

Improving Secure and Robust Patient Service Delivery

Eduard Baranov, Thomas Given-Wilson, Axel Legay

Université Catholique de Louvain, Belgium
`firstname.lastname@uclouvain.be`

Abstract. The Wablieft project explores how to improve medical service delivery through a shared marketplace for service providers. This shared marketplace allows patients to choose services from providers and so support improved service delivery and patient satisfaction. Having a shared marketplace raises some service reliability and correctness challenges, as well as creates opportunities for improved information gathering. This work formalises the shared marketplace to prove correct behaviour and properties of the marketplace behaviour. The information available to the shared marketplace is also used to improve predictions of medical scenarios such as pandemics, and thus improve service delivery.

1 Introduction

The Wablieft project considers how to improve the delivery of services to patients by using a shared marketplace [6]. Current service providers may be over- or under-utilised due to patients being assigned to a specific service provider by their hospital. By pooling the information on services and allowing patients to choose the service provider (and hence timing, locations, etc.) of the service that is best for them, the goal is to improve the utilisation and overall delivery of services. This pooling of information is done through a shared marketplace where service providers offer their services, and patients are able to choose the service and service provider that best matches them

This shared marketplace is designed to benefit all parties involved, hospitals, patients, service providers (and others such as payment options, legislation, etc.). For hospitals they do not need to manage a single or multiple service providers directly and can instead use the marketplace to offer many services (and service providers). For patients this provides options to choose service providers according to patient preferences, e.g. more convenient location, better appointment times, etc. For service providers this allows many more potential patients to find services, and to more efficiently schedule services (ensuring more consistent utilisation). There are also advantages for other actors in the Wablieft project related to patient compliance, auditing, financial services, and legal protections, although these are not detailed in this work.

Having a shared marketplace raises some service reliability and correctness challenges, as well as creates opportunities for improved information gathering.

By using this new approach to service provisioning there are potential concerns with implementation.

One of the main challenges is to ensure that the shared marketplace behaves correctly and fairly toward all users. This requires some delicacy in how the different users can interact with the shared marketplace, and how to ensure this cannot be exploited. Also by having a shared marketplace system, the common behaviours to all users of the same class should be handled in a consistent manner, e.g. all service providers should have the same experience. There are also questions of accountability in a shared marketplace, where all the users may wish to have consistent, immutable, and reliable records of the usage of the system.

The information available to the shared marketplace is also potentially able to provide benefits to users of the system. The shared marketplace has a global view of how the market is behaving and can use this to improve outcomes for all parties. For example, consider when service provisioning may be interrupted by emergency service requirements (e.g. an operating theatre that was scheduled for one patient is required for an emergency). Then in the case of some larger occurrence like a pandemic, predicting the spread to other regions can improve service delivery by preventing too many services being offered and cancelled.

This work formalises the shared marketplace in Uppaal SMC [1] and uses statistical model checking [4, 7, 9] to analyse key behavioural properties. The models were made easily reconfigurable, thus allowing many experiments with differing input parameters (number of patients, services, kind of blockchain, kind of pandemic model etc.). To model the occurrence of a pandemic and predictive power of the shared marketplace, we used a simplified version of the model, optimised to improve performance while considering a large number of patients.

The main contributions of this work are as follows.

- A formal model for a shared marketplace of the Wablieft project.
- Prove the correctness of the model of the Wablieft project.
- Demonstrate the predictive advantage of global knowledge from a shared marketplace as in the Wablieft project.

The structure of the paper is as follows. Section 2 recalls important background for understanding this paper. Section 3 overviews the Wablieft project and its goals. Section 4 presents the models of the Wablieft project. Section 5 experiments with how to improve service delivery in the face of a pandemic. Section 6 concludes and considers future work.

2 Background

This paper makes use of formal verification techniques. The key concepts are briefly introduced here for those unfamiliar with the field. In formal verification both the system and the requirements are represented via mathematical models and formulas on which (mathematical) operations are applied. The usage of a model allows for validation at design time, in particular early in the project life

cycle. One common approach to verification is *Model Checking* (MC) (see [3] for a detailed survey) where the system is represented by a *Transition System* (TS) or graph where nodes represent states of the system and ordered edges between nodes represent transitions between those states. The execution of the system is represented by a sequence $n_0 e_0 n_1 e_1 n_2 \dots$, where n_i s are nodes and e_i s are edges from n_i to n_{i+1} . Requirements are represented by temporal logic formulas [3]. For example, both linear and branching temporal logics (resp. LTL/CTL) are sufficient to represent requirements that make (temporal) hypothesis on a given set of executions. Such logics extend classical Boolean logics with temporal operators over sequences of states. As an example, the LTL formula “ $\Box a$ ” says that proposition “ a ” must be true in each state of each execution, while “ $\Diamond a$ ” says that for each execution there must be a state in the execution where “ a ” is true.

The advantages of MC are that 1. it can be deployed at all steps of conception time, and 2. it is exhaustive as it explores the entire behaviors of the system.

A wide range of works have focused on quantitative systems. Such systems are TS where transitions are equipped with quantitative values such as cost or probabilities. This allows us to express quantitative measures, e.g., duration of an execution or probability/cost of an action. In such cases, logics are themselves extended with quantitative operators, which allows us to pose quantitative questions like “what is the probability of termination”, or “what is the probability that we avoid a deadlock”, or “what is the timing cost of a specific execution”.

Unfortunately, MC requires exploration of all executions and this is infeasible for large models. This problem is also known as the state-space explosion problem. To address this problem, an alternative approach was proposed based on algorithms from statistics. The core idea of *Statistical Model Checking* (SMC) [7–9] is to make many simulations of the model during which properties are monitored. Then, the statistical algorithm (e.g. Monte-Carlo) is used to decide the probability of the property to be satisfied with some degree of confidence. The level of confidence can be tuned with the number of simulations. Being based on simulation techniques, SMC is known to be less time and memory consuming than exhaustive methods. SMC is usually used to monitor bounded executions, therefore Bounded LTL/CTL logics are used - versions that can be decided on finite executions.

Uppaal SMC [1, 5] is a statistical model checker using stochastic timed automata models [2]. The stochastic extension adds probabilistic choice between transitions and probability distributions for time delays. Uppaal SMC provides several queries for statistical model checking: “probability estimation” - probability of the property to be satisfied within the given timebound; “hypothesis testing” - comparing the probability of the property to be satisfied with a threshold; “probability comparison” of two properties. In addition Uppaal SMC supports evaluation of expected values of an expression.

3 The Wablieft Project

The motivation for the Wablieft project is to improve the delivery of healthcare services through a shared marketplace.

3.1 The Wablieft Marketplace

The Wablieft shared marketplace is designed to bring together various actors in the healthcare sector, including: hospitals, patients, and service providers amongst others. (Note this work will focus on the interactions between these three and the shared marketplace.)

A typical interaction of these actors without the shared marketplace is as follows. The patient visits a hospital and is prescribed to receive a treatment. The hospital then assigns a service provider to provide the treatment to the patient. The patient arranges an appointment for the treatment with the service provider. The patient receives the treatment at the appointed time from the (hospital) chosen service provider.

One main inefficiency in this approach is that the hospital chooses the service provider. In practice other service providers may also be able to provide the same service, often at a more convenient time or location for the patient. This is where the shared marketplace is designed to improve the service.

A typical interaction of the above actors with the shared marketplace is as follows. The patient visits a hospital and prescribed to receive a treatment. The hospital provides the patient with a voucher to use in the shared marketplace. The patient uses this voucher to arrange an appointment with the service provider they prefer (and receive a coupon for this appointment). The patient receives the treatment at the appointed time from their chosen service provider.

The flow of interactions with the shared marketplace is not significantly changed, except that now the patient can choose a service provider taking into account their preferences. The hospital can also benefit by no longer needing to have a priori knowledge of all service providers, or working with many service providers (or being limited by the service providers they have a prior relationship with). Similarly the service providers benefit by having a larger pool of patients they can offer services to, and also not requiring direct relationships with many hospitals.

3.2 Safe and Secure Behaviour

There are several aspects of service delivery where safety and security are highly desirable. Here these do not necessarily depend upon the use of a shared marketplace. However, since the goal of the Wablieft project is to gain the advantages of a shared marketplace, this is an opportunity to ensure desired behaviours are guaranteed by this new approach to service delivery.

One central requirement from all of the actors is a fair use of the shared marketplace. This manifests in several ways. One is that all actors are treated equally in their role (all patients are equal to each other, all service providers are

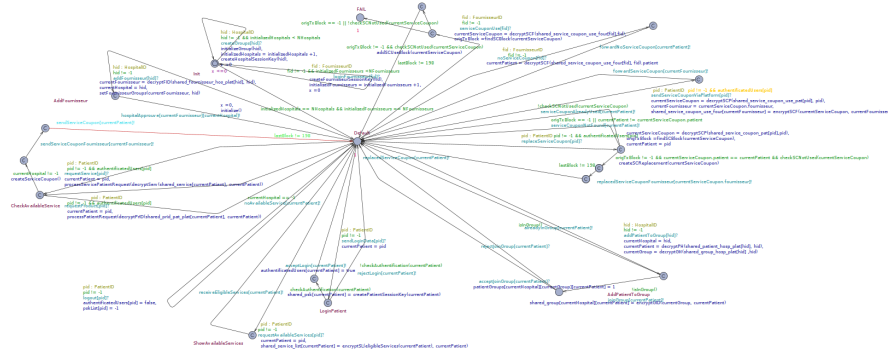


Fig. 1: Marketplace template for wablief model.

equal to each other, all hospitals are equal to each other, etc.). In particular this means that no patient, service provider, or hospital is prioritised over another. Another is the capability to inspect and audit all sales done by the marketplace. This allows for all actors to examine the actions taken and ensure their actions are correctly recorded. This motivated the desire for an immutable record of the use of the shared marketplace.

There are also several properties related to how the marketplace supports coupons for patients to use services. The following are related to correct usage, and the inability for malicious usage of the marketplace.

- A coupon can be used only once. That is, a coupon can only be used to gain a service and never re-used.
- A revoked coupon cannot be used. Since a coupon may be lost or replaced, this coupon can be revoked. Once a coupon is revoked it cannot be used to gain a service.
- A forged coupon (i.e. not issued by the marketplace) cannot be used. This ensures that only coupons created by the shared marketplace can be used on the shared marketplace.

There are also privacy related properties, for example a patient cannot receive coupons issued to other patients. Similarly, service providers cannot see patients that are not using their service without a coupon, and hospitals only have knowledge about their own patients.

4 Modelling Wablief

This section presents the models of the Wablief project and the properties to ensure safe and secure operation of the market.

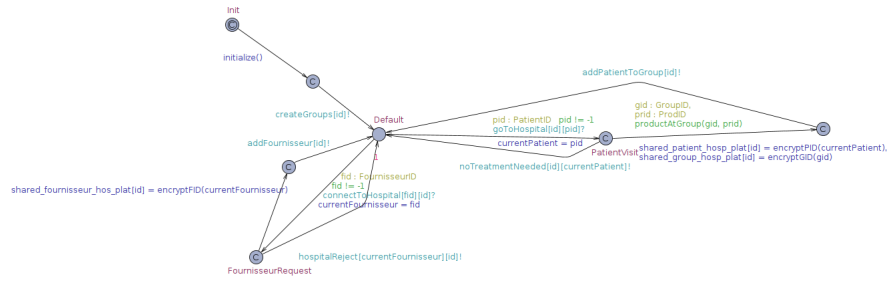


Fig. 2: Hospital template for wablieft model.

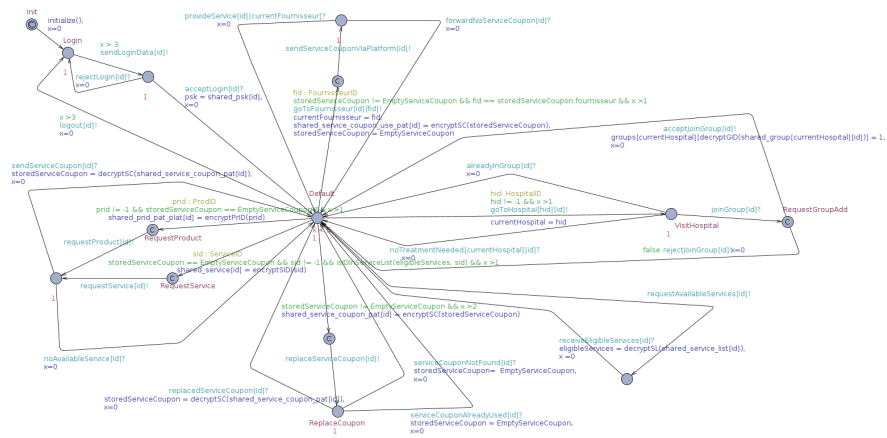


Fig. 3: Patient template for wablieft model.

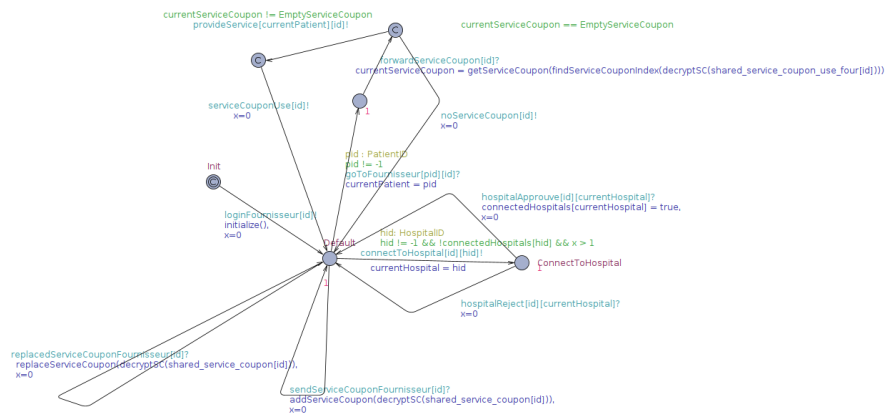


Fig. 4: Service Provider template for wablief model.

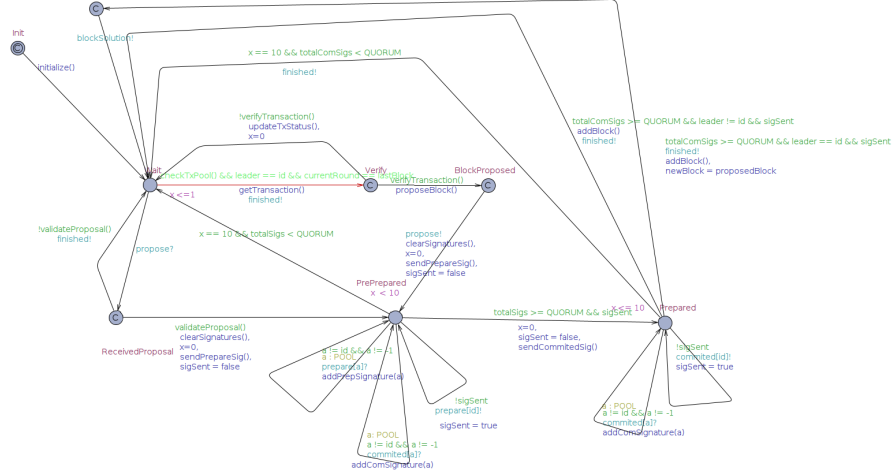


Fig. 5: Blockchain validator template for wablief model.

4.1 The Wablief Model

The marketplace and other actors have been modelled in Uppaal SMC. There are four templates for each of the actors: marketplace Fig. 1, hospitals Fig. 2, patients Fig. 3, and service providers Fig. 4.

The general workflow is as follows. The first step is when a patient goes to a hospital and if the patient needs treatment then the hospital adds information to the marketplace that the patient requires a particular medical service. Note that at this stage the patient must opt-in to using the shared marketplace and thus the addition of their information. The next step is when the patient goes to the shared marketplace and requests the prescribed medical service from any service provider of the patient's choice. The marketplace generates a coupon that is securely transferred to the patient which can be used at a service provider. As an additional option, a patient can request a coupon be reissued, e.g. in case of losing the coupon.

There are several mechanisms included in the model that help ensure required safety and security properties. Patients and service providers are required to “login” to the marketplace and obtain a session key. This key allows encryption of all communication between the actors and the marketplace preventing other actors from reading this communication. In the model this is abstracted to simply adding the key to the message and assuming the possibility of decryption only in case of knowing the key.

In order to keep track of all coupon issuing and usage, we have a blockchain that stores all this information. The presence of a blockchain keeps the data immutable and has two uses. The first is allowing actors to check the origin of each coupon, in particular whether it was really issued by the shared marketplace and whether the coupon has already been used or revoked. The second is allowing a later audit of the shared marketplace ensuring the proper behaviour.

There are several architectural decisions that have to be made during the project. The modular nature of Uppaal SMC models allows us to check different options without the considerable effort for complete remodelling the entire system. As an example of such required decision is the type of blockchain used in the project. The default option considered by industrial partners is to use a private blockchain, i.e. only the shared marketplace can create blocks in the blockchain. Another approach is to use a “consortium” blockchain that requires an agreement from several trusted partners (e.g. hospitals and service providers) to add a block to the blockchain. By simply adding two templates with validators Fig. 5 and leader controller necessary for Istanbul Byzantine Fault Tolerance consensus algorithm, it is possible to evaluate both options and prove their correct function.

4.2 The Wablieft Properties

Due to the size of the model, full verification of the properties is not feasible. Therefore, we use statistical model checking that is available at Uppaal SMC.

For the privacy properties we checked the probability that service coupon received by a patient (and decrypted) does indeed belong to this patient, in particular we check that it always holds that the patient can either have his ID or is empty.

$$Pr[\leq 1000](\bigwedge_{p:Patient} (p.storedServiceCoupon.patient == p.id \vee p.storedServiceCoupon.patient == -1)) \quad (1)$$

For the properties related to coupon misuse, we added a malicious patient template that intentionally attempts to reuse or forge a coupon. In case of a success, the malicious patient goes into a successful state, and we can evaluate the probability of reaching such state.

For the evaluation of properties we use an instantiation with two hospitals, ten patients, and two service providers. The blockchain size is bounded by two hundred blocks and the simulation lasts until the blockchain is full. Each property requires approximately 21 seconds to evaluate and the satisfaction probability is above 98% with 99% confidence.

5 Marketplace Prediction Capabilities

In this section we consider two evolutions of the shared marketplace and a pandemic scenario.

The first evolution of the Wablieft model is that services may have multiple contributing components. Each service may require one or more doctors, one or more nurses, specialised room, and some medical devices. For one example,

an operation service may require an operating theater, an anesthesiologist, and two nurses. Another example is a dialysis service that requires only a nurse and a dialysis machine. Clearly if a service provider has only a limited number of nurses, this may prevent offering too many services. If they had only six nurses, then they could offer 3 operation services, or 6 dialysis services, but not both. Ideally the service provider would like to be able to offer the maximum of both, and then reduce as these services are purchased through the marketplace.

The second evolution of the Wablieft model is that emergency patients may appear, i.e. a patient who was not predicted to appear scheduled but requires immediate medical service. In practice service providers may be required to provide services for emergency patients who cannot wait to go through the marketplace (or any other waiting list) to be treated. This creates a potential conflict for the service provider who would like to offer the maximum number of services possible, but also have facilities available for emergency patients. Here this is considered as a trade-off where normal services may be cancelled if too many emergency patients appear, but only if the service provider does not have enough components (i.e. doctors, nurses, etc.). Of course service providers can somewhat predict emergencies and so do not sell all possible services assuming no emergency patients, as service cancellations are extremely poor outcomes for patients. Normally, number of emergency cases are expected to have some stable pattern, e.g. n emergencies per day with slightly more on Friday and Saturday night.

Observe that the two evolutions above together add some interesting complexity to service delivery. In particular the ability to predict how many services can be offered to normal patients without causing too many cancellations. This is made more complex (and realistic) here by the reality that some medical services may require more equipment than others, and that with limited resources (e.g. doctors, nurses, etc.) there is motivation to maximise service delivery.

This section considers how to improve service delivery, but also the advantages that can be gained by having a shared marketplace. Here we consider how the shared marketplace can improve outcomes for patients by using better information to predict emergency patient patterns. In particular we consider the scenario of a pandemic, where emergencies can grow exponentially requiring more and more resources each day. Here a single service provider may notice the fast growth and adapt their scheduling, however during the first days of pandemic multiple services could be cancelled due to unexpected number of emergency cases.

In such a scenario the marketplace's global knowledge of the population's health can better evaluate current needs. A pandemic does not start everywhere simultaneously, some locations are affected later than others. The marketplace has a capability to detect the start of the pandemic in the first location and notify service providers in other locations to be prepared.

To explore this scenario we developed a Uppaal SMC model focusing on three actors: a (shared) marketplace, several service providers, and many patients. Each day service providers select how much resources shall be kept for emergency

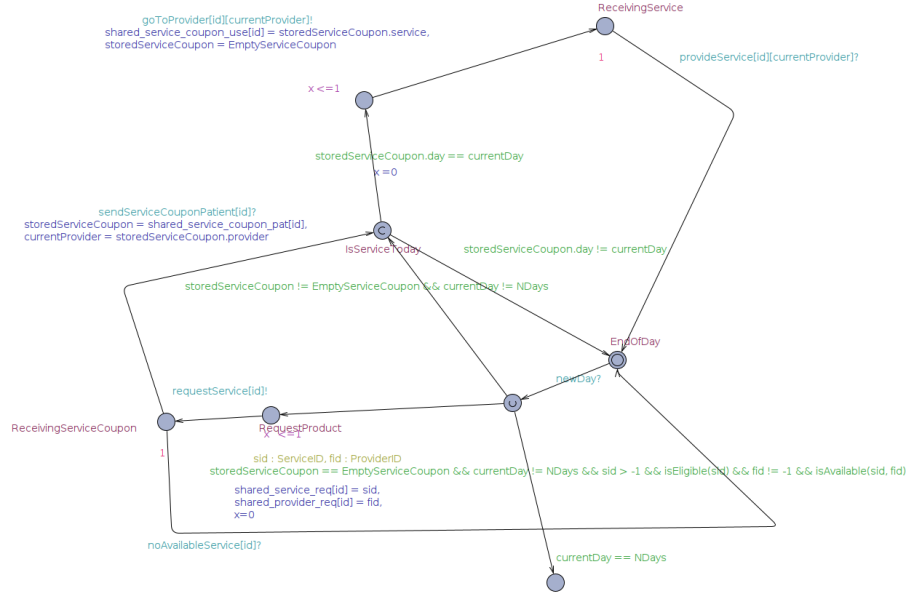


Fig. 6: Patient template for pandemic model.

cases and offer the remaining resources to be sold via the marketplace. Patients (as shown in Fig. 6) in turn can book services from the marketplace.

To focus on the predictive power of the service providers and the marketplace, the model is restricted to only selling services for the current or next day. This restriction is introduced in order to prevent the case when lots of services are already sold for several days in advance and so not allowing service providers to reserve more resources for emergency cases.

Each service provider encounters a number of emergencies during each day. This was originally modelled by a large pool of patients (who may require a normal service, or may have an emergency). However, this approach was too computationally expensive since to reasonably model both the normal services and emergency cases, a large pool of patients (with potential for these services) was required. To address this computational cost the model was simplified: each service provider model selects the number of emergencies (according to some function) and processes the emergencies at the beginning of each day. (This processing of emergencies first in effect preempts normal service delivery.) Note that while patients can book services from any service provider, emergency cases are considered to be cared for by the “nearest” service provider.

Number of patients, service providers, services and their requirements are parameters of the model. We have created a python script that is capable to modify these parameters of the model based on the desired configuration. There are two controlled sources of randomness in the model: the rates of service purchase and the rate of emergencies are parameters of the model. Both are selected with normal distribution where parameters define the mean value.

We consider several options for the service providers to predict the number of emergencies in the future.

1. Baseline - There is a fixed number of resources reserved for emergency cases. That is, each day the prediction is a fixed value.
2. Providers Separately - Each service provider makes predictions based on their local knowledge. Here the service providers can look at their history of emergency cases and attempt to predict future emergencies.
3. Total Emergencies - The shared marketplace makes prediction based on the total number of emergencies. Here the marketplace considers all the emergency cases across all service providers and uses this for prediction.
4. Worst Scenario Among Providers - The shared marketplace makes separate predictions based on data from each service provider then selects the worst scenario and this is propagated to all service providers. Here the marketplace finds the worst emergency numbers from any service provider uses this to predict a worst case scenario for all service providers.

The prediction function is identical in all cases except Baseline. The prediction function compares the growth of emergencies over the last 3 days and chooses among the constant, linear and exponential scenarios. The only difference between the options is the input data (source) given to the prediction function.

To evaluate the predictions we consider several outcomes.

1. Cancelled - This is the number of services that had to be cancelled due to emergency patients. The idea here is to measure how often patients must be turned away due to service providers over-selling their capabilities. This number also include undelivered emergency services in case of provider facilities overflow by emergency patients (although no special penalty is imposed here since emergency services are prioritised and this would overflow regardless of prediction).
2. Sold - This is the number of services sold through the marketplace. This is to balance against a naive approach that could only provide emergency services and so have almost no cancelled services, but also deliver very few services (i.e. only emergency services).
3. Delivered - This is the number of services that were successfully delivered. This measures *normal* services delivered, in practice this adjusts for the number of cancelled services except for cases when a service provider is overflowed by emergency patients. Note that emergency services are not counted here since we are considering the shared marketplace delivery improvements.

The outcomes are evaluated with SMC engine of Uppaal SMC. Each service provider has counters of cancelled and delivered services, while the marketplace computes the sales. By running the simulation multiple times (100 in our experiments) Uppaal SMC can then report the expected values of outcomes.

5.1 Experiments

Our first experiment is a model of service providers that provide a simple set of services: a physician visit, several types of surgeries, a blood analysis and an x-ray

scan. In total we have 5 medical professional types: physician, surgeon, anesthesiologist, radiographer, and nurse, also 4 types of equipment: x-ray, laser, pacemaker, and analysis laboratory. We consider an exponential emergency growth. We consider three service providers that are hit by the exponential growth not simultaneously but in consequent days. At the peak almost 80% of service provider resources would be required by emergency patients. Only one service is unaffected since it does not require professionals and tools involved into the emergency services. In order to reduce the computation complexity we simplified the model by replacing a single standard patient with 6, i.e. each patient would book and consume 6 units of resource instead of 1. This allows to gradually reduce the number of processes while, as we believe, having only minor effect on model applicability. In the experiment we considered 900 standard patients transformed into 150 processes after simplification.

Prediction	Cancelled	Delivered	Sold
Baseline	3677.4	7328.64	11006
Providers separately	22.92	7230.18	7253.1
Total emergencies	66.18	7077.78	7143.96
Worst scenario among providers	14.4	6096.36	6110.76

Fig. 7: Results for exponential scenario

We run 100 simulations of 20 days of marketplace work. The mean values for considered outcomes are shown in Table 7. The baseline is selling 80% of service provider resources resulting in 11006 bookings by patients. As expected, since emergency services require more than 20% of providers resources starting from day 8, the baseline has lots of overbookings resulted in high number of cancelled services. Using means for prediction, more resources are reserved for emergencies when the growth is detected and, even if the sales are lower, the number of cancelled services decreases immensely. Notice that despite the decrease in sales the number of delivered services is close to the baseline for 2 out of 3 prediction options. The best result in cancelled services is achieved when considering worst case scenario among providers. This is compensated by considerably lower level of sales due to high reservation of resources. The prediction based on counting total emergencies does not perform well in this experiment: large growth of emergencies for one provider is compensated by the stable situation of others. Simulation for each prediction option takes 15 minutes.

In order to see if the model can work in a more realistic scenario we take the number of COVID-19 cases per Belgian province for the period 01.03.2020-13.04.2020 reported by Belgian Institute of Public Health¹. We assume that all patients used service providers of their municipality and we also assume that all patients added some load to the service providers. At each municipality we fixed the number of resources proportional to the population of the region such that

¹ <https://epistat.wiv-isp.be/covid/>

the COVID-19 patients would add a significant resource consumption. In addition we added other patients that are trying to receive standard services. In this experiment we have 11 service providers, 2 standard services plus an emergency service, and 200 standard patients (as before each books and consumes 6 units of resource). Simulation for each prediction option takes about 2 hours.

Prediction	Cancelled	Delivered	Sold
Baseline	645.53	49410.7	50027.44
Providers separately	307.35	47057.9	47336.12
Total emergencies	303.61	48465.9	48740.64
Worst scenario among providers	135.58	44429.1	44535.78

Fig. 8: Results for COVID-19 scenario

The results are shown in Table 8 and in Figs. 9 & 10.

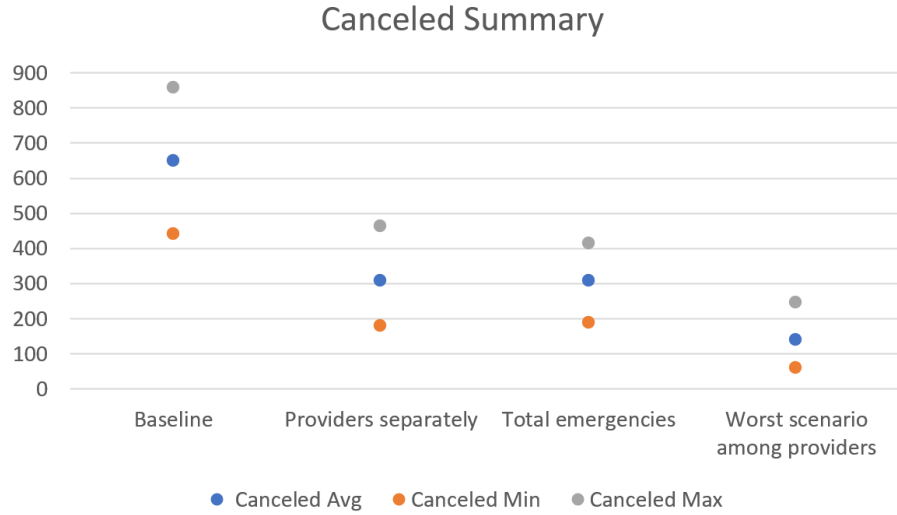


Fig. 9: Number of cancelled services or missed emergencies for COVID-19 scenario.

An overview of the cancelled results can be seen in Fig. 9. Clearly the number of cancelled medical services decreases with each choice of prediction. This is as expected, with the baseline having high cancelled numbers, localised and global knowledge marketplace prediction performing similarly (although the global knowledge prediction has tighter bounds), and worst case scenario having the lowest number of cancelled services.

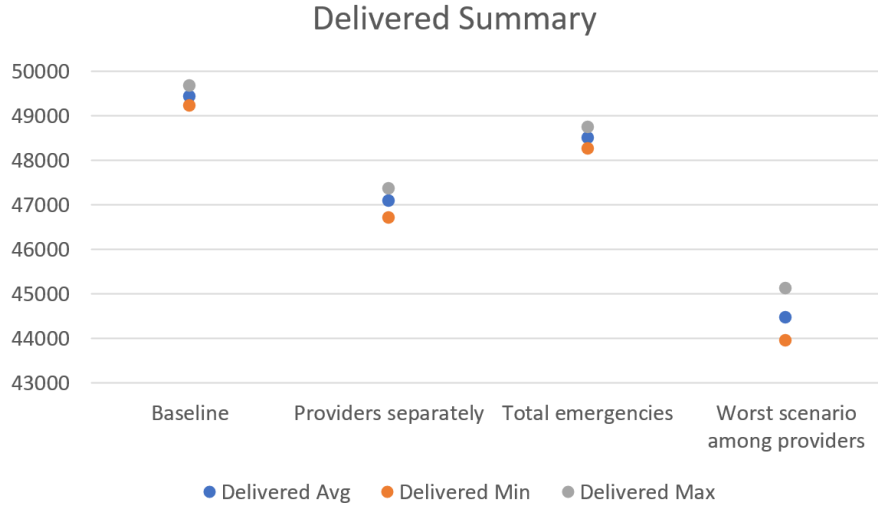


Fig. 10: Number of delivered services for COVID-19 scenario.

A graph of the delivered results can be seen in Fig. 10. As expected the baseline here performs the best. This is due to over-selling of services and then having a very high cancellation rate (as seen above). The prediction per service provider performs significantly worse than the global knowledge marketplace. This is most significant since they achieve approximately the same cancelled rate, and so the global prediction increases service delivery while also achieving a (very small) improvement in cancelled medical services. Finally, as expected the worst case scenario delivers the least medical services.

Overall these results show that using predictions can significantly reduce the number of cancelled services (more than 335 or 52% reduction for all prediction models). Also the global knowledge of the shared marketplace has a significant improvement on relative service delivery; reducing from the baseline by approximately 950 or < 2%, compared with approximately 2350 or < 5% less services delivered for local predictions. (That is, the global knowledge of the shared marketplace has an approximately 60% improvement in the reduction of delivered services.) The worst case scenario of course performs the worst in delivery, but was used here to indicate a (somewhat) reasonable bound for worst performance.

6 Conclusions

The Wablief project proposes to use a shared marketplace to improve medical service delivery. The shared marketplace provides benefits for all actors; hospitals, patients, and service providers. By formalising this model in Uppaal SMC it is possible to prove that desirable properties about the shared marketplace

can be proven. This in turn ensures that an implementation can meet these properties.

The shared marketplace also has access to greater information than individual service providers. We explore how this can be used to improve responses to events that span multiple service providers and impact their normal ability to offer and deliver services. Here we demonstrate the advantages of shared information from a shared marketplace by using this knowledge to improve prediction in a pandemic scenario. This is considered with two different models for the pandemic; simple exponential growth, and using real data from COVID-19 incidences in Belgium. Four different approaches to prediction are compared, demonstrating that the shared marketplace can improve both emergency and non-emergency service delivery.

Future Work. Future work on the Wabliet project will be to extend the Wabliet model with more actors to consider other entities such as financial and government services, and also blockchain-based records keeping. This will also require the development of more complex properties that address concerns related to GDPR regulations.

There are also opportunities to consider other ways that knowledge from the shared marketplace can be exploited to improve medical service delivery. Other kinds of larger scale responses or scenarios can be considered. Also the possibility to learn from patterns in one service provider or medical service, and apply the knowledge to other providers or medical services in a more fine-grained manner.

References

1. Uppaal. <http://www.uppaal.org/>
2. Alur, R., Dill, D.L.: A theory of timed automata. *Theoretical computer science* **126**(2), 183–235 (1994)
3. Baier, C., Katoen, J.P.: *Principles of Model Checking (Representation and Mind Series)*. The MIT Press (2008)
4. David, A., Larsen, K.G., Legay, A., Mikučionis, M., Poulsen, D.B.: Uppaal smc tutorial. *International Journal on Software Tools for Technology Transfer* **17**(4), 397–415 (2015)
5. David, A., Larsen, K.G., Legay, A., Mikučionis, M., Poulsen, D.B.: Uppaal SMC tutorial. *International Journal on Software Tools for Technology Transfer* **17**(4), 397–415 (jan 2015). <https://doi.org/10.1007/s10009-014-0361-y>
6. Given-Wilson, T., Baranov, E., Legay, A.: Building user trust of critical digital technologies. In: 2020 IEEE International Conference on Industrial Technology, ICIT 2020, Buenos Aires, Argentina, February 26-28, 2020. pp. 1199–1204. IEEE (2020). <https://doi.org/10.1109/ICIT45562.2020.9067154>, <https://doi.org/10.1109/ICIT45562.2020.9067154>
7. Hérault, T., Lasseigne, R., Magniette, F., Peyronnet, S.: Approximate probabilistic model checking. In: *Proceedings of the 5th International Conference on Verification, Model Checking, and Abstract Implementations, Lecture Notes in Computer Science*, vol. 2937, pp. 73–84. Springer Berlin Heidelberg (2004)

8. Legay, A., Delahaye, B., Bensalem, S.: Statistical model checking: An overview. In: International Conference on Runtime Verification. pp. 122–135. Springer (2010)
9. Sen, K., Viswanathan, M., Agha, G.: On statistical model checking of stochastic systems. In: Etessami, K., Rajamani, S.K. (eds.) 17th International Conference on Computer Aided Verification, Lecture Notes in Computer Science, vol. 3576, pp. 266–280. Springer Berlin Heidelberg (2005)