)

We now prove Proposition 5.20, which characterizes the pseudo-PDF corresponding to the Gossip interaction model as:

$$f_{j \rightarrow i}^{t, \text{Gossip}}(s) = \mathbf{w}^\top A e^{As} \mathbf{e}_1, \quad (5.58)$$

where  $\mathbf{w} \in \mathbb{R}^n$  with  $w_k = \frac{k-1}{n-1}$ , and  $A \in \mathbb{R}^{n \times n}$  is a tridiagonal matrix with  $[A]_{k,k} = -\left( \frac{k(n-k)}{n-1} \lambda_c + (k-1) \lambda_r \right)$ ,  $[A]_{k,k+1} = k \lambda_r$  and  $[A]_{k+1,k} = \frac{k(n-k)}{n-1} \lambda_c$ .

*Proof.* By definition,  $T_{j \rightarrow i}^{\mathcal{H}^{\text{Gossip}}}(t) \leq s$  corresponds to the existence of some information of the type  $(x, j, \tau)$  in  $\omega_i^{\text{Gossip}}(t)$ , with  $\tau \in [t-s, t]$ , so that

$$\begin{aligned} F_{j \rightarrow i}^{t, \text{Gossip}}(s) &= \mathbb{P} \left[ T_{j \rightarrow i}^{\mathcal{H}^{\text{Gossip}}}(t) \leq s \right] \\ &= \mathbb{P} \left[ \exists \tau \in [t-s, t] : (x_j(\tau), j, \tau) \in \omega_i^{\text{Gossip}}(t) \right]. \end{aligned}$$

Let us use  $\eta_j(t, t') \in \mathbb{N}$  to denote the number of agents having access to at least one information about the value held by  $j$  more recent than  $t'$  in their knowledge set at time  $t$ . This number is random for some stochastic event sequence  $\mathcal{H}^{\text{Gossip}}$ . Because no distinction is made between the agents at replacements and information exchanges, for any  $k \in \{1, \dots, n\}$  we have

$$\mathbb{P} \left[ \exists \tau \in [t-s, t] : (x_j(\tau), j, \tau) \in \omega_i^{\text{Gossip}}(t) \mid \eta_j(t, t-s) = k \right] = \frac{k-1}{n-1},$$

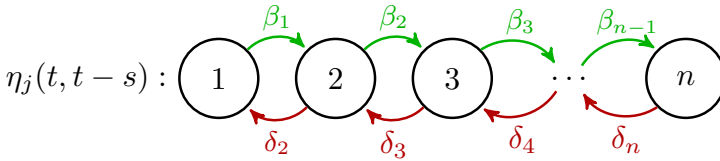
so that there holds

$$F_{j \rightarrow i}^{t, \text{Gossip}}(s) = \sum_{k=1}^n \frac{k-1}{n-1} \mathbb{P} [\eta_j(t, t-s) = k]. \quad (5.59)$$

Observe that  $\eta_j(t, t-s)$  is a Markov jump process in the sense of Definition 2.15 (defined in Section 2.3.2). Namely, it remains constant between events and is instantaneously modified at replacements and information exchanges. More specifically, if  $\eta_j(t, t-s) = k$ , then

- $\eta_j(t, t-s)$  increases by one with rate  $\frac{k(n-k)}{n-1} \lambda_c$  through interactions;
- $\eta_j(t, t-s)$  decreases by one with rate  $(k-1) \lambda_r$  through replacements.

It thus evolves according to the birth-death process represented in Fig. 5.C.1 (see Example 2.21 for details about such processes).



**Fig. 5.C.1** Birth-death process characterizing  $\eta_j(t, t-s)$ : for  $j = 1, \dots, n$ , we have  $\beta_j = \frac{j(n-j)}{n-1} \lambda_c$  and  $\delta_j = (j-1) \lambda_r$ .

Let  $\pi(s) \in \mathbb{R}^n$  be the vector such that  $\pi_k(s) = \mathbb{P} [\eta_j(t, t-s) = k]$  (where the dependence on  $t$  is omitted). Then, using standard properties of Markov jump processes and of Birth-death processes, respectively given in (2.21) and (2.31), we have

$$\frac{d}{ds} \pi(s) = A \pi(s),$$

for the matrix  $A \in \mathbb{R}^{n \times n}$  defined above. It thus follows that

$$\pi(s) = e^{As} \pi(0) \quad \text{with} \quad \pi(0) = \mathbf{e}_1.$$

Injecting that result into (5.59) then yields

$$F_{j \rightarrow i}^{t, \text{Gossip}}(s) = \sum_{k=1}^n \frac{k-1}{n-1} \pi_k(s) = \mathbf{w}^\top e^{As} \mathbf{e}_1$$

with  $w_k = \frac{k-1}{n-1}$ . Finally, one finds that

$$f_{j \rightarrow i}^{t, \text{Gossip}}(s) = \frac{d}{ds} F_{j \rightarrow i}^{t, \text{Gossip}}(s) = \mathbf{w}^\top A e^{As} \mathbf{e}_1,$$

which concludes the proof.  $\square$

## 5.D Proof of Corollary 5.22 (XYZ approximation)

We now prove Corollary 5.22, whose result reads for the Gossip model:

$$\liminf_{t \rightarrow \infty} \mathbb{E}[\mathcal{M}(t)] \geq \frac{n-1}{n^2} \left( \frac{7}{2} + \log\left(\frac{n-2}{2}\right) + h_n \right) \rho_c^{-1} \sigma^2 + \mathcal{O}\left(\frac{n}{\rho_c^2}\right) \sigma^2, \quad (5.60)$$

for some known polynomial  $h_n$  in  $\mathcal{O}(n^{-1})$

We first prove a few preliminary results that will allow us to build the proof. Note that for this proof, we use the notation “:” to refer to a whole row or column of a matrix, e.g. for some matrix  $A$ ,  $[A]_{i,:}$  stands for the whole  $i$ -th row of  $A$  and  $[A]_{:,i}$  for the whole  $i$ -th column of  $A$ .

Let us first define the quantity

$$E := 1 - \mathbf{w}^\top A (2\lambda_r - A)^{-1} \mathbf{e}_1, \quad (5.61)$$

such that (5.43) is equivalent to  $\frac{n-1}{n^2} E \sigma^2$ .

**Lemma 5.25.** *Let  $\tilde{\rho} := \frac{1}{n-1} \frac{\lambda_c}{\lambda_r}$  and let  $C \in \mathbb{R}^{n \times n}$  and  $R \in \mathbb{R}^{n \times n}$  be two matrices such that  $\frac{1}{\lambda_r} A = C\tilde{\rho} + R$ . Moreover, let  $\mathbf{v} \in \mathbb{R}^n$  be such that  $v_k = \frac{k(n-k)}{n-1}$ . The expression (5.61) is equivalent to*

$$E = 1 - \mathbf{w}^\top \left( \mathbf{v} - \mathbf{w}\tilde{\rho}^{-1} \right)^\top \left( \tilde{\rho}^{-1}(2I - R) - C \right)^{-1} \mathbf{e}_1. \quad (5.62)$$

*Proof.* Let us define  $\tilde{A} := \frac{1}{\lambda_r} A$ , then one has

$$E = 1 - \mathbf{w}^\top \tilde{A} (2I - \tilde{A})^{-1} \mathbf{e}_1.$$

Moreover, by definition of  $A$ , there holds

$$\tilde{A} = C\tilde{\rho} + R,$$

## 5 | Performance limitations for average consensus in OMAS

where  $C$  and  $R$  are bidiagonal matrices defined as

$$\begin{aligned} [C]_{k,k} &= -[C]_{k+1,k} = -(N-k) & (1 \leq k \leq N-1); \\ [R]_{k,k} &= -[R]_{k-1,k} = -(k-1) & (2 \leq k \leq N). \end{aligned}$$

Hence, one has

$$\begin{aligned} E &= 1 - \mathbf{w}^\top (C\tilde{\rho} + R) (2I - C\tilde{\rho} - R)^{-1} \mathbf{e}_1 \\ &= 1 - \mathbf{w}^\top (C + R\tilde{\rho}^{-1}) (\tilde{\rho}^{-1}(2I - R) - C)^{-1} \mathbf{e}_1. \end{aligned}$$

Let  $\mathbf{v}$  be the vector such that  $v_k = \frac{k(n-k)}{n-1}$ , then there holds

$$\begin{aligned} (\mathbf{w}^\top C)_k &= \mathbf{w}^\top [C]_{:,k} = v_k [C]_{k,k} + v_{k+1} [C]_{k+1,k} \\ &= \frac{k-1}{n-1} (-k(n-k)) + \frac{k}{n-1} (k(n-k)) = \frac{k(n-k)}{n-1} = v_k. \end{aligned}$$

Moreover, observe that

$$\begin{aligned} (\mathbf{w}^\top R)_k &= \mathbf{w}^\top [R]_{:,k} = v_k [R]_{k,k} + v_{k-1} [R]_{k-1,k} \\ &= -\frac{k-1}{N-1} (k-1) + \frac{k-2}{N-1} (k-1) = -\frac{k-1}{N-1} = w_k. \end{aligned}$$

Injecting these two results in the previous one then yields

$$\begin{aligned} E &= 1 - (\mathbf{w}^\top C + \mathbf{w}^\top R\tilde{\rho}^{-1}) (\tilde{\rho}^{-1}(2I - R) - C)^{-1} \mathbf{e}_1 \\ &= 1 - (\mathbf{v}^\top - \mathbf{w}^\top \tilde{\rho}^{-1}) (\tilde{\rho}^{-1}(2I - R) - C)^{-1} \mathbf{e}_1, \end{aligned}$$

which concludes the proof.  $\square$

The following proposition then shows that the second factor of (5.62) can be expressed as a polynomial in  $\tilde{\rho}$ .

**Proposition 5.26.** *Considering expression (5.62) from Lemma 5.25, there holds*

$$(\tilde{\rho}^{-1}(2I - R) - C)^{-1} = X\tilde{\rho} + Y + Z\tilde{\rho}^{-1} + \mathcal{O}(\tilde{\rho}^{-2}), \quad (5.63)$$

for some matrices  $X, Y, Z \in \mathbb{R}^{n \times n}$ .

*Proof.* Let us define  $\eta := \tilde{\rho}^{-1}$  and  $M := (2I - R)\eta - C$ , as well as the matrix

$$S_{ij} := \begin{bmatrix} M & \vdots & \mathbf{e}_i \\ \dots & & \dots \\ \mathbf{e}_j^\top & \vdots & 0 \end{bmatrix}.$$

Then  $-[M^{-1}]_{j,i} = -\mathbf{e}_j^\top M \mathbf{e}_i$  is the Schur complement of  $M$  in  $S_{ij}$ . Hence, from the properties of the Schur complement, there holds

$$[M^{-1}]_{j,i} = \det \left( \mathbf{e}_j^\top M \mathbf{e}_i \right) = -\frac{\det(S_{ij})}{\det(M)}.$$

Observe  $\det(M)$  and  $\det(S_{ij})$  can be written as polynomials such that

$$\det(M) = \eta^q p(\eta) \quad \text{with } p(0) \neq 0; \quad (5.64)$$

$$\det(S_{ij}) = \eta^{q_{ij}} p_{ij}(\eta) \quad \text{with } p_{ij}(0) \neq 0 \quad \forall i, j. \quad (5.65)$$

Hence, for some coefficients  $c_0, c_1, \dots$ , we have

$$[M^{-1}]_{j,i} = c_0 \eta^{q-q_{ij}} + c_1 \eta^{q-q_{ij}+1} + c_2 \eta^{q-q_{ij}+2} + \dots,$$

and thus for some matrices  $X, Y$  and  $Z$

$$M^{-1} = X \eta^{q-\tilde{q}_{ij}} + Y \eta^{q-\tilde{q}_{ij}+1} + Z \eta^{q-\tilde{q}_{ij}+2} + \dots,$$

with  $\tilde{q}_{ij} := \min_{i,j} \{q_{ij}\}$ .

Let  $M_k$  denote the  $k \times k$  upper-left sub-matrix of  $M$ . By definition of  $M$ , and from the properties of tridiagonal matrices, there holds

$$\begin{aligned} \det(M_k) &= [k(n-k) + (k+1)\eta] \det(M_{k-1}) \\ &\quad - (k-1)^2(n-k+1)\eta \det(M_{k-2}), \end{aligned}$$

with  $\det(M_0) = 1$  and  $\det(M_{-1}) = 0$ . Hence, by induction, one shows that for  $k < n$

$$\det(M_k) = p_k(\eta)$$

for some polynomial  $p_k$  such that  $p_k(0) \neq 0$ . As a consequence, we have

$$\det(M) = (n+1)\eta p_{n-1}(\eta) - (n-1)^2 \eta p_{n-2}(\eta) = \eta p(\eta),$$

## 5 | Performance limitations for average consensus in OMAS

for some polynomial  $p$  such that  $p(0) \neq 0$ , and it follows that the exponent of  $\eta$  in (5.64) is  $q = 1$ . Moreover, it appears from the structure of  $M$  that the determinant of any cofactor of  $M$  is a polynomial of  $\eta$ , and for all  $i, j$ , one thus has  $q_{ij} \geq 0$ . Hence,

$$\begin{aligned} M^{-1} &= X\eta^{-1} + Y + Z\eta + \dots \\ &= X\tilde{\rho} + Y + Z\tilde{\rho}^{-1} + \mathcal{O}(\tilde{\rho}^{-2}), \end{aligned}$$

which concludes the proof.  $\square$

Starting from the previous proposition, we investigate the structure of the matrices  $X$ ,  $Y$  and  $Z$  in the following proposition, which is the last intermediate result of the proof of Corollary 5.22.

**Proposition 5.27.** *Expression (5.61) reduces to*

$$E = \frac{\tilde{\rho}^{-1}}{n-1} \left[ \sum_{k=2}^{n-2} \binom{n}{k(n-k)} + \frac{3n-1}{N-1} \right] + \mathcal{O}(\tilde{\rho}^{-2}). \quad (5.66)$$

*Proof.* Let us assume that  $\tilde{\rho}^k \approx 0$  for all  $k \leq -2$ . Then, reminding that

$$\left( \tilde{\rho}^{-1}(2I - R) - C \right)^{-1} = X\tilde{\rho} + Y + Z\tilde{\rho}^{-1} + \mathcal{O}(\tilde{\rho}^{-2})$$

from Proposition 5.26 for some matrices  $X$ ,  $Y$  and  $Z$ , we have

$$\begin{aligned} I &= \left( X\tilde{\rho} + Y + Z\tilde{\rho}^{-1} \right) \left( \tilde{\rho}^{-1}(2I - R) - C \right) \\ &= \left( \tilde{\rho}^{-1}(2I - R) - C \right) \left( X\tilde{\rho} + Y + Z\tilde{\rho}^{-1} \right). \end{aligned}$$

This yields six equations that  $X$ ,  $Y$  and  $Z$  must satisfy for the expression above to hold, allowing their identification. For concision matters, we do not detail the whole algebra in what follows.

(i)  $CX = 0$  and  $XC = 0$ : the identification yields

$$X = \alpha \mathbf{e}_N \mathbf{1}^\top \quad (5.67)$$

for some  $\alpha \in \mathbb{R}$ .

(ii)  $(2I - R)X - I = CY$ : the identification yields  $\alpha = \frac{1}{2}$ , so that

$$X = \frac{1}{2} \mathbf{e}_N \mathbf{1}^\top. \quad (5.68)$$

Moreover, it gives

$$[Y]_{k,:} = \begin{cases} \frac{1}{k(n-k)} \sum_{j=1}^k \mathbf{e}_j^\top, & \text{for } k < n-1 \\ \frac{n+1}{2(n-1)} \mathbf{1}^\top - \frac{\mathbf{e}_n^\top}{n-1}, & \text{for } k = n-1. \end{cases} \quad (5.69)$$

(iii)  $(2I - R)X - I = YC$ : the identification shows that for some  $\beta \in \mathbb{R}$ :

$$[Y]_{n,k} = \beta + \sum_{j=1}^{k-1} \frac{1}{j(n-j)}. \quad (5.70)$$

(iv)  $(2I - R)Y - CZ = 0$ : the identification yields  $\beta = -\frac{1}{2} - \sum_{j=1}^{n-1} \frac{1}{j(n-j)}$ , so that

$$[Y]_{n,k} = -\frac{1}{2} - \sum_{j=k}^{n-1} \frac{1}{j(n-j)}. \quad (5.71)$$

Moreover, for  $1 \leq k \leq n-1$ , the identification gives

$$[Z]_{k,:} = \frac{1}{k(n-k)} \left( k[Y]_{k+1,:} - \sum_{j=1}^k [Y]_{j,:} \right). \quad (5.72)$$

(v)  $Y(2I - R) - ZC = 0$ : the identification yields the first column of the matrix  $Z$ , which we will see later is sufficient to entirely characterize the wanted expression:

$$[Z]_{k,1} = \begin{cases} \frac{1}{k(n-k)} \left( \frac{k}{(k+1)(n-k-1)} - 2 \sum_{j=1}^k \frac{1}{j(n-j)} \right), & k < n-2 \\ \frac{1}{2(n-2)} \left( \frac{(n-2)(n+1)}{2(n-1)} - 2 \sum_{j=1}^{n-2} \frac{1}{j(n-j)} \right) & k = n-2 \\ -\frac{1}{n-1} \left( \frac{n+1}{2} + (n+1) \sum_{j=1}^{n-1} \frac{1}{j(n-j)} \right) & k = N-1 \end{cases}. \quad (5.73)$$

We can now compute  $E$ . Using Proposition 5.26, (5.62) reduces to

$$E = 1 - \mathbf{v}^\top X \mathbf{e}_1 \tilde{\rho} - \mathbf{v}^\top Y \mathbf{e}_1 - \mathbf{v}^\top Z \mathbf{e}_1 \tilde{\rho}^{-1} + \mathbf{w}^\top X \mathbf{e}_1 + \mathbf{w}^\top Y \mathbf{e}_1 \tilde{\rho}^{-1} + \mathcal{O}(\tilde{\rho}^{-2}).$$

We can then identify each term of the sum above using the identified matrices  $X$ ,  $Y$  and  $Z$  from (5.67) to (5.73). Once more, for the sake of concision

we do not detail the algebra of what follows:

$$\mathbf{v}^\top \mathbf{X} \mathbf{e}_1 = \mathbf{v}^\top [X]_{:,1} = \frac{1}{2} v_n = 0;$$

$$\mathbf{v}^\top \mathbf{Y} \mathbf{e}_1 = \mathbf{v}^\top [Y]_{:,1} = 3/2;$$

$$\mathbf{v}^\top \mathbf{Z} \mathbf{e}_1 = \mathbf{v}^\top [Z]_{:,1} = \sum_{k=1}^{n-1} \frac{k(n-k)}{n-1} [Z]_{k,1} = \frac{1}{n-1} \left( - \sum_{k=2}^{n-2} \binom{n+k}{k(n-k)} - \frac{3}{2} \frac{3n-1}{n-1} \right);$$

$$\mathbf{w}^\top \mathbf{X} \mathbf{e}_1 = \mathbf{w}^\top [X]_{:,1} = 1/2;$$

$$\mathbf{w}^\top \mathbf{Y} \mathbf{e}_1 = \mathbf{w}^\top [Y]_{:,1} = \sum_{k=1}^n \frac{k-1}{n-1} [Y]_{k,1} = -\frac{1}{n-1} \left( \frac{1}{2} + \sum_{k=1}^{n-1} \frac{1}{k} \right).$$

Injecting these expressions into that of  $E$  from above finally gives

$$\begin{aligned} E &= 1 - \frac{3}{2} + \frac{1}{2} - 0\tilde{\rho} + \left( \mathbf{w}^\top \mathbf{Y} \mathbf{e}_1 - \mathbf{v}^\top \mathbf{Z} \mathbf{e}_1 \right) \tilde{\rho}^{-1} + \mathcal{O}(\tilde{\rho}^{-2}) \\ &= \frac{\tilde{\rho}^{-1}}{n-1} \left( \sum_{k=2}^{n-2} \binom{n+k}{k(n-k)} + \frac{3}{2} \frac{3n-1}{n-1} - \frac{1}{2} - \sum_{k=1}^{n-1} \frac{1}{k} \right) + \mathcal{O}(\tilde{\rho}^{-2}) \\ &= \frac{\tilde{\rho}^{-1}}{n-1} \left( \sum_{k=2}^{n-2} \binom{n}{k(n-k)} + \frac{3}{2} \frac{3n-1}{n-1} - \frac{3}{2} - \frac{1}{n-1} \right) + \mathcal{O}(\tilde{\rho}^{-2}) \\ &= \frac{\tilde{\rho}^{-1}}{n-1} \left( \sum_{k=2}^{n-2} \binom{n}{k(n-k)} + \frac{3n-1}{n-1} \right) + \mathcal{O}(\tilde{\rho}^{-2}), \end{aligned}$$

which concludes the proof.  $\square$

We can now finally give the proof of Corollary 5.22, which builds on the preliminary results given above.

*Proof of Corollary 5.22.* We start from the result of Proposition 5.27, which we bound from below. The composite trapezoid rule for integrating convex functions allows writing

$$\int_2^{n-2} \frac{1}{x(n-x)} dx \leq \sum_{k=2}^{n-3} \left( \frac{1}{k(n-k)} + \frac{1}{(k+1)(n-k-1)} \right),$$

where the inequality is guaranteed by the convexity of the function  $\frac{1}{x(n-x)}$ . Hence, it follows that

$$\sum_{k=2}^{n-2} \frac{1}{k(n-k)} \geq \int_2^{n-2} \frac{1}{x(n-x)} dx + \frac{1}{2(n-2)},$$

where the last term compensates those lacking from the trapezoid rule. Moreover, one can compute the integral as follows, using partial fractions decomposition

$$\begin{aligned} \int_2^{n-2} \frac{1}{x(n-x)} dx &= \int_2^{n-2} \frac{1}{nx} dx + \int_2^{n-2} \frac{1}{n(n-x)} dx \\ &= \frac{1}{n} \left[ \log(x) - \log(n-x) \right]_2^{n-2} = \frac{2}{n} \log\left(\frac{n-2}{2}\right). \end{aligned}$$

Injecting the above inequality into (5.66) from Proposition 5.27 then gives

$$\begin{aligned} E &\geq \frac{\tilde{\rho}}{n-1} \left( \frac{3n-1}{N-1} + \frac{n}{2(n-2)} + 2 \log\left(\frac{n-2}{2}\right) \right) + \mathcal{O}(\tilde{\rho}^{-2}) \\ &\geq \frac{\tilde{\rho}}{n-1} \left( \frac{7}{2} + 2 \log\left(\frac{n-2}{2}\right) + \frac{n-5}{(n-2)(n-1)} \right) + \mathcal{O}(\tilde{\rho}^{-2}). \end{aligned}$$

The conclusion then follows from injecting this expression into (5.43).  $\square$



## **PART II**

# **Decentralized optimization: algorithms for resource allocation**



# 6

## Decentralized optimization

So far we only focused on consensus problems in OMAS: Thanks to their rather basic nature, they allowed investigating quite in depths some specific mechanisms related to OMAS in general. In this part, we take some distance from consensus and consider the more advanced problem of *decentralized optimization*. The more complex structure and nature of the interactions related to these problems make their analysis even more challenging in OMAS. In this chapter, we properly define some standard decentralized optimization problems and review corresponding results in closed systems. We also give insights on existing results related to open systems, or to similar settings.

### 6.1 General definition

In decentralized optimization each agent  $i$  in the system holds an estimate  $x_i \in \mathbb{R}^d$  and a local cost function  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ . Let  $x \in \mathbb{R}^{nd}$  denote the vector concatenating all estimates  $x_i$ ; the objective of the agents is to collaboratively minimize some function  $f(x)$ , which depends on the local costs  $f_i$ , given some constraints. A common definition for  $f$ , which we will use throughout this thesis, requires it to be separable and defined as the sum of all local cost functions, leading to the following problem definition.

**Definition 6.1** (Decentralized optimization). Consider a multi-agent system whose agent set is  $\mathcal{V}$ , and where each agent  $i \in \mathcal{V}$  holds a local cost function  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ . Let us define some set  $X \subseteq \mathbb{R}^{nd}$ , the decentralized optimization problem consists in solving

$$\min_{x \in \mathbb{R}^{nd}} f(x) = \sum_{i \in \mathcal{V}} f_i(x_i) \quad \text{subject to} \quad x \in X. \quad (6.1)$$

Problems of this type typically appear in the context of machine learning [XKK20, Tia21, CNS17b], where a loss function must be minimized with respect to some large dataset spread over several individual computing entities. It also appears in cooperative control [RTB04], and in more general multi-vehicle systems [CYRC12], where agents aim at minimizing some measure of the total deviation based on local interactions. Consensus is sometimes formulated as a decentralized optimization problem as well, *e.g.*, for signal processing [CS12] or sensor networks [RN04]. Similar problems can be found for topics related to social networks [BMG14] and wireless networks in general [CNS17a]. More generally, decentralized optimization can become crucial for optimization over large datasets [CDAJW10], or using private information [YSVNVQ10].

Decentralized optimization problems have received a lot of attention in the last decade, and have been declined in several variations. First-order methods, which rely on exploiting the gradient of each local cost function, have been largely explored in particular, see *e.g.*, [PN18, SBB<sup>+</sup>17], or [XPNK20] for a survey. More complex techniques have also been considered, such as in [NOR18] which uses tools inspired from integral control.

## 6.2 Decentralized optimization in closed systems

The challenges and solutions related to problems of the type of Definition 6.1 strongly depend on the constraint  $x \in X$ . Several definitions of  $X$  can be found in the literature, depending on the nature of the application that is considered. We present below two variations of Definition 6.1 corresponding to two different characterizations of  $X$ .

### 6.2.1 Consensus constraint

One well-known particularization of decentralized optimization consists in requiring the agents to reach consensus at the same time as minimizing a cost function (for instance in the context of decentralized estimation [RN04]). The corresponding optimization problem then reads

$$\min_{\mathbf{x} \in \mathbb{R}^{nd}} f(\mathbf{x}) = \sum_{i \in \mathcal{V}} f_i(x_i) \quad \text{subject to} \quad x_i = x_j \quad \forall i \neq j. \quad (6.2)$$

The problem above is sometimes introduced as a reformulation of

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x}), \quad (6.3)$$

where the agents each have their own copy of the common state  $\mathbf{x}$  and aim at agreeing on the minimizer of  $f$ . Methods for solving this problem often consist in alternating optimization and consensus steps. One of the simplest and most standard (first-order) methods of this type is the well-known *Decentralized Gradient Descent* (DGD) algorithm [NO09], defined as follows.

**Decentralized Gradient Descent (DGD):** The DGD algorithm consists in successively performing a consensus step and a local gradient descent step. For some sequence of nonnegative step sizes  $(\eta_k)_{k \in \mathbb{N}}$  and some symmetric doubly stochastic matrix  $W$  encapsulating the network topology, it can be summarized as the following update rule for agent  $i$  at iteration  $k$ :

$$\mathbf{x}_i(k+1) = \sum_{j \in \mathcal{V}} [W]_{i,j} \mathbf{x}_j(k) - \eta_k \nabla f_i(\mathbf{x}_i(k)). \quad (6.4)$$

The first term of the expression above corresponds to the consensus component of the iteration, and the second one is a gradient descent step with step size  $\eta_k$ . The update rule (6.4) thus aims at achieving a trade-off between the consensus and the optimization processes. More generally, for closed systems, one can show that by choosing an appropriate decreasing sequence of step sizes  $(\eta_k)_{k \in \mathbb{N}}$ , then one solves (6.2) exactly [SLWY14].

The DGD algorithm is one of the most basic algorithms in the context of decentralized optimization, and has consequently received a lot of attention from various angles of attack, see *e.g.*, [TBA86]. It was typically considered in several of the applications previously mentioned, and stud-

ied for instance in [YLY13] in terms of convergence, or more recently in [CH22], where the authors use it as a case study for an automated worst-case performance analysis. As we will see later, it is also one of the few algorithms for which open systems have been considered.

### 6.2.2 Budget constraint: resource allocation

Another variation of Definition 6.1, which is the focus of this part, is the so-called *Resource allocation* problem, where a fixed amount of budget must be distributed in an optimal way among a set of agents [IK88, Pat08]. This formulation yields the following optimization problem definition.

**Definition 6.2** (Resource allocation). *A Resource Allocation (RA) problem is a decentralized optimization problem in the sense of Definition 6.1 subject to an equality constraint on the weighted sum of the states of the agents with respect to some budget  $\mathbf{b} \in \mathbb{R}^d$  and some positive weight distribution  $\mathbf{a} \in \mathbb{R}_{>0}^n$  (i.e., with  $a_i > 0$  for all  $i$ ), which reads*

$$\min_{\mathbf{x} \in \mathbb{R}^{nd}} f(\mathbf{x}) = \sum_{i \in \mathcal{V}} f_i(\mathbf{x}_i) \quad \text{subject to} \quad \sum_{i \in \mathcal{V}} a_i \mathbf{x}_i = \mathbf{b}. \quad (6.5)$$

Problems of this type can be encountered in various fields of application, including distributed computer systems [KS89], smart grids [DYC21], power systems [YHL16], actuator networks [TASJ13], games [LYH17], or energy distribution [DGCH12]. Consequently, several variations of the formulation above can be found for this problem depending on the considered application. For instance, the budget  $\mathbf{b}$  can sometimes be defined by the agents in the system, i.e.,  $\mathbf{b} = \sum_{i \in \mathcal{V}} \mathbf{d}_i$  for some quantities  $\mathbf{d}_i$ , such as in [YHL16, TASJ13].

Most of the strategies for solving the resource allocation problem are gradient-based, such as in [XB06], where a weighted version of the gradient descent is proposed to maintain the constraint valid. Nevertheless, as for many decentralized optimization problems, the computation of the whole gradient at each iteration requires significant computational capabilities. Hence, alternatives relying only on local interactions have been proposed, such as the *Coordinate Descent* introduced in [Nes12] where each optimization step is performed only along one direction. In the context of multi-agent systems, this amounts to performing the optimization step over one pair of connected agents, and thus only requires the computation of two local gradients per iteration.

In such solution, the choice of the sequence of pairs of agents is crucial for the algorithm to correctly perform. While deterministic versions of this choice can be found in the literature, it is well-known that randomization typically yields convergence guarantees. Hence, the *Random Coordinate Descent* (RCD) algorithm was proposed in [Nec13] where this sequence is chosen randomly, and which is defined as follows.

**Random Coordinate Descent (RCD):** The RCD algorithm consists in randomly choosing two connected agents  $i, j \in \mathcal{V}$  to follow each other's gradient in such way that the constraint is preserved. For some nonnegative step size  $h \geq 0$ , this can be summarized as the following iteration rule, where  $i, j \in \mathcal{V}$  are the interacting agents, and with  $\mathbf{x}_\ell^+ = \mathbf{x}_\ell$  for  $\ell \neq i, j$ :

$$\mathbf{x}_i^+ = \mathbf{x}_i - h \frac{a_j}{a_i^2 + a_j^2} (a_j \nabla f_i(\mathbf{x}_i) - a_i \nabla f_j(\mathbf{x}_j)); \quad (6.6)$$

$$\mathbf{x}_j^+ = \mathbf{x}_j - h \frac{a_i}{a_i^2 + a_j^2} (a_i \nabla f_j(\mathbf{x}_i) - a_j \nabla f_i(\mathbf{x}_i)). \quad (6.7)$$

The iteration rule (6.6)-(6.7) essentially amounts for agents  $i$  and  $j$  to perform a gradient step over the function  $f_i(\mathbf{x}_i) + f_j(\mathbf{x}_j)$  under the constraint that  $a_i \mathbf{x}_i + a_j \mathbf{x}_j$  does not change. Since the other agents keep their estimate constant, this rule ensures the preservation of the constraint during the entire process: It must thus be satisfied at its initialization for the DGD algorithm to converge to the solution of (6.5). This also means that there is no need for the step size  $h$  to decay over the iterations here, by contrast with the DGD algorithm previously introduced where a consensus process was coupled with the optimization one, which thus required a decreasing importance. In the specific case where the agents are homogeneous, *i.e.*, if  $\mathbf{a} = \mathbf{1}_n$ , then (6.6) reduces to  $\mathbf{x}_i^+ = \mathbf{x}_i - \frac{h}{2} (\nabla f_i(\mathbf{x}_i) - \nabla f_j(\mathbf{x}_j))$ , which exactly amounts to following both gradients with equal importance.

**Remark 6.1.** *The update rule (6.6)-(6.7) can be formally obtained under the assumption that the local cost functions are  $\beta$ -smooth by computing*

$$\begin{bmatrix} \mathbf{v}_i \\ \mathbf{v}_j \end{bmatrix} = \arg \min_{\mathbf{s}_i, \mathbf{s}_j \in \mathbb{R}^d: a_i \mathbf{s}_i + a_j \mathbf{s}_j = \mathbf{0}} \left\langle \begin{bmatrix} \nabla f_i(\mathbf{x}_i) \\ \nabla f_j(\mathbf{x}_j) \end{bmatrix}, \begin{bmatrix} \mathbf{s}_i \\ \mathbf{s}_j \end{bmatrix} \right\rangle + \frac{\beta}{2} \left\| \begin{bmatrix} \mathbf{s}_i \\ \mathbf{s}_j \end{bmatrix} \right\|^2. \quad (6.8)$$

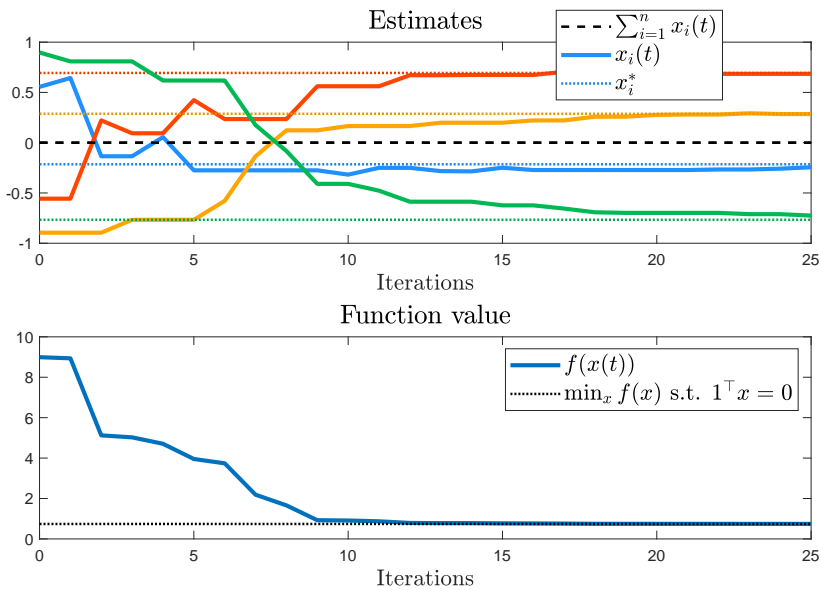
*Solving (6.8) then yields the iteration  $\mathbf{x}_i^+ = \mathbf{x}_i + \mathbf{v}_i$  and  $\mathbf{x}_j^+ = \mathbf{x}_j + \mathbf{v}_j$  that minimizes an upper bound on  $f_i(\mathbf{x}_i) + f_j(\mathbf{x}_j)$  under the assumption that the constraint*

does not change, consistently with the interpretation given above. We refer to [Nec13] for details about this. Moreover, following the approach used in [Nec13], one can show by solving (6.8) that the optimal step size  $h$  for the iteration rule with respect to the function value is given by  $h^* = \frac{1}{\beta}$ .

The RCD algorithm allows reducing the computational complexity by using only pairwise interactions. Moreover, [Nec13] shows that randomly choosing the pair of interacting agents guarantees convergence in terms of the function value for smooth cost functions, as illustrated in Fig. 6.2.1. In particular, let  $f^*$  be the solution of (6.5), then [Nec13, Theorem 4.1] shows that the RCD algorithm applied with  $h = \frac{1}{2\beta}$  yields

$$\mathbb{E} [f(x_k)] - f^* \leq C^k (f(x_0) - f^*), \quad (6.9)$$

where  $C < 1$  is a constant parameter which depends on the properties of the network, of the local cost functions, and on the probability distribution related to the selection of the interacting edge during an RCD step.



**Fig. 6.2.1** Realization of the RCD algorithm in a fully connected system of 4 agents holding 1-dimensional cost functions with  $b = 0$ . Each color in the top plot corresponds to the estimate held by an agent (plain line) and the minimizer it converges to (dotted line). The bottom plot shows the function value.

The RCD algorithm is thus a simple and efficient algorithm solving the resource allocation problem and its variations. Several extensions of it can be found in the literature, such as in [RT14] which studies *block coordinate descent* strategies, namely where several pairs are considered at once.

## 6.3 Decentralized optimization in open systems

Decentralized optimization has been considered in settings subject to time-varying topologies, *i.e.*, where the connections between the agents change over time [SL19, NO13]. Similarly, harsh network models characterized by asynchronous updates with directed communications subject to message delays or losses are considered in [OPS18]: Network-independent results are then derived, and could be applied to time-varying topologies.

However, if these analyses allow for connections between agents to evolve with time, they do not consider the possibility for the agents themselves to change during the process. Yet, as previously discussed, the study of decentralized optimization in open systems as we define them in this thesis, *i.e.*, by considering actual arrivals and departures of agents and aiming at understanding their impact on the system, is becoming crucial regarding several applications nowadays. This is typically the case for federated learning where clients come and go to collaboratively train a model, or can undergo variations in their availability, especially with the ever growing size of such systems [ITW<sup>+</sup>22]. It similarly follows the emergence of *elastic* algorithms for large scale machine learning models, where servers involved in the computations may fail, join or leave [MSWS13].

The possibility for agents to actually join and leave the system in the context of decentralized optimization has started receiving attention very recently. In [HR20] the authors study the stability of the DGD algorithm adapted for systems subject to replacements of agents (and thus of some of the cost functions and estimates), yielding

$$\mathbf{x}_i^+ = \rho \sum_{j \in \mathcal{V}} [W]_{i,j} \mathbf{x}_j - \eta \nabla f_i(\mathbf{x}_i), \quad (6.10)$$

where  $\rho$  and  $\eta$  are the parameters defining the trade-off between optimization in consensus. Observe that in that case no decreasing sequence for the step size  $\eta$  is considered for the optimization steps: As replacements

permanently perturb the system, convergence can never be achieved and reducing the effect of the optimization process would ultimately result in aiming at achieving a consensus process in open system. Arrivals and departures were also considered in [HIMM21], but they were modelled as the activation and deactivation of some of the nodes at each time instant. Finally, [DRFL21] proposes *self-healing* first-order methods, robust to arrivals and departures of agents using tools inspired from control theory. As far as we know, no similar research was conducted for resource allocation problems in open systems, which typically relate to machine learning activities [MMRB20], or to energy distribution problems where a fixed amount of energy must be supplied by devices who might suffer failures or change their operating points due to the environmental conditions. Chapter 7 is thus dedicated to an analysis of the RCD algorithm for solving resource allocation problems in systems subject to replacements of agents. These results are presented in a conference paper [MVH<sup>+</sup>21] whose extension as a journal paper is currently under review [MVH<sup>+</sup>22], following an approach similar to that used in [HR20].

## 6.4 Online optimization

Another interesting line of work related to optimization in open systems is the so-called *online optimization* [Haz16, Bub11]. This framework was first defined in the context of machine learning [Haz06, SS12], and relies on tools borrowed from game theory (in particular the metric). It deals with optimization processes in complex environments which cannot be efficiently modelled, encouraging the use of algorithms learning and adapting as the problem is observed.

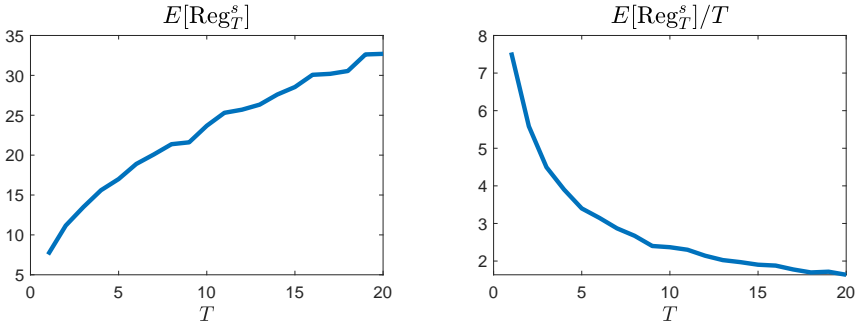
More specifically, in online optimization, an online player iteratively makes decisions while the outcomes of these choices are unknown. After each decision, the player suffers an unknown loss, which can be adversarial or depend on the decision of the player. It can thus be seen as a structured repeated game where the player aims at minimizing a time-varying convex function  $f^t$  by making decisions  $x_t$  without knowing  $f^t$  over a finite number  $T$  of iterations of the game. Observe that the function and decision set must satisfy some conditions, typically in terms of boundedness, in order for the game to be make sense (*i.e.*, to be “fair”).

Following the game-theoretic inspiration of online optimization, an ap-

appropriate metric for analyzing algorithms in that setting is found in game theory: The (static) *regret* is defined as the difference between the total cost paid by the player and that one would get by choosing the best possible fixed decision in hindsight, that is, by knowing the functions  $f^t$ . Typically, for a sequence of estimates  $\{x^t\}_{t=1,\dots,T}$ , the static regret after  $T$  iterations is given by

$$\text{Reg}_T^s := \sum_{t=1}^T f^t(x^t) - \min_{x \in X} \sum_{t=1}^T f^t(x). \tag{6.11}$$

This definition is directly adapted from game theory, typically in the context of stochastic games which involve a repeated game with probabilistic transitions, and where performance is usually measured with respect to the difference between the total payoff of the player and the optimal cost that could have been paid given the knowledge of the game afterwards. For an algorithm generating the sequence  $x^t$ , the static regret corresponds to the accumulated error it achieves with respect to some fixed estimate  $x^* := \min_{x \in X} \sum_{t=1}^T f^t(x)$  over a finite horizon  $T$ . Such algorithm performs well if its regret grows sublinearly with  $T$ , *i.e.*,  $\text{Reg}_T^s = o(T)$  so that  $\lim_{T \rightarrow \infty} \frac{\text{Reg}_T^s}{T} = 0$ , which implies that on average the algorithm ends up performing as well as the best fixed estimate  $x^*$  [Haz16].



**Fig. 6.4.1** Evolution of the static regret (left) and the averaged static regret (right) with  $T$  achieved by the online gradient descent over  $T = 20$  iterations under the constraint that  $x \in \mathcal{B}((,0),1)$  with  $x \in \mathbb{R}^5$  and  $f^t(x) = \sum_{i=1}^5 f_i^t(x_i)$ , where the  $f_i^t$  are random quadratic functions satisfying some conditions. The simulation is averaged over 500 realizations.

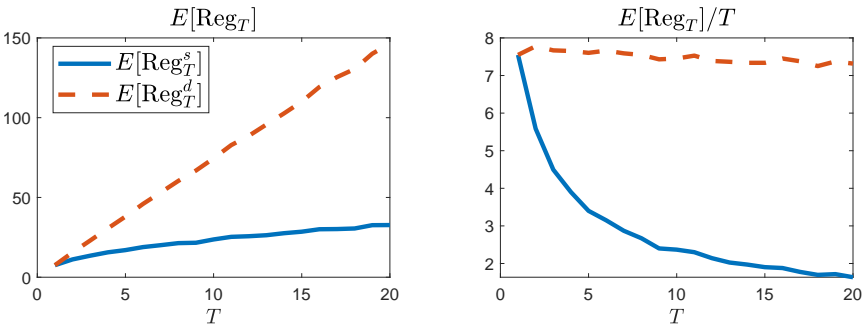
As an illustration, we show in Fig. 6.4.1 the evolution of the regret for the online gradient descent detailed in [Haz16, Section 3.1, Algorithm 6] in a specific setting where the constraint is given by  $x \in \mathcal{B}(0,1)$ . The fig-

ure indeed highlights the expected sublinear growth of the static regret, resulting in the decaying averaged regret over  $T$ .

Ideas from online optimization were used *e.g.*, in [MHP21] in the context of federated learning, or in [HMK<sup>+</sup>21] where the regret is put in perspective with the notion of dynamic coverage. In [SJ16], the mirror descent algorithm is analyzed under the scope of online optimization. The authors of [SJ16] actually study an alternative to the usual static regret called *dynamic regret*, denoted  $\text{Reg}_T^d$ , which evaluates the accumulated error with respect to an optimal trajectory rather than an optimal fixed estimate:

$$\text{Reg}_T^d := \sum_{t=1}^T \left( f^t(x^t) - \min_{x \in X} f^t(x) \right). \quad (6.12)$$

The dynamic regret (6.12) thus differs from the static one (6.11) in the definition of the estimate with which that of the algorithm we consider is compared. In both cases it is defined in hindsight, with the knowledge of the function  $f^t$ , yet whereas the former is fixed over  $T$ , the latter is time-varying, and thus corresponds to the instantaneous optimal estimate for  $f^t$ . By definition, the static regret (6.11) is thus more restrictive than the dynamic one (6.12). Hence, one has  $\text{Reg}_T^s \leq \text{Reg}_T^d$ , and results obtained on the dynamic regret are therefore stronger. This is illustrated in Fig. 6.4.2, where we compare the static and dynamic regrets in the same setting as for Fig. 6.4.1. Observe that in that case the sublinear growth is not achieved for the dynamic regret: This follows its definition, which leaves less room for the algorithm to improve it throughout iterations.



**Fig. 6.4.2** Comparison of the static (plain blue) and dynamic (dashed red) regrets for the same setting as in Fig. 6.4.1 with respect to  $T$ .

The regret (static or dynamic) is a rather natural way to measure the performance of an algorithm in an ever-changing environment, as asymptotic convergence cannot be achieved anyway. Its analysis typically differs from *e.g.*, [HR20], where the focus is on instantaneous performance rather than accumulated errors. More generally, the tools used in online optimization can be exploited and adapted for analyzing optimization problems in open systems, in particular regarding the way it handles time-varying objective functions. In Chapter 8, we use performance metrics related to those used in online optimization to analyze algorithms solving the resource allocation problem with time-varying budget in open systems, and in particular the RCD algorithm. These results appear in [VMH<sup>+</sup>22].

Online optimization is thus an interesting perspective when it comes to analyzing optimization in open systems, as both aim at dealing with the minimization of a time-varying function, *i.e.*,  $\min_{x \in X} f^t(x)$ . Nevertheless, two important aspect should be taken into account regarding online optimization, which make it differ from optimization in open settings: (i) only disruptions in terms of the objective function are considered, and not in terms of the agents themselves; and (ii) very few assumptions are made on the way the objective function can change between iterations (those changes can actually be adversarial), whereas in open systems the changes are characterized by the arrivals and departures of agents, and are thus more structured and “stable”. In that sense, online optimization provides interesting tools, but is not sufficient alone to handle all the challenges inherently arising from open settings.



# 7

## Random Coordinate Descent in OMAS

*The results of this chapter were obtained in collaboration with Renato Vizuite; They were submitted to IEEE Transactions on Automatic Control (TAC) [MVH<sup>+</sup>22]; Moreover a preliminary version was presented at the IEEE Conference on Decision and Control (CDC) 2021 [MVH<sup>+</sup>21], which will receive the 2022 Outstanding Student Paper Prize awarded by the Networks and Communication Systems TC during CDC2022.*

FOR the same reasons as for consensus, decentralized optimization algorithms are poised to never converge in open systems to the (time-varying) optimal solution of the corresponding problem. Instead, the analysis of algorithms in that setting typically aims at showing stability results, such as in [HR20], or the convergence of the error towards some steady-state. In fact, the derivation of such results requires the introduction of assumptions on the cost functions held by the agents in order to condition the impact of replacements on the error. These results then allow characterizing the trade-off between the performance and robustness of algorithms.

In this chapter we focus on the *Resource Allocation* problem (RA) in OMAS, where we restrict to fixed-size systems with strongly convex and smooth functions whose local minimizer is located in a given ball. We ana-

lyze the convergence of the Random Coordinate Descent algorithm in that setting, and show that the expected distance between the estimate and the minimizer reaches some steady-state in expectation. From this we characterize the trade-off between convergence speed for the error and robustness, based on the parameters defining the problem.

We first formulate the problem in Section 7.1, where we properly describe the system and its evolution through replacements and with the RCD algorithm. This section also introduces the assumptions that will be necessary to derive our results. In particular, they require the replacement and optimization processes to be independent, allowing the decoupled analysis of their respective effects on the error. On the one hand, we study the impact of a single replacement on the location of the minimizer, and consequently on the error in Section 7.2. On the other hand, we characterize the linear convergence of each RCD step in Section 7.3. This will require the introduction of an alternative norm that depends on the properties of the underlying network of the system. From these components, we can then obtain stability conditions and characterize the evolution of the error in expectation in Section 7.4.

## 7.1 Problem formulation: open RA with fixed budget

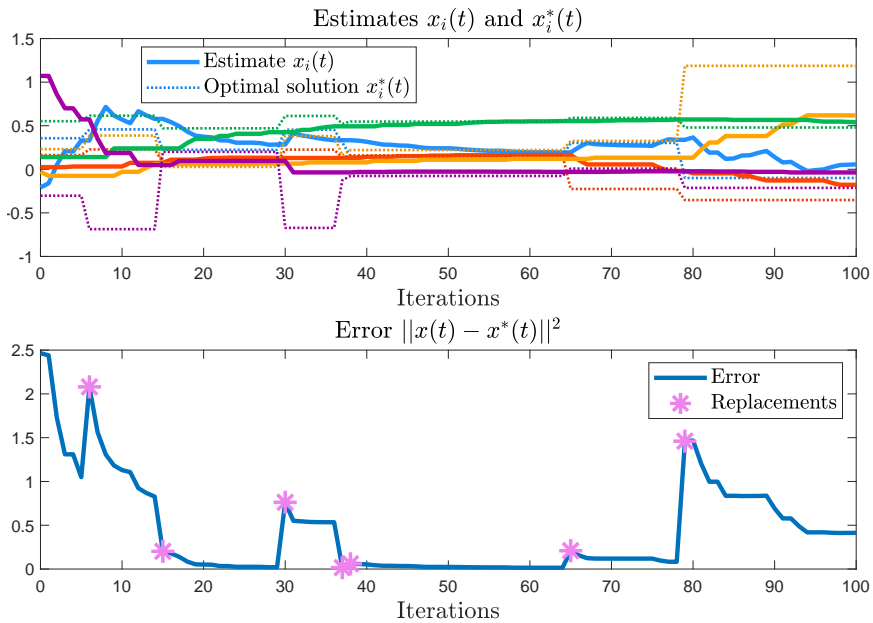
The resource allocation problem was introduced in Definition 6.2. It consists in minimizing a cost function  $f$  separable into local costs  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$  held by the agents, while guaranteeing a budget constraint on the states  $x_i$  for some budget  $\mathbf{b} \in \mathbb{R}^d$  and some nonnegative weight distribution  $\mathbf{a} \in \mathbb{R}_{>0}^n$ . It is formulated as follows:

$$\min_{x \in \mathbb{R}^{nd}} f(x) = \sum_{i \in \mathcal{V}} f_i(x_i) \quad \text{subject to} \quad \sum_{i \in \mathcal{V}} a_i x_i = \mathbf{b}. \quad (7.1)$$

This means that each agent aims at minimizing its cost while meeting the budget constraint, thus requiring some collaboration between them.

In open systems, the agents constituting the system, and hence the local cost functions composing  $f$ , change with time. As a consequence, the solution of (7.1) is a time-varying vector denoted  $x^*(t) \in \mathbb{R}^{nd}$ , which the agents aim at tracking. In the context of distributed energy resources, such changes of the local cost functions can be assimilated to the instantaneous failures experienced by the devices supplying energy, or to the variation in

their operating point because of *e.g.*, environmental conditions. Our goal is to analyze the behavior of the RCD algorithm defined in Section 6.2 for solving (7.1) in systems of fixed size, and whose convergence is prevented by replacements of agents. This is illustrated in Fig. 7.1.1, which shows that agents efficiently track the optimal values with the RCD, but are hindered by replacements.



**Fig. 7.1.1** Realization of the RCD algorithm for solving (7.1) in a system of 5 agents subject to replacements (once every 20 RCD steps on average) as described in Section 7.1: Each color of the top plot represents the estimate held by a given agent (plain line) and the optimal solution it tracks (dotted line); the bottom plot shows the evolution of the distance between the estimates and the minimizer, where each purple star highlights a replacement.

Methods such as the RCD algorithm build on pairwise interactions, which are conditioned by the topology of the underlying network of the system. This feature actually plays an important role in the present analysis. Typically, sparse and dense graphs have drastically different performance [CLRS09]. Moreover, specific graph topologies tend to naturally arise in the design of real systems, such as ring or star topologies which are common in many Ethernet systems [WA11] or some networks whose

design is based on expander graphs [HLW06]. Furthermore, problems of the type of (7.1) are often considered with the assumption that the agents are homogeneous (*i.e.*, that the constraint vector is  $\mathbf{a} = \mathbf{1}_n$ ). This convenient property where each agent has the same importance is however not always true, for instance for energy supply problems where priority can be given *e.g.*, to renewable energy sources [BML13].

The features described above will have a significant impact on both the definition of the RCD algorithm and on the way we obtain our results. Hence, in the remainder of this section, we properly define the properties of the problem and of the system, as well as the assumptions that will be needed to derive our results.

### 7.1.1 Resource allocation problem

We consider the resource allocation problem defined in (7.1), where a budget  $\mathbf{b} \in \mathbb{R}^d$  must be distributed among  $n$  agents in an optimal way with respect to a separable cost function  $f$  and according to a positive weight distribution  $\mathbf{a} \in \mathbb{R}_{>0}^n$  (*i.e.*, with  $a_i > 0$  for all  $i$ ). For the sake of simplicity, we first describe the problem in closed system, and later extend this formulation to open systems of fixed size.

First observe that the constraint  $\sum_{i=1}^n a_i x_i = \mathbf{b}$  from (7.1) can equivalently be written as  $(\mathbf{a}^\top \otimes I_d) \mathbf{x} = \mathbf{b}$ , where the symbol  $\otimes$  denotes the Kronecker product. Then the feasible set of (7.1) is defined as

$$\mathcal{S}_{\mathbf{a},\mathbf{b}} := \left\{ \mathbf{x} \in \mathbb{R}^{nd} : \left( \mathbf{a}^\top \otimes I_d \right) \mathbf{x} = \mathbf{b} \right\}. \quad (7.2)$$

We make the following standard assumption on the local cost functions.

**Assumption 7.1.** *Each local cost function  $f_i$  is continuously differentiable and*

- *$\alpha$ -strongly convex, *i.e.*, for all  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ :*

$$f_i(\mathbf{y}) \geq f_i(\mathbf{x}) + \nabla f_i(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{\alpha}{2} \|\mathbf{y} - \mathbf{x}\|^2; \quad (7.3)$$

- *$\beta$ -smooth, *i.e.*, for all  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ :*

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq \beta \|\mathbf{x} - \mathbf{y}\|. \quad (7.4)$$

More generally, the set of  $d$ -dimensional functions  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$  satisfying these conditions is denoted  $\mathcal{F}_{\alpha,\beta}^d$ .

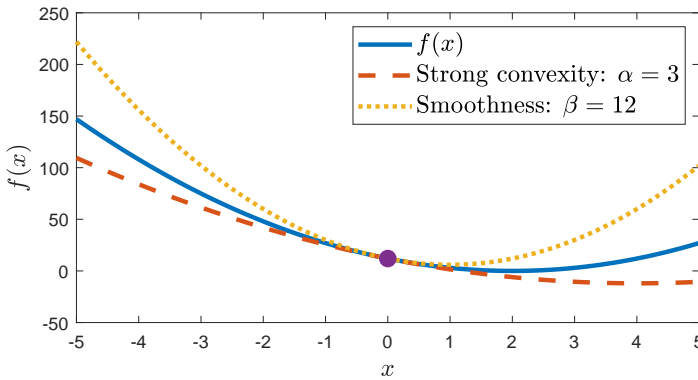
Assumption 7.1 actually provides an upper and a lower bound on the curvature of any function  $f \in \mathcal{F}_{\alpha,\beta}^d$ , as illustrated in Fig. 7.1.2. In particular, for  $f$  twice differentiable, then  $f \in \mathcal{F}_{\alpha,\beta}^d$  if and only if for all  $x \in \mathbb{R}^d$

$$\alpha I_d \preceq \nabla^2 f(x) \preceq \beta I_d. \tag{7.5}$$

We also define the value

$$\kappa := \frac{\beta}{\alpha}, \tag{7.6}$$

called *condition number* of any function  $f \in \mathcal{F}_{\alpha,\beta}^d$ .



**Fig. 7.1.2** Function  $f \in \mathcal{F}_{\alpha,\beta}^1$  with  $\alpha = 3$  and  $\beta = 12$ : At any point (in purple), the curvature of  $f$  is bounded by two parabolas of respective curvatures  $\alpha$  and  $\beta$ .

We already give the following proposition which shows that under Assumption 7.1, then the total cost function in (7.1) satisfies  $f \in \mathcal{F}_{\alpha,\beta}^{nd}$ .

**Proposition 7.2.** *If  $f_1, \dots, f_n \in \mathcal{F}_{\alpha,\beta}^d$  holds in (7.1), then  $f \in \mathcal{F}_{\alpha,\beta}^{nd}$ .*

*Proof.* Since the function  $f$  is separable, we have for all  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{nd}$ :

$$\begin{aligned} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 &= \sum_{i=1}^n \|\nabla f_i(\mathbf{x}_i) - \nabla f_i(\mathbf{y}_i)\|^2 \\ &\leq \beta^2 \sum_{i=1}^n \|\mathbf{x}_i - \mathbf{y}_i\|^2 = \beta^2 \|\mathbf{x} - \mathbf{y}\|^2, \end{aligned}$$

which establishes the  $\beta$ -smoothness of  $f$ . Similarly, for all  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{nd}$ :

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{y}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle &= \sum_{i=1}^n (f_i(\mathbf{x}_i) - f_i(\mathbf{y}_i) - \langle \nabla f_i(\mathbf{x}_i), \mathbf{y}_i - \mathbf{x}_i \rangle) \\ &\geq \sum_{i=1}^n \frac{\alpha}{2} \|\mathbf{y}_i - \mathbf{x}_i\|^2 = \frac{\alpha}{2} \|\mathbf{y} - \mathbf{x}\|^2, \end{aligned}$$

which establishes the  $\alpha$ -strong convexity of  $f$ , and concludes the proof.  $\square$

It follows from Proposition 7.2 (and thus from Assumption 7.1) that the solution of (7.1), denoted  $\mathbf{x}^*$ , is unique. Therefore, for some  $\lambda^* \in \mathbb{R}^d$ , a necessary and sufficient condition for the optimality of  $\mathbf{x}^*$  is the following:

$$\nabla f(\mathbf{x}^*) = \left( \mathbf{a}^\top \otimes I_d \right)^\top \lambda^* = \mathbf{a} \otimes \lambda^*. \quad (7.7)$$

**Remark 7.1.** For the specific case of one-dimensional local cost functions, i.e., where  $f_i : \mathbb{R} \rightarrow \mathbb{R}$  for all  $i$ , then  $b \in \mathbb{R}$ , and the feasible set can be more simply written as  $\mathcal{S}_{\mathbf{a},b} := \{\mathbf{x} \in \mathbb{R}^n : \langle \mathbf{a}, \mathbf{x} \rangle = b\}$ . Similarly, the optimality condition (7.7) can also be written as  $\nabla f(\mathbf{x}^*) = \mathbf{a}\lambda^*$ , for some  $\lambda^* \in \mathbb{R}$ .

### 7.1.2 System description

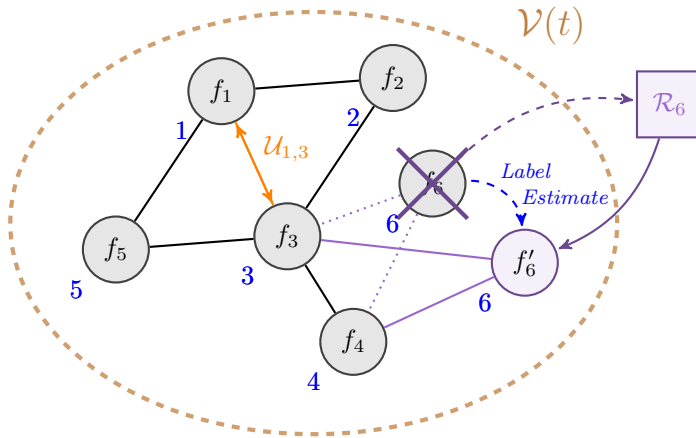
In addition to problem (7.1), we define an undirected and connected graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where the set of nodes  $\mathcal{V} = \{1, \dots, n\}$  corresponds to the agents, and where the set of edges is  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . Each agent  $i \in \mathcal{V}$  has access to a local cost function  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$  and to a local variable  $\mathbf{x}_i \in \mathbb{R}^d$ .

We assume the evolution of the system is event-based in the sense of Section 2.4, meaning that its evolution is characterized by instantaneous modifications triggered by *events*, and remains unchanged in between. In particular, the system is subject to two types of events, described below.

1. A *pairwise update*, denoted  $\mathcal{U}$ , consists in the selection of  $(i, j) \in \mathcal{E}$  with some fixed probability  $p_{ij}$  (such that  $\sum_{(i,j) \in \mathcal{E}} p_{ij} = 1$ ), resulting in  $i$  and  $j$  exchanging information in an undirected manner to update their respective estimates  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . The specific pairwise update between agents  $i$  and  $j$  is denoted  $\mathcal{U}_{i,j}$ , and there holds  $\mathcal{U} = \bigcup_{(i,j) \in \mathcal{E}} \mathcal{U}_{i,j}$ .
2. A *replacement of agent  $i$* , denoted  $\mathcal{R}_i$ , consists in agent  $i$  leaving the system and being immediately replaced by a new joining agent holding a new local cost function  $f'_i$ , resulting in a change of the global

minimizer  $x^*$ . The joining agent retrieves the label  $i$  and estimate  $x_i$  of the leaving agent, so that the constraint in (7.1) remains valid.

The description of replacements above actually corresponds to the attribution of a new local cost function  $f'_i$  to the replaced agent, whereas it keeps its label and estimate unchanged. In the context of energy distribution, this choice actually corresponds to the possible instantaneous variations of cost functions that energy sources can suffer from, due to *e.g.*, environmental conditions, even though the amount of energy they produce remains the same. The system described above is illustrated in Fig. 7.1.3.



**Fig. 7.1.3** Representation of the OMAS considered in this chapter, subject to pairwise updates  $\mathcal{U}_{i,j}$  (in orange) and to replacements  $\mathcal{R}_i$  (in purple), where the joining agent receives a new local cost function  $f'_i$ , and inherits the label  $i$  and the estimate  $x_i$  of the leaving agent.

### 7.1.3 Discretization of the time and reformulation

Based on the system description above, one can define the *event set* of the system containing all the events that can take place in the system as

$$\mathcal{E} := \mathcal{U} \cup \mathcal{R}, \tag{7.8}$$

where  $\mathcal{R} = \bigcup_{i \in \mathcal{V}} \mathcal{R}_i$  is the event that any agent gets replaced. We assume that two distinct events never happen simultaneously. Hence, following the event-based evolution of the system, we consider a discrete evolution

of the time introduced in Section 2.4, such that each time step  $k$  corresponds to the time-instant  $t_k$  at which an event  $\zeta \in \Xi$  takes place. This allows defining the *history* of the process up to iteration  $k$  as

$$\mathcal{H}^k := \{(t_1, \zeta_1), \dots, (t_k, \zeta_k)\}, \quad (7.9)$$

where  $\zeta_i \in \Xi$  for all  $i = 1, \dots, k$ . Moreover, we consider the following assumption, which guarantees that replacements and pairwise updates are independent processes.

**Assumption 7.3.** *The events  $\zeta_i$  constituting the history of the process  $\mathcal{H}^k$  are independent of each other, and of any information in the system prior to time  $t_i$ , such as the estimates or cost functions held by the agents.*

Assumption 7.3 thus guarantees that the events occurring in the system are completely independent of its history and of each other. As a consequence, for any time step  $k$ , the event  $\zeta_k$  is

- a pairwise update (i.e.,  $\zeta_k \in \mathcal{U}$ ) with fixed probability  $p_U$ ;
- a replacement (i.e.,  $\zeta_k \in \mathcal{R}$ ) with fixed probability  $p_R = 1 - p_U$ .

Using the discrete-time evolution of the system described above, and using  $f_i^k$  to denote the local cost function held by agent  $i$  at iteration  $k$ , one can then reformulate (7.1) as

$$\min_{x \in \mathcal{S}_{a,b}} f^k(x) = \sum_{i=1}^n f_i^k(x_i). \quad (7.10)$$

The solution of (7.10) thus potentially differs from an iteration to another, and we denote the instantaneous solution of (7.10) at time step  $k$  by

$$\mathbf{x}^{*,k} := \arg \min_{x \in \mathcal{S}_{a,b}} f^k(x). \quad (7.11)$$

We finally introduce the following assumption, which particularizes Assumption 7.1 following the approach used in [HR20]. It imposes a certain level of uniformity among the local cost functions  $f_i^k$  held by the agents at any time step  $k$ : This prevents arbitrary changes of those functions during replacements, which otherwise could induce arbitrarily large variations in the instantaneous solution  $\mathbf{x}^{*,k}$  of (7.11).

**Assumption 7.4.** For all  $i \in \mathcal{V}$ , the local cost function  $f_i^k$  held by agent  $i$  at any time step  $k$  satisfies  $f_i^k \in \mathcal{F}_{\alpha,\beta}^d$ . Moreover, its minimizer denoted  $\bar{\mathbf{x}}_i^* := \arg \min_{\mathbf{x} \in \mathbb{R}^d} f_i^k(\mathbf{x})$  (where we omit the dependency to  $k$ ) satisfies

$$\bar{\mathbf{x}}_i^* \in \mathcal{B}(\mathbf{0}_d, c) \quad (7.12)$$

for some  $c > 0$ . Moreover, without loss of generality, we assume that

$$f_i^k(\bar{\mathbf{x}}_i^*) = 0. \quad (7.13)$$

More generally, we use to  $\hat{\mathcal{F}}_{\alpha,\beta,c}^d$  to denote the set of all functions satisfying the conditions listed above.

The objective of the agents is therefore to track  $\mathbf{x}^{*,k}$  from (7.11), and our goal is to characterize the evolution of the distance between the estimates of the agents  $\mathbf{x}^k$  and  $\mathbf{x}^{*,k}$ . One possible choice to measure that distance is e.g.,  $\|\mathbf{x}^k - \mathbf{x}^{*,k}\|^2$  (however we will see later that this specific metric is not appropriate when considering general graph topologies for the system).

#### 7.1.4 Random Coordinate Descent (RCD) algorithm

In order to compute the solution of (7.10), we consider that when they perform pairwise updates, the agents implement the RCD algorithm described in Section 6.2. Namely, when agents  $i$  and  $j$  interact at an event  $\mathcal{U}_{i,j}$ , they perform a RCD iteration, resulting in the following update rule for some nonnegative step size  $h \geq 0$ :

$$\mathbf{x}^+ = \mathbf{x} - h\mathbf{Q}^{ij}\nabla f(\mathbf{x}), \quad (7.14)$$

where  $\mathbf{Q}^{ij} \in \mathbb{R}^{nd \times nd}$  is the matrix obtained from  $Q^{ij} \otimes I_d$ , with  $Q^{ij} \in \mathbb{R}^{n \times n}$  the matrix filled with zeroes except for the entries:

$$\begin{aligned} [Q^{ij}]_{i,i} &= \frac{a_j^2}{a_i^2 + a_j^2}; & [Q^{ij}]_{i,j} &= -\frac{a_i a_j}{a_i^2 + a_j^2}; \\ [Q^{ij}]_{j,i} &= -\frac{a_i a_j}{a_i^2 + a_j^2}; & [Q^{ij}]_{j,j} &= \frac{a_i^2}{a_i^2 + a_j^2}. \end{aligned}$$

We also introduce the matrices  $L_p$  and  $\mathbf{L}_p$  which will be used later as follows, and where we remind  $p_{ij}$  is the probability that  $(i,j) \in \mathcal{E}$  are the

interaction agents during a pairwise update:

$$\mathbf{L}_p = \sum_{(i,j) \in \mathcal{E}} p_{ij} \mathbf{Q}^{ij} = \left( \sum_{(i,j) \in \mathcal{E}} p_{ij} \mathbf{Q}^{ij} \right) \otimes I_d = L_p \otimes I_d. \quad (7.15)$$

Observe that by definition there holds

$$L_p \mathbf{a} = \mathbf{Q}^{ij} \mathbf{a} = \mathbf{0}_n, \quad (7.16)$$

meaning that 0 is an eigenvalue of both  $\mathbf{Q}^{ij}$  (for all  $i, j$ ) and  $L_p$ , with the corresponding eigenvector  $\mathbf{a}$ . We respectively denote by  $\lambda_2$  and  $\lambda_n$  the second smallest and largest eigenvalues of  $L_p$ . Using the fact that the graph  $\mathcal{G}$  is connected, one can show that  $\lambda_2 > 0$  (we refer to [Nec13, Lemma3.3] for a detailed proof). Observe that  $L_p$  encapsulates both the properties of the network and the probabilities  $p_{ij}$  to select an edge  $(i, j) \in \mathcal{E}$  during an event  $\mathcal{U}$ , and hence so do  $\lambda_2$  and  $\lambda_n$  which both depend on these.

**Remark 7.2.** For homogeneous agents (i.e.,  $\mathbf{a} = \mathbf{1}_n$ ) and uniformly distributed probabilities  $p_{ij}$ , one can show that  $L_p = \frac{1}{2|\mathcal{E}|} L$ , where we remind  $L$  is the Laplacian matrix of the graph  $\mathcal{G}$  as in Definition 2.3. Hence, we refer to the matrix  $L_p$  as the “scaled Laplacian” of the graph, since it enjoys similar properties, in particular in terms of eigenvalues.

In the subsequent sections, we characterize the evolution in expectation of the distance between the sequence of estimate  $\mathbf{x}^k$  obtained using the RCD algorithm during pairwise updates and the optimal solution  $\mathbf{x}^{*,k}$ . From Assumption 7.3, we can perform this analysis by studying their respective effects on the distance separately, thus allowing the characterization of the evolution of the distance in expectation over a single time step.

## 7.2 Effect of replacements

We first study the effect of replacements on our performance metric. This requires bounding the distance by which the minimizer  $\mathbf{x}^{*,k}$  can change after a single replacement, which we remind amounts to the modification of one local cost function  $f_i^k$ . Hence, we first provide the following preliminary lemmas that will allow deriving our result, and whose proofs are in

Appendix 7.A: Lemma 7.5 studies the location of the whole minimizer  $\mathbf{x}^{*,k}$ , and Lemma 7.6 that of the minimizer  $x_i^{*,k}$  held by a given agent  $i$ .

**Lemma 7.5.** *Let  $\mathbf{x}^* := \arg \min_{\mathbf{x} \in \mathcal{S}_{a,b}} f(\mathbf{x}) = \sum_{i=1}^n f_i(x_i)$ , where all functions  $f_i \in \hat{\mathcal{F}}_{\alpha,\beta,c}^d$  (in the sense of Assumption 7.4), then  $\mathbf{x}^* \in \mathcal{B}(\mathbf{0}_{nd}, R_{\mathbf{b},\kappa})$  with*

$$R_{\mathbf{b},\kappa} = \sqrt{n\kappa} \left( c + \frac{c}{\sqrt{\kappa}} + \frac{\|\mathbf{b}\|}{\sqrt{n}\|\mathbf{a}\|} \right), \quad (7.17)$$

where  $c$  was defined in Assumption 7.4.

*Proof.* The proof is given in Appendix 7.A.1. □

**Lemma 7.6.** *Let  $\mathbf{x}^* := \arg \min_{\mathbf{x} \in \mathcal{S}_{a,b}} f(\mathbf{x}) = \sum_{i=1}^n f_i(x_i)$ , where all functions  $f_i \in \hat{\mathcal{F}}_{\alpha,\beta,c}^d$  (in the sense of Assumption 7.4), then for  $\lambda^*$  defined in (7.7) there holds*

$$\|\lambda^*\| \leq \frac{\beta}{\|\mathbf{a}\|^2} (\|\mathbf{b}\| + c \|\mathbf{a}\|_1); \quad (7.18)$$

and therefore

$$\|\mathbf{x}^*\| \leq \frac{a_i}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c. \quad (7.19)$$

*Proof.* The proof is given in Appendix 7.A.2. □

We can now use these two lemmas to characterize the evolution of the distance between the estimate  $\mathbf{x}^k$  and the minimizer  $\mathbf{x}^{*,k}$  upon a replacement event. Without loss of generality, we assume that agent  $n$  is being replaced, so that for the  $n + 1$  functions  $f_1, \dots, f_{n-1}, f_n^{(1)}, f_n^{(2)} \in \hat{\mathcal{F}}_{\alpha,\beta,c}^d$  (in the sense of Assumption 7.4) we can define  $\mathbf{x}^{(1)}$  and  $\mathbf{x}^{(2)}$  the minimizers respectively before and after the replacement as

$$\begin{aligned} \mathbf{x}^{(1)} &:= \arg \min_{\mathbf{x} \in \mathcal{S}_{a,b}} \left( \sum_{i=1}^{n-1} f_i(x_i) + f_n^{(1)}(x_n) \right); \\ \mathbf{x}^{(2)} &:= \arg \min_{\mathbf{x} \in \mathcal{S}_{a,b}} \left( \sum_{i=1}^{n-1} f_i(x_i) + f_n^{(2)}(x_n) \right). \end{aligned} \quad (7.20)$$

Then we have the following result, proved in Appendix 7.B.

**Proposition 7.7.** *Let  $a_+$  and  $a_-$  respectively denote the largest and smallest values in  $\mathbf{a}$  and let  $\rho_a := \frac{a_+^2}{\|\mathbf{a}\|^2 - a_+^2}$ , where we remind  $\mathbf{a}$  is defined in (7.2), then for*

$\mathbf{x}^{(1)}$  and  $\mathbf{x}^{(2)}$  as defined in (7.20) there holds:

$$\left\| \mathbf{x}^{(1)} - \mathbf{x}^{(2)} \right\|^2 \leq \bar{M}_{n,\kappa}^2 := \min \{ \psi_{n,\kappa}, \chi_{n,\kappa}, \theta_{n,\kappa} \}, \quad (7.21)$$

with

$$\psi_{n,\kappa} = 4n\kappa \left( c + \frac{c}{\sqrt{\kappa}} + \frac{\|\mathbf{b}\|}{\sqrt{n}\|\mathbf{a}\|} \right)^2; \quad (7.22)$$

$$\chi_{n,\kappa} = 8 \left( \frac{a_+}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c \right)^2; \quad (7.23)$$

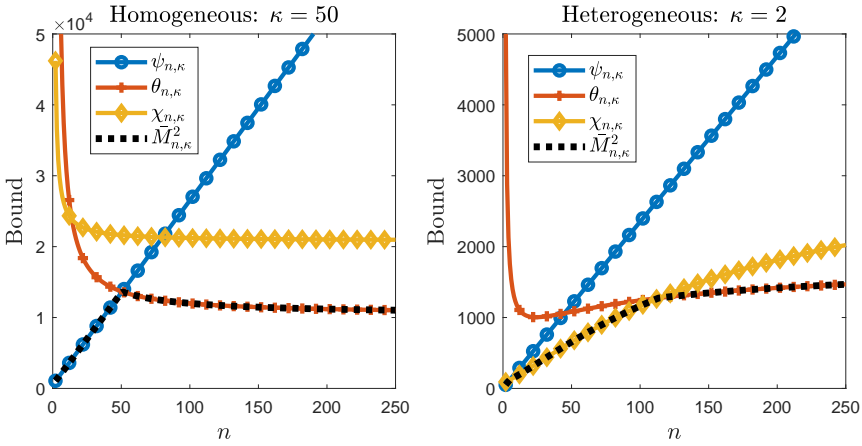
$$\theta_{n,\kappa} = 4 \left( 1 + \frac{(\kappa+1)^2}{4\kappa} \rho_a \right) \left( \frac{a_+}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c \right)^2. \quad (7.24)$$

*Proof.* The proof is given in Appendix 7.B. □

### 7.2.1 Interpretation and illustration

The bound  $\bar{M}_{n,\kappa}^2$  from Proposition 7.7 is obtained by taking the minimum between three quantities:  $\psi_{n,\kappa}$ ,  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$ . The first one is the largest possible distance existing between two minimizers, which follows from the region in which they can be located (defined by Lemma 7.5). The second and third ones are based on the largest possible distance separating the local minimizer of the replaced agent before and after its replacement (obtained using Lemma 7.6). While the derivation of  $\psi_{n,\kappa}$  and  $\chi_{n,\kappa}$  rely on the use of inequalities related to  $\alpha$ -strongly convex functions, that of  $\theta_{n,\kappa}$  also involves the use of additional properties corresponding to  $\beta$ -smooth functions and the determination of the maximum value of a concave function. As a consequence,  $\theta_{n,\kappa}$  shows a strong dependency on the weights of the agents through the coefficient  $\rho_a$ , which is not present in the other two bounds. Interestingly, the bounds  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$  coincide when  $\frac{(\kappa+1)^2}{4\kappa} \rho_a = 1$ .

We illustrate the result of Proposition 7.7 in Fig. 7.2.1. Let  $\bar{a} = \frac{1}{n} \sum_{i=1}^n a_i$  and  $\overline{a^2} = \frac{1}{n} \sum_{i=1}^n a_i^2$  respectively stand for the average value and average of the squared values of  $\mathbf{a}$ . One can more generally highlight the dependencies of the three quantities with the parameters using standard algebraic



**Fig. 7.2.1** Bound obtained with Proposition 7.7 with respect to  $n$  for  $b = 1$  and  $c = 1$ , where we distinguish  $\psi_{n,\kappa}$ ,  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$ . The left plot corresponds to a system with homogeneous agents (*i.e.*,  $a = \mathbb{1}_n$ ) for  $\kappa = 50$ , and the right plot to a system with heterogeneous agents (*i.e.*,  $a_1 = 10$  and  $a_j = 1$  for all  $j > 1$ ) for  $\kappa = 2$ .

manipulations, yielding

$$\psi_{n,\kappa} \leq 4n\kappa \left( 2c + \frac{\|\mathbf{b}\|}{n\bar{a}} \right)^2 = \mathcal{O}(n\kappa);$$

$$\chi_{n,\kappa} \leq 8 \left( \frac{a_+^2}{a_-^2} \left( \frac{\|\mathbf{b}\|}{n\bar{a}} + 2c \right) \kappa \right)^2 = \mathcal{O}(\kappa^2);$$

$$\theta_{n,\kappa} \leq 4 \left( \frac{a_+^2}{a_-^2} \left( \frac{\kappa}{2(n-1)} + 2 \right) \right) \left( \frac{a_+^2}{a_-^2} \left( \frac{\|\mathbf{b}\|}{n\bar{a}} + 2c \right) \kappa \right)^2 = \mathcal{O} \left( \kappa^2 + \frac{\kappa^3}{n} \right).$$

Observe that  $\psi_{n,\kappa}$  scales linearly in both  $n$  and  $\kappa$ . By contrast,  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$  both scale only in  $\kappa$  with a higher order. This suggests that  $\psi_{n,\kappa}$  is tighter for small values of  $n$  and large values of  $\kappa$ , whereas  $\theta_{n,\kappa}$  and  $\chi_{n,\kappa}$  are tighter otherwise.

The main difference between  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$  lies in a multiplicative factor, constant for the former, and depending of the parameters and the values in  $a$  for the latter. Hence,  $\chi_{n,\kappa} < \theta_{n,\kappa}$  only when  $\frac{(\kappa+1)^2}{4\kappa} \rho_a > 1$ , *i.e.*, for large values of  $\kappa$  and small values of  $n$ . This is however significant only for heterogeneous agents; In the homogeneous case under the same condition,

which then becomes  $\frac{(\kappa+1)^2}{4\kappa} > n - 1$ , it seems there always holds

$$\psi_{n,\kappa} = 4n\kappa \left( c \left( 1 + \frac{1}{\sqrt{\kappa}} \right) + \frac{\|b\|}{n} \right)^2 \leq 8 \left( \frac{\kappa}{n} \|b\| + c(\kappa + 1) \right)^2 = \chi_{n,\kappa}.$$

This last inequality was however only empirically observed so far, and a theoretical proof remains to be obtained.

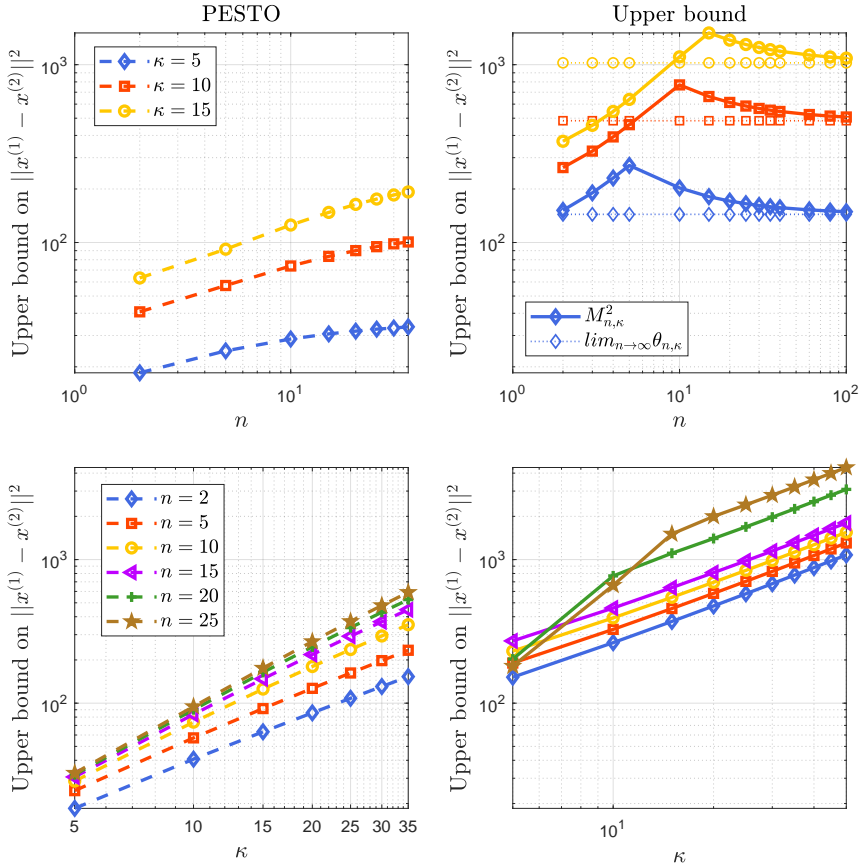
**Remark 7.3.** *The interpretation of the quantities  $\psi_{n,\kappa}$ ,  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$  from Proposition 7.7 actually strongly depends on the implicit assumption that  $\|b\|$  is fixed and  $\|a\|_1$  scales with  $n$  (i.e.,  $\bar{a}$  is fixed). This particular modelling choice is arbitrary, and implies that the solution held by an agent  $x_i^{*,k}$  becomes smaller for large values of  $n$ . Other choices might have different implications on the interpretation, and in particular on the scaling of these quantities. For instance one could choose to either fix  $\|b\|$  and  $\|a\|_1$ , or that both  $\|b\|$  and  $\|a\|_1$  scale with  $n$ , so that the  $x_i^{*,k}$  remain mostly the same no matter  $n$  (observe that the latter yields the same scalings than those presented above).*

## 7.2.2 PEP analysis using the PESTO toolbox

The result of Proposition 7.7 can also be analyzed with respect to empirical results obtained in the PEP framework (“Performance Estimation Problems”), and more specifically with the PESTO toolbox [THG17], which allows the computation of exact empirical bounds for questions related to convex functions. Using PESTO, we can obtain a numerical upper bound for a generalization of  $\left\| x^{(1)} - x^{(2)} \right\|$ . Details can be found in Appendix 7.C.

For the sake of simplicity, we only present here the PESTO results for homogeneous agents, and compare them with the theoretical bounds of Proposition 7.7. As a consequence, the latter do not involve  $\chi_{n,\kappa}$ , which happens to be more significant *e.g.*, for heterogeneous agents. Similar conclusions could nevertheless be drawn the same way for such settings.

We can observe in Fig. 7.2.2 that even though there is some quantitative gap between the theoretical result and that obtained using PESTO, the scaling of the bounds with respect to  $n$  and  $\kappa$  is well captured. In particular, the top-right plot shows that  $\bar{M}_{n,\kappa}^2 = \theta_{n,\kappa}$  when  $n$  becomes large, resulting in the convergence of  $\bar{M}_{n,\kappa}^2$  towards a constant, consistently with the result obtained with PESTO. In parallel, the bottom-left plot shows that the bounds from PESTO end up scaling linearly with  $\kappa$  as it gets large, consistently with  $\bar{M}_{n,\kappa}$ , which becomes  $\psi_{n,\kappa}$  in that case.



**Fig. 7.2.2** Evolution of the upper bound on  $\|x^{(1)} - x^{(2)}\|^2$  for homogeneous agents, respectively with respect to  $n$  for several values of  $\kappa$  (top) and with respect to  $\kappa$  for several values of  $n$  (bottom). For each plot, the results of Proposition 7.7 (right, plain line) are compared with the empirical results obtained using the PESTO toolbox (left, dashed line). The top-right plot also depicts in dotted line the asymptotic value that  $\theta_{n,\kappa}$  should reach according to (7.24).

## 7.3 Effect of RCD steps

We can now study the effect of the second type of event occurring in the system, namely pairwise updates resulting in an RCD iteration. This actually amounts to analyze the convergence of the RCD algorithm in closed system over one iteration.

7.3.1 The  $L_p^\dagger$ -seminorm

Let us introduce the following standard definitions [OR00].

**Definition 7.8** (Linear Convergence). *Let  $\{\mathbf{x}^k\}$  be a sequence of points converging to some fixed point  $\mathbf{x}^* \in \mathbb{R}^d$ . We say the convergence is linear if there exists  $r \in (0, 1)$ , called convergence rate, such that for all  $k$*

$$\|\mathbf{x}^{k+1} - \mathbf{x}^*\| \leq r \|\mathbf{x}^k - \mathbf{x}^*\|. \quad (7.25)$$

**Definition 7.9** (Exponential Convergence). *Let  $\{\mathbf{x}^k\}$  be the sequence of points converging to some fixed point  $\mathbf{x}^* \in \mathbb{R}^d$ . We say the convergence is exponential if there exists  $r \in (0, 1)$  and some positive constant  $C$  such that for all  $k$*

$$\|\mathbf{x}^k - \mathbf{x}^*\| \leq Cr^k. \quad (7.26)$$

In optimization, exponential convergence is also known as *R-linear convergence*. Linear convergence differs from exponential convergence (and is actually stronger) in the sense that it is characterized over an individual iteration of the sequence rather than over its entirety [NW06]. In that sense, it is more suited to our analysis, which requires the characterization of the convergence over one iteration of the RCD algorithm.

In [Nec13], the authors prove linear convergence for the RCD algorithm in closed system with respect to the function value, *i.e.*, for the metric  $f(\mathbf{x}^k) - f(\mathbf{x}^*)$ . Therefore, using the properties of strongly convex and smooth functions [Nes18, Bub15], exponential convergence with respect to the distance between the estimate and the minimizer can directly be

proved starting from [Nec13, Theorem 4.1], resulting in

$$\mathbb{E} \left[ \left\| \mathbf{x}^k - \mathbf{x}^* \right\| \right] \leq \kappa \left( 1 - \frac{\alpha}{\beta} \lambda_2 \right)^k \left\| \mathbf{x}_0 - \mathbf{x}^* \right\|, \quad (7.27)$$

where  $\frac{\alpha}{\beta} \lambda_2 \leq 1$  is guaranteed in the proof of [Nec13, Theorem 4.1]. However, as explained above, this cannot be directly used in our analysis. Linear convergence for the metric above can be obtained for the Euclidean norm under the assumption that the graph is complete with uniform probabilities  $p_{ij}$  and homogeneous agents, as established in [MVH<sup>+</sup>21]. Nevertheless, this is not the case anymore without this assumption, as the following example shows.

**Example 7.10.** Consider a line graph with 3 agents holding one-dimensional local cost functions  $f_i : \mathbb{R} \rightarrow \mathbb{R}$ , and satisfying the constraint  $\langle \mathbb{1}_3, \mathbf{x} \rangle = -3$ . Consider moreover that the probabilities  $p_{ij}$  of activation of the different edges under an update event are given by  $p_{12} = 0.9$ ,  $p_{23} = 0.1$  (and hence  $p_{13} = 0$ ). Finally, let the local cost functions and estimates at iteration  $k$  be given by

$i$	$f_i(x_i)$	$x_i^*$	$x_i^k$
1	$50(x_1 - 2)^2$	2	10
2	$20(x_2 + 2)^2$	-2	7
3	$(x_3 + 3)^2$	-3	-20

Then, starting from  $\mathbf{x}^k$ , the expected result of an iteration of the RCD algorithm with step size  $h = 1/\beta = 0.01$  is

$$\mathbb{E} \left[ \left\| \mathbf{x}^{k+1} - \mathbf{x}^* \right\|^2 \right] = 437.204 > 434 = \left\| \mathbf{x}^k - \mathbf{x}^* \right\|^2, \quad (7.28)$$

meaning that linear convergence cannot be achieved in that case.

For this reason, we consider a different norm to study the convergence of the RCD algorithm, which allows achieving linear convergence. Since the RCD algorithm is performed along a specific interaction network, a natural choice is to consider a norm induced by matrices related to that network, as in [WZGO21]. We thus consider the following seminorm.

**Definition 7.11** ( $\mathbf{L}_p^\dagger$ -seminorm). Let  $\mathbf{L}_p^\dagger$  be the Moore-penrose pseudoinverse (from Definition 2.7) of the matrix  $\mathbf{L}_p$  defined in (7.15). The  $\mathbf{L}_p^\dagger$ -seminorm of a

## 7 | Random Coordinate Descent in OMAS

vector  $x \in \mathbb{R}^{nd}$  is defined as

$$\|x\|_{\mathbf{L}_p^\dagger} := \sqrt{x^\top \mathbf{L}_p^\dagger x}. \quad (7.29)$$

In the next proposition, we show that the  $\mathbf{L}_p^\dagger$ -seminorm is actually a norm on  $\mathcal{S}_{a,0}$ , which we remind is defined in (7.2), and corresponds to the kernel of  $\mathbf{a}^\top \otimes I_d$ . In the specific case where  $d = 1$ ,  $\mathcal{S}_{a,0}$  thus corresponds to the orthogonal complement of  $\mathbf{a}$ .

**Proposition 7.12.** *The  $\mathbf{L}_p^\dagger$ -seminorm  $\|\cdot\|_{\mathbf{L}_p^\dagger}$  is a norm on  $\mathcal{S}_{a,0}$ .*

*Proof.* First, from Definition 7.11 and since  $\mathbf{L}_p^\dagger$  is symmetric there holds

$$\|x\|_{\mathbf{L}_p^\dagger} = 0 \quad \text{iif} \quad x \in \text{Ker}(\mathbf{L}_p^\dagger) := \left\{ x \in \mathbb{R}^{nd} \mid x = \mathbf{a} \otimes w, w \in \mathbb{R}^d \right\}.$$

Second, from (7.2), there holds

$$x \in \mathcal{S}_{a,0} \quad \text{iif} \quad (\mathbf{a}^\top \otimes I_d) x = \mathbf{0}_{nd}.$$

Therefore,  $x \in \text{Ker}(\mathbf{L}_p^\dagger) \cap \mathcal{S}_{a,0}$  if and only if

$$(\mathbf{a}^\top \otimes I_d) x = (\mathbf{a}^\top \otimes I_d) (\mathbf{a} \otimes w) = \|\mathbf{a}\|^2 w = \mathbf{0}_{nd},$$

which happens only if  $w = \mathbf{0}_d$ , and thus for  $x = \mathbf{0}_{nd}$ . Hence, for  $x \in \mathcal{S}_{a,0}$ , one has  $\|x\|_{\mathbf{L}_p^\dagger} = 0$  if and only if  $x = \mathbf{0}_{nd}$ , which concludes the proof.  $\square$

Proposition 7.12 implies that the  $\mathbf{L}_p^\dagger$ -seminorm has all the properties of a norm for any  $z \in \mathcal{S}_{a,0}$ . In particular, for any  $x, y \in \mathcal{S}_{a,b}$  (for any  $b$ ), then  $z = x - y$  satisfies  $z \in \mathcal{S}_{a,0}$ . Therefore, the  $\mathbf{L}_p^\dagger$ -seminorm is a suitable criterion for measuring the distance between any two feasible points.

### 7.3.2 Contraction rate for the RCD algorithm in closed system

Let us remind the update rule of the RCD algorithm (7.14) defined in Section 7.1.4 for some positive step size  $h$  as

$$x^+ = x - h\mathbf{Q}^{ij}\nabla f(x). \quad (7.30)$$

In the following proposition, proved in Appendix 7.D, we analyze the convergence of (7.30) with respect to the  $\mathbf{L}_p^\dagger$ -seminorm from Definition 7.11,

and which we remind is a norm when applied to e.g.,  $\mathbf{x}^k - \mathbf{x}^*$  as stated in Proposition 7.12.

**Proposition 7.13.** Consider the function  $f(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x}_i)$ , and let us define  $\mathbf{x}^* := \arg \min_{\mathbf{x} \in \mathcal{S}_{a,b}} f(\mathbf{x})$ . Consider moreover the update rule (7.30) defined in Section 7.1.4. Then, under Assumption 7.1, for any positive step size

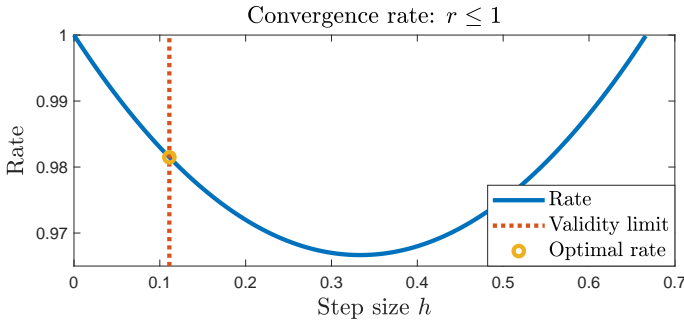
$$h \leq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha + \beta}, \quad (7.31)$$

the update rule (7.30) applied on any initial point  $\mathbf{x} \in \mathcal{S}_{a,b}$  and on the randomly selected pair of agents  $(i, j) \in \mathcal{E}$  satisfies

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^+}^2 \right] \leq \left( 1 - 2h\alpha\lambda_2 + h^2\alpha^2\lambda_n \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^+}^2. \quad (7.32)$$

*Proof.* The proof is given in Appendix 7.D. □

Observe that the rate of convergence in (7.32) is less than one for step sizes satisfying  $h \leq \frac{2\lambda_2}{\alpha\lambda_n}$ . This condition is thus satisfied on all its range of validity, since (7.31) implies  $h \leq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha + \beta} \leq \frac{2\lambda_2}{\alpha\lambda_n}$ . This additional restriction on the step size actually follows the approach used in the proof, and is necessary to obtain (7.32); In Remark 7.5 we discuss the alternative contraction rate for a different range of step sizes, and which thus relies on a different development.



**Fig. 7.3.1** Convergence rate (7.32) with  $\alpha = 1$ ,  $\beta = 5$ ,  $\lambda_2 = 0.1$  and  $\lambda_n = 0.3$  with respect to the step size  $h$  such that the rate is less than 1: The dotted red line limits the validity range of  $h$  according to (7.31).

Hence, as illustrated in Fig. 7.3.1, one can directly determine the optimal step size  $h^*$  in the range defined by (7.31) that minimizes the conver-

gence rate in (7.32), presented in the following corollary.

**Corollary 7.14.** *The convergence rate in (7.32) is minimal under (7.31) for the step size*

$$h^* = \frac{2\lambda_2}{(\alpha + \beta)\lambda_n}, \quad (7.33)$$

which yields

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 \right] \leq \left( 1 - \frac{\lambda_2^2}{\lambda_n \kappa} \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2. \quad (7.34)$$

**Remark 7.4** (Complete graph). *For the particular case of a complete graph with 1-dimensional homogeneous agents and uniform probabilities, the eigenvalues of  $L_p$  are  $\lambda_2 = \lambda_n = \frac{1}{n-1}$  and the  $\mathbf{L}_p^\dagger$ -seminorm then coincides with the Euclidean norm for all  $\mathbf{z} = \mathbf{x} - \mathbf{y}$ , with  $\mathbf{x}, \mathbf{y} \in \mathcal{S}_{a,b}$ . Then, the (7.32) becomes*

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|^2 \right] \leq \left( 1 - \frac{\alpha h}{n-1} (2 - \alpha h) \right) \|\mathbf{x} - \mathbf{x}^*\|^2.$$

In particular, since in that case by definition  $h \leq \frac{2}{\alpha+\beta} \leq \frac{1}{\alpha}$ , it follows that

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|^2 \right] \leq \left( 1 - \frac{\alpha h}{n-1} \right) \|\mathbf{x} - \mathbf{x}^*\|^2,$$

which coincides with [MVH<sup>+</sup>21, Eq. (13)].

**Remark 7.5** (Alternative rate). *Starting from (7.83) in the proof of Proposition 7.13, one can use a similar argument to derive the following alternative convergence rate, valid for  $\frac{\lambda_2}{\lambda_n} \frac{2}{\alpha+\beta} \leq h \leq \frac{\kappa^{-1} + \kappa_L}{\kappa^{-1} + 1} \frac{\lambda_2}{\lambda_n^2 \beta}$ , with  $\kappa_L = \frac{\lambda_n}{\lambda_2}$ :*

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 \right] \leq \left( 1 - 2 \frac{\kappa^{-1} + \kappa_L}{\kappa^{-1} + 1} \beta \lambda_2 h + h^2 \beta^2 \frac{\lambda_n^2}{\lambda_2} \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2. \quad (7.35)$$

This result could be used in the rest of this work in the same way as that of Proposition 7.13 for the corresponding range of step size. This development is however omitted in this work as it is parallel to that we propose and yields similar results.

### 7.3.3 Homogeneous agents with uniform probabilities

As previously mentioned in Remark 7.2, in the particular case where the agents are homogeneous (*i.e.*, when  $\mathbf{a} = \mathbf{1}_n$ ) and where the probabilities

$p_{ij}$  are uniform (i.e., for all  $(i, j) \in \mathcal{E}$ ,  $p_{ij} = p$  for some  $p$ ), then the matrix  $L_p$  such that  $\mathbf{L}_p = L_p \otimes I_d$  can be expressed as  $L_p = \frac{p}{2}L$ , where  $L$  is the usual Laplacian matrix associated to the graph. Moreover, in that case  $L_p$  can also be associated with an electrical circuit [DSPB18], so that the notion of effective resistance from Definition 2.8 can be used to define an alternative range of validity for the step size  $h$ , independently of  $\lambda_2$ .

Hence, the following proposition provides an alternative bound for the convergence of the RCD algorithm in the specific case described above. It can thus be used the same way as Proposition 7.13 in the subsequent results for that case. However, for the sake of generality, we express the main result in the next section only in terms of Proposition 7.13.

**Proposition 7.15.** *In the same setting as that of Proposition 7.13, where we assume  $\mathbf{a} = \mathbb{1}_n$  and  $p_{ij} = p = \frac{1}{|\mathcal{E}|}$  for all  $(i, j) \in \mathcal{E}$ , then for any positive step size*

$$h \leq \frac{2p}{\lambda_n} \frac{2}{\alpha + \beta}, \quad (7.36)$$

the update rule (7.30) applied on any initial point  $\mathbf{x} \in \mathcal{S}_{\mathbf{a}, \mathbf{b}}$  and on the randomly selected pair of agents  $(i, j) \in \mathcal{E}$  satisfies

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^+}^2 \right] \leq \left( 1 - 2h\alpha\lambda_2 + \frac{h^2\alpha^2\lambda_2\lambda_n}{2p} \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^+}^2. \quad (7.37)$$

*Proof.* The proof is given in Appendix 7.E. □

In the same way as for Proposition 7.13, the convergence rate in (7.37) is less than one for  $h \leq \frac{4p}{\alpha\lambda_n}$ , which is satisfied on all its range since  $h \leq \frac{4p}{(\alpha+\beta)\lambda_n} \leq \frac{4p}{\alpha\lambda_n}$  from (7.36). Hence, we can similarly obtain the optimal step size and convergence rate, in the following corollary.

**Corollary 7.16.** *The convergence rate in (7.37) is minimal under (7.36) for the step size*

$$h^* = \frac{4p}{(\alpha + \beta)\lambda_n}, \quad (7.38)$$

which yields

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^+}^2 \right] \leq \left( 1 - \frac{2p\lambda_2}{\lambda_n} \frac{1}{\kappa} \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^+}^2. \quad (7.39)$$

The result obtained in Proposition 7.15 is more suited for graphs with small algebraic connectivity  $\lambda_2$ , i.e., which get easily disconnected [JVM08].

In particular, as soon as  $\lambda_2 \leq 2p$ , then (i) the validity range of the step size  $h$  becomes more favorable in (7.36) as  $\frac{2p}{\lambda_n} \frac{2}{\alpha+\beta} \geq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha+\beta}$ , and (ii) the contraction rate in (7.37) gets better as  $\frac{\lambda_2 \lambda_n}{2p} \leq \lambda_n$ .

In fact, since in the setting of this section one has  $L_p = \frac{p}{2}L$ , there holds  $\lambda_2 = \frac{p}{2}\tilde{\lambda}_2$ , where  $\tilde{\lambda}_2$  is the second smallest eigenvalue of  $L$ . This means that  $\lambda_2 \leq 2p$  is equivalent to  $\tilde{\lambda}_2 \leq 4$ , which corresponds to sparse graphs, *i.e.*, with small algebraic connectivity. We give a few examples for different graph topologies in Table 7.3.1 for  $n = 5$  agents: Whereas the sparse topologies such as the ring, star and line graphs satisfy the condition, it is not the case of the complete graph (and more general denser graphs), for which we will prefer Proposition 7.13 to Proposition 7.15. As the system size increases then this distinction becomes even more clear.

	Complete	Star	Ring	Line
$\lambda_2$	0.25	0.125	0.1382	0.0477
$2p$	0.2	0.5	0.4	0.5
$\tilde{\lambda}_2$	5	1	1.382	0.382

**Table 7.3.1** Values of the parameters  $\lambda_2$ ,  $2p$  and  $\tilde{\lambda}_2$  for several types of network topologies (*i.e.*, complete, star, ring and line graphs, see Example 2.2 for definitions) constituted of 5 agents, where the agents are homogeneous and the probabilities associated to each edge is uniform, *i.e.*,  $p = \frac{1}{|\mathcal{E}|}$ .

## 7.4 Convergence of the error in open system

Using the results of the two previous sections, we now give the following theorem, which is the main result of this chapter and characterizes the convergence of the error in expectation for the RCD algorithm in our open system. Thanks to Assumption 7.3, its derivation relies on the separate analysis of the effects of replacements and RCD iterations, from which we characterize the evolution of the error in expectation.

**Theorem 7.17.** *Let  $M_{n,\kappa} = \frac{1}{\lambda_2} \bar{M}_{n,\kappa}$ , with  $\bar{M}_{n,\kappa}$  defined in (7.21). In the setting of Section 7.1, the iteration rule (7.14) with step size  $h \leq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha+\beta}$  generates a*

sequence of estimates  $\mathbf{x}^k$  which satisfies for any  $\eta > 0$ :

$$\mathbb{E} \left[ \left\| \mathbf{x}^{k+1} - \mathbf{x}^{*,k+1} \right\|_{\mathbf{L}_p^\dagger}^2 \right] - \Gamma_\eta \leq A_\eta \left( \mathbb{E} \left[ \left\| \mathbf{x}^k - \mathbf{x}^{*,k} \right\|_{\mathbf{L}_p^\dagger}^2 \right] - \Gamma_\eta \right), \quad (7.40)$$

with

$$A_\eta := 1 - p_U \alpha h (2\lambda_2 - \alpha \lambda_n h) + (1 - p_U) \frac{M_{n,\kappa}}{\eta}; \quad (7.41)$$

$$\Gamma_\eta := \frac{(1 - p_U) M_{n,\kappa} (\eta + M_{n,\kappa}) \eta}{p_U \eta \alpha h (2\lambda_2 - \alpha \lambda_n h) - (1 - p_U) M_{n,\kappa}}. \quad (7.42)$$

*Proof.* Let us define  $\mathcal{C}^k := \left\| \mathbf{x}^k - \mathbf{x}^{*,k} \right\|_{\mathbf{L}_p^\dagger}^2$ . Moreover, let  $\mathcal{U}^k$  and  $\mathcal{R}^k$  respectively denote the occurrence of a pairwise update and of a replacement at the time step  $k$ . From Assumption 7.3, there holds

$$\mathbb{E} \left[ \mathcal{C}^{k+1} \right] = p_U \mathbb{E} \left[ \mathcal{C}^{k+1} | \mathcal{U}^k \right] + (1 - p_U) \mathbb{E} \left[ \mathcal{C}^{k+1} | \mathcal{R}^k \right]. \quad (7.43)$$

For  $h \leq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha + \beta}$ , Proposition 7.13 yields

$$\mathbb{E} \left[ \mathcal{C}^{k+1} | \mathcal{U}^k \right] \leq \left( 1 - 2\lambda_2 \alpha h + \lambda_n \alpha^2 h^2 \right) \mathbb{E} \left[ \mathcal{C}^k \right]. \quad (7.44)$$

For the replacement case, one has  $\mathbf{x}^{k+1} = \mathbf{x}^k$ , and Proposition 7.7 yields

$$\begin{aligned} \mathbb{E} \left[ \mathcal{C}^{k+1} | \mathcal{R}^k \right] &= \mathbb{E} \left[ \left\| \mathbf{x}^{k+1} - \mathbf{x}^{*,k} + \mathbf{x}^{*,k} - \mathbf{x}^{*,k+1} \right\|_{\mathbf{L}_p^\dagger}^2 \right] \\ &\leq \mathbb{E} \left[ \left( \left\| \mathbf{x}^k - \mathbf{x}^{*,k} \right\|_{\mathbf{L}_p^\dagger} + \left\| \mathbf{x}^{*,k} - \mathbf{x}^{*,k+1} \right\|_{\mathbf{L}_p^\dagger} \right)^2 \right] \\ &= \mathbb{E} \left[ \left( \sqrt{\mathcal{C}^k} + M_{n,\kappa} \right)^2 \right]. \end{aligned} \quad (7.45)$$

Injecting (7.44) and (7.45) into (7.43) then yields the following nonlinear recurrence in  $\mathcal{C}^k$ :

$$\begin{aligned} \mathbb{E} \left[ \mathcal{C}^{k+1} \right] &\leq (1 - p_U \alpha h (2\lambda_2 - \lambda_n \alpha h)) \mathbb{E} \left[ \mathcal{C}^k \right] \\ &\quad + (1 - p_U) \left( 2M_{n,\kappa} \mathbb{E} \left[ \sqrt{\mathcal{C}^k} \right] + M_{n,\kappa}^2 \right). \end{aligned} \quad (7.46)$$

Since  $2x \leq \eta + \frac{x^2}{\eta}$  holds for all  $x \geq 0$  and  $\eta > 0$ , one has  $\sqrt{\mathcal{C}^k} \leq \frac{\eta}{2} + \frac{\mathcal{C}^k}{2\eta}$  for

any  $\eta > 0$ , and hence

$$\begin{aligned} \mathbb{E} [C^{k+1}] &\leq \left( 1 - p_U \alpha h (2\lambda_2 - \lambda_n \alpha h) + (1 - p_U) \frac{M_{n,k}}{\eta} \right) \mathbb{E} [C^k] \\ &\quad + (1 - p_U) M_{n,k} (\eta + M_{n,k}). \end{aligned}$$

Solving the linear recurrence then yields the conclusion. □

Let us define the rate ratio

$$\rho_R := \frac{1 - p_U}{p_U}, \tag{7.47}$$

which characterizes the expected number of replacements happening in the system between two RCD updates. We give the following corollary.

**Corollary 7.18.** *Let  $\bar{\eta} := \rho_R \frac{M_{n,k}}{\alpha h (2\lambda_2 - \lambda_n \alpha h)}$ . For any  $\eta > \bar{\eta}$  one has  $A_\eta < 1$ . Moreover,*

$$\limsup_{k \rightarrow \infty} \mathbb{E} \left[ \left\| \mathbf{x}^k - \mathbf{x}^{*,k} \right\|_{\mathbf{L}_p^+}^2 \right] \leq \Gamma_\eta = \frac{\bar{\eta} (M_{n,k} + \eta)}{1 - \bar{\eta} / \eta}. \tag{7.48}$$

*Proof.* The result directly follows from observing that  $A_\eta < 1$  if and only if

$$p_U \alpha h (2\lambda_2 - \lambda_n \alpha h) + (1 - p_U) M_{n,k} < 0,$$

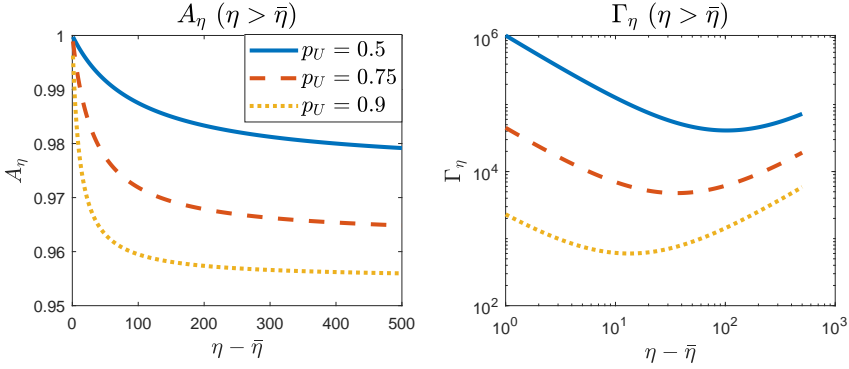
and from rewriting  $\Gamma_\eta$  as

$$\Gamma_\eta = \frac{\rho_R M_{n,k} (M_{n,k} + \eta) \eta}{\alpha h (2\lambda_2 - \lambda_n \alpha h) \eta - \rho_R M_{n,k}} = \frac{\frac{\rho_R M_{n,k}}{\alpha h (2\lambda_2 - \lambda_n \alpha h)} (M_{n,k} + \eta)}{1 - \frac{\rho_R M_{n,k}}{\alpha h (2\lambda_2 - \lambda_n \alpha h) \eta}},$$

which yields the conclusion. □

From Corollary 7.18, one can always choose a value for  $\eta$  such that the contraction rate in (7.40) satisfies  $A_\eta < 1$  (as long as RCD iterations take place, *i.e.*, if  $\rho_R < \infty$ ). Hence, the asymptotic error is guaranteed to be bounded by  $\Gamma_\eta$  as defined in (7.42). We illustrate these quantities in Fig. 7.4.1, which captures their evolution with  $\eta$ .

When  $\eta \rightarrow \infty$ , then the contraction rate  $A_\eta$  becomes minimal but  $\Gamma_\eta$  becomes extremely large, *i.e.*,  $A_\eta \rightarrow 1 - p_U \alpha h (2\lambda_2 - \lambda_n \alpha h)$  and  $\Gamma_\eta \rightarrow \infty$ . Observe moreover that  $\Gamma_\eta$  is convex for  $\eta > \bar{\eta}$ , meaning that one can determine the specific value of  $\eta$  that minimizes the upper bound on the



**Fig. 7.4.1** Evolution of  $A_\eta$  (left) and  $\Gamma_\eta$  (right) as defined in Theorem 7.17 for values of  $\eta > \bar{\eta}$  (with  $\bar{\eta}$  defined in Corollary 7.18) and for several probabilities  $p_U$ : We consider  $\kappa = 5$ ,  $\lambda_2 = \lambda_n = \frac{1}{n-1}$  and some arbitrary value of  $M_{n,\kappa}$ .

asymptotic error  $\Gamma_\eta$ . This is presented in the following corollary, proved in Appendix 7.F.

**Corollary 7.19.** *When  $\eta = \eta^* = \bar{\eta} \left(1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}}\right)$ , then the convergence of the error achieved in expectation by the RCD algorithm in open system is guaranteed with minimal upper bound on the asymptotic error  $\Gamma_{\eta^*}$ , and there holds:*

$$A_{\eta^*} = 1 - p_U \alpha h (2\lambda_2 - \lambda_n \alpha h) \frac{\sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}}}{1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}}}; \quad (7.49)$$

$$\Gamma_{\eta^*} = (\eta^*)^2 = \bar{\eta}^2 \left(1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}}\right)^2, \quad (7.50)$$

where we remind  $\bar{\eta} := \frac{\rho_R M_{n,\kappa}}{\alpha h (2\lambda_2 - \lambda_n \alpha h)}$  from Corollary 7.18.

*Proof.* The proof is given in Appendix 7.F. □

**Remark 7.6.** *The methodology we used in this section can easily be extended to other algorithms than the RCD algorithm. In particular the results of Theorem 7.17, and hence of Corollaries 7.18 and 7.19, can be adapted to any algorithm with linear convergence in closed system, that is, such that*

$$\|x^+ - x^*\|_{\mathbf{L}_p^+}^2 \leq K \|x - x^*\|_{\mathbf{L}_p^+}^2, \quad (7.51)$$

with some positive  $K \leq 1$ . In that case, the same convergence rate as that presented in Theorem 7.17 is obtained with

$$A_\eta := 1 - p_U(1 - K) + (1 - p_U) \frac{M_{n,\kappa}}{\eta}; \quad (7.52)$$

$$\Gamma_\eta := \frac{(1 - p_U)M_{n,\kappa}(\eta + M_{n,\kappa})\eta}{p_U\eta(1 - K) - (1 - p_U)M_{n,\kappa}}. \quad (7.53)$$

Hence, using a similar argument as that from Corollary 7.18, one shows that convergence (i.e.,  $A_\eta < 1$ ) can always be guaranteed if  $K < 1$ . This analysis can be applied e.g., to the results presented in Proposition 7.15 or in Remark 7.5.

### 7.4.1 Interpretation

We provide pieces of interpretation for Theorem 7.17 and its subsequent corollaries, and in particular regarding the expected asymptotic error  $\Gamma_\eta$ .

When replacements become rare (i.e., for  $\rho_R \rightarrow 0$ ), then the system behaves as a closed system. Hence, convergence of the error is achieved with the asymptotic error decaying to  $\Gamma_\eta \rightarrow 0$ , and the contraction rate reduces to that of the RCD algorithm in closed system i.e.,  $A_\eta \rightarrow 1 - \alpha h(2\lambda_2 - \lambda_n \alpha h)$  for all  $\eta > 0$ , consistently with Proposition 7.13. By contrast, with replacements becoming more frequent (i.e., for  $\rho_R > 0$ ), then both  $A_\eta$  and  $\Gamma_\eta$  increase, as the noxious effect of replacements becomes more important (observe  $A_\eta < 1$  remains true for all  $\eta > \bar{\eta}$  as long as  $\rho_R < \infty$ ). In the particular case where  $\rho_R \rightarrow \infty$ , then  $A_{\eta^*} \rightarrow 1$ , and  $\Gamma_{\eta^*} \rightarrow 4\bar{\eta} \rightarrow \infty$ .

Clearly, some conservatism is observed in the results, especially as  $\rho_R$  grows. It mostly follows from Proposition 7.7, which bounds the additive effect of a single replacement on the error. Those thus add up along the iterations; Directly bounding the total sum of those perturbations instead might yield tighter bounds. This is thus one limitation of our model.

Interestingly, within the range of values for  $h$  allowed by Theorem 7.17,  $\Gamma_{\eta^*}$  decays as  $h$  increases so that choosing  $h$  as large as possible yields the smallest upper bound on the expected asymptotic error. This means that the choice of the step size  $h$  is only limited by the analysis of the RCD algorithm in closed system (in Proposition 7.13), and no particular precaution should be taken regarding the open character of the system.

## 7.4.2 Illustration of the results

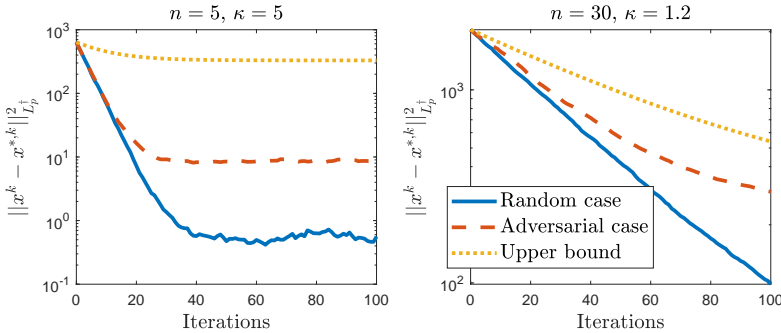
To illustrate the results of Theorem 7.17, we consider the local cost functions held by the agents are piecewise quadratic. Namely, for  $\mu_i \in [-1, 1]$  and  $\varphi_{i1}, \varphi_{i2} \in \left[\frac{\alpha}{2}, \frac{\beta}{2}\right]$ , then the local cost of agent  $i$  is defined as

$$f_i(x_i) = \begin{cases} \varphi_{i1}(x_i - \mu_i)^2, & \text{if } x_i < \mu_i \\ \varphi_{i2}(x_i - \mu_i)^2, & \text{if } x_i \geq \mu_i \end{cases}. \quad (7.54)$$

Such functions clearly satisfy Assumptions 7.1 and 7.4 with  $c = 1$ . Observe that no assumption on the way we choose the local cost function  $f_i$  of a joining agent at replacements is required in the derivation of our results. Hence, we arbitrarily consider two cases for a replacement at time step  $k$ :

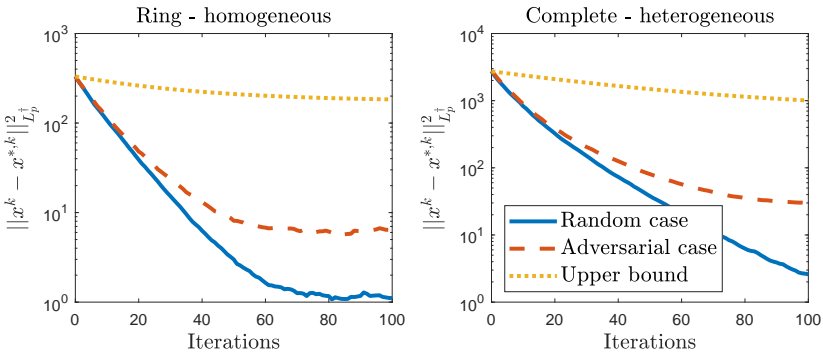
- (1) *random*, where  $\varphi_{i1}$  and  $\varphi_{i2}$  are uniformly randomly chosen in  $\left[\frac{\alpha}{2}, \frac{\beta}{2}\right]$ ;
- (2) *adversarial*, where they are chosen among 100 realizations of the random choice as the one that maximizes  $\left\|x^{k+1} - x^{*,k+1}\right\|_{\mathbf{L}_p^\dagger}^2$ .

Observe that those only concern the choice of the function at replacements; the choice of *the agent  $i$  that is replaced* remains uniformly random.



**Fig. 7.4.2** Performance of the RCD algorithm in the setting of Section 7.1 for a complete graph with with homogeneous agents and uniform probabilities  $p_{ij}$ , where the local cost functions are as defined in (7.54),  $p_U = 0.95$ ,  $b = 1$ , and respectively with (left)  $n = 5$  agents and  $\kappa = 5$  and (right)  $n = 30$  agents and  $\kappa = 1.2$ . The plain blue line and red dashed line correspond to the actual simulated performance of the algorithm averaged over 500 realizations, respectively in the random and adversarial replacements cases. The yellow dotted line corresponds to the upper bound (7.40) obtained with  $A_{\eta^*}$  and  $\Gamma_{\eta^*}$  from Corollary 7.19.

In Fig. 7.4.2 we illustrate the convergence of the RCD algorithm in open systems with respect to our metric for a specific setting with a complete graph, homogeneous agents and uniform probabilities  $p_{ij}$ . We consider two parametrizations for  $\kappa$  and  $n$ , and compare the simulated results with (7.40) using  $A_{\eta^*}$  and  $\Gamma_{\eta^*}$  from Corollary 7.19. The figure shows the convergence of the error, consistently with Corollary 7.18, but also some conservatism inherited from Proposition 7.7, as previously discussed. Moreover, consistently with its interpretation in Section 7.2.1, both parametrizations use different characterizations of  $\bar{M}_{n,\kappa}^2$ :  $\theta_{n,\kappa}$  for  $n = 30$ ,  $\kappa = 1.2$ , and  $\psi_{n,\kappa}$  for  $n = 5$ ,  $\kappa = 5$ . This highlights the impact of those parameters in the tightness of the bound  $\bar{M}_{n,\kappa}^2$ , used to derive our main results. Finally, observe that the bound is tighter for the adversarial case, which is less favorable than the random case, but still encapsulated by our bound. In fact, other cases less favorable than the specific definition of the adversarial one we consider could be defined, for which the bound would be less conservative.



**Fig. 7.4.3** Performance of the RCD algorithm in the setting of Section 7.1, where the local cost functions are as defined in (7.54), with  $n = 5$ ,  $\kappa = 1.2$ ,  $p_U = 0.95$ , and  $b = 1$ , respectively in (left) a ring graph with homogeneous agents, and (right) a complete graph with heterogeneous agents (with  $a_1 = 10$  and  $a_i = 1$  for  $i > 1$ ). The plain blue line and red dashed line correspond to the actual simulated performance of the algorithm averaged over 500 realizations, respectively in the random and adversarial replacements cases. The yellow dotted line corresponds to the upper bound (7.40) obtained with  $A_{\eta^*}$  and  $\Gamma_{\eta^*}$  from Corollary 7.19.

In Fig. 7.4.3 we perform a similar comparison as in Fig. 7.4.2, but for a ring graph with homogeneous agents and a complete graph with heterogeneous agents. Contrasting with the previous illustrations, the ring graph setting implies a different, sparse, topology. This reduces the range of va-

lidity for the step size  $h$  due to the small value of  $\lambda_2$ , which does not scale with  $n$ . Similarly, the heterogeneous setting impacts  $\lambda_2$  and consequently reduces the range of  $h$ , due to the imbalance in  $L_p$ . Those moreover affect the behavior of the norm  $\|\cdot\|_{L_p}^2$ , whose definition relies on  $L_p$ . Furthermore, the heterogeneous setting influences  $\bar{M}_{n,\kappa}$ , which depends on the values in  $\mathbf{a}$ . For all of these reasons, our bound tends to be more conservative in such settings. Nevertheless, even though they differ quantitatively, the results of Fig. 7.4.3 are qualitatively similar to the case of the complete graph with homogeneous agents presented in Fig. 7.4.2.

## 7.5 Discussion and conclusions

In this chapter, we took distance from consensus problems to investigate decentralized optimization problems in open systems. In particular, we focused on the analysis of *resource allocation* problems subject to replacements of agents, resulting in variations of the local cost functions held by the agents in the system. We studied the performance of the Random Coordinate Descent (RCD) algorithm in such setting, whose convergence is prevented because of the replacements. Rather, we showed that the RCD algorithm is stable in that context, and that the expected distance between the estimate it generates and the instantaneous solution of the problem converges to some steady state. We then derived an upper bound for the evolution of that error in expectation, valid for general graph topologies and possibly heterogeneous agents.

The derivation of our results was made possible by the assumption that the RCD algorithm and the replacements in the system are independent processes, allowing their separate analysis. Hence, we studied on the one hand the effect of replacements, resulting in some additive term on the error, and on the other hand that of RCD iterations, which bring the estimate closer to the optimum. These then allow the characterization of the evolution of the error in expectation.

Interestingly, the consensus problem we studied in Part I can be formulated as a resource allocation problem, as

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^n x_i^2 \quad \text{s.t.} \quad \sum_{i=1}^n x_i = \mathbf{b}. \quad (7.55)$$

By imposing  $\mathbf{b} = \sum_{i=1}^n x_i(0)$ , the solution of (7.55) is  $\mathbf{x}^* = \frac{1}{n} \mathbf{1}_n^\top \mathbf{x}(0)$ , and consensus to the average is then achieved. Hence, the consensus problem can be studied through the scope of the resource allocation problem with  $f_i(x_i) = x_i^2$ , and  $\mathbf{a} = \mathbf{1}_n$ . This characterization implies that  $\alpha = \beta = 2$ , and an iteration of the RCD algorithm for agents  $i$  and  $j$  then reads

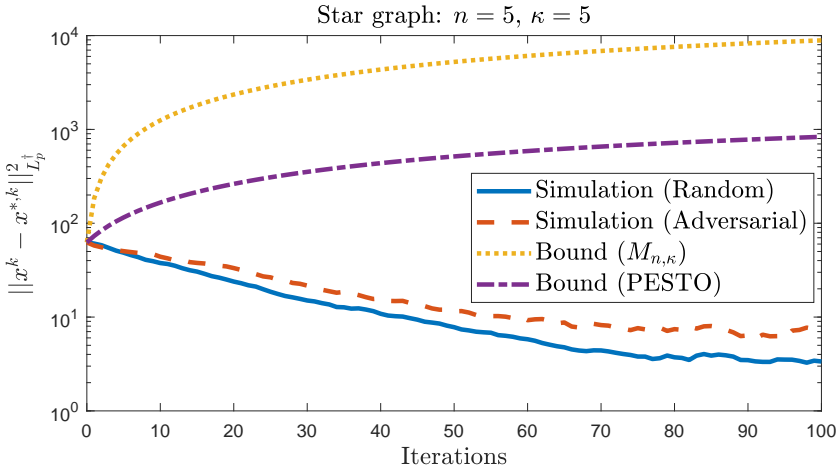
$$x_i^+ = (1 - h)x_i + hx_j; \quad x_j^+ = (1 - h)x_j + hx_i, \quad (7.56)$$

which exactly corresponds to the Gossip algorithm (3.7) if  $h = \frac{1}{2}$ . Proposition 7.13, which characterizes the convergence of the RCD algorithm, can thus be used to study the consensus problem with that specific formulation. It actually requires  $h \leq \frac{\lambda_2}{\lambda_n} \frac{1}{2}$  in that setting: Whereas this allows considering the Gossip algorithm with complete graphs, it reduces the validity range of  $h$  for different topologies, which could potentially induce additional conservatism with respect to dedicated analyses. More generally, one could extend this analysis to open systems using Theorem 7.17; The problem however slightly differs from that studied in *e.g.*, Chapter 5, as the time-varying intrinsic values held by the agents should then be modelled as variations in the budget  $\mathbf{b}$  in the constraint during replacements. This constitutes an interesting line of work to extend and exploit the analyses of this chapter. In fact, such comparisons might allow highlighting the possible room for improvement in our results for specific classes of functions or settings.

Aside from considering time-varying constraints, several other extensions of the results presented in this chapter could be considered. Typically, a natural continuation is the adaptation of our analyses to systems subject to decoupled arrivals and departures: This would however require properly defining how these events impact the estimates held by the agents and the constraint itself, as it could then get violated. Other classes of functions might also be considered, *e.g.*, by considering probability distributions for the local minimizer of each function, or by lifting the strong convexity and smoothness assumptions; These are however at the basis of our analysis, and lifting them would require a consequent adaptation. Finally, another step that can be taken for the future is to consider other types of interactions, *e.g.*, by considering updates over more than one edge per iteration, or simply different algorithms. This is typically encouraged by Remark 7.6, which paves the way towards the generalization of Theorem 7.17 for other algorithms sharing properties with the RCD. More generally, one could also consider algorithms which do not require the constraint to remain

valid all along the process, such as the Dual Ascent algorithm [BPC<sup>+</sup>11], and which might help considering arrivals and departures instead of only replacements.

Finally, a last question to be addressed in this chapter is that of the conservatism in our bounds, which can be observed in the numerical results of Section 7.4.2. This limitation becomes even more clear with sparse graphs, as illustrated in Fig. 7.5.1 where we consider the particularly limiting star graph with  $\kappa = 5$ .



**Fig. 7.5.1** Performance of the RCD algorithm in a similar setting as in Fig. 7.4.3, for a star graph of  $n = 5$  homogeneous agents with  $\kappa = 5$  and  $p_U = 0.95$ . The plain blue and red dashed lines correspond to the simulated performance, respectively in the random and adversarial replacements cases. They are compared with the upper bound (7.40) obtained using  $\bar{M}_{n,\kappa}$  from Proposition 7.21 (in dashed red line) and using the exact numerical result from PESTO (in dash-dotted purple line), where we used  $A_{\eta^*}$  and  $\Gamma_{\eta^*}$  from Corollary 7.19.

This conservatism actually mostly follows from that introduced when bounding the effect of a replacement in proposition 7.7. On the one hand, as the PEP analysis of Section 7.2.2 suggests, there is still some room for improvement regarding the tightness of this bound. This is illustrated in the figure, where we also plot the corresponding upper bound one would obtain with the result of the PESTO computations, which clearly improves its tightness. On the other hand, the figure shows it is not the only source of conservatism into play, which lies in our methodology itself.

Our approach amounts to bounding the effect of every single replacement on the error metric; As a consequence, the upper bound we obtain over several iterations potentially considers the accumulation of the worst possible effect of each individual replacement on the error. This is however very unlikely to happen, as the importance of the effect of a replacement strongly depends on the state of the system when it happens, so that consecutive replacements are expected to counteract each other regarding their effect on the error. This highlights one important limitation of our approach, and in particular of the metric we used throughout this chapter. A potential alternative to address this issue is to *e.g.*, consider another metric which would directly encapsulate the accumulated effect of all replacements at once.

# Appendix of Chapter 7

## 7.A Proofs of Lemmas 7.5 and 7.6 (Minimizer's variation)

This section aims at proving the intermediate results from which we bound the effect of replacements on our error metric. The following results thus study the effect of replacements of a single function on the location of the minimizer, and for all the results that follow, there holds

$$\mathbf{x}^* := \arg \min_{\mathbf{x} \in \mathcal{S}_{a,b}} f(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x}_i). \quad (7.57)$$

### 7.A.1 Proof of Lemma 7.5

We remind Lemma 7.5 reads that  $\mathbf{x}^* \in \mathcal{B}(\mathbf{0}_{nd}, R_{b,\kappa})$ , with

$$R_{b,\kappa} = \sqrt{n\kappa} \left( c + \frac{c}{\sqrt{\kappa}} + \frac{\|\mathbf{b}\|}{\sqrt{n}\|\mathbf{a}\|} \right). \quad (7.58)$$

*Proof of Lemma 7.5.* Let us consider  $\mathbf{x} \in \mathcal{S}_{a,b}$  such that  $\mathbf{x} \notin \mathcal{B}(\mathbf{0}_{nd}, R_{b,\kappa})$ , and let  $\bar{\mathbf{x}}^* = \arg \min_{\mathbf{x} \in \mathbb{R}^{nd}} f(\mathbf{x})$  denote the global minimizer without constraint. By definition, we have  $\|\mathbf{x}\| > R_{b,\kappa}$ . Moreover, since  $\bar{\mathbf{x}}^* \in \mathcal{B}(\mathbf{0}_d, c)^n$ , we have  $\|\bar{\mathbf{x}}^*\| \leq \sqrt{nc}$ . Hence, it follows that

$$\|\mathbf{x} - \bar{\mathbf{x}}^*\| > R_{b,\kappa} - \sqrt{nc}.$$

Since  $f$  is  $\alpha$ -strongly convex, and reminding that  $f(\bar{\mathbf{x}}^*) = 0$  from Assump-

tion 7.4, there holds

$$f(\mathbf{x}) \geq \frac{\alpha}{2} \|\mathbf{x} - \bar{\mathbf{x}}^*\|^2 > \frac{\alpha}{2} \kappa \left( \sqrt{nc} + \frac{\|\mathbf{b}\|}{\|\mathbf{a}\|} \right)^2 = \frac{\beta}{2} \left( \sqrt{nc} + \frac{\|\mathbf{b}\|}{\|\mathbf{a}\|} \right)^2. \quad (7.59)$$

Let us now define  $\mathbf{x}_b := \frac{1}{\|\mathbf{a}\|^2} (\mathbf{a} \otimes I_d) \mathbf{b}$ , which is a feasible point with norm  $\|\mathbf{x}_b\| = \frac{\|\mathbf{b}\|}{\|\mathbf{a}\|}$ . Since  $f$  is  $\beta$ -smooth, and since  $f(\bar{\mathbf{x}}^*) = 0$  from Assumption 7.4, there holds

$$f(\mathbf{x}_b) \leq \frac{\beta}{2} \|\mathbf{x}_b - \bar{\mathbf{x}}^*\|^2 \leq \frac{\beta}{2} \left( \sqrt{nc} + \frac{\|\mathbf{b}\|}{\|\mathbf{a}\|} \right)^2. \quad (7.60)$$

Finally, since  $\mathbf{x}_b \in \mathcal{S}_{\mathbf{a},\mathbf{b}}$  there holds  $f(\mathbf{x}^*) \leq f(\mathbf{x}_b)$  by definition, and combining this with (7.59) and (7.60) then yields

$$f(\mathbf{x}) > \frac{\beta}{2} \left( \sqrt{nc} + \frac{\|\mathbf{b}\|}{\|\mathbf{a}\|} \right)^2 \geq f(\mathbf{x}_b) \geq f(\mathbf{x}^*),$$

which implies that  $\mathbf{x}$  cannot be the minimizer of the problem, and thus concludes the proof.  $\square$

### 7.A.2 Proof of Lemma 7.6

We remind Lemma 7.6 reads that

$$\|\boldsymbol{\lambda}^*\| \leq \frac{\beta}{\|\mathbf{a}\|^2} (\|\mathbf{b}\| + c \|\mathbf{a}\|_1); \quad (7.61)$$

and

$$\|\mathbf{x}_i^*\| \leq \frac{a_i}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c. \quad (7.62)$$

*Proof of Lemma 7.6.* Let us denote  $\bar{\mathbf{x}}_i^*$  the minimizer of  $f_i$  without constraint which satisfies  $f_i(\bar{\mathbf{x}}_i^*) = 0$  (from Assumption 7.4 and  $\nabla f_i(\bar{\mathbf{x}}_i^*) = \mathbf{0}_d$ ). Since all functions  $f_i$  are  $\beta$ -smooth from Assumption 7.1, we have for all  $i$

$$\|\nabla f_i(\mathbf{x}_i^*)\|^2 \leq \beta \langle \nabla f_i(\mathbf{x}_i^*), \mathbf{x}_i^* - \bar{\mathbf{x}}_i^* \rangle.$$

Then, using the optimality condition (7.7), there holds

$$a_i^2 \|\boldsymbol{\lambda}^*\|^2 \leq \beta \langle \boldsymbol{\lambda}^*, a_i(\mathbf{x}_i^* - \bar{\mathbf{x}}_i^*) \rangle.$$

By summing the inequality above over all  $i$ , we then get

$$\|\mathbf{a}\|^2 \|\boldsymbol{\lambda}^*\|^2 \leq \beta \left\langle \boldsymbol{\lambda}^*, \sum_{i=1}^n a_i (\mathbf{x}_i^* - \bar{\mathbf{x}}_i^*) \right\rangle,$$

and from the Cauchy-Schwarz inequality it becomes

$$\|\mathbf{a}\|^2 \|\boldsymbol{\lambda}^*\|^2 \leq \beta \|\boldsymbol{\lambda}^*\| \left\| \sum_{i=1}^n a_i (\mathbf{x}_i^* - \bar{\mathbf{x}}_i^*) \right\|.$$

Hence, using the triangle inequality and the fact that  $\sum_{i=1}^n a_i \mathbf{x}_i^* = \mathbf{b}$ ,

$$\|\boldsymbol{\lambda}^*\| \leq \frac{\beta}{\|\mathbf{a}\|^2} \left( \|\mathbf{b}\| + \left\| \sum_{i=1}^n a_i \bar{\mathbf{x}}_i^* \right\| \right) \leq \frac{\beta}{\|\mathbf{a}\|^2} (\|\mathbf{b}\| + c \|\mathbf{a}\|_1),$$

which corresponds to (7.61). Now, since all functions  $f_i$  are  $\alpha$ -strongly convex from Assumption 7.1, we have for all  $i$  and all  $\mathbf{x}_i$ :

$$\alpha \|\mathbf{x}_i - \bar{\mathbf{x}}_i^*\| \leq \|\nabla f(\mathbf{x}_i)\|.$$

The reverse triangle inequality and the optimality condition (7.7) then yield

$$\|\mathbf{x}_i^*\| \leq \frac{1}{\alpha} \|a_i \boldsymbol{\lambda}^*\| + \|\bar{\mathbf{x}}_i^*\| \leq \frac{a_i}{\alpha} \|\boldsymbol{\lambda}^*\| + c,$$

and (7.62) finally follows from combining the inequality above with (7.61), which concludes the proof.  $\square$

## 7.B Proof of Proposition 7.7 (Effect of replacement)

We now provide the proof of Proposition 7.7, which we remind characterizes the effect of a replacement on the location of the minimizer. For  $\mathbf{x}^{(1)}$  and  $\mathbf{x}^{(2)}$  denoting the minimizers before and after the replacement, Proposition 7.7 then reads the following for specific quantities  $\psi_{n,\kappa}$ ,  $\chi_{n,\kappa}$  and  $\theta_{n,\kappa}$ :

$$\left\| \mathbf{x}^{(1)} - \mathbf{x}^{(2)} \right\|^2 \leq \bar{M}_{n,\kappa}^2 := \min \{ \psi_{n,\kappa}, \chi_{n,\kappa}, \theta_{n,\kappa} \}. \quad (7.63)$$

*Proof.* We prove the inequality for each of the three quantities.

**Proof of  $\psi_{n,\kappa}$ :** We first prove that

$$\left\| \mathbf{x}^{(1)} - \mathbf{x}^{(2)} \right\|^2 \leq \psi_{n,\kappa} = 4n\kappa \left( c + \frac{c}{\sqrt{\kappa}} + \frac{\|\mathbf{b}\|}{\sqrt{n}\|\mathbf{a}\|} \right)^2. \quad (7.64)$$

Observe that  $\mathbf{x}^{(1)}, \mathbf{x}^{(2)} \in \mathcal{B}(\mathbf{0}_{nd}, R_{b,\kappa})$  from Lemma 7.5, so that there holds

$$\left\| \mathbf{x}^{(1)} - \mathbf{x}^{(2)} \right\|^2 \leq 2 \left( \left\| \mathbf{x}^{(1)} \right\|^2 + \left\| \mathbf{x}^{(2)} \right\|^2 \right) \leq 4R_{b,\kappa}^2$$

which yields (7.64).

**Proof of  $\chi_{n,\kappa}$ :** Second, we prove that

$$\left\| \mathbf{x}^{(1)} - \mathbf{x}^{(2)} \right\|^2 \leq \chi_{n,\kappa} = 8 \left( \frac{a_+}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c \right)^2. \quad (7.65)$$

Remember from the optimality condition (7.7) that for  $i = 1, \dots, n-1$ , and for  $q = 1, 2$ , there holds  $\nabla f_i(\mathbf{x}_i^{(q)}) = a_i \boldsymbol{\lambda}^{(q)}$  for some vectors  $\boldsymbol{\lambda}^{(q)}$ . Hence, since the functions  $f_i$  are  $\alpha$ -strongly convex, we have for all  $i = 1, \dots, n-1$ :

$$a_i \left\langle \boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}, \mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)} \right\rangle \geq \alpha \left\| \mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)} \right\|^2.$$

Let us define  $\mathbf{y}^{(q)} \in \mathbb{R}^{d(n-1)}$  the vector such that  $\mathbf{y}_i^{(q)} = \mathbf{x}_i^{(q)}$  for  $q = 1, 2$  and for  $i = 1, \dots, n-1$ . Summing up the above inequalities over  $i = 1, \dots, n-1$ , and using the fact that  $\sum_{i=1}^n a_i \mathbf{x}_i^{(q)} = \mathbf{b} - a_n \mathbf{x}_n^{(q)}$ , we get

$$a_n \left\langle \boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}, \mathbf{x}_n^{(2)} - \mathbf{x}_n^{(1)} \right\rangle \geq \alpha \left\| \mathbf{y}^{(1)} - \mathbf{y}^{(2)} \right\|^2.$$

Using Cauchy-Schwarz inequality and triangle inequality we then obtain

$$\left\| \mathbf{y}^{(1)} - \mathbf{y}^{(2)} \right\|^2 \leq \frac{a_n}{\alpha} \left( \left\| \boldsymbol{\lambda}^{(1)} \right\| + \left\| \boldsymbol{\lambda}^{(2)} \right\| \right) \left( \left\| \mathbf{x}_n^{(1)} \right\| + \left\| \mathbf{x}_n^{(2)} \right\| \right),$$

and using (7.18) and (7.19) from Lemma 7.6 yields

$$\left\| \mathbf{y}^{(1)} - \mathbf{y}^{(2)} \right\|^2 \leq 4 \left( \frac{a_n}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c \right). \quad (7.66)$$

By definition of  $\mathbf{y}^{(q)}$ , we have

$$\|\mathbf{x}^{(1)} - \mathbf{x}^{(2)}\|^2 = \|\mathbf{y}^{(1)} - \mathbf{y}^{(2)}\|^2 + \|\mathbf{x}_n^{(1)} - \mathbf{x}_n^{(2)}\|^2. \quad (7.67)$$

We apply the triangle inequality and (7.19) from Lemma 7.5 to obtain

$$\begin{aligned} \|\mathbf{x}^{(1)} - \mathbf{x}^{(2)}\|^2 &\leq \|\mathbf{y}^{(1)} - \mathbf{y}^{(2)}\|^2 + \left( \|\mathbf{x}_n^{(1)}\| + \|\mathbf{x}_n^{(2)}\| \right)^2 \\ &\leq \|\mathbf{y}^{(1)} - \mathbf{y}^{(2)}\|^2 + 2 \left( \|\mathbf{x}_n^{(1)}\|^2 + \|\mathbf{x}_n^{(2)}\|^2 \right) \\ &\leq \|\mathbf{y}^{(1)} - \mathbf{y}^{(2)}\|^2 + 4 \left( \frac{a_n}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c \right). \end{aligned} \quad (7.68)$$

Finally, (7.65) follows from combining (7.66) and (7.68) and using the fact that  $a_n \leq a_+$  by definition.

**Proof of  $\theta_{n,\kappa}$ :** Finally, we prove that  $\|\mathbf{x}^{(1)} - \mathbf{x}^{(2)}\|^2 \leq \theta_{n,\kappa}$  with

$$\theta_{n,\kappa} = 4 \left( 1 + \frac{(\kappa + 1)^2}{4\kappa} \rho_a \right) \left( \frac{a_+}{\|\mathbf{a}\|^2} \kappa (\|\mathbf{b}\| + c \|\mathbf{a}\|_1) + c \right)^2. \quad (7.69)$$

Since the functions  $f_i$  are  $\alpha$ -strongly convex and  $\beta$ -smooth, there holds that for all  $i = 1, \dots, n-1$ :

$$a_i(1 + \kappa^{-1}) \langle \boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}, \mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)} \rangle \geq \beta^{-1} a_i \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}\|^2 + \alpha \|\mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)}\|^2.$$

Let  $m := \sum_{i=1}^{n-1} a_i^2$ , and remember the definition of  $\mathbf{y}^{(q)}$  from above. By summing up the above inequalities over all  $i = 1, \dots, n-1$ , we get

$$a_n(1 + \kappa^{-1}) \langle \boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}, \mathbf{x}_n^{(2)} - \mathbf{x}_n^{(1)} \rangle \geq m\beta^{-1} \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}\|^2 + \alpha \|\mathbf{y}^{(1)} - \mathbf{y}^{(2)}\|^2.$$

Cauchy-Schwarz inequality then yields:

$$\begin{aligned} a_n(1 + \kappa^{-1}) \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}\| \|\mathbf{x}_n^{(2)} - \mathbf{x}_n^{(1)}\| \\ \geq m\beta^{-1} \|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}\|^2 + \alpha \|\mathbf{y}^{(1)} - \mathbf{y}^{(2)}\|^2, \end{aligned}$$

## 7 | Random Coordinate Descent in OMAS

which can be equivalently written as

$$\alpha \left\| \mathbf{y}^{(1)} - \mathbf{y}^{(2)} \right\|^2 \leq \phi \left( \left\| \boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)} \right\| \right), \quad (7.70)$$

where

$$\phi(z) := -m\beta^{-1}z^2 + a_n(1 + \kappa^{-1}) \left\| \mathbf{x}_n^{(2)} - \mathbf{x}_n^{(1)} \right\| z.$$

Since  $\phi$  is a concave parabola, there exists  $\phi^* = \max_z \phi(z) < \infty$  such that  $\phi(z) \leq \phi^*$  for all  $z$ , and which is given by

$$\phi^* = \frac{a_n^2(1 + \kappa^{-1})^2 \left\| \mathbf{x}_n^{(2)} - \mathbf{x}_n^{(1)} \right\|^2}{4m\beta^{-1}}.$$

Hence, it follows by using (7.70) that

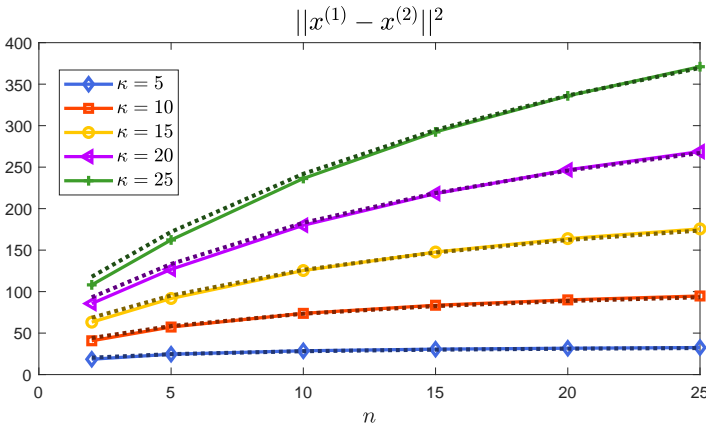
$$\begin{aligned} \left\| \mathbf{y}^{(1)} - \mathbf{y}^{(2)} \right\|^2 &\leq \frac{a_n^2(1 + \kappa^{-1})^2}{\alpha\beta^{-1}} \frac{\left\| \mathbf{x}_n^{(1)} - \mathbf{x}_n^{(2)} \right\|^2}{4m} \\ &= \left( \sqrt{\kappa} + \frac{1}{\sqrt{\kappa}} \right)^2 \frac{a_+^2}{4 \left( \left\| \mathbf{a} \right\|^2 - a_+^2 \right)} \left\| \mathbf{x}_n^{(1)} - \mathbf{x}_n^{(2)} \right\|^2. \end{aligned}$$

Finally, (7.69) follows from (7.67) combined with the inequality above, and from using (7.62) from Lemma 7.6 to bound  $\left\| \mathbf{x}_n^{(1)} \right\|$  and  $\left\| \mathbf{x}_n^{(2)} \right\|$ . This also finally concludes the proof.  $\square$

### 7.C PESTO computations for the effect of replacements

In this section, we provide additional details on the way PESTO was used to obtain the numerical results presented in Section 7.2.2. The PESTO toolbox was initially developed to numerically compute exact worst-case performance of first-order convex optimization algorithms, and more generally allows deriving exact empirical bounds for questions related to convex functions.

To obtain the numerical results of Section 7.C, we considered a general setting with functions  $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$  satisfying Assumptions 7.1 and 7.4, *i.e.*, that are  $\alpha$ -strongly convex and  $\beta$ -smooth, and with  $\arg \min_{\mathbf{x} \in \mathbb{R}^d} f_i(\mathbf{x}) = \mathbf{0}_d$  and  $\min_{\mathbf{x} \in \mathbb{R}^d} f_i(\mathbf{x}) = 0$ . Let us remind that  $\mathbf{x}^{(1)}$  and  $\mathbf{x}^{(2)}$  correspond to the



**Fig. 7.C.1** Result of the PESTO simulation solving (7.71) for several values of both  $n$  and  $\kappa$ , with respect to  $n$  and with  $\|\mathbf{b}\| = 1$  (plain line). The results are compared with the conjecture (7.73) with  $c_1 = c_2 = 1$  (dotted line).

minimizers before and after a replacement, as defined in (7.20). PESTO is then used to evaluate

$$\max \left\| \mathbf{x}^{(2)} - \mathbf{x}^{(1)} \right\|^2 \tag{7.71}$$

under the resource allocation constraint, where we restrict to the case of homogeneous agents (*i.e.*, with  $\mathbf{a} = \mathbf{1}_n$ ). Yet, PESTO does not allow directly implementing this constraint as  $\sum_{i=1}^n x_i^{(1)} = \sum_{i=1}^n x_i^{(2)} = \mathbf{b}$ , but rather as

$$\sum_{i=1}^n x_i^{(1)} = \sum_{i=1}^n x_i^{(2)} = \mathbf{v}_b \tag{7.72}$$

for any vector  $\mathbf{v}_b$  such that  $\|\mathbf{v}_b\| = \|\mathbf{b}\|$ . This means that the problem solved by PESTO is *more general* than that we consider, and consequently the upper bound it generates remains valid, though it is less tight. Observe however that when  $d = 1$ , this constraint is actually exactly the same.

It is important to notice that the variables over which the optimization problem (7.71) is solved by PESTO are *the functions  $f_i$ , the vector  $\mathbf{v}_b$  and the decision variables  $x_i$* . Hence, the result is the worst case over all the instances of the specific problem (7.71).

We show in Fig. 7.C.1 the results obtained with PESTO for our specific problem with  $\|\mathbf{b}\| = 1$ . Additional numerical exploration actually sug-

gests the following conjecture for an analytic expression for the result of the PESTO simulation:

$$\left\| \mathbf{x}^{(1)} - \mathbf{x}^{(2)} \right\|^2 = (\kappa + 1)^2 - \frac{c_1 \kappa^3}{n + \kappa + c_2}, \quad (7.73)$$

for some constants  $c_1, c_2 \in \mathbb{R}$ , which is also presented in the figure.

## 7.D Proof of Proposition 7.13 (Effect of an RCD iteration)

We prove here Proposition 7.13, which establishes the convergence rate of the RCD algorithm in closed system, yielding for  $h \leq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha + \beta}$ :

$$\mathbb{E} \left[ \left\| \mathbf{x}^+ - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2 \right] \leq \left( 1 - 2h\alpha\lambda_2 + h^2\alpha^2\lambda_n \right) \left\| \mathbf{x} - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2. \quad (7.74)$$

*Proof of Proposition 7.13.* From (7.30), there holds

$$\begin{aligned} \mathbb{E} \left[ \left\| \mathbf{x}^+ - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2 \right] &= \sum_{(i,j) \in \mathcal{E}} p_{ij} \left\| \mathbf{x} - h\mathbf{Q}^{ij}\nabla f(\mathbf{x}) - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2 \\ &= \left\| \mathbf{x} - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2 + h^2 \sum_{(i,j) \in \mathcal{E}} p_{ij} \left\| \mathbf{Q}^{ij}\nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 \\ &\quad - 2h \sum_{(i,j) \in \mathcal{E}} p_{ij} \left\langle \mathbf{Q}^{ij}\nabla f(\mathbf{x}), \mathbf{L}_p^\dagger(\mathbf{x} - \mathbf{x}^*) \right\rangle. \end{aligned} \quad (7.75)$$

Since  $\mathbf{L}_p = \sum_{(i,j) \in \mathcal{E}} p_{ij}\mathbf{Q}^{ij}$ , it follows that

$$\begin{aligned} \mathbb{E} \left[ \left\| \mathbf{x}^+ - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2 \right] &= \left\| \mathbf{x} - \mathbf{x}^* \right\|_{\mathbf{L}_p^\dagger}^2 + h^2 \sum_{(i,j) \in \mathcal{E}} p_{ij} \left\| \mathbf{Q}^{ij}\nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 \\ &\quad - 2h \left\langle \mathbf{L}_p\nabla f(\mathbf{x}), \mathbf{L}_p^\dagger(\mathbf{x} - \mathbf{x}^*) \right\rangle. \end{aligned} \quad (7.76)$$

We first treat the second term of the right-hand side of (7.76). Remember from (7.16) that  $Q^{ij}\mathbf{a} = \mathbf{0}_n$ , and from (7.7) that  $\nabla f(\mathbf{x}^*) = \mathbf{a} \otimes \boldsymbol{\lambda}^*$  for some  $\boldsymbol{\lambda}^* \in \mathbb{R}^d$ . Hence, since  $\mathbf{Q}^{ij} = Q^{ij} \otimes I_d$  by definition, there holds

$$\mathbf{Q}^{ij}\nabla f(\mathbf{x}^*) = (Q^{ij} \otimes I_d)(\mathbf{a} \otimes \boldsymbol{\lambda}^*) = (Q^{ij}\mathbf{a}) \otimes \boldsymbol{\lambda}^* = \mathbf{0}_{nd}. \quad (7.77)$$

It thus follows that

$$\begin{aligned} \left\| \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 &= \left\| \mathbf{Q}^{ij} (\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)) \right\|_{\mathbf{L}_p^\dagger}^2 \\ &\leq \frac{1}{\lambda_2} \left\| \mathbf{Q}^{ij} (\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)) \right\|^2, \end{aligned} \quad (7.78)$$

where the inequality follows from the fact that the eigenvalues of  $\mathbf{L}_p$  are exactly those of  $L_p$  repeated  $d$  times (by Theorem 13.12 of [Lau04]), so that the smallest and largest nonzero eigenvalues of  $\mathbf{L}_p^\dagger$  are respectively  $1/\lambda_n$  and  $1/\lambda_2$ , yielding for all  $\mathbf{z} \in \mathbb{R}^{nd}$ :

$$\|\mathbf{z}\|_{\mathbf{L}_p^\dagger}^2 \leq \frac{1}{\lambda_2} \|\mathbf{z}\|^2. \quad (7.79)$$

Therefore, since  $\mathbf{Q}^{ij} = (\mathbf{Q}^{ij})^\top = (\mathbf{Q}^{ij})^2$ , and since  $\|\mathbf{z}\|_{\mathbf{L}_p}^2 \leq \lambda_n \|\mathbf{z}\|^2$  holds for all  $\mathbf{z} \in \mathbb{R}^{nd}$ , there holds from (7.78):

$$\begin{aligned} \sum_{(i,j) \in \mathcal{E}} p_{ij} \left\| \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 &\leq \frac{1}{\lambda_2} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)\|_{\mathbf{L}_p}^2 \\ &\leq \frac{\lambda_n}{\lambda_2} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)\|^2. \end{aligned} \quad (7.80)$$

We now analyze the third term of the right-hand side of (7.76). From (7.77) there holds

$$\mathbf{L}_p \nabla f(\mathbf{x}^*) = \sum_{(i,j) \in \mathcal{E}} p_{ij} \mathbf{Q}^{ij} \nabla f(\mathbf{x}^*) = \mathbf{0}_{nd}, \quad (7.81)$$

yielding

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \rangle \geq \frac{\beta^{-1} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)\|^2}{1 + \kappa^{-1}} + \frac{\alpha \|\mathbf{x} - \mathbf{x}^*\|^2}{1 + \kappa^{-1}}.$$

Hence, using the result above and (7.79), it follows that

$$\begin{aligned} &-2h \left\langle \mathbf{L}_p \nabla f(\mathbf{x}), \mathbf{L}_p^\dagger (\mathbf{x} - \mathbf{x}^*) \right\rangle \\ &\leq -2h \frac{\beta^{-1} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)\|^2}{1 + \kappa^{-1}} - 2h \frac{\alpha \lambda_2 \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2}{1 + \kappa^{-1}}. \end{aligned} \quad (7.82)$$

Injecting (7.80) and (7.82) into (7.75) yields

$$\begin{aligned} \mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 \right] &\leq \left( 1 - 2h \frac{\alpha \lambda_2}{1 + \kappa^{-1}} \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 \\ &\quad + \left( h^2 \frac{\lambda_n}{\lambda_2} - 2h \frac{\beta^{-1}}{1 + \kappa^{-1}} \right) \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)\|^2. \end{aligned} \quad (7.83)$$

Observe that if  $h \leq \frac{\lambda_2}{\lambda_n} \frac{2}{\alpha + \beta}$  then  $h^2 \frac{\lambda_n}{\lambda_2} - 2h \frac{\beta^{-1}}{1 + \kappa^{-1}} \leq 0$ , and hence we can use the definition of  $\alpha$ -strongly convex functions to find that

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*)\|^2 \geq \alpha^2 \lambda_2 \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2,$$

where we used the fact that  $\|\mathbf{x} - \mathbf{x}^*\|^2 \geq \lambda_2 \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2$ . It follows that

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 \right] \leq \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 + (h^2 \alpha^2 \lambda_n - 2h \alpha \lambda_2) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2,$$

which concludes the proof.  $\square$

## 7.E Proof of Proposition 7.15 (Complete homogeneous case)

We prove here Proposition 7.15, which establishes the convergence rate of the RCD algorithm in closed system for homogeneous agents and uniform probabilities  $p_{ij}$ , yielding for  $h \leq \frac{2p}{\lambda_n} \frac{2}{\alpha + \beta}$ :

$$\mathbb{E} \left[ \|\mathbf{x}^+ - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2 \right] \leq \left( 1 - 2h \alpha \lambda_2 + \frac{h^2 \alpha^2 \lambda_2 \lambda_n}{2p} \right) \|\mathbf{x} - \mathbf{x}^*\|_{\mathbf{L}_p^\dagger}^2. \quad (7.84)$$

*Proof of Proposition 7.15.* The proof is the same as that of Proposition 7.13 up to (7.76). Since the matrices  $\mathbf{Q}^{ij}$  are idempotent, the summation term of the second element in (7.76) can be expressed as:

$$\left\| \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 = \left\langle \mathbf{Q}^{ij} \nabla f(\mathbf{x}), \mathbf{Q}^{ij} \mathbf{L}_p^\dagger \mathbf{Q}^{ij} \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\rangle.$$

Then we can use an upper bound for the quadratic form to obtain

$$\left\| \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 \leq \left\| \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\|^2 \lambda_{\max}(\mathbf{Q}^{ij} \mathbf{L}_p^\dagger \mathbf{Q}^{ij}).$$

Now, the matrix  $\mathbf{Q}^{ij}\mathbf{L}_p^\dagger\mathbf{Q}^{ij}$  is given by

$$\mathbf{Q}^{ij}\mathbf{L}_p^\dagger\mathbf{Q}^{ij} = \left(\mathbf{Q}^{ij}\mathbf{L}_p^\dagger\mathbf{Q}^{ij}\right) \otimes \mathbf{I}_d,$$

which implies that  $\lambda_{\max}(\mathbf{Q}^{ij}\mathbf{L}_p^\dagger\mathbf{Q}^{ij}) = \lambda_{\max}(\mathbf{Q}^{ij}\mathbf{L}_p^\dagger\mathbf{Q}^{ij})$ . Then, we have

$$\mathbf{Q}^{ij}\mathbf{L}^\dagger\mathbf{Q}^{ij} = \frac{1}{2} \left( [\mathbf{L}_p^\dagger]_{i,i} + [\mathbf{L}_p^\dagger]_{j,j} - 2[\mathbf{L}_p^\dagger]_{i,j} \right) \mathbf{Q}^{ij} = \frac{1}{2} r_{i,j}^{\text{eff}} \mathbf{Q}^{ij},$$

where  $r_{i,j}^{\text{eff}}$  is the *effective resistance* between the agents  $i$  and  $j$  from (2.7) in Definition 2.8. Since there is an edge between  $i$  and  $j$ , we have  $r_{i,j}^{\text{eff}} \leq \frac{1}{p}$ . Then one obtains that

$$\lambda_{\max}(\mathbf{Q}^{ij}\mathbf{L}^\dagger\mathbf{Q}^{ij}) \leq \frac{1}{2p} \text{ for all } (i,j) \in \mathcal{E}, \quad (7.85)$$

and it follows

$$\begin{aligned} \sum_{(i,j) \in \mathcal{E}} p_{ij} \left\| \mathbf{Q}^{ij} \nabla f(\mathbf{x}) \right\|_{\mathbf{L}_p^\dagger}^2 &\leq \frac{1}{2p} \left\| \nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*) \right\|_{\mathbf{L}_p^\dagger}^2 \\ &\leq \frac{\lambda_n}{2p} \left\| \nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^*) \right\|^2. \end{aligned} \quad (7.86)$$

The remainder of the proof then follows the exact same steps as in that of Proposition 7.13, where we use (7.86) instead of (7.80).  $\square$

## 7.F Proof of Corollary 7.19 (Minimal upper bound)

We prove here Corollary 7.19, which we reminds establishes the minimal upper bound on the asymptotic error achieved by the RCD algorithm in open systems when  $\eta = \eta^* = \bar{\eta} \left( 1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}} \right)$ , and yielding

$$A_{\eta^*} = 1 - p_U \alpha h (2\lambda_2 - \lambda_n \alpha h) \frac{\sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}}}{1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}}}; \quad (7.87)$$

$$\Gamma_{\eta^*} = (\eta^*)^2 = \bar{\eta}^2 \left( 1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}} \right)^2, \quad (7.88)$$

*Proof of Corollary 7.19.* Observe that  $\Gamma_\eta$  is convex for  $\eta > \bar{\eta}$ , and there holds  $\frac{d}{d\eta}\Gamma|_{\eta=\eta^*} = 0$  with  $\eta^* = \arg \min_{\eta > \bar{\eta}} \Gamma_\eta$ . Hence we compute

$$\frac{d}{d\eta}\Gamma_\eta = \frac{\bar{\eta}}{(\eta - \bar{\eta})^2}(\eta^2 - 2\bar{\eta}\eta - M_{n,\kappa}\bar{\eta}) = 0, \quad (7.89)$$

which is satisfied for

$$\eta_1^* = \bar{\eta} + \sqrt{\bar{\eta}^2 + M_{n,\kappa}\bar{\eta}} \quad ; \quad \eta_2^* = \bar{\eta} - \sqrt{\bar{\eta}^2 + M_{n,\kappa}\bar{\eta}}.$$

Since  $\eta_2^* < \bar{\eta}$ , it must be rejected, and it follows that

$$\eta^* = \bar{\eta} + \sqrt{\bar{\eta}^2 + M_{n,\kappa}\bar{\eta}} = \bar{\eta} \left( 1 + \sqrt{1 + \frac{M_{n,\kappa}}{\bar{\eta}}} \right).$$

We can then compute

$$A_{\eta^*} = 1 - p_U \alpha h (2\lambda_2 - \lambda_n \alpha h) + \frac{(1 - p_U) M_{n,\kappa}}{\bar{\eta} \left( 1 + \sqrt{1 + M_{n,\kappa}/\bar{\eta}} \right)},$$

and a few algebraic manipulations yield (7.87). Observe now that  $\Gamma_\eta = \eta^2$  if and only if

$$\eta^2 - 2\bar{\eta}\eta - M_{n,\kappa}\bar{\eta} = 0,$$

which is equivalent to (7.89) for  $\eta > \bar{\eta}$ , so that the solution is  $\eta^*$ . Hence,  $\Gamma_{\eta^*} = (\eta^*)^2$ , which yields (7.88).  $\square$

# 8

## An Online Approach for Resource Allocation in OMAS

*The results of this chapter were obtained in collaboration with Renato Vizuete; They were accepted and will be presented at the IEEE Conference on Decision and Control (CDC) 2022 [VMH<sup>+</sup>22].*

**T**HIS last chapter lies in the continuity of the previous one, as we consider a variation of the same problem where the budget to be allocated varies with the replacements. Motivated by problems in the context of *e.g.*, power systems or task distribution, such settings usually assume that the cost to minimize must be paid regularly. This typically induces a shift in the metrics that are used to analyze the performance of algorithms to directly consider the total cost accumulated over several iterations. Studying problems of this type with such metrics thus allows considering the total effect of arrivals and departures of agents over all the iterations at once, yielding a different characterization of the performance of algorithms as that conducted in Chapter 7.

In this chapter we focus on resource allocation problems subject to replacements of agents, where the budget to be allocated is defined by the

agents in the system and thus varies with replacements as well. This variation in the problem definition actually requires defining a way to handle the possible violations of the resource allocation constraint, especially during replacements. We analyze the performance of algorithms, and in particular of the RCD algorithm, by studying the error accumulated over the iterations with respect to some prescribed strategies. Such metrics happen to be strongly connected with usual metrics from online optimization, from which we thus retrieve tools.

In Section 8.1, we first establish the problem formulation, and in particular we properly describe the system and define the different assumptions that we will consider. Then, in Section 8.2 we introduce the metrics we will use to analyze the performance of algorithms, and which are based on the definition of reference strategies. Finally, taking advantage of these metrics to directly study the accumulated impact of replacements on the error, we derive upper bounds for them in Section 8.3, from which we characterize the performance of the RCD algorithm.

## 8.1 Problem formulation: open RA with varying budget

We consider the alternative formulation of the resource allocation problem from Definition 6.2 where each agent holds some quantity  $d_i \in \mathbb{R}^d$ , called *demand* of agent  $i$ , from which we build the budget  $\mathbf{b}$  in the constraint. With homogeneous agents, the problem then reads

$$\min_{x \in \mathbb{R}^{nd}} f(x) = \sum_{i=1}^n f_i(x_i) \quad \text{subject to} \quad \sum_{i=1}^n x_i = \sum_{i=1}^n d_i. \quad (8.1)$$

It thus amounts for the agents to minimize the total cost  $f$  while ensuring that the total demand  $\mathbf{b} = \sum_{i=1}^n d_i$  is met. Formulations of this type typically arise in the context of actuator networks [TASJ13] or power systems [YHL16]: The quantity  $d_i$  then typically corresponds to the local resource available to agent  $i$ , so that  $\mathbf{b} = \sum_{i=1}^n d_i$  corresponds to the total network resource which must be allocated. Similar settings include for instance task distribution problems in *e.g.*, processor networks, where each processor is assigned an amount of tasks  $d_i$  to fulfill and holds some cost function  $f_i$  defining its individual performance; Their goal is then to share the total workload  $\sum_{i=1}^n d_i$  in order to optimize their global performance.

In open systems, not only do the local cost functions  $f_i$  in (8.1) change with time, but also the demands  $d_i$ , and as a consequence the total budget  $\mathbf{b}$ . Problems involving time-varying budgets were explored *e.g.*, in [BSFH18, WSR22], but usually with only quadratic functions, and not in the context of open systems where these variations follow those of the system composition. Analyses inspired from online optimization using metrics measuring the level of violation of time-varying optimization constraints can also be found *e.g.*, in [YLY<sup>+</sup>21].

In the remainder of this section, we properly define the properties of the problem and system we consider, as well as the nature of the interactions happening in the system in order to solve the problem.

### 8.1.1 General formulation

We consider the variation of the resource allocation problem in (8.1) where we restrict to nonnegative states  $x_i \in \mathbb{R}_+^p$ . Moreover, we consider the following assumption, which particularizes Assumption 7.1 from Chapter 7.

**Assumption 8.1.** *The local cost functions  $f_i$  in (8.1) satisfy  $f_i : \mathbb{R}_+^p \rightarrow \mathbb{R}$  and  $f_i \in \mathcal{F}_{\alpha,\beta}^p$  (in the sense of Assumption 7.1, i.e., they are continuously differentiable,  $\alpha$ -strongly convex and  $\beta$ -smooth over  $\mathbb{R}_+^p$ ). Moreover, they satisfy*

$$\arg \min_{x \in \mathbb{R}_+^p} f_i(x) = \mathbf{0}_p \quad \text{and} \quad f_i(\mathbf{0}_p) = 0. \quad (8.2)$$

More generally, we use  $\mathcal{Z}_{\alpha,\beta}^p$  to denote the set of all functions satisfying the conditions above.

Assumption 8.1 implies that the cost paid by the agents is always non-negative. Moreover it is zero, and thus minimal, only if they do not contribute at all to satisfy the demand (*i.e.*, when  $x_i = \mathbf{0}_p$ ). In the same way as in the previous chapter, we use  $\kappa = \beta/\alpha$  to denote the *condition number* of the functions  $f_i$ . Furthermore, following the same argument as in Proposition 7.2, one has that  $f = \sum_{i=1}^n f_i \in \mathcal{Z}_{\alpha,\beta}^{np}$ .

To this problem we associate a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  with  $\mathcal{V} = \{1, \dots, n\}$ , and  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  by which the agents in the system interact. At random times a pair of agents  $(i, j) \in \mathcal{E}$  is uniformly randomly chosen (with probability  $\frac{1}{|\mathcal{E}|}$ ) to exchange information and update their respective estimates.

Moreover, we assume that the system is subject to:

- arrivals, resulting in a new agent *in* joining the system with a new cost function  $f_{in}$  and demand  $d_{in}$ ;
- departures, where an agent *out* leaves the system to never come back, with the possibility to send a last message to its neighbours.

Consequently, since the composition of the system varies, so does the optimal solution of (8.1), which is thus denoted  $x^{*,t}$ .

### 8.1.2 Simplifying assumptions and reformulation

In this chapter, we consider a simplified version of the general problem described above, and derive preliminary results to pave the way for future research. Hence, we introduce the following simplifying assumptions.

**Assumption 8.2.** *The local cost functions  $f_i$  are one-dimensional:  $f_i \in \mathcal{Z}_{\alpha,\beta}^1$ .*

**Assumption 8.3.** *The demand  $d_i$  associated with the agents is always  $d_i = 1$ .*

**Assumption 8.4.** *The graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is complete.*

From Assumptions 8.2 and 8.3, one can write the budget constraint as  $\sum_{i=1}^n x_i = n$ , and then define the feasible set of the problem as

$$\mathcal{S}_n := \left\{ x \in \mathbb{R}_+^n : \mathbf{1}_n^\top x = n \right\}. \quad (8.3)$$

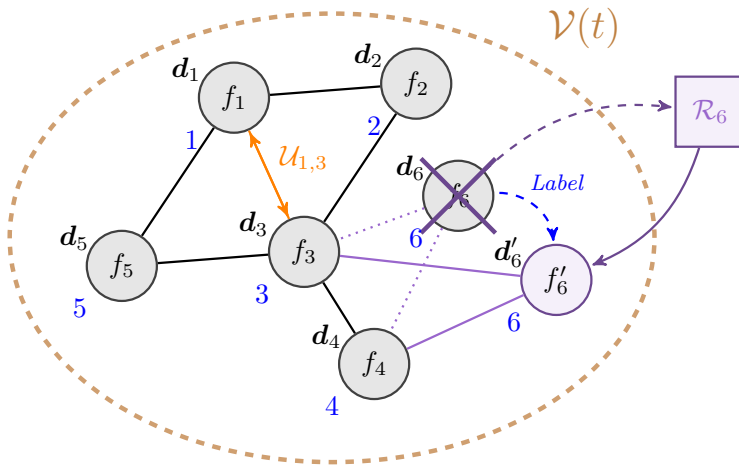
Assumption 8.4 implies that at the occurrence of a pairwise update, then any pair of agents  $(i, j)$  with  $i, j \in \mathcal{V}$  interacts with probability  $\frac{2}{n(n-1)}$ .

Moreover, we restrict our analysis to the specific case where the openness of the system is based solely on replacements (*i.e.*, the simultaneous occurrence of arrivals and replacements). As a consequence the system size  $n$  is constant, and the replacement of an agent  $i$  can be seen as the attribution of a new function, demand and estimate to that agent. Furthermore, the evolution of the system is event-based in the sense of Section 2.4, so that it is characterized by instantaneous modifications triggered by two types of events:

- A pairwise update between agents  $i$  and  $j$ , denoted  $\mathcal{U}_{i,j}$ , where two connected agents  $(i, j) \in \mathcal{E}$  exchange information and update their respective estimates  $x_i$  and  $x_j$ ;

- A replacement of agent  $i$ ,  $\mathcal{R}_i$ , which consists in the simultaneous departure of agent  $i$  and arrival of a new agent holding a new local function  $f'_i$  and demand  $d'_i$ ; the leaving agent has the possibility to send a last message to its neighbours, and the joining agent then takes its place and retrieves its label (but not necessarily its estimate).

The system described above is illustrated in Fig. 8.1.1. Observe that, by contrast with the previous chapter, at replacements *the joining agent does not retrieve the estimate of the agent it replaces*. This is due to the demand  $d_i$  which might change as well in general, so that keeping the same estimate would result in violating the budget constraint most of the time. Rather, we will define later some correction mechanism to ensure it remains valid.



**Fig. 8.1.1** Representation of the OMAS considered in this chapter, subject to either pairwise updates  $\mathcal{U}_{ij}$  (in orange) or replacements  $\mathcal{R}_i$  (in purple), where the joining agent receives a new cost function  $f'_i$  and demand  $d'_i$ , and retrieves the label of the leaving agent (in blue), but not its estimate.

Observe that the evolution of the system described above actually corresponds to that of Section 7.1.2 in the previous section (up to an alternative definition for the effect of replacements). Hence, the same discretization of the time can be performed, where each time step is chosen at the time instant at which an event takes place (see Section 7.1.3 for details). In particular, we consider the following assumption, which guarantees that at any time step  $k$ , the event  $\zeta_k$  taking place is either a pairwise update with fixed probability  $p_U$ , or a replacement with fixed probability  $1 - p_U$ .

**Assumption 8.5.** *The event  $\xi_k$  taking place at time step  $k$  is independent of any other event  $\xi_j$  with  $j \neq k$ , and of any information prior to the time step  $k$ , such as estimates, demands or cost functions.*

Hence, by denoting  $f_i^t$  the local cost function held by agent  $i$  at the time step  $t$ , one can finally reformulate (8.1) in our simplified setting as

$$\min_{\mathbf{x} \in \mathcal{S}_n} f^t(\mathbf{x}) = \sum_{i=1}^n f_i^t(x_i), \quad (8.4)$$

where we remind  $\mathcal{S}_n := \{\mathbf{x} \in \mathbb{R}_+^n : \mathbf{1}_n^\top \mathbf{x} = n\}$  from (8.3), and where the following assumption holds.

**Assumption 8.6.** *The local cost function  $f_i^t$  held by any agent  $i \in \mathcal{V}$  at any time step  $t$  satisfies  $f_i^t \in \mathcal{Z}_{\alpha, \beta}^1$  in the sense of Assumption 8.1.*

### 8.1.3 Update rules and well-posedness

**Pairwise updates:** In order to solve (8.4), we consider the RCD algorithm described in Section 6.2. We consider the optimal step size in our setting with respect to the function value discussed in Remark 6.1, given by  $\frac{1}{\beta}$ . Under the simplified assumptions described in Section 8.1.2, whenever to agents  $i$  and  $j$  interact, their respective estimates thus become

$$x_i^+ = x_i - \frac{1}{\beta} \left( f_i'(x_i) - f_j'(x_j) \right); \quad x_j^+ = x_j - \frac{1}{\beta} \left( f_j'(x_j) - f_i'(x_i) \right), \quad (8.5)$$

where  $f_i'$  denotes the derivative of  $f_i$  as it is one-dimensional.

**Replacements:** Since the agents are allowed to send a last message to their neighbours (*i.e.*, to all the agents in our case from Assumption 8.4) upon departure, we will consider the following update rule for the agents remaining the system when they receive a last message from a leaving agent *out*:

$$x_i^+ = x_i + \frac{x_{out} - x_i}{n} = \left( 1 - \frac{1}{n} \right) x_i + \frac{1}{n} x_{out}. \quad (8.6)$$

When an agent joins the system it only knows itself. Hence, since by default agents aim at satisfying their own demand, we consider that any

agent *in* joining the system satisfies

$$x_{in}^+ = d_{in}^+ = 1, \quad (8.7)$$

where the last equality follows from Assumption 8.3. Replacements thus consist in the simultaneous occurrence of both rules (8.6) and (8.7).

**Well-posedness:** In the next proposition, proved in Appendix 8.A, we show that the problem is well-posed with the rules described above, namely that they never break the budget constraint nor have negative states.

**Proposition 8.7** (Well-posedness). *For a system as described in Section 8.1.2 with the rules (8.5) to (8.7), if  $x^0 \in \mathcal{S}_n$ , then  $x^t \in \mathcal{S}_n$  for all  $t$ .*

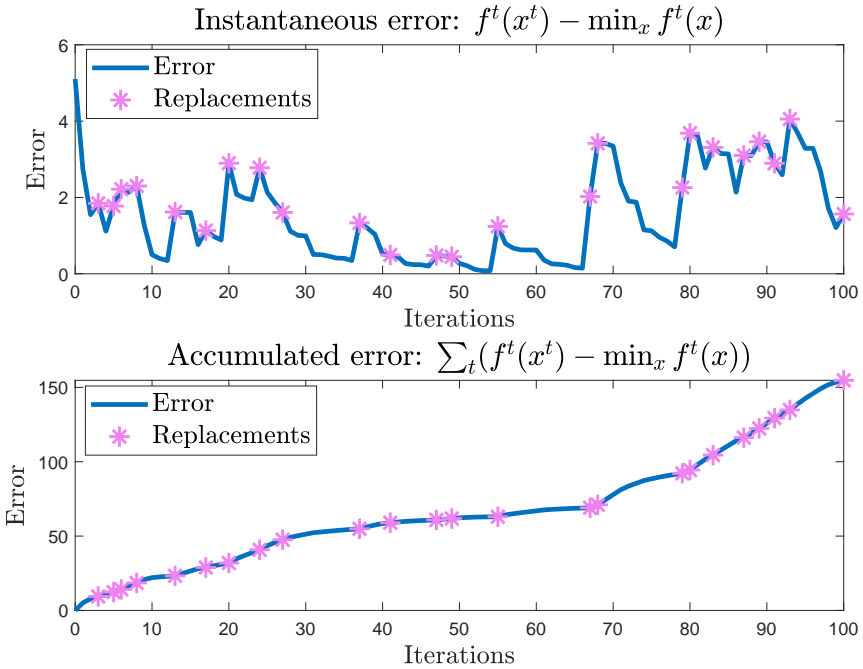
*Proof.* The proof is given in Appendix 8.A. □

Our goal is to analyze the performance of the RCD algorithm in the open system described above using suited metrics, which we characterize in the next section.

## 8.2 Performance metrics

In the settings described in Section 8.1, *i.e.*, corresponding to problems where the cost function must be paid on a regular basis, it is natural to aim at minimizing the total cost accumulated over the iterations rather than the instantaneous one at a given specific time step. In that sense, one typically aims at measuring the performance evaluated over some time horizon  $T$  instead on focusing on instantaneous performance. This is for instance observed in online learning [LXL22], or power grids where the performance of prediction-correction algorithms is evaluated in terms of the so-called *mean cumulative tracking error* defined as  $\frac{1}{T} \sum_{t=1}^T \|x^t - x^{*,t}\|$  [SDP<sup>+</sup>20].

In the same line, a natural index for measuring the performance of algorithms solving (8.4) is given by  $\sum_{t=1}^T (f^t(x^t) - f^t(x^{*,t}))$ , which corresponds to the accumulated error in terms of the function value over a time horizon  $T$ . In Fig. 8.2.1, we compare the evolution of this metric with its analogous instantaneous metric for a specific realization of the RCD algorithm in a system subject to replacements as described in Section 8.1. In this setting, the figure shows that the instantaneous metric suffers from more abrupt variations than the accumulated one at replacements.



**Fig. 8.2.1** Evolution of the instantaneous (top) and accumulated (bottom) errors with respect to the function value for a realization of the RCD algorithm in the setting of Section 8.1, where replacements (purple stars) take place on average once every four RCD iterations.

The accumulated error of the figure above exactly corresponds to the *dynamic regret* (6.12) used in online optimization and introduced in Section 6.4: It corresponds to the sum of the differences between the estimate  $x^t$  and the instantaneous solution of (8.4)  $x^{*,t}$ , which we call “best estimate”. One can actually generalize this idea to other similar indexes comparing an estimate  $x^t$  with any other strategy of interest in order to highlight the accumulated gain or loss it induces with respect to it. In particular, for this chapter we define the following reference strategies:

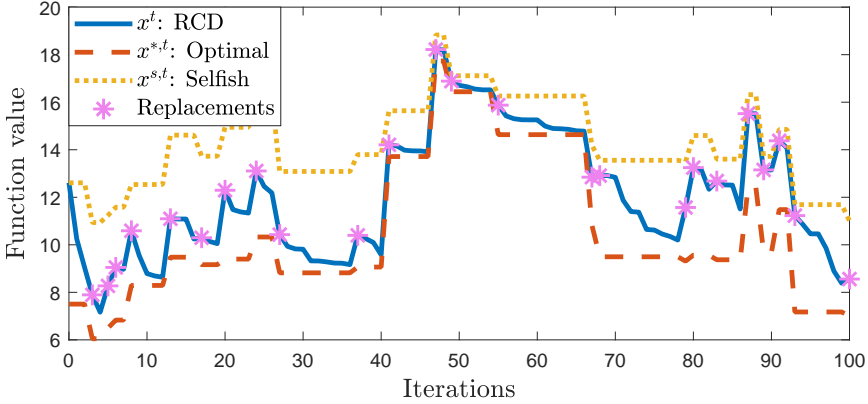
1. **Optimal strategy:** at any time  $t$ , the agents know the instantaneous solution of (8.4), so that the estimate with that strategy is  $x^{*,t}$ ;
2. **Selfish strategy:** the agents do not collaborate and only operate at their own demand, so that the estimate is  $x^{s,t} := \mathbb{1}_n$  at all  $t$ .

Regarding the definition of these strategies, a well-designed algorithm is

expected to generate a sequence of estimates  $\mathbf{x}^t$  which satisfies for all  $t$ :

$$f^t(\mathbf{x}^{*,t}) \leq f^t(\mathbf{x}^t) \leq f^t(\mathbf{x}^{s,t}). \quad (8.8)$$

The behavior of the reference strategies is illustrated and compared with that of the RCD algorithm in Fig. 8.2.2, showing that (8.8) is satisfied for that realization.



**Fig. 8.2.2** Evolution of the estimate obtained with the RCD algorithm (plain blue line) and of the optimal (red dashed line) and selfish (yellow dotted line) strategies for the same realization as in Fig. 8.2.1. Each purple star corresponds to a replacement.

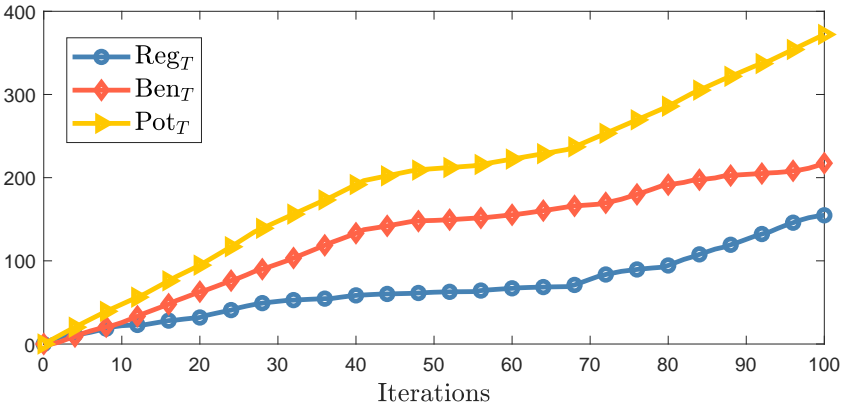
We can now define the following performance metrics, based on the different strategies of interest defined above (in our case  $\mathbf{x}^t$  is generated by the RCD algorithm). To avoid conflicts in the notations, and since we will only consider the dynamic regret and not the static one, in this chapter we refer to the dynamic regret (denoted  $\text{Reg}_T^d$  in Section 6.4) as  $\text{Reg}_T$ :

$$\text{Dynamic regret:} \quad \text{Reg}_T := \sum_{t=1}^T (f^t(\mathbf{x}^t) - f^t(\mathbf{x}^{*,t})); \quad (8.9)$$

$$\text{Benefit:} \quad \text{Ben}_T := \sum_{t=1}^T (f^t(\mathbf{x}^{s,t}) - f^t(\mathbf{x}^t)); \quad (8.10)$$

$$\text{Potential benefit:} \quad \text{Pot}_T := \sum_{t=1}^T (f^t(\mathbf{x}^{s,t}) - f^t(\mathbf{x}^{*,t})). \quad (8.11)$$

The “dynamic regret” and “benefit” respectively measure the accumulated error from using a given algorithm with respect to the optimal solution  $x^{*,t}$  and the accumulated gain from using it instead of the selfish strategy  $x^{s,t}$ . The “potential benefit” is independent of the algorithm; it represents the accumulated advantage of the optimal strategy with respect to the selfish one, and satisfies  $\text{Pot}_T = \text{Ben}_T + \text{Reg}_T$ . They are illustrated in Fig. 8.2.3.



**Fig. 8.2.3** Evolution of the dynamic regret (8.9) in blue, the benefit (8.10) in red and the potential benefit (8.11) in yellow for the same realization as in Fig. 8.2.2.

Remember from Section 6.4 that the dynamic regret differs from the usual static regret which is more common in standard online optimization: Whereas the former compares  $x^t$  with the time-varying instantaneous solution  $x^{*,t}$ , the latter is more restrictive and compares it with some fixed estimate  $x^* = \arg \min_{x \in S_n} \sum_{t=1}^T f^t(x)$  which is optimal over the entire horizon. Hence, the conclusions which are usually desired in online optimization using the static regret should not be expected with the dynamic one.

In the remainder of this chapter, we derive upper bounds on the metrics above in expectation in order to characterize their evolution.

**Remark 8.1.** *An implicit assumption made in this section is that the cost  $f^t$  is paid at the time instants at which replacements and interactions happen. Hence, the metrics are defined over these specific time instants as well. One could more generally define a specific timescale for the time instants at which the cost is paid, and evaluate the evolution of the metrics with respect to all the events potentially happening in between. This however requires the use of more complex tools to handle different timescales at the same time, and is thus left for future research.*

### 8.3 Upper bounds on the metrics

In this section, we derive upper bounds on the evolution of the potential benefit (8.11) and dynamical regret (8.9) in expectation, in the setting described in Section 8.1. We first provide the following lemma.

**Lemma 8.8.** *Let  $\mathbf{x} \in \mathcal{S}_n$ , then there holds*

$$n \leq \|\mathbf{x}\|^2 \leq n^2. \quad (8.12)$$

Moreover, for  $f(\mathbf{x}) := \sum_{i=1}^n f_i(x_i)$  with all  $f_i$  satisfy Assumptions 8.1 and 8.2,

$$\frac{\alpha}{2}n \leq f(\mathbf{x}) \leq \frac{\beta}{2}n^2. \quad (8.13)$$

*Proof.* The first result is a direct consequence of the equivalence of the norms. Recall that  $f(\mathbf{0}_n) = 0$  and  $\nabla f(\mathbf{0}_n) = \mathbf{0}_n$  from Assumptions 8.1 and 8.2. Hence, since  $f$  is  $\beta$ -smooth and using (8.12) there holds

$$f(\mathbf{x}) \leq \frac{\beta}{2} \|\mathbf{x}\|^2 \leq \frac{\beta}{2}n^2,$$

which establishes the upper bound in (8.13). Similarly, since  $f$  is  $\alpha$ -strongly convex and using (8.12) there holds

$$f(\mathbf{x}) \geq \frac{\alpha}{2} \|\mathbf{x}\|^2 \geq \frac{\alpha}{2}n,$$

which establishes the lower bound and concludes the proof.  $\square$

Lemma 8.8 provides a global upper bound on the difference between any two solutions  $\mathbf{x}, \mathbf{y} \in \mathcal{S}_n$ :

$$|f^t(\mathbf{x}) - f^t(\mathbf{y})| \leq \frac{n}{2}(n\beta - \alpha). \quad (8.14)$$

This bound is thus valid for any of the metrics defined in Section 8.2, and one has e.g.,

$$\text{Ben}_T \leq \frac{n}{2}(n\beta - \alpha)T. \quad (8.15)$$

It is however very conservative, especially for the metrics involving  $\mathbf{x}^t$ , as it does not make use of the algorithm at all.

## 8.3.1 Upper bound for the potential benefit

In the following theorem, we obtain an upper bound for the potential benefit, which we remind quantifies the accumulated advantage of the optimal strategy with respect to the selfish one. This quantity thus grows independently of the algorithm that is used by definition.

**Theorem 8.9.** *In the setting of Section 8.1, there holds*

$$\text{Pot}_T \leq \frac{n}{2} \alpha (\kappa - 1) T, \quad (8.16)$$

and in particular

$$\lim_{T \rightarrow \infty} \frac{\text{Pot}_T}{T} \leq \frac{n}{2} \alpha (\kappa - 1). \quad (8.17)$$

*Proof.* Recall that  $\mathbf{x}^{s,t} = \mathbb{1}_n$  by definition and  $f^t(\mathbf{0}_n) = 0$  and  $\nabla f^t(\mathbf{0}_n) = \mathbf{0}_n$  for all  $t$  from Assumption 8.6. Hence, since  $f^t$  is  $\beta$ -smooth there holds

$$f^t(\mathbf{x}^{s,t}) \leq \nabla f^t(\mathbf{0}_n)^\top (\mathbf{x}^{s,t}) + \frac{\beta}{2} \|\mathbf{x}^{s,t}\|^2 = \frac{\beta}{2} \|\mathbb{1}_n\|^2 = \frac{\beta}{2} n.$$

Similarly, since  $f^t$  is  $\alpha$ -strongly convex, we have

$$f^t(\mathbf{x}^{*,t}) \geq \nabla f^t(\mathbf{0}_n)^\top (\mathbf{x}^{*,t}) + \frac{\alpha}{2} \|\mathbf{x}^{*,t}\|^2 = \frac{\alpha}{2} \|\mathbf{x}^{*,t}\|^2 \geq \frac{\alpha}{2} n,$$

where the last inequality follows from Lemma 8.8. Hence

$$f^t(\mathbf{x}^{s,t}) - f^t(\mathbf{x}^{*,t}) \leq \frac{\beta}{2} n - \frac{\alpha}{2} n = \frac{n}{2} (\beta - \alpha)$$

holds. Injecting it into (8.11) then yields (8.16), and (8.17) finally follows from dividing (8.16) by  $T$ .  $\square$

Since the upper bound above is independent of the algorithm, it positively grows the same way at every iteration. Moreover, as we will see later it is rather conservative regarding simulations of the system: Since the proof is rather simple, this suggests that there remains room for improvement in that sense.

## 8.3.2 Upper bound for the dynamic regret

Second, we obtain an upper bound on the dynamic regret (8.9) in expectation, which we remind measures the accumulated error achieved from

using the RCD algorithm  $\mathbf{x}^t$  with respect to the optimal solution  $\mathbf{x}^{*,t}$ . To that end, we first introduce the following quantity:

$$\mathcal{C}_t := f^t(\mathbf{x}^t) - f^t(\mathbf{x}^{*,t}), \quad (8.18)$$

which corresponds to the instantaneous error achieved by the RCD algorithm with respect to the instantaneous optimal solution at iteration  $t$ . Moreover, we define

$$\Delta f_t := f^{t+1}(\mathbf{x}^{t+1}) - f^t(\mathbf{x}^t); \quad (8.19)$$

$$\Delta f_t^* := f^{t+1}(\mathbf{x}^{*,t+1}) - f^t(\mathbf{x}^{*,t}). \quad (8.20)$$

These quantities stand for the instantaneous variation over one time step of the total estimated cost from the RCD algorithm and of the optimal cost respectively. Hence, it follows that for any time step  $t$ :

$$\mathcal{C}_{t+1} = \mathcal{C}_t + \Delta f_t - \Delta f_t^*. \quad (8.21)$$

By studying the evolution of  $\Delta f_t$  and  $\Delta f_t^*$ , and thus of  $\mathcal{C}_t$  in expectation, we can characterize that of  $\text{Reg}_T := \sum_{t=1}^T \mathcal{C}_t$ . The following proposition proved in Appendix 8.B characterizes the behavior of  $\Delta f_t$ .

**Proposition 8.10.** *Let  $\gamma := 1 - \frac{1}{\kappa(n-1)}$  and  $\theta_{\alpha,\beta} := \frac{5}{2}\beta - \frac{3}{2}\alpha$ . In the setting of Section 8.1, there holds for any time step  $t$ :*

$$\mathbb{E}[\Delta f_t | \mathcal{C}_t] \leq p_U(\gamma - 1)\mathcal{C}_t + (1 - p_U)\theta_{\alpha,\beta}. \quad (8.22)$$

Using this result we now give the following theorem which studies the evolution of the dynamic regret in expectation.

**Theorem 8.11.** *Let  $\mathcal{C}_0 = f^0(\mathbf{x}^0) - f^0(\mathbf{x}^{*,0})$  be the initial error at the initialization of the system. Moreover, let  $\eta := 1 - \frac{p_U}{\kappa(n-1)}$ ,  $M_f := \frac{\eta}{2}(\beta n - \alpha)$  and  $\theta_{\alpha,\beta} := \frac{5}{2}\beta - \frac{3}{2}\alpha$ . In the setting of Section 8.1 there holds for any time step  $t$ :*

$$\mathbb{E}[\text{Reg}_T] \leq \mathcal{C}_0 \sum_{t=1}^T \eta^t + \sum_{t=0}^{T-1} \eta^t \left( M_f + (T-t)(1 - p_U)\theta_{\alpha,\beta} \right). \quad (8.23)$$

*Proof.* First recall that from (8.21) one has for any  $t$ :

$$\mathbb{E}[\mathcal{C}_{t+1}] = \mathbb{E}[\mathcal{C}_t + \Delta f_t - \Delta f_t^*]. \quad (8.24)$$

Then, from Proposition 8.10 there holds

$$\mathbb{E} [\Delta f_t] \leq p_U (\gamma - 1) \mathbb{E} [\mathcal{C}_t] + (1 - p_U) \theta_{\alpha, \beta}, \quad (8.25)$$

where we remind  $\gamma = 1 - \frac{1}{\kappa(n-1)}$ . Injecting (8.25) in (8.24) then yields

$$\mathbb{E} [\mathcal{C}_{t+1}] \leq \eta \mathbb{E} [\mathcal{C}_t] + (1 - p_U) \theta_{\alpha, \beta} - \mathbb{E} [\Delta f_t^*],$$

with  $\eta = 1 + p_U (\gamma - 1) = 1 - \frac{p_U}{\kappa(n-1)}$ . This describes the evolution of a discrete-time dynamical system of the form  $v[k+1] \leq Av[k] + Bu[k]$  where  $A = \eta$ , and  $Bu[k] = (1 - p_U) \theta_{\alpha, \beta} - \mathbb{E} [\Delta f_k^*]$ . Standard results in that framework yield  $v[k] \leq A^k v[0] + \sum_{j=0}^{k-1} A^{k-j-1} Bu[j]$ , so that we obtain

$$\mathbb{E} [\mathcal{C}_t] \leq \eta^t \mathbb{E} [\mathcal{C}_0] + \sum_{j=0}^{t-1} \eta^{t-j-1} \left( (1 - p_U) \theta_{\alpha, \beta} - \mathbb{E} [\Delta f_j^*] \right). \quad (8.26)$$

Injecting (8.26) into the definition of  $\text{Reg}_T$  in (8.9) then gives

$$\mathbb{E} [\text{Reg}_T] \leq \mathcal{C}_0 \sum_{t=1}^T \eta^t + \sum_{t=1}^T \left( \sum_{j=0}^{t-1} \eta^{t-j-1} \left( (1 - p_U) \theta_{\alpha, \beta} - \mathbb{E} [\Delta f_j^*] \right) \right).$$

After some term re-organization, it becomes

$$\mathbb{E} [\text{Reg}_T] \leq \mathcal{C}_0 \sum_{t=1}^T \eta^t + (1 - p_U) \sum_{t=0}^{T-1} (T-t) \eta^t \theta_{\alpha, \beta} - \sum_{t=0}^{T-1} \eta^t \left( \sum_{j=0}^{T-t} \mathbb{E} [\Delta f_j^*] \right).$$

Finally, using Lemma 8.8, one has that

$$\begin{aligned} - \sum_{j=0}^{T-t} \mathbb{E} [\Delta f_j^*] &= \mathbb{E} \left[ - \sum_{j=0}^{T-t} \Delta f_j^* \right] = \mathbb{E} \left[ f^1(\mathbf{x}^{*,1}) - f^{T-t+1}(\mathbf{x}^{*,T-t+1}) \right] \\ &\leq \frac{n}{2} (\beta n - \alpha) = M_f. \end{aligned}$$

Injecting this in the previous result concludes the proof.  $\square$

Let us define the rate ratio  $\rho_R := \frac{1-p_U}{p_U}$  which corresponds to the expected number of replacements taking place in the system between two RCD updates. We analyze below the asymptotic behavior of the averaged dynamic regret based on Theorem 8.11.

**Corollary 8.12.** *In the same setting as in Theorem 8.11, there holds*

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E} [\text{Reg}_T]}{T} \leq \rho_R (n-1) \theta_{\alpha, \beta} = \frac{1-p_U}{p_U} (n-1) \alpha \frac{5\kappa-3}{2}. \quad (8.27)$$

*Proof.* Starting from (8.23) we have

$$\frac{\mathbb{E} [\text{Reg}_T]}{T} \leq C_0 \sum_{t=1}^T \frac{\eta^t}{T} + \sum_{t=0}^{T-1} \left( M_f \frac{\eta^t}{T} + (1-p_U) \left(1 - \frac{t}{T}\right) \eta^t \theta_{\alpha, \beta} \right).$$

Recall that  $\eta = 1 - \frac{p_U}{\kappa(n-1)} < 1$  by definition. Hence  $\sum_{t=1}^T \eta^t < T$  for all  $T$ , and it follows that  $\lim_{T \rightarrow \infty} \frac{\eta^t}{T} = 0$ . As a consequence we have

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E} [\text{Reg}_T]}{T} \leq \lim_{T \rightarrow \infty} (1-p_U) \sum_{t=0}^{T-1} \left(1 - \frac{t}{T}\right) \eta^t \theta_{\alpha, \beta}.$$

Moreover, since  $\eta < 1$  one has  $\sum_{j=0}^{\infty} \eta^j 1 \frac{1}{1-\eta}$  and  $\sum_{j=0}^{\infty} j \eta^j = \frac{\eta}{(1-\eta)^2}$ . The latter implies that  $\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} t \eta^t = 0$ , and therefore

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E} [\text{Reg}_T]}{T} \leq (1-p_U) \frac{1}{1-\eta} \theta_{\alpha, \beta} = \frac{1-p_U}{p_U} \kappa (n-1) \theta_{\alpha, \beta},$$

and the conclusion follows from  $\theta_{\alpha, \beta} = \frac{5}{2} \beta - \frac{3}{2} \alpha$ .  $\square$

**Remark 8.2.** *The proofs of Theorem 8.11 and Corollary 8.12 can directly be adapted to any algorithm achieving linear convergence in closed system with some contraction rate  $\gamma < 1$ . In that case, the factor  $\eta = 1 + p(\gamma - 1)$  in Theorem 8.11 remains smaller than 1, and one can then generalize the result of Corollary 8.12 for any such algorithm as*

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E} [\text{Reg}_T]}{T} \leq \rho_R \frac{\theta_{\alpha, \beta}}{1-\gamma}. \quad (8.28)$$

*It is however not clear whether other algorithms might yield tighter bounds.*

## 8.4 Illustrations and interpretation

The results obtained above show some interpretation, which we give below. At the same time we provide some numerical results to compare them.

### 8.4.1 Setting for the simulations

To illustrate Theorems 8.9 and 8.11, we consider a system of 5 agents with  $\alpha = 1$  and  $\beta = 10$  (so that  $\kappa = 10$ ) and  $\rho_R = 0.0125$  (i.e., such that replacements happen once every 80 time steps on average). Observe that the derivation of these results is independent of the way we select the functions held by the agents that join the system at replacements, as long as they satisfy Assumption 8.6. Therefore, in the numerical results that follows, we consider two different ways to make that choice: *random* and *disadvantageous*:

- *Random choice (RC)*: The local cost  $f_i$  is a piecewise quadratic function such that  $f_i(x) = \phi_1 x^2$  for  $x \leq 0$  and  $f_i(x) = \phi_2 x^2$  for  $x > 0$ , with  $\phi_1, \phi_2$  uniformly randomly chosen in  $\left[\frac{\alpha}{2}, \frac{\beta}{2}\right]$ ;
- *Disadvantageous choice (DC)*: The local cost is a quadratic function  $f_i(x) = \phi x^2$  with  $\phi \in \left\{\frac{\alpha}{2}, \frac{\beta}{2}\right\}$ , in such way that there is (roughly) as many functions with  $\phi = \frac{\alpha}{2}$  and  $\phi = \frac{\beta}{2}$  at all times in the system.

Our results thus hold for both choices, but also for many others. This typically includes the worst possible case that would maximize the impact of replacements on our metrics. It is however not clear what this worst case is, and instead we have designed the *disadvantageous* case in an attempt to approach it. Still, other less favorable choices can probably be made, resulting in worse performance.

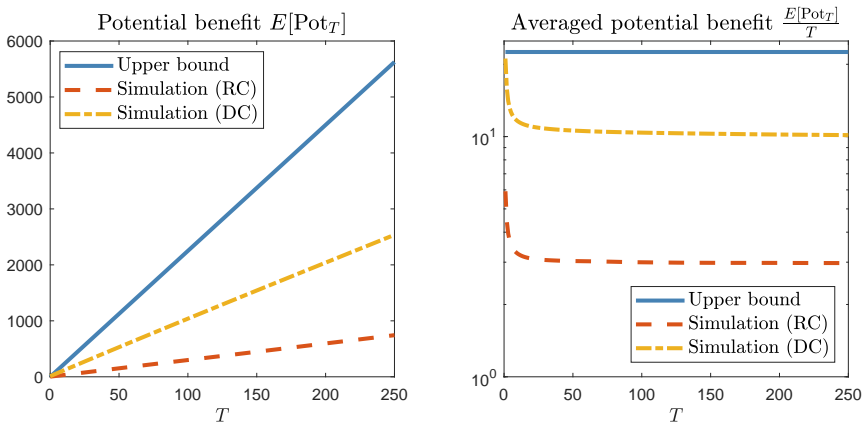
We compare our bounds with numerical results in that setting in the remainder of this section. The figures below show that although they are rather conservative, our bounds capture well the qualitative behavior of the metrics. Moreover, because the DC case is less favorable than the RC one, our bounds are tighter for the former. This suggests our bounds might be tight for some specific choices for the joining functions, in particular regarding the potential benefit.

### 8.4.2 Potential benefit

As illustrated in Fig. 8.4.1 the upper bound (8.16) linearly scales with  $T$ , so that there remains an asymptotic error in (8.17). This behavior is actually natural as  $\text{Pot}_T$  is independent of the algorithm by definition: It is thus not expected to decay with the interactions in the network, rather it positively grows (constantly) at every time step. The asymptotic error (8.17) then scales with  $n$  and  $\alpha$ , consistently with the setting described in Section 8.1. Finally, notice that because the dynamical regret  $\text{Reg}_T$  (8.9) is nonnegative by definition, the bounds obtained in Theorem 8.9 are also valid for the benefit, since  $\text{Ben}_T = \text{Pot}_T - \text{Reg}_T$ . Hence we have

$$\text{Ben}_T \leq \frac{n}{2}(\beta - \alpha)T, \quad (8.29)$$

which improves on (8.15).

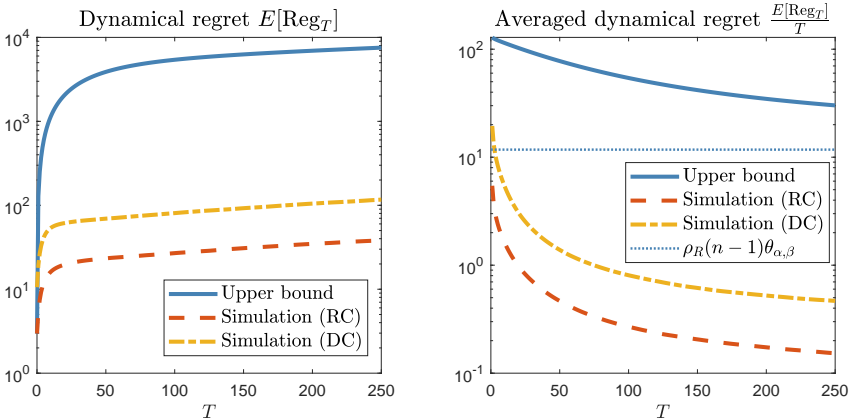


**Fig. 8.4.1** Evolution of  $E[\text{Pot}_T]$  and  $\frac{E[\text{Pot}_T]}{T}$  for the system described in Section 8.4.1, *i.e.*, with  $\alpha = 1$ ,  $\beta = 10$  and  $\rho_R = 0.0125$ . Both replacement cases are considered: the random choice RC (in dashed red line) and the disadvantageous choice DC (in yellow dash-dotted line). These are compared with the upper bounds from (8.16) on the left and (8.17) on the right (in plain blue line).

### 8.4.3 Dynamic regret

As illustrated in Fig. 8.4.2 and similarly to the the potential benefit, the upper bound (8.23) on the expected dynamic regret linearly scales with  $T$ , so

that there remains an asymptotic error in (8.27). This behavior contrasts with usual results expected in the context of online optimization discussed in Section 6.4, where the objective is to show that the regret grows sublinearly with  $T$  in such way that the averaged regret asymptotically decays to zero with  $T$  [Haz16, Ch. 1.1]. In fact, this expected behavior usually holds true when considering the *static* regret, where the estimate  $x^t$  is compared with a fixed solution  $x^*$  optimal with respect to the whole horizon  $T$ . Yet, our analysis builds on the *dynamic* regret, where it is compared with the instantaneous solution  $x^{*,t}$ . This metric is more restrictive, and does not allow the usual sublinear growth in general [LXL22]. More specifically, in our case, each replacement introduces a perturbation of non-decaying magnitude on both the cost functions and the estimates. Hence, it is very likely that no algorithm can instantaneously compensate these perturbations, so that the linear growth in  $T$  is most probably unavoidable.



**Fig. 8.4.2** Evolution of  $\mathbb{E}[\text{Reg}_T]$  and  $\frac{\mathbb{E}[\text{Reg}_T]}{T}$  for the system described in Section 8.4.1, *i.e.*, with  $\alpha = 1$ ,  $\beta = 10$  and  $\rho_R = 0.0125$ . Both replacement cases are considered: the random choice RC (in dashed red line) and the disadvantageous choice DC (in dash-dotted yellow line). These are compared with the bounds from (8.23) on the left and (8.27) on the right (in plain blue line); the dotted blue line is to the upper bound on  $\lim_{T \rightarrow \infty} \frac{\mathbb{E}[\text{Reg}_T]}{T}$  from (8.27).

Observe that the asymptotic upper bound (8.27) linearly grows with  $n - 1$  and  $\alpha$ , consistently with the setting described in Section 8.1. In particular, the dependency in  $n - 1$  directly follows from the contraction rate  $\gamma$  of the RCD algorithm. Moreover, (8.27) is proportional to  $\rho_R$ , which guarantees that the asymptotic averaged regret remains reasonable bounded

for small values of  $\rho_R$  (*i.e.*, for rare replacements). In particular it decays to zero when  $\rho_R \rightarrow 0$ , namely for closed systems.

Interestingly, it scales with  $\rho_R(n-1) = (1-p_U)\frac{n-1}{p_U}$ , so that the bound can be seen as the ratio between the impact of replacements, which happen at the system level with probability  $1-p_U$  (independently of  $n$ ), and the probability of a given agent getting involved in a RCD update  $\frac{p_U}{n-1}$ . This is actually consistent with Proposition 8.10, which shows that the effect of replacements on  $\Delta f_t$  is independent of  $n$ . By contrast, one could consider alternative situations *e.g.*, where all agents would be reset during replacements, whose impact would then be expected to grow with  $n$ . Notice however that this observation strongly depends on the way the scaling of the different quantities are defined, similarly to the discussion of Remark 7.3.

Finally, observe that the asymptotic upper bound (8.27) is proportional to  $\frac{5\kappa-3}{2}$ , implying that a larger possible interval for the curvature of the functions in the system generates a larger expected error. This factor is thus consistent with our setting, but is nevertheless a potential source of conservatism, especially regarding the potential benefit from (8.17) where the scaling is in  $\frac{1}{2}(\kappa-1)$ . Typically, tighter bounds can be obtained by specifying *e.g.*, the shape of the functions, such as in Appendix 8.C where we refine the result of Theorem 8.11 for quadratic functions, achieving a dependency in  $\frac{3}{2}(\kappa-1)$ . Observe moreover that this conservatism solely comes from Proposition 8.10, namely from the impact of replacements on the function values. Interestingly, the coefficient  $M_f$  in Theorem 8.11, which encapsulates their impact on the optimal function value, does not appear in the asymptotic result (8.27). This was not the case in the previous chapter, where it yielded additional conservatism, and directly follows from the use of accumulated metrics instead of instantaneous ones.

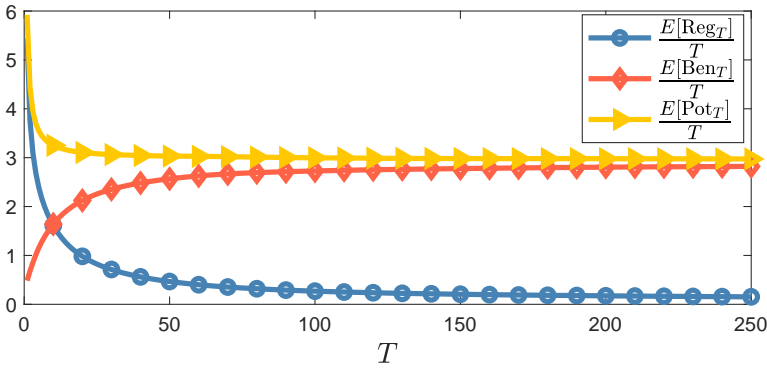
## 8.5 Discussion and conclusions

In this chapter we considered a variation of the resource allocation problem from the previous chapter where replacements also influence the budget to be allocated. Inspired by tools taken from online optimization, we used different metrics based on the accumulated error achieved by the RCD algorithm with respect to reference strategies, and applied them on a simple preliminary setting. We showed in particular that convergence to the optimal solution most probably cannot be achieved with these metrics, but

that the asymptotic error then remains reasonably bounded in expectation.

The metrics we consider here allow handling the noxious effect of replacements in an efficient way, by directly considering their total effect over a period of time rather than by accumulating the instantaneous ones. In that sense, one could carry over the analysis of the previous chapter using the tools introduced in this one with the aim of deriving tighter results, easier to interpret.

More generally, improving the tightness of some of our bounds is a natural follow-up for this chapter, especially regarding the benefit. Moreover, our results suggest that our averaged metrics never completely decay to zero asymptotically, as replacements introduce an error which cannot be fully compensated. This interpretation is supported *e.g.*, by Fig. 8.5.1, and the next step would be to obtain lower bounds also scaling linearly with  $T$ .



**Fig. 8.5.1** Evolution of the averaged metrics (*i.e.*, the dynamical regret in blue line, the benefit in red line, and the potential benefit in yellow line) for the system described in Section 8.4.1, *i.e.*, with  $\alpha = 1$ ,  $\beta = 10$  and  $\rho_R = 0.0125$ , in expectation over 1000 realizations.

So far this chapter only gives preliminary results for our approach, and extensions to make it more general are to be envisioned. In particular, our approach could be extended to handle variable-size systems (*i.e.*, where arrivals and departures are decoupled) with general graph topologies and heterogeneous demands. This would however require properly defining the arrival and departure rules to keep the resource allocation constraint valid, which can introduce additional conservatism. It would also require adapting the metrics, *e.g.*, by normalizing them with respect to the (time-varying) system size.

# Appendix of Chapter 8

## 8.A Proof of Proposition 8.7 (Well-posedness)

We provide here the proof of Proposition 8.7, which shows that if  $x^0 \in \mathcal{S}_n$ , then the estimates  $x^t$  remain feasible when subject to the update rules from Section 8.1.3 (i.e., for RCD updates, arrivals and departures).

*Proof of Proposition 8.7.* We analyze the effect of arrivals, departures and RCD updates separately, and show that they all preserve the constraints, i.e., that  $x_i \geq 0$  and  $\sum_{i=1}^n x_i = n$  at all times.

**Arrival:** Let  $in$  be the label of the joining agent so that  $\mathcal{V}^+ = \mathcal{V} \cup \{in\}$  with  $|\mathcal{V}| = n$ , then  $x_{in} = 1 \geq 0$  is a direct consequence of (8.7), and

$$\sum_{i \in \mathcal{V}^+} x_i^+ = \sum_{i \in \mathcal{V}} x_i + x_{in} = n + 1$$

guarantees that the resource allocation constraint is met.

**Departure:** Let  $out \in \mathcal{V}$  be the label of the leaving agent with  $|\mathcal{V}| = n$ , so that  $\mathcal{V}^+ = \mathcal{V} \setminus \{out\}$ , then (8.6) yields

$$x_i^+ = \left(1 - \frac{1}{n}\right) x_i + \frac{1}{n} x_{out} \geq 0$$

for all  $i \in \mathcal{V}^+$ . The resource allocation constraint is thus met since

$$\sum_{i \in \mathcal{V}^+} x_i^+ = \left(1 - \frac{1}{n}\right) (n - x_{out}) + \frac{n-1}{n} x_{out} = n - 1.$$

**RCD update:** By definition, an RCD iteration keeps the resource allocation constraint valid (see e.g., (6.8)). Observe moreover that from Assumptions 8.1 and 8.2, if  $x_i \geq 0$  we have

$$x_i f'_i(x_i) \geq \alpha |x_i|^2 \geq 0,$$

so that  $f'_i(x_i) \geq 0$ . Since  $f_i$  is  $\beta$ -smooth, one has  $f_i(x_i) \leq \beta x_i$ , and therefore at each update of the RCD algorithm between agents  $i$  and  $j$  there holds

$$x_i^+ = x_i - \frac{1}{2\beta} (f'_i(x_i) - f'_j(x_j)) \geq x_i - \frac{1}{2\beta} (\beta x_i) = \frac{x_i}{2} \geq 0.$$

The same analysis can be conducted on  $x_j^+$ , concluding the proof.  $\square$

## 8.B Proof of Proposition 8.10 (Function variation)

We prove the result of Proposition 8.10, which we remind reads:

$$\mathbb{E} [\Delta f_t | \mathcal{C}_t] \leq p(\gamma - 1)\mathcal{C}_t + (1 - p)\theta_{\alpha, \beta} \quad (8.30)$$

with  $\gamma = 1 - \frac{1}{\kappa(n-1)}$  and  $\theta_{\alpha, \beta} = \frac{5}{2}\beta - \frac{3}{2}\alpha$ .

*Proof of Proposition 8.10.* Let  $\mathcal{R}_t$  and  $\mathcal{U}_t$  respectively denote the (probabilistic) events that a replacement and a RCD update happen at time step  $t$ . Then, from assumption 8.5, there holds

$$\mathbb{E} [\Delta f_t | \mathcal{C}_t] = p_U \mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{U}_t] + (1 - p_U) \mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{R}_t]. \quad (8.31)$$

Let us analyze the effect of RCD updates and replacements separately.

**RCD update:** In that case  $\mathbf{x}^{*,t+1} = \mathbf{x}^{*,t}$ , so that  $\Delta f_t^* = 0$  and hence  $\Delta f_t = \mathcal{C}_{t+1} - \mathcal{C}_t$ . Moreover, the convergence rate of the RCD algorithm in [Nec13] yields  $\mathbb{E} [\mathcal{C}_{t+1} | \mathcal{C}_t, \mathcal{U}_t] \leq \gamma \mathcal{C}_t$ . Hence we have

$$\mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{U}_t] = \mathbb{E} [\mathcal{C}_{t+1} - \mathcal{C}_t | \mathcal{C}_t, \mathcal{U}_t] \leq (\gamma - 1)\mathcal{C}_t. \quad (8.32)$$

**Replacement:** A replacement  $\mathcal{R}_t$  consists in an arrival and a departure happening simultaneously at time step  $t$ , which we also analyze separately.

- **Arrival:** Let  $f_{new}$  denote the local cost function of the agent joining the system, then from (8.7) we have

$$\Delta f_t = f^t(\mathbf{x}^t) + f_{new}(1) - f^t(\mathbf{x}^t) = f_{new}(1) \leq \frac{\beta}{2}, \quad (8.33)$$

where the last inequality is a consequence of the  $\beta$ -smoothness of  $f_{new}$  from Assumption 8.1.

- **Departure:** Let  $\ell$  denote the label of the leaving agent, such that  $f^{t+1}(\mathbf{x}^{t+1}) = \sum_{i \neq \ell} f_i(x_i^{t+1})$ . From (8.6), we have  $x_i^{t+1} = x_i^t + \frac{x_\ell^t - x_i^t}{n}$ . Moreover, by definition of departures, the agent  $\ell$  is uniformly randomly chosen, thus with probability  $\frac{1}{n}$ . Let  $\mathcal{D}_t$  denote the occurrence of a departure at time step  $t$ , there thus holds

$$\begin{aligned} \mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{D}_t] &= \sum_{\ell=1}^n \frac{1}{n} \left( \sum_{i \neq \ell} f_i^t \left( x_i^t + \frac{x_\ell^t - x_i^t}{n} \right) - f^t(\mathbf{x}^t) \right) \\ &= \frac{1}{n} \sum_{\ell=1}^n \left( f^t \left( \frac{n-1}{n} \mathbf{x}^t + \frac{x_\ell^t}{n} \mathbf{1}_n \right) - f_\ell^t(x_\ell^t) \right) - f^t(\mathbf{x}^t) \\ &= \frac{1}{n} \left( \sum_{\ell=1}^n f^t \left( \frac{n-1}{n} \mathbf{x}^t + \frac{x_\ell^t}{n} \mathbf{1}_n \right) \right) - \frac{n+1}{n} f^t(\mathbf{x}^t). \end{aligned}$$

Since  $f^t$  is  $\beta$ -smooth from Assumption 8.1, we have

$$\begin{aligned} f^t \left( \frac{n-1}{n} \mathbf{x}^t + \frac{x_\ell^t}{n} \mathbf{1}_n \right) &\leq f^t(\mathbf{x}^t) + \frac{1}{n} \langle \nabla f^t(\mathbf{x}^t), \mathbf{1}_n x_\ell^t - \mathbf{x}^t \rangle \\ &\quad + \frac{\beta}{2n^2} \|\mathbf{1}_n x_\ell^t - \mathbf{x}^t\|^2, \end{aligned}$$

and it thus follows that

$$\begin{aligned} \mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{D}_t] &\leq \frac{1}{n^2} \sum_{\ell=1}^n \langle \nabla f^t(\mathbf{x}^t), \mathbf{1}_n x_\ell^t - \mathbf{x}^t \rangle \\ &\quad + \frac{\beta}{2n^3} \sum_{\ell=1}^n \|\mathbf{1}_n x_\ell^t - \mathbf{x}^t\|^2 - \frac{1}{n} f^t(\mathbf{x}^t). \quad (8.34) \end{aligned}$$

Recall that  $\alpha \|\mathbf{x}\|^2 \leq \langle \nabla f^t(\mathbf{x}), \mathbf{x} \rangle \leq \beta \|\mathbf{x}\|^2$  for any  $\mathbf{x}$  from Assump-

## 8 | An Online Approach for Resource Allocation in OMAS

tion 8.1. Hence, since  $\sum_{\ell=1}^n x_\ell = n$  from Assumption 8.3, we have

$$\sum_{\ell=1}^n \langle \nabla f^t(\mathbf{x}^t), \mathbb{1}_n x_\ell^t - \mathbf{x}^t \rangle = n \langle \nabla f^t(\mathbf{x}^t), \mathbb{1}_n - \mathbf{x}^t \rangle \leq n(\beta n - \alpha \|\mathbf{x}^t\|^2).$$

Moreover, one finds

$$\begin{aligned} \sum_{\ell=1}^n \|\mathbb{1}_n x_\ell^t - \mathbf{x}^t\|^2 &= \sum_{\ell=1}^n \sum_{i=1}^n \left( (x_\ell^t)^2 - 2x_\ell^t x_i^t + (x_i^t)^2 \right) \\ &= \sum_{\ell=1}^n \left( n(x_\ell^t)^2 - 2nx_\ell^t + \|\mathbf{x}^t\|^2 \right) = 2n \left( \|\mathbf{x}^t\|^2 - n \right). \end{aligned}$$

Injecting those two results in (8.34) then yields

$$\mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{D}_t] \leq \frac{1}{n} \left( \beta n - \alpha \|\mathbf{x}^t\|^2 \right) + \frac{\beta}{n^2} \left( \|\mathbf{x}^t\|^2 - n \right) - \frac{1}{n} f^t(\mathbf{x}^t).$$

From Lemma 8.8, we have  $\|\mathbf{x}^t\| r \leq n^2$  and  $f^t(\mathbf{x}^t) \geq \frac{\alpha}{2} n$ , so that

$$\mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{D}_t] \leq 2\beta - \frac{3}{2}\alpha. \quad (8.35)$$

Combining (8.33) and (8.35) then yields

$$\mathbb{E} [\Delta f_t | \mathcal{C}_t, \mathcal{R}_t] \leq \frac{5}{2}\beta - \frac{3}{2}\alpha, \quad (8.36)$$

and injecting (8.32) and (8.36) into (8.31) concludes the proof.  $\square$

### 8.C Dynamical regret: the case of quadratic functions

We show here that the results of Theorem 8.11 and Corollary 8.12 can be refined by using additional assumptions on the shape of the local cost functions. In particular we consider the following assumption.

**Assumption 8.13.** *The local cost function of any agent  $i$  at time  $t$  is of the form*

$$f_i^t(x_i) = \phi_i x_i^2, \quad \phi_i \in \left[ \frac{\alpha}{2}, \frac{\beta}{2} \right], \quad (8.37)$$

With this assumption, one refines Proposition 8.10 as follows.

**Proposition 8.14.** Let  $\gamma := 1 - \frac{1}{\kappa(n-1)}$  and  $\tilde{\theta}_{\alpha,\beta} := \frac{3n^2-3n+1}{2n^2}(\beta - \alpha)$ . In the setting of Section 8.1 and under Assumption 8.13, there holds for any time step  $t$ :

$$\mathbb{E}[\Delta f_t | \mathcal{C}_t] \leq p_U(\gamma - 1)\mathcal{C}_t + (1 - p)\tilde{\theta}_{\alpha,\beta}. \quad (8.38)$$

*Proof.* The proof follows the same steps as that of Proposition 8.10 in Appendix 8.B. Only the departure case is modified as follows, where the first steps are the same with  $f_i(x) = \phi_i x^2$ , and where we omit the dependence of the functions and estimates to the time to lighten the notations, yielding

$$\begin{aligned} \mathbb{E}[\Delta f_t | \mathcal{C}_t, \mathcal{R}_t] &= \sum_{\ell=1}^n \frac{1}{n} \left( \sum_{i \neq \ell} \phi_i \left( x_i + \frac{x_\ell - x_i}{n} \right)^2 - f(x) \right) \\ &= \frac{1}{n} \sum_{\ell=1}^n \left( \sum_{i \neq \ell} \phi_i \left( \left( x_i + \frac{x_\ell - x_i}{n} \right)^2 - x_i^2 \right) - \phi_\ell x_\ell^2 \right) \\ &= \frac{1}{n^2} \sum_{\ell=1}^n \sum_{i \neq \ell} \phi_i \left( \frac{1-2n}{n} x_i^2 + 2x_i x_\ell \frac{n-1}{n} + \frac{x_\ell^2}{n} \right) - \frac{f(x)}{n}. \end{aligned}$$

Using the fact that  $\sum_{\ell=1}^n \sum_{i \neq \ell} \phi_i x_i^2 = (n-1)f(x)$  and that  $\phi_i \leq \frac{\beta}{2}$  for all  $i$ , we obtain

$$\mathbb{E}[\Delta f_t | \mathcal{C}_t, \mathcal{R}_t] \leq \left( \frac{1-2n}{n^3}(n-1) - \frac{1}{n} \right) f(x) + \frac{\beta}{2n^2} \sum_{\ell=1}^n \sum_{i \neq \ell} \left( 2x_i x_\ell \frac{n-1}{n} + \frac{x_\ell^2}{n} \right).$$

Observe that  $\sum_{\ell=1}^n \sum_{i \neq \ell} x_i x_\ell = n^2 - \|\mathbf{x}\|^2$  and  $\sum_{\ell=1}^n \sum_{i \neq \ell} x_\ell^2 = (n-1)\|\mathbf{x}\|^2$ , so that a few algebraic manipulations yield

$$\mathbb{E}[\Delta f_t | \mathcal{C}_t, \mathcal{R}_t] \leq -\frac{3n^2 - 3n + 1}{n^3} f(x) + \beta \frac{n-1}{2n^3} (2n^2 - \|\mathbf{x}\|^2).$$

From Lemma 8.8, we have  $\|\mathbf{x}\|^2 \geq n$  and  $f(x) \geq \frac{\alpha}{2}n$ , and it follows that

$$\mathbb{E}[\Delta f_t | \mathcal{C}_t, \mathcal{R}_t] \leq -\frac{3n^2 - 3n + 1}{2n^3} \alpha + \beta \frac{2n^2 - 3n + 1}{2n^3}.$$

The conclusion is then obtained the same way as for Proposition 8.10 using the expression above.  $\square$

We then obtain the following alternative theorem, which particularizes Theorem 8.11 and Corollary 8.12 for quadratic functions.

**Theorem 8.15.** Let  $\mathcal{C}_0 = f^0(\mathbf{x}^0) - f^0(\mathbf{x}^{*,0})$  be the initial error at the initialization of the system. Moreover, let  $\eta := 1 - \frac{p_U}{\kappa(n-1)}$ ,  $M_f := \frac{n}{2}(\beta n - \alpha)$  and  $\tilde{\theta}_{\alpha,\beta} := \frac{3n^2-3n+1}{2n^2}(\beta - \alpha)$ . In the setting of Section 8.1 and under Assumption 8.13 there holds for any time step  $t$ :

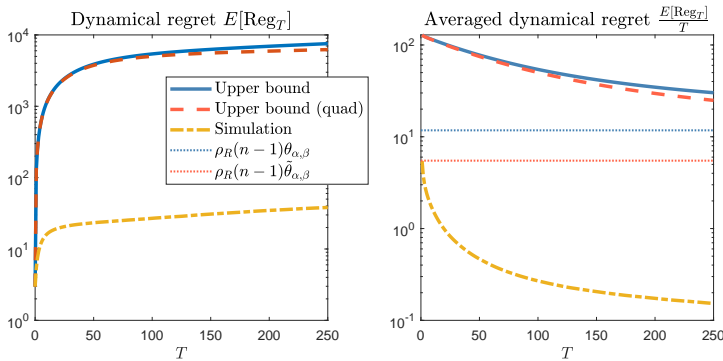
$$\mathbb{E} [\text{Reg}_T] \leq \mathcal{C}_0 \sum_{t=1}^T \eta^t + \sum_{t=0}^{T-1} \eta^t \left( M_f + (T-t)(1-p_U)\tilde{\theta}_{\alpha,\beta} \right). \quad (8.39)$$

In particular,

$$\lim_{T \rightarrow \infty} \frac{\mathbb{E} [\text{Reg}_T]}{T} \leq \rho_R(n-1)\tilde{\theta}_{\alpha,\beta} = \frac{1-p_U}{p_U}(n-1)\beta \frac{3n^2-3n+1}{2n^2}(\kappa-1). \quad (8.40)$$

*Proof.* The proof is equivalent to those of Theorem 8.11 and Corollary 8.12 using Proposition 8.14 instead of Proposition 8.10.  $\square$

We compare the results obtained here with those of Section 8.3.2 in Fig. 8.C.1 in the setting described in Section 8.4.1 with only quadratic functions. As their derivation relies on additional information on the local cost functions, the upper bounds obtained here are slightly tighter than the general ones for that case. In particular, the asymptotic bound here scales with  $\frac{3}{2}(\kappa-1)$  instead of  $\frac{5\kappa-3}{2}$ . This thus illustrates the potential improvement of the bounds that can be achieved by specifying additional information regarding the functions in the system.



**Fig. 8.C.1** Evolution of  $\mathbb{E} [\text{Reg}_T]$  and  $\frac{\mathbb{E} [\text{Reg}_T]}{T}$  for the system described in Section 8.4.1 for random replacement (dash-dotted yellow) compared with the bounds obtained in Section 8.3.2 (plain blue) and in Appendix 8.C (dashed red); the dotted lines are the corresponding upper bounds on  $\lim_{T \rightarrow \infty} \frac{\mathbb{E} [\text{Reg}_T]}{T}$ .

**PART III**  
**Conclusion**



# 9

## Conclusions and perspectives

IN this thesis we considered the possibility for agents to join and leave in the analysis of multi-agent systems, resulting in the so-called “open” multi-agent systems (OMAS). Motivated by the ever growing size of systems nowadays, making even small individual rates of arrivals and departures significant at the system level, and because of the emergence of systems naturally showing such property, the study of OMAS has started receiving a significant attention recently. Nevertheless, very few dedicated analyses can be found in the literature yet, and most of the studies initially tailored for closed systems do not easily extend to open ones.

An important consequence of the challenges arising from analysing OMAS is the impossibility for algorithms to achieve usual convergence. In particular, two major challenges are the variable size of the system, and the ever-changing nature of the information in the system. By studying the specific problems of consensus and decentralized optimization in OMAS, we investigated these challenges and highlighted some of their properties and consequences on algorithm design, paving the way towards future analyses in more complex settings or for more complex problems.

In what follows, we give a quick overview of the contributions of this thesis, and some insights on possible further researches.

## 9.1 What did we achieve?

In **Chapter 2**, we set up the scene for deriving our main results: We introduced the needed mathematical tools and the different choices we made all along the thesis to model open multi-agent systems. In particular, we properly characterized the *type of arrivals and departures* we considered and the *stochastic properties* of these processes defining the evolution of our systems. Moreover, we identified the properties of the problems we considered throughout this thesis, as these strongly influence the challenges to be faced in their analysis. This chapter more generally highlights the variety of *design choices* and problems that can be defined when studying OMAS, and their potential implication on the corresponding analyses.

**Part I** was dedicated to the analysis of *consensus problems* in OMAS, which we defined and discussed in **Chapter 3**. By investigating how to deal with both the variable size of OMAS and the ever-changing nature of the information they are subject to for that specific problem, we highlighted some general properties of OMAS related to these challenges.

We tackled the first one in **Chapter 4**, where we investigated the question of the representation of OMAS in the context of consensus. In particular, we showed that their behavior can be studied *via* scale-independent quantities called *descriptors* whose evolution is characterized by a finite-dimensional dynamical system. We proposed a general methodology in Section 4.2 relying on tools taken from *Markov theory* to handle the possibly variable size of systems. We then applied it in Section 4.3 to study the behavior of gossip interactions in both systems of fixed and variable sizes, and showed that the evolution in expectation of the variance, and hence of the level of disagreement, is bounded for such systems.

Then, in **Chapter 5**, we derived lower bounds on the performance that can be achieved by any algorithm for solving the average consensus problem in OMAS of fixed size for some given model defining the interactions in the system. The obtained lower bounds, called *fundamental performance limitations*, then constitute a suited performance criterion for algorithms, as convergence is not appropriate in OMAS. Their derivation required the definition in Section 5.2 of the so-called *knowledge sets* to model the information made potentially available to an agent at some time for a given model, thus handling the second challenge previously mentioned. We then applied the general bound obtained in Section 5.3 to study the Gossip al-

algorithm in Section 5.4, by defining appropriate *interaction models*.

In **Part II**, we focused on the analysis of decentralized optimization algorithms in OMAS. More specifically we considered the *resource allocation problem* introduced in **Chapter 6** subject to replacements, resulting in variations of the objective functions, and at some extent of the budget to be allocated.

We first studied the convergence of the error achieved by the Random Coordinate Descent (RCD) algorithm in **Chapter 7**, in terms of the distance between the estimate and the minimizer. By analysing separately the effects of the algorithm and of replacements on the *instantaneous error*, we showed that it converges in expectation to some steady state which strongly depends on the properties of the system. The methodology we used required (i) the definition of a new norm in Section 7.3 in order to have linear convergence for the RCD algorithm in closed system, and (ii) the analysis of the effect of replacements on the instantaneous error in Section 7.2, from which a significant part of the conservatism of the result is originated.

In **Chapter 8** we considered a slight variation of the same problem where the budget varies with replacements. By contrast with the previous chapter, we used performance metrics based on the *error accumulated* over the iteration with respect to a given reference strategy. This approach is strongly related to online optimization, and in particular to the notion of *regret*, from which we defined our metrics in Section 8.2. Using this approach, we obtained upper bounds in Section 8.3 for the accumulated error achieved by the RCD algorithm with respect to two reference strategies: the optimal collaboration and the non-collaborating strategy. These bounds happen to be less conservative than those of the previous chapter regarding the effect of replacements, which is directly evaluated over all the iterations at once with the approach of this chapter.

## 9.2 Is this all over?

The framework of open multi-agent systems is extremely rich and dense: The present thesis intended at paving the way regarding basic questions in that rather young domain, which remains largely unexplored nonetheless.

As discussed in **Chapter 2**, different design choices result in different types of systems and problems, with different challenges. The analyses

of this thesis thus hold for some specific relevant system descriptions and problem formulations. Yet, many other settings which we do not detail here could be considered, leading to extensions or variations of our results.

We detail below some perspectives for the specific scope of this thesis.

Part I investigates general challenges related to OMAS in the specific context of consensus problems. In particular, Chapters 4 and 5 present general methodologies, respectively for representing OMAS and evaluating the performance of algorithms, applied to consensus. Hence, perspectives for this part are mostly centered on their application to more general and advanced settings and problems. This typically includes their generalization to more restrictive or structured types of interactions through *e.g.*, the use of specific graph topologies. This might require slight adaptations, *e.g.*, with relevant choices of *descriptors* for Chapter 4 or by properly adapting the concepts of *optimal algorithm* and *knowledge sets* for Chapter 5. The methodologies would nonetheless remain fundamentally the same. In the same line, the analyses could be extended to different algorithms or advanced problems building on consensus and which could directly exploit the results obtained in this part, such as decentralized optimization or formation control.

As far as Part II is concerned, conceivable perspectives are twofold. First, one might seek for tighter results, as the bounds derived in both Chapters 7 and 8 show conservatism. Yet, the improvement margin appears to be limited, and we strongly suspect the noxious effect of replacements cannot be fully compensated. In that sense, the derivation of lower bounds *e.g.*, in the context of Chapter 8, is a relevant follow-up to validate this suspicion (this actually relates with the fundamental performance limitations of Chapter 5). Second, the analyses conducted in this part can be extended to more general settings, both regarding the properties of the system or of the problem itself, or to other algorithms, as the general methodology mostly remains the same under consistent assumptions. Finally, the generalization of the analysis to different metrics might lead to significant improvements or variations of the results. This is clear regarding the gain from using the metrics of Chapter 8 with respect to Chapter 7.

Overall, except in Chapter 4, we always considered open systems of fixed-size. A natural extension is thus to consider decoupled arrivals and departures, which is a particularly complex aspect of OMAS. One might actually exploit the tools introduced in Chapter 4 to handle systems of variable size in the other chapters. This however requires adapting many aspects of these analyses, in particular regarding the metrics that are used.

Interestingly, Chapter 8 already sets the scene for extending the analysis to variable size systems, up to a slight adaptation of the arrivals and departure rules and to the normalization of the metrics by the system size.

Finally, the natural next step for the analyses presented in this thesis is the design of algorithms tailored for open systems that would exploit them. Typically, designing algorithms that would efficiently react to variations in the system size and whose performance meets the lower bounds derived in Chapter 5 remains an open question that deserves attention, and we believe the work presented in this thesis will help in that direction.

## 9.3 Final word

We are eventually reaching the end of this manuscript. I like considering this thesis as an exploration of the extremely rich and mostly unknown world of *open* multi-agent systems. It is always astonishing how the “simple” addition of a feature – in our case the openness – to a well-known domain can lead to significant changes up to its core.

At the end of the day, the results we present here mostly aim at opening doors for future researches. By answering questions related to basic problems, we investigated the intrinsic properties of open systems and highlighted the underlying mechanisms that have to be taken into account when working with them. In that sense, we hope the research presented in this thesis does not only yield relevant results for standard problems in open systems, but also constitutes a solid baseline for future advances in that field, both in general and regarding high level applications.

There remains a lot of unanswered questions in the context of OMAS, and we hope we contributed to clear the way towards the deep understanding of these evermore realistic and accurate systems.



# List of Publications

## ★ Journal papers

- [MH20] Charles Monnoyer de Galland and Julien M. Hendrickx, “**Fundamental Performance Limitations for Average Consensus in Open Multi-Agent Systems**”, in IEEE Transactions on Automatic Control (*to be published*)
- [MMH20] Charles Monnoyer de Galland, Samuel Martin and Julien M. Hendrickx, “**Modelling Gossip Interactions in Open Multi-Agent Systems**”, in IEEE Transactions on Automatic Control (*submitted*)
- [MVH<sup>+</sup>22] Charles Monnoyer de Galland, Renato Vizuete, Julien M. Hendrickx, Elena Panteley and Paolo Frasca, “**Random Coordinate Descent for Resource Allocation in Open Multi-Agent Systems**”, in IEEE Transactions on Automatic Control (*submitted*)

## ★ Conference papers

- [MH19] Charles Monnoyer de Galland and Julien M. Hendrickx, “**Lower bound performances for average consensus in open multi-agent systems**”, in Proceedings of the 58th IEEE Conference on Decision and Control (CDC 2019), Nice (France), pages 7429-7434, December 2019 (*published*)
- [MVH<sup>+</sup>21] Charles Monnoyer de Galland, Renato Vizuete, Julien M. Hendrickx, Paolo Frasca and Elena Panteley, “**Random coordinate descent algorithm for open multi-agent systems with complete topology and homogeneous agents**”, in Proceedings of the 60th IEEE Conference on Decision and Control (CDC 2021), Austin (Texas, USA), December 2021 (*to be published*)

[VMH<sup>+</sup>22] Renato Vizuete, Charles Monnoyer de Galland, Julien M. Hendrickx, Paolo Frasca, Elena Panteley, **“Resource allocation in open multi-agent systems: an online optimization analysis”**, in Proceedings of the 61st IEEE Conference on Decision and Control (CDC 2022), to be held in Cancun (Mexico), December 2022 (*to be published*)



## Posters

- **“Fundamental Performance Limitations for averaging in Open Multi-Agent Systems”**, ICTEAM Welcome Day (2021), held on October 15, 2021 in Louvain-la-Neuve (Belgium)



## Abstracts

- **“Fundamental performance limitations for intrinsic averaging in open multi-agent systems”**, 38th Benelux Meeting on Systems and Control (2019), held from March 19 to March 21, 2019 in Lommel (Belgium)
- **“Lower Bound Performance for Averaging Algorithms in Open Multi-Agent Systems”**, at the 39th Benelux Meeting on Systems and Control (2020), held from March 10 to March 12, 2020 in Elspeet (The Netherlands)
- **“Random Coordinate Descent algorithm for Open Multi-Agent Systems”**, at the Benelux Workshop on Systems and Control (2021), held on June 29, 2021 in Rotterdam (The Netherlands)
- **“Resource allocation in open multi-agent systems: an online optimization analysis”**, 41st Benelux Meeting on Systems and Control, held from July 5 to July 7, 2022 in Brussels (Belgium)
- **“Fundamental performance limitations for intrinsic averaging in open multi-agent systems”**, 25th International Symposium on Mathematical Theory of Networks and Systems, to be held from September 12 to September 17, 2022 in Bayreuth, Germany
- **“Resource allocation in open multi-agent systems: an online optimization analysis”**, 25th International Symposium on Mathematical Theory of Networks and Systems, to be held from September 12 to September 17, 2022 in Bayreuth, Germany

# Bibliography

- [AAFJ06] Dana Angluin, James Aspnes, Michael J. Fischer, and Hong Jiang. Self-stabilizing population protocols. In *Principles of Distributed Systems*, pages 103–117. Springer Berlin Heidelberg, 2006.
- [AB06] David Angeli and Pierre-Alexandre Bliman. Stability of leaderless discrete-time multi-agent systems. *MCSS*, 18:293–322, 10 2006. doi:10.1007/s00498-006-0006-0.
- [AHH17] Mahmoud Abdelrahim, Julien M. Hendrickx, and W.P.M.H. (Maurice Heemels). Max-consensus in open multi-agent systems with gossip interactions. In *Proceedings of the 56th IEEE CDC*, pages 4753–4758, 12 2017. doi:10.1109/CDC.2017.8264362.
- [BCC01] Fred Brauer and Carlos Castillo-Chávez. *Mathematical Models in Population Biology and Epidemiology*, volume 40. Springer, 01 2001. doi:10.1007/978-1-4614-1686-9.
- [BGPS06] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah. Randomized gossip algorithms. *IEEE/ACM Trans. Netw.*, 14(SI):2508–2530, June 2006. doi:10.1109/TIT.2006.874516.
- [BMG14] Brian Baingana, Gonzalo Mateos, and Georgios B. Giannakis. Proximal-gradient algorithms for tracking cascades over social networks. *IEEE Journal on Selected Topics in Signal Processing*, 8(4):563–575, August 2014. doi:10.1109/JSTSP.2014.2317284.

- [BMHT09] Vincent Blondel, Julien M. Hendrickx, and John Tsitsiklis. Continuous-time average-preserving opinion dynamics with opinion-dependent communications. *SIAM Journal on Control and Optimization*, 48, 07 2009. doi:10.1137/090766188.
- [BML13] Lori Bird, Michael Milligan, and Debra Lew. Integrating variable renewable energy: Challenges and solutions. Technical report, National Renewable Energy Lab.(NREL), Golden, CO (United States), 2013.
- [BPC<sup>+</sup>11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3:1–122, 01 2011. doi:10.1561/22000000016.
- [Bra18] Baptiste Brasseur. Algorithms in open multi-agent systems : gossiping with random replacements. *Ecole polytechnique de Louvain, Université catholique de Louvain*, 2018. Prom. : Hendrickx, Julien M. URL: <http://hdl.handle.net/2078.1/thesis:16470>.
- [BSFH18] Lu Bai, Chao Sun, Zhi Feng, and Guoqiang Hu. Distributed continuous-time resource allocation with time-varying resources under quadratic cost functions. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 823–828. IEEE, 2018.
- [Bub11] Sébastien Bubeck. Introduction to online optimization, December 2011. URL: <https://www.microsoft.com/en-us/research/publication/introduction-online-optimization/>.
- [Bub15] Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4), 2015. doi:10.1561/22000000050.
- [Bul22] F. Bullo. *Lectures on Network Systems*. Kindle Direct Publishing, 1.6 edition, 2022. URL: <http://motion.me.ucsb.edu/book-lns>.

- [CDAJW10] John C. Duchi, Alekh Agarwal, and Martin J. Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic Control*, 57, 05 2010. doi:10.1109/TAC.2011.2161027.
- [CGJ<sup>+</sup>09] Carlos Carrascosa, Adriana Giret, Vicente Julian, Miguel Rebollo, Estefania Argente, and V Botti. Service oriented mas: an open architecture. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 1291–1292. International Foundation for Autonomous Agents and Multiagent Systems, 2009.
- [CH22] Sebastien Colla and Julien M. Hendrickx. Automatic performance estimation for decentralized optimization, 2022. URL: <https://arxiv.org/abs/2203.05963>, doi:10.48550/ARXIV.2203.05963.
- [CHW11] Rutger Claes, Tom Holvoet, and Danny Weyns. A decentralized approach for anticipatory vehicle routing using delegate multiagent systems. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):364–373, 2011. doi:10.1109/TITS.2011.2105867.
- [CKFL05] Iain Couzin, Jens Krause, Nigel Franks, and Simon Levin. Effective leadership and decision-making in animal groups on the move. *Nature*, 433:513–6, 03 2005. doi:10.1038/nature03236.
- [CLRS09] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT press, 2009.
- [CNDP<sup>+</sup>21] Dinh C. Nguyen, Ming Ding, Pubudu Pathirana, Aruna Seneviratne, Jun Li, and H. Vincent Poor. Federated learning for internet of things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 04 2021. doi:10.1109/COMST.2021.3075439.
- [CNS17a] Kobi Cohen, Angelia Nedić, and R. Srikant. Distributed learning algorithms for spectrum sharing in spatial random

- access wireless networks. *IRE Transactions on Automatic Control*, 62(6):2854–2869, June 2017. doi:10.1109/TAC.2016.2626578.
- [CNS17b] Kobi Cohen, Angelia Nedić, and R. Srikant. On projected stochastic gradient descent algorithm with weighted averaging for least squares regression. *IEEE Transactions on Automatic Control*, 62(11):5974–5981, 2017. doi:10.1109/TAC.2017.2705559.
- [Col20] S. Colla. Data fusion in open multi-agent systems for decentralized estimation. *Ecole polytechnique de Louvain, Université catholique de Louvain*, 2020. Prom. : Hendrickx, Julien M. URL: <https://dial.uclouvain.be/memoire/ucl/object/thesis:25154>.
- [CS12] Jianshu Chen and Ali H. Sayed. Diffusion adaptation strategies for distributed optimization and learning over networks. *IEEE Transactions on Signal Processing*, 60(8):4289–4305, 2012. doi:10.1109/TSP.2012.2198470.
- [CYRC12] Yongcan Cao, Wenwu Yu, Wei Ren, and Guanrong Chen. An overview of recent progress in the study of distributed multi-agent coordination. *IEEE Transactions on Industrial Informatics*, 9, 07 2012. doi:10.1109/TII.2012.2219061.
- [DBB14] M. Dorigo, M. Birattari, and M. Brambilla. Swarm robotics. *Scholarpedia*, 9(1):1463, 2014. revision #138643. doi:10.4249/scholarpedia.1463.
- [DeG74] Morris H. DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974. URL: <http://www.jstor.org/stable/2285509>.
- [Dep21] Diego Deplano. Coordination of multi-agent systems: stability via nonlinear perron-frobenius theory and consensus for desynchronization and dynamic estimation. *Università degli Studi di Cagliari*, 2021. Adv. : Franceschelli, Mauro and Giua, Alessandro URL: <http://hdl.handle.net/11584/310628>.

- [Des02] Abhijit Deshmukh. *Knowledge Management, Organizational Intelligence and Learning, and Complexity*, volume 1. IJDAI, 2002.
- [DFG20] Diego Deplano, M. Franceschelli, and A. Giua. Dynamic max-consensus and size estimation of anonymous multi-agent networks. *ArXiv*, abs/2009.03858, 2020.
- [DGCH12] Alejandro D Dominguez-Garcia, Stanton T Cady, and Christoforos N Hadjicostis. Decentralized optimal dispatch of distributed energy resources. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 3688–3693. IEEE, 2012.
- [DGFG06] Carole Delporte-Gallet, Hugues Fauconnier, Rachid Guerraoui, and Eric Ruppert. When birds die: Making population protocols fault-tolerant. In *International Conference on Distributed Computing in Sensor Systems*, pages 51–66. Springer, 2006.
- [DKJ18] Ali Dorri, Salil S. Kanhere, and Raja Jurdak. Multi-agent systems: A survey. *IEEE Access*, 6:28573–28593, 2018. doi: 10.1109/ACCESS.2018.2831228.
- [DRFL21] Israel L. Donato Ridgley, Randy A. Freeman, and Kevin M. Lynch. Self-healing first-order distributed optimization. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 3850–3856, 2021. doi:10.1109/CDC45484.2021.9683487.
- [DSPB18] Florian Dörfler, John W Simpson-Porco, and Francesco Bullo. Electrical networks and algebraic graph theory: Models, properties, and applications. *Proceedings of the IEEE*, 106(5):977–1005, 2018.
- [DYC21] Pengcheng Dai, Wenwu Yu, and Duxin Chen. Distributed Q-learning algorithm for dynamic resource allocation with unknown objective functions and application to microgrid. *IEEE Transactions on Cybernetics*, 2021.
- [Ebe20] Andreas Eberle. *Markov Processes*. University of Bonn, 2020. URL: <https://wt.iam.uni-bonn.de/fileadmin/WT/>

Inhalt/people/Andreas\_Eberle/MarkovProcesses1920/MarkovProcesses1920.pdf.

- [FC18] M. Farina and R. Carli. Partition-based distributed kalman filter with plug and play features. *IEEE Transactions on Control of Network Systems*, 5(1):560–570, 2018. doi:10.1109/TCNS.2016.2633786.
- [FF18] Mauro Franceschelli and Paolo Frasca. Proportional Dynamic Consensus in Open Multi-Agent Systems. In *57th IEEE Conference on Decision and Control (CDC 2018)*, pages 900–905, Miami, FL, United States, December 2018. URL: <https://hal.archives-ouvertes.fr/hal-01945840>, doi:10.1109/CDC.2018.8619639.
- [FF21] Mauro Franceschelli and Paolo Frasca. Stability of open multiagent systems and applications to dynamic consensus. *IEEE Transactions on Automatic Control*, 66(5):2326–2331, 2021. doi:10.1109/TAC.2020.3009364.
- [FJ99] Noah Friedkin and Eugene Johnsen. Social influence networks and opinion change. *Advances in Group Processes*, 16, 01 1999.
- [FM04a] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004. doi:10.1109/TAC.2004.834433.
- [FM04b] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004. doi:10.1109/TAC.2004.834433.
- [FZ07] F. Fagnani and S. Zampieri. Randomized consensus algorithms over large scale networks. In *Information Theory and Applications Workshop, 2007*, pages 150–159, 2007.
- [Fö07] Anna Förster. Machine learning techniques applied to wireless ad-hoc networks: Guide and survey. In *2007 3rd International Conference on Intelligent Sensors, Sensor Networks and Information*, pages 365–370. IEEE, 01 2007. doi:10.1109/ISSNIP.2007.4496871.

- [GH18] Balazs Gerencser and Julien M. Hendrickx. Push sum with transmission failures. *IEEE Transactions on Automatic Control*, 64(3):1019–1033, 2018.
- [GJ12] Sébastien Grauwin and Pablo Jensen. Opinion group formation and dynamics: Structures that last from nonlasting entities. *Phys. Rev. E*, 85:066113, Jun 2012. URL: <https://link.aps.org/doi/10.1103/PhysRevE.85.066113>, doi: 10.1103/PhysRevE.85.066113.
- [GJR<sup>+</sup>10] A Giret, Vicente Julián, M Rebollo, Estefanía Argente, Carlos Carrascosa, and V Botti. An open architecture for service-oriented virtual organizations. In *Lecture Notes in Computer Science*, vol 5919, pages 118–132, 09 2010. doi:10.1007/978-3-642-14843-9\_8.
- [Gla06] Mevludin Glavic. Agents and multi-agent systems: A short introduction for power engineers. *University of Liège*, 05 2006.
- [Haz06] Elad Hazan. Efficient algorithms for online convex optimization and their applications. *Princeton University*, 2006.
- [Haz16] Elad Hazan. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2:157–325, 01 2016. doi:10.1561/2400000013.
- [HB14] Sabine Hauert and Sangeeta N. Bhatia. Mechanisms of cooperation in cancer nanomedicine: towards systems nanotechnology. *Trends in biotechnology*, 32(9):448–455, 2014.
- [Hen14] Julien Hendrickx. Views in a graph: To which depth must equality be checked? *Parallel and Distributed Systems, IEEE Transactions on*, 25:1907–1912, 07 2014. doi:10.1109/TPDS.2013.232.
- [Hew91] Carl Hewitt. Open information systems semantics for distributed artificial intelligence. *Artificial Intelligence*, 47(1):79–106, 1991. URL: <https://www.sciencedirect.com/science/article/pii/000437029190051K>, doi:[https://doi.org/10.1016/0004-3702\(91\)90051-K](https://doi.org/10.1016/0004-3702(91)90051-K).

- [HIMM21] Yu-Guan Hsieh, Franck Iutzeler, Jérôme Malick, and Panayotis Mertikopoulos. Optimization in open networks via dual averaging. *arXiv preprint arXiv:2105.13348*, 2021.
- [HJS06] Trung Dong Huynh, Nicholas R Jennings, and Nigel R Shadbolt. An integrated trust and reputation model for open multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.
- [HK02] Rainer Hegselmann and Ulrich Krause. Opinion dynamics and bounded confidence models, analysis and simulation. *Journal of Artificial Societies and Social Simulation*, 5, 07 2002.
- [HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.
- [HM16] J. M. Hendrickx and S. Martin. Open multi-agent systems: Gossiping with deterministic arrivals and departures. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1094–1101, 2016. doi: 10.1109/ALLERTON.2016.7852357.
- [HM17] J. M. Hendrickx and S. Martin. Open multi-agent systems: Gossiping with random arrivals and departures. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 763–768, 2017. doi:10.1109/CDC.2017.8263752.
- [HMK<sup>+</sup>21] Ben Haydon, Kirti D. Mishra, Patrick Keyantuo, Dimitra Panagou, Fotini Chow, Scott Moura, and Chris Vermillion. Dynamic coverage meets regret: Unifying two control performance measures for mobile agents in spatiotemporally varying environments. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 521–526, 2021. doi:10.1109/CDC45484.2021.9682826.
- [HMQS16] Jonathan Hermon, Ben Morris, Chuan Qin, and Allan Sly. The social network model on infinite graphs, 2016. URL: <https://arxiv.org/abs/1610.04293>, doi:10.48550/ARXIV.1610.04293.

- [HOT11] J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Distributed anonymous discrete function computation. *IEEE Transactions on Automatic Control*, 56(10):2276–2289, 2011.
- [HR20] Julien M. Hendrickx and Michael G. Rabbat. Stability of decentralized gradient descent in open multi-agent systems. In *59th IEEE Conference on Decision and Control, CDC 2020, Jeju Island, South Korea, December 14-18, 2020*, pages 4885–4890. IEEE, 2020. doi:10.1109/CDC42340.2020.9304049.
- [IK88] Toshihide Ibaraki and Naoki Katoh. *Resource Allocation Problems: Algorithmic Approaches*. MIT press, 1988.
- [ITW<sup>+</sup>22] Ahmed Imteaj, Urmish Thakker, Shiqiang Wang, Jian Li, and M. Hadi Amini. A survey on federated learning for resource-constrained iot devices. *IEEE Internet of Things Journal*, 9(1):1–24, 2022. doi:10.1109/JIOT.2021.3095077.
- [ITY<sup>+</sup>14] Gerardo Iñiguez, János Török, Taha Yasseri, Kimmo Kaski, and János Kertész. Modeling social dynamics in a collaborative environment. *EPJ Data Science*, 3(1):7, Sep 2014. doi:10.1140/epjds/s13688-014-0007-z.
- [JLM03] A. Jadbabaie, Jie Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003. doi:10.1109/TAC.2003.812781.
- [JVM08] A Jamakovic and Piet Van Mieghem. On the robustness of complex networks by using the algebraic connectivity. In *International conference on research in networking*, pages 183–194. Springer, 2008.
- [KDG03] David Kempe, Alin Dobra, and Johannes Gehrke. Gossip-based computation of aggregate information. *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 482–491, 2003.
- [KMA<sup>+</sup>21] Peter Kairouz, Brendan H. McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael Gregorio Lucas d’Oliveira, Salim El

- Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konečný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrede Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Ozgür, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian Urban Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. *Advances and Open Problems in Federated Learning*. *Foundations and Trends in Machine Learning*, 14(1-2):1–210, 2021. URL: <https://hal.inria.fr/hal-02406503>.
- [KR02] Jens Krause and Graeme Ruxton. *Living in Groups*. OUP Oxford, 01 2002.
- [KRK09] Jens Krause, Graeme Ruxton, and Stefan Krause. Swarm intelligence in animals and humans. *Trends in ecology & evolution*, 25:28–34, 10 2009. doi:10.1016/j.j.tree.2009.06.016.
- [KS89] James F. Kurose and Rahul Simha. A microeconomic approach to optimal resource allocation in distributed computer systems. *IEEE Transactions on Computers*, 38(5):705–717, 1989.
- [KVC<sup>+</sup>19] Solmaz S. Kia, Bryan Van Scoy, Jorge Cortes, Randy A. Freeman, Kevin M. Lynch, and Sonia Martinez. Tutorial on dynamic average consensus: The problem, its applications, and the algorithms. *IEEE Control Systems*, 39(3):40–72, June 2019. doi:10.1109/MCS.2019.2900783.
- [Lau04] Alan J. Laub. *Matrix Analysis For Scientists And Engineers*. Society for Industrial and Applied Mathematics, USA, 2004.
- [Lov12] L. Lovász. *Large Networks and Graph Limits*. American Mathematical Society, 2012.

- [LRC01] Herbert Levine, Wouter-Jan Rappel, and Inon Cohen. Self-organization in systems of self-propelled particles. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 63:017101, 02 2001. doi:10.1103/PhysRevE.63.017101.
- [LSN<sup>+</sup>11] Naomi Leonard, Tian Shen, Benjamin Nabet, Luca Scardovi, Iain Couzin, and Simon Levin. Decision versus compromise for animal groups in motion. *Proceedings of the National Academy of Sciences of the United States of America*, 109:227–32, 12 2011. doi:10.1073/pnas.1118318108.
- [LVDH15] Riccardo Lucchese, Damiano Varagnolo, Jean-Charles Delvenne, and Julien Hendrickx. Network cardinality estimation using max consensus: The case of bernoulli trials. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 895–901, 2015. doi:10.1109/CDC.2015.7402342.
- [LXL22] Xiuxian Li, Lihua Xie, and Na Li. A survey of decentralized online learning, 2022. URL: <https://arxiv.org/abs/2205.00473>, doi:10.48550/ARXIV.2205.00473.
- [LYH17] Shu Liang, Peng Yi, and Yiguang Hong. Distributed Nash equilibrium seeking for aggregative games with coupled constraints. *Automatica*, 85:179–185, 2017.
- [MA22] José Ángel Morell and Enrique Alba. Dynamic and adaptive fault-tolerant asynchronous federated learning using volunteer edge devices. *Future Generation Computer Systems*, 133, 03 2022. doi:10.1016/j.future.2022.02.024.
- [MH19] Charles Monnoyer de Galland and Julien M. Hendrickx. Lower bound performances for average consensus in open multi-agent systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 7429–7434, Dec 2019. doi:10.1109/CDC40024.2019.9029319.
- [MH20] Charles Monnoyer de Galland and Julien M. Hendrickx. Fundamental performance limitations for average consensus in open multi-agent systems. *arXiv e-prints*, 2020. arXiv:2004.06533.

- [MHP21] Aritra Mitra, Hamed Hassani, and George J. Pappas. Online federated learning. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 4083–4090, 2021. doi:10.1109/CDC45484.2021.9683589.
- [MLB20] W. M. Haddad, J. Lee, and P. Bhat. Asymptotic and finite time semistability for nonlinear discrete-time systems with application to network consensus. *Transactions on Automatic Control*, 2020.
- [MMH20] Charles Monnoyer de Galland, Samuel Martin, and Julien M. Hendrickx. Modelling gossip interactions in open multi-agent systems. *arXiv e-prints*, 2020. arXiv:2009.02970.
- [MMRB20] Cristina Morariu, Octavian Morariu, Silviu Răileanu, and Theodor Borangiu. Machine learning for predictive scheduling and resource allocation in large scale manufacturing systems. *Computers in Industry*, 120:103244, 2020. URL: <https://www.sciencedirect.com/science/article/pii/S0166361519311595>, doi:<https://doi.org/10.1016/j.compind.2020.103244>.
- [Mon18] Charles Monnoyer de Galland. Efficiency bounds in computations on open multi-agent systems. *Ecole polytechnique de Louvain, Université catholique de Louvain*, 2018. Prom. : Hendrickx, Julien M. URL: <http://hdl.handle.net/2078.1/thesis:14719>.
- [MR19] F. Molinari and J. Raisch. Efficient consensus-based formation control with discrete-time broadcast updates. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4172–4177, Dec 2019. doi:10.1109/CDC40024.2019.9029346.
- [MS02] Alfred Müller and Dietrich Stoyan. *Comparison methods for stochastic models and risks*. Chichester ; New York, NY : Wiley, 2002. Includes bibliographical references (p. [303]-323) and index.
- [MSWS13] Dhruv Mahajan, Sundararajan Sellamanickam, Markus Weimer, and Keerthi Selvaraj. Towards resource-

- elastic machine learning, January 2013. URL: <https://www.microsoft.com/en-us/research/publication/towards-resource-elastic-machine-learning/>.
- [MVH<sup>+</sup>21] Charles Monnoyer de Galland, Renato Vizuete, Julien M. Hendrickx, Paolo Frasca, and Elena Panteley. Random coordinate descent algorithm for open multi-agent systems with complete topology and homogeneous agents. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 1701–1708, 2021. doi:10.1109/CDC45484.2021.9683049.
- [MVH<sup>+</sup>22] Charles Monnoyer de Galland, Renato Vizuete, Julien M. Hendrickx, Elena Panteley, and Paolo Frasca. Random coordinate descent for resource allocation in open multi-agent systems, 2022. URL: <https://arxiv.org/abs/2205.10259>, doi:10.48550/ARXIV.2205.10259.
- [NAR10] Behrang Monajemi Nejad, Sid Ahmed Attia, and Jörg Raisch. Max-consensus in a max-plus algebraic setting: The case of switching communication topologies. *IFAC Proceedings Volumes*, 43(12):173–180, 2010. 10th IFAC Workshop on Discrete Event Systems. URL: <https://www.sciencedirect.com/science/article/pii/S1474667015324526>, doi:<https://doi.org/10.3182/20100830-3-DE-4013.00030>.
- [Nec13] I. Necoara. Random coordinate descent algorithms for multi-agent convex optimization over networks. *IEEE Transactions on Automatic Control*, 58(8):2001–2012, 2013.
- [Nes12] Yurii Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.
- [Nes18] Yurii Nesterov. *Lectures on Convex Optimization*. Springer, 2018.
- [NO09] A. Nedić and A. Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54:48–61, 2009.

- [NO13] Angelia Nedic and Alex Olshevsky. Distributed optimization over time-varying directed graphs. *Automatic Control, IEEE Transactions on*, 60, 03 2013. doi:10.1109/TAC.2014.2364096.
- [NOP10] Angelia Nedic, Asuman Ozdaglar, and Pablo A. Parrilo. Constrained consensus and optimization in multi-agent networks. *IEEE Transactions on Automatic Control*, 55(4):922–938, 2010. doi:10.1109/TAC.2010.2041686.
- [NOR18] Angelia Nedić, Alexander Olshevsky, and Michael G. Rabbat. Network topology and communication-computation tradeoffs in decentralized optimization. *Proceedings of the IEEE*, 106:953–976, 2018.
- [NW06] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer Science & Business Media, 2006.
- [NZJY21] S. Novoth, Q. Zhang, K. Ji, and D. Yu. Distributed formation control for multi-vehicle systems with splitting and merging capability. *IEEE Control Systems Letters*, 5(1):355–360, 2021. doi:10.1109/LCSYS.2020.3002218.
- [Oli18] Alexandre Olikier. Open multi-agent systems with fixed-sized and possibly not complete topologies. *Ecole polytechnique de Louvain, Université catholique de Louvain*, 2018. Prom. : Hendrickx, Julien M. URL: <http://hdl.handle.net/2078.1/thesis:14693>.
- [OPS18] Alex Olshevsky, Ioannis Paschalidis, and Artin Spiridonoff. Robust asynchronous stochastic gradient-push: Asymptotically optimal and network-independent performance for strongly convex functions. *Journal of Machine Learning*, 11 2018.
- [OR00] James M Ortega and Werner C Rheinboldt. *Iterative solution of nonlinear equations in several variables*. SIAM, 2000.
- [OSFM07] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007. doi:10.1109/JPROC.2006.887293.

- [OSFS08] Reza Olfati-Saber and Nils F. Sandell. Distributed tracking in sensor networks with limited sensing range. In *2008 American Control Conference*, pages 3157 – 3162, 07 2008. doi:10.1109/ACC.2008.4586978.
- [OSM04] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004. doi:10.1109/TAC.2004.834113.
- [Ota06] Jun Ota. Multi-agent robot systems as distributed autonomous systems. *Advanced Engineering Informatics*, 20(1):59–70, jan 2006. doi:10.1016/j.aei.2005.06.002.
- [OYM<sup>+</sup>01] Ryokichi Onishi, Saneyasu Yamaguchi, Hiroaki Morino, H. Aida, and T. Saito. The multi-agent system for dynamic network routing. In *Proceedings of the Fifth International Symposium on Autonomous Decentralized Systems*, volume E84B, pages 375 – 382, 02 2001. doi:10.1109/ISADS.2001.917442.
- [Pac98] Baburao Pachpatte. *Inequalities for Differential and Integral Equations*. Academic Press, 1998.
- [Par18] Kun Park. *Fundamentals of Probability and Stochastic Processes with Applications to Communications*. Springer, 01 2018. doi:10.1007/978-3-319-68075-0.
- [Pat08] Michael Patriksson. A survey on the continuous nonlinear resource allocation problem. *European Journal of Operational Research*, 185, 2008.
- [Pav09] G.A. Pavliotis. *Applied Stochastic Processes*. Springer New York, NY, 2009. URL: [https://www.ma.imperial.ac.uk/~pavl/lecture\\_notesM4A42.pdf](https://www.ma.imperial.ac.uk/~pavl/lecture_notesM4A42.pdf).
- [PKP06] Joel Predd, S.B. Kulkarni, and H. Vincent Poor. Distributed learning in wireless sensor networks. *Signal Processing Magazine, IEEE*, 23:56–69, 07 2006. doi:10.1109/MSP.2006.1657817.
- [PMS93] Maretsugu Yamasaki Paolo M. Soardi. Classification of infinite networks and its application. *Circuits, Systems and Signal Processing*, 1993.

- [PN18] Shi Pu and Angelia Nedic. Distributed stochastic gradient tracking methods. *Mathematical Programming*, 05 2018.
- [Pri18] N. Privault. *Understanding Markov Chains: Examples and Applications*. Springer, 2018.
- [PT17] Anton Proskurnikov and Roberto Tempo. A tutorial on modeling and analysis of dynamic social networks. part i. *Annual Reviews in Control*, 43:65–79, 05 2017. doi:10.1016/j.arcontrol.2017.03.002.
- [PW02] Simon Parsons and Michael Wooldridge. Game theory and decision theory in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 5:243–254, 09 2002. doi:10.1023/A:1015575522401.
- [RBA05] Wei Ren, Randal Beard, and Ella Atkins. A survey of consensus problems in multiagent coordination. In *Proceedings of the 2005, American Control Conference, 2005*, volume 3, pages 1859 – 1864 vol. 3, 07 2005. doi:10.1109/ACC.2005.1470239.
- [Rey87] Craig Reynolds. Flocks, herds, and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21:25–34, 07 1987. doi:10.1145/280811.281008.
- [RFF12] S. Rivero, M. Farina, and G. Ferrari-Trecate. Plug-and-play decentralized model predictive control. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 4193–4198, 2012. doi:10.1109/CDC.2012.6426730.
- [RFFt15] Stefano Rivero, Marcello Farina, and Giancarlo ferrari trecate. Plug-and-play state estimation and application to distributed output-feedback model predictive control. *European Journal of Control*, 25, 04 2015. doi:10.1016/j.ejcon.2015.04.001.
- [RN04] M. Rabbat and R. Nowak. Distributed optimization in sensor networks. In *Third International Symposium on Information Processing in Sensor Networks, 2004. IPSN 2004*, pages 20–27, April 2004. doi:10.1109/IPSN.2004.238733.

- [RT14] Peter Richtárik and Martin Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1):1–38, 2014.
- [RTB04] R.L. Raffard, C.J. Tomlin, and S.P. Boyd. Distributed optimization for cooperative agents: application to formation flight. In *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, volume 3, pages 2453–2459 Vol.3, 2004. doi:10.1109/CDC.2004.1428778.
- [SBB<sup>+</sup>17] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. *arXiv e-prints*, 2017. arXiv:1702.08704.
- [SC13] P. Sen and B. K. Chakrabarti. *Sociophysics: an introduction*. Oxford University Press, 2013.
- [SDP<sup>+</sup>20] Andrea Simonetto, Emiliano Dall’Anese, Santiago Paterlain, Geert Leus, and Georgios B. Giannakis. Time-varying convex optimization: Time-structured algorithms and applications. *Proceedings of the IEEE*, 108(11):2032–2048, 2020. doi:10.1109/JPROC.2020.3003156.
- [SJ16] Shahin Shahrampour and Ali Jadbabaie. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, PP, 09 2016. doi:10.1109/TAC.2017.2743462.
- [SL19] Bryan Van Scoy and Laurent Lessard. A distributed optimization algorithm over time-varying graphs with efficient gradient evaluations. *IFAC-PapersOnLine*, 52:357–362, 2019.
- [SLWY14] Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25, 04 2014. doi:10.1137/14096668X.
- [Soa94] Paolo M. Soardi. *Potential Theory on Infinite Networks*. Elsevier, 1994.

- [Son08] Isaac M. Sonin. The decomposition-separation theorem for finite nonhomogeneous markov chains and related problems, 2008.
- [SS12] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012. URL: <http://dx.doi.org/10.1561/22000000018>, doi:10.1561/22000000018.
- [SSF19] Z. A. Z. Sanai Dashti, C. Seatzu, and M. Franceschelli. Dynamic consensus on the median value in open multi-agent systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3691–3697, Dec 2019. doi:10.1109/CDC40024.2019.9029451.
- [Sta18] Tracy Staedter. 100,000 iot sensors monitor a 1,400-km canal in china. *IEEE Spectrum, TechTalk Blog,*, 2018.
- [TASJ13] André Teixeira, José Araújo, Henrik Sandberg, and Karl H Johansson. Distributed actuator reconfiguration in networked control systems. *IFAC Proceedings Volumes*, 46(27):61–68, 2013.
- [TBA86] J. Tsitsiklis, D. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, 1986. doi:10.1109/TAC.1986.1104412.
- [THG17] A. B. Taylor, J. M. Hendrickx, and F. Glineur. Performance estimation toolbox (PESTO): Automated worst-case analysis of first-order optimization methods. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 1278–1283. IEEE, 2017. doi:10.1109/CDC.2017.8263832.
- [Tia21] Ye Tian. Distributed optimization for machine learning: Guarantees and tradeoffs. *Purdue University Graduate School*, 2021. URL: <https://doi.org/10.25394/PGS.15029829.v1>.
- [TIY<sup>+</sup>13] János Török, Gerardo Iñiguez, Taha Yasseri, Maxi San Miguel, Kimmo Kaski, and János Kertész. Opinions, conflicts, and consensus: modeling social dynamics in a collaborative environment. *Physical review letters*, 110(8):088701, 2013.

- [Ton18] O. K. Tonguz. Red light, green light—no light: Tomorrow’s communicative cars could take turns at intersections. *IEEE Spectrum*, 55(10):24–29, Oct 2018. doi:10.1109/MSPEC.2018.8482420.
- [TT98] John Toner and Yuhai Tu. Flocks, herds, and schools: A quantitative theory of flocking. *Phys. Rev. E*, 58:4828–4858, Oct 1998. URL: <https://link.aps.org/doi/10.1103/PhysRevE.58.4828>, doi:10.1103/PhysRevE.58.4828.
- [VCB]<sup>+</sup>95] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. Novel type of phase transition in a system of self-driven particles. *Phys. Rev. Lett.*, 75:1226–1229, Aug 1995. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.75.1226>, doi:10.1103/PhysRevLett.75.1226.
- [VFG20] R. Vizuete, P. Frasca, and F. Garin. Graphon-based sensitivity analysis of sis epidemics. *IEEE Control Systems Letters*, 4(3):542–547, 2020.
- [VFP21] R. Vizuete, P. Frasca, and E. Panteley. On the influence of noise in randomized consensus algorithms. *IEEE Control Systems Letters*, 5(3):1025–1030, 2021.
- [VMH18] Vineeth Varma, Irinel-Constantin Morarescu, and Yezekael Hayel. Continuous time opinion dynamics of agents with multi-leveled opinions and binary actions. In *IEEE Conference on Computer Communications, INFOCOM 2018*, Honolulu, United States, April 2018. URL: <https://hal.archives-ouvertes.fr/hal-01756831>.
- [VMH]<sup>+</sup>22] Renato Vizuete, Charles Monnoyer de Galland, Julien M. Hendrickx, Paolo Frasca, and Elena Panteley. Resource allocation in open multi-agent systems: an online optimization analysis, 2022. URL: <https://arxiv.org/abs/2207.09316>, doi:10.48550/ARXIV.2207.09316.
- [VMN18] Vineeth S. Varma, Irinel-Constantin Morărescu, and Dragan Nešić. Open multi-agent systems with discrete states and stochastic interactions. *IEEE Control Systems Letters*, 2(3):375–380, 2018. doi:10.1109/LCSYS.2018.2840431.

- [VZ12] Tamas Vicsek and Anna Zafeiris. Collective motion. *Physics Reports*, 517(3):71–140, 2012. Collective motion. URL: <https://www.sciencedirect.com/science/article/pii/S0370157312000968>, doi:<https://doi.org/10.1016/j.physrep.2012.03.004>.
- [WA11] Craig Wester and Mark Adamiak. Practical applications of Ethernet in substations and industrial facilities. In *IEEE/IAS Pulp & Paper Industry Technical Paper Conference*, pages 55–66. IEEE, 2011.
- [WBH<sup>+</sup>21] L. Wang, C. Bernardo, Y. Hong, F. Vasca, G. Shi, and C. Altafini. Achieving consensus in spite of stubbornness: time-varying concatenated friedkin-johnsen models. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 4964–4969, 2021. doi:10.1109/CDC45484.2021.9683466.
- [Wei99] Gerhard Weiss. Multiagent systems a modern approach to distributed artificial intelligence. *AI Magazine*, 3, 10 1999.
- [Woo02] Michael Wooldridge. *An Introduction to MultiAgent Systems*, pages –348. John Wiley & Sons, 06 2002.
- [WSR22] Bo Wang, Shan Sun, and Wei Ren. Distributed time-varying quadratic optimal resource allocation subject to nonidentical time-varying Hessians with application to multiquadrotor hose transportation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2022.
- [WZGO21] Bao Wang, Difan Zou, Quanquan Gu, and Stanley J Osher. Laplacian smoothing stochastic gradient markov chain monte carlo. *SIAM Journal on Scientific Computing*, 43(1):A26–A53, 2021.
- [XB06] Lin Xiao and Stephen Boyd. Optimal scaling of a gradient method for distributed resource allocation. *Journal of Optimization Theory and Applications*, 129(3):469–488, 2006.
- [XKK20] Ran Xin, Soumya Kar, and Usman A. Khan. Decentralized stochastic optimization and machine learning: A unified variance-reduction framework for robust performance and fast convergence. *IEEE Signal Processing Magazine*, 37(3):102–113, 2020. doi:10.1109/MSP.2020.2974267.

- [XPNK20] Ran Xin, Shi Pu, Angelia Nedic, and Usman Khan. A general framework for decentralized optimization with first-order methods. *Proceedings of the IEEE*, 108:1869–1889, 11 2020. doi:10.1109/JPROC.2020.3024266.
- [YHL16] Peng Yi, Yiguang Hong, and Feng Liu. Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems. *Automatica*, 74:259–269, 2016.
- [YK96a] M. Yamashita and T. Kameda. Computing on anonymous networks. i. characterizing the solvable cases. *IEEE Transactions on Parallel and Distributed Systems*, 7(1):69–89, Jan 1996. doi:10.1109/71.481599.
- [YK96b] Masafumi Yamashita and Tiko Kameda. Computing on anonymous networks: Part ii - decision and membership problems. *Parallel and Distributed Systems, IEEE Transactions on*, 7:90 – 96, 02 1996. doi:10.1109/71.481600.
- [YLY13] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26, 10 2013. doi:10.1137/130943170.
- [YLY<sup>+</sup>21] Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and Karl Johansson. Regret and cumulative constraint violation analysis for online convex optimization with long term constraints. In *International Conference on Machine Learning*, pages 11998–12008. PMLR, 2021.
- [YQG<sup>+</sup>18] Mengbin Ye, Yuzhen Qin, Alain Govaert, Brian Anderson, and Ming Cao. An influence network model to study discrepancies in expressed and private opinions. *Automatica*, 06 2018.
- [YSVNVQ10] Feng Yan, Shreyas Sundaram, S V. N. Vishwanathan, and Yuan Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. *IEEE Transactions on Knowledge and Data Engineering*, 25, 06 2010. doi:10.1109/TKDE.2012.191.

★ | Bibliography

- [YTL<sup>+</sup>18] Mengbin Ye, Minh Trinh, Young-Hun Lim, Brian Anderson, and Hyo-Sung Ahn. Continuous-time opinion dynamics on multiple interdependent topics. *Automatica*, 05 2018.

# Index

- $L_p^\dagger$ -seminorm, 183
- Accumulated error, 217
- Age of information, 107
- Agent, 2
- Agreement state, 40
- Algebraic connectivity, 17
- Arrival, 52, 100
- Asynchronous algorithm, 41
- Average consensus, 42, 98
  
- Benefit, 219
- Birth-death process, 28
  
- Complementary event, 22
- Complete bipartite graph, 15
- Complete graph, 15
- Condition number, 171
- Conditional expectation, 24
- Conditional probability, 22
- Connected graph, 15
- Consensus, 39
- Continuous-time Markov chain, 24
- Continuous-time Markov process, 24
- Convergence rate, 182
- Coordinate Descent, 158
- Counting process, 26
- Cumulative distribution function, 23
  
- Decentralized Gradient Descent, 157
- Decentralized optimization, 156
- Degree, 15
- Demand, 212
- Departure, 100
- Departure (adversarial), 52
- Departure (random), 52
- Descriptor vector, 55
- Directed graph, 14
- Dynamic composition, 20
- Dynamic regret, 164, 219
  
- Edge, 14
- Effective resistance, 18
- Event (of a clock), 30, 58
- Event (probabilistic), 22
- Event sequence (deterministic), 104
- Event sequence (stochastic), 108
- Event set, 31, 58, 104, 173
- Expectation/expected value, 23
- Exponential convergence, 182
- Exponential distribution, 27
- External problem, 31
  
- Feasible set, 214
- Flooding, 33, 44
  
- Gossip algorithm, 42
- Gossip interaction, 52

- Gossip model, 124
- Graph, 14
- Heterogeneous agents, 180
- History of a process, 31, 174
- Homogeneous agents, 180
- Indicator function, 28
- Infinitesimal Generator Operator, 92
- Infinitesimal generator operator, 26
- Information exchange, 100
- Instantaneous error, 217
- Instantaneous jump rate, 26, 92
- Intrinsic problem, 31
- Knowledge set, 105
- Knowledge set function, 111
- Laplacian matrix, 16
- Line graph, 15
- Linear convergence, 182
- Markov jump process, 24
- Max consensus, 32, 44
- Mean, 53
- Mean of squares, 54
- Mean Squared Error, 102
- Multi-agent system, 1
- Neighbour, 15
- Node, 14
- Open multi-agent system, 2, 6
- Optimal estimate, 116
- Optimal strategy, 218
- Optimality condition, 172
- Pairwise gossip interactions, 42
- Pairwise update, 172
- Path, 15
- Ping model, 119
- Ping update, 119
- Ping' model, 123
- Poisson clock, 26
- Poisson process, 26
- Potential benefit, 219
- Probability, 22
- Probability density function, 23
- Probability space, 22
- Pseudo-CDF, 113
- Pseudo-PDF, 113
- Pseudoinverse, 17
- Push-sum algorithm, 43
- Random Coordinate Descent, 159
- Random variable, 23
- Replacement, 172
- Replacement (adversarial), 52
- Replacement (random), 52
- Replacement (single), 100
- Resource allocation, 158
- Ring graph, 15
- Scaled Laplacian, 176
- Selfish strategy, 218
- SI model, 128
- SIS process, 125
- Smooth, 170
- Squared mean, 54
- Star graph, 15
- State space, 24
- Statistical independence, 22
- Stochastic process, 24
- Strongly convex, 170
- Superset of agents, 18
- Symmetric pairwise update, 41
- Synchronization, 40
- Synchronous algorithm, 41
- System, 1
- Time-homogeneous CTMC, 24

Transition kernel, 25

Transition rate, 25

Transition rate matrix, 25

Usual stochastic order, 115

Variance, 54