Chapter 8
Simulation and Machine Learning Based Real-Time Delay Prediction for Complex Queuing Systems

Najiya Fatma, Pranav Shankar Girish, and Varun Ramamohan

Abstract Real-time delay prediction involves providing entities arriving at a queue with an estimate of their wait time at the time of their arrival at the queue. This wait-time estimate is generated as a function of the state of the system at the time of the entity's arrival to the queue. In this chapter, we present a hybrid simulation and machine learning-based approach toward real-time delay prediction for complex queuing systems. The approach involves using a discrete-event simulation of the queuing system under consideration to generate system state data that is in turn used to train machine learning methods that generate real-time delay predictions. We provide a case study illustrating this approach that involves generating real-time delay predictions for end-stage renal disease patients registering on kidney transplantation waitlists.

Keywords Hybrid modeling · Discrete-event simulation · Machine learning · Kidney transplantation · Real-time delay prediction

8.1 Introduction

Overcrowding is one of the most important issues faced by the healthcare facilities in India and elsewhere [1]. For example, long queues have become a major concern at state-of-the-art tertiary care centers, with 10,000 patients visiting each day [2]. Limited medical resources relative to the demand and the highly unpredictable nature of the demand for care are a few of the many factors yielding long waits at healthcare

N. Fatma · P. S. Girish · V. Ramamohan (✉)
III-351, Department of Mechanical Engineering, Indian Institute of Technology Delhi, New Delhi 110016, India
e-mail: varunr@mech.iitd.ac.in

N. Fatma
e-mail: mez188287@mech.iitd.ac.in

P. S. Girish
e-mail: me1190823@mech.iitd.ac.in

M. Fakhimi and N. Mustafee (eds.), *Hybrid Modeling and Simulation*, Simulation Foundations, Methods and Applications, https://doi.org/10.1007/978-3-031-59999-6_8

facilities [3]. Empirical evidence suggests that delays not only lead to poor health outcomes for patients, but also cause unnecessary anxiety and inconvenience to patients and overburden healthcare providers [4, 5]. Delays are also expensive; for instance, among patients with the most acute conditions, a one-hour increase in wait time leads to an approximately 30% increase in medical expenses [6]. One of the most frequent discussions in healthcare operations research is to minimize service delays. Informing arriving patients about their expected delay—*at the time of their arrival to the queue*, as opposed to providing patients with the average wait-time estimate—is a relatively inexpensive technique of reducing wait-time uncertainty for the patient and for service-seeking entities in queueing systems in general. This process is called real-time delay prediction.

Studies on customer psychology in waiting situations reported that real-time prediction of waiting time not only improved patient satisfaction and service quality, but also helped in effective planning of medical resources for healthcare administrators [7]. Effective and accurate prediction tools to forecast demand, predict congestion level, queue-lengths, real-time delays, and lengths of stay have been developed to manage patient flow and improve patient service levels [8–11]. In this work, we present a method for generating real-time predictions of wait times for complex queueing systems for which analytical approaches are likely to be intractable. These typically include queueing systems whose queue disciplines are not among those commonly encountered, such as first-come first-served, last in first out, and so on. In such systems, data-driven approaches are often used. For example, queue log data recording the state of the system at or near the point in time when each arrival occurs is used to train statistical and/or machine learning (ML) models for predicting the delay. However, in many queueing systems, such queue log data is either not available or difficult to record. Our approach is suitable for such systems, and its development was motivated by our experience in predicting real-time delays for patients seeking admission to a neurosurgery ward in a large tertiary care hospital where such data was not available [12].

Our approach is hybrid: We use a combination of discrete-event simulation (DES) and ML to generate these predictions. This approach first involves developing a validated DES of the queueing system under consideration and then generating data capturing the state of the system at the point in time at which a service-seeking entity arrives in the system along with its delay prior to starting service. Such system state data can involve, for example, the number of entities already in the queue and the elapsed service time of the entity currently receiving service. This data is then used to train an ML model for generating real-time estimates of delay for each new arriving service-seeking entity.

In the realm of hybrid simulation (HS) and hybrid modeling (HM) literature, the above approach aligns with the HM paradigm. HS primarily centers on the integrated use of methods originating within the modeling and simulation domain, involving the simultaneous application of various simulation techniques to represent the system more effectively under examination. In contrast, HM is concerned with the integration of simulation methodologies (including DES, system dynamics, agent-based simulation, or HS) with other modeling and optimization techniques derived from

the broader fields of operations research and management sciences. In other words, HM serves as an extension of HS, enhancing its capabilities [13–16]. The healthcare industry is experiencing a growing trend of employing hybrid models, driven by their ability to effectively depict complex service systems.

To demonstrate our HM approach, we consider a case study in the Indian context involving real-time delay prediction for end-stage renal disease (ESRD) patients registering on a kidney transplantation waitlist. We first predict whether patients registering on the waitlist will receive a transplant or not. If they are predicted to receive a transplant, we then predict the wait time before the organ is allocated. This approach can be used by both patients as well as medical care providers (in case doctors deem it appropriate to not disclose this information to patients) in their decision-making regarding seeking care at the queueing system under consideration. For instance, if a patient is predicted to not receive an organ, they may immediately start exploring options for receiving a kidney from a living donor. For those predicted to receive an organ, they can discuss plans for continuing dialysis for the foreseeable future. For such patients, having an estimate of the actual wait time (the regression problem) will be useful. Finally, the wait times for many patients are in the order of months, and both types of predictions are likely to be particularly useful for such patients.

Overall, the HM real-time delay prediction approach that we propose is suitable in the context of complex queuing systems where the queue log data often lacks the granularity required to generate accurate real-time delay predictions. The patient waitlisting and kidney allocation system that we model is such a queueing system. Further, the proposed hybrid approach is advantageous in comparison to real-time simulation for generating real-time delay predictions, especially in terms of speed of generation of the delay prediction. The proposed approach requires a single function evaluation, whereas the real-time simulation may need to be executed multiple times to generate the average real-time delay prediction. Note that in both cases—the proposed hybrid approach as well as the real-time simulation case, the simulation driving the delay prediction would have to be reprogrammed if the queuing system configuration changes.

The remainder of the chapter is organized as follows. In Sect. 8.2, we provide a more detailed introduction to the concept of real-time delay prediction and discuss the related literature. In Sect. 8.3, we briefly describe the kidney allocation process and discuss the development of the simulation, including input parameter estimation. We later illustrate the use of the validated simulation model of the waitlist registration and kidney allocation process to generate real-time delay predictions for ESRD patients. We make concluding remarks in Sect. 8.4 wherein we summarize this work, describe its advantages and limitations, and discuss avenues of future research.

8.2 Background and Literature Review

8.2.1 *Real-Time Delay Prediction: Overview and Proposed Approach*

Real-time delay prediction involves providing each entity with an estimate of their expected wait time—at the point in time at which they arrive at the queueing system—until the start of the service [17]. These estimates may be generated by either: (a) analyzing the delay history data for previous arrivals to the queue; (b) utilizing the system state information to develop closed-form expressions for mathematically tractable queuing systems (where possible); (c) using queue log data where such data is available to train statistical/ML models; or (d) as proposed here, using validated simulation models of the queueing system in question to generate system state data for training statistical/ML delay prediction models. To illustrate the concept of real-time delay prediction, we take the example of the simple $M/G/1$ queuing system with a generally distributed service time. The real-time delay is estimated by the expression below [18].

$$P(T \leq t|x) = \frac{P(x \leq X \leq t+x)}{P(X \geq x)} \Rightarrow G(t|x) = \frac{G(t+x) - G(x)}{1 - G(x)}. \tag{8.1}$$

In Eq. (8.1), T is the random variable representing the remaining service time of the entity currently in service given the elapsed service time x; that is, it is the delay assuming no other entities are in the queue, and t is a realization of T. X is the random variable representing the service time itself and $G(x)$ represents its cumulative distribution function (cdf). Now the expected remaining service time given an elapsed service time of x can be found as the expected value of T, given by:

$$E[T] = \int_{S_T} tg(t|x)\mathrm{d}t. \tag{8.2}$$

In Eq. (8.2), S_T represents the support of T. Once $E[T]$, referred to henceforth as w for economy of notation, is estimated, then the real-time predicted delay for the arriving entity, denoted by d, can be found as follows:

$$d = w + L_q E[s]. \tag{8.3}$$

In Eq. (8.3), L_q represents the length of the queue at the time the delay prediction is generated, and $E[s]$ is the expected value of the service time random variable X. As an example, for uniformly distributed service times with parameters $U(a, b)$, w can be estimated as follows:

$$w = \int_0^{b-x} \left(\frac{1}{b-x}\right) t \,\mathrm{d}t = \frac{b-x}{2}. \tag{8.4}$$

Depending upon the functional form of $G(x)$ in Eq. (8.1), the computation of w may be straightforward or tedious. For example, computing w for the triangular distribution requires working with its piecewise cdf, and for the Gaussian distribution, one has to work with integrals of the Gaussian error function, which require numerical evaluation. Fatma and Ramamohan [8] propose an approximate real-time delay predictor that is agnostic to the specific service time distribution as long as it is symmetric and unimodal, while still using the distributional information of the service time. This predictor is given below in Eq. (8.5).

$$w = \begin{cases} G^{-1}(0.5) - x, & 0 \leq x < G^{-1}(0.5) \\ G^{-1}(0.75) - x, & G^{-1}(0.5) \leq x < G^{-1}(0.75) \\ \frac{G^{-1}(\text{ext}) - x}{2}, & G^{-1}(0.75) \leq x \leq G^{-1}(\text{ext}) \end{cases}. \tag{8.5}$$

In Eq. (8.5), G^{-1} represents the quantile function of the service time and $G^{-1}(\text{ext})$ represents an extreme right quantile. The logic underlying the development of the above predictor and the exact expressions of w for few symmetric and unimodal service time distributions can be understood from [8]. We also refer readers to Table 8.5 in [8] to learn about the wide variety of other queuing systems for which analytical expressions for real-time delays have been developed.

Developing analytical expressions for real-time delay prediction can become challenging for complex queueing systems where queue disciplines are complex (for example, if it is not FCFS or LIFO), or significant non-stationarity is present in multiple queueing aspects—for example, if balking and/or reneging behavior in addition to arrival/service processes are non-stationary. For such systems, queue log data, if available, may be used to train statistical/ML methods for predicting delays. However, in cases where queue log or system state data is not available or not captured in a manner suitable for training statistical/ML methods, the hybrid approach that we propose in this chapter may be used. This requires the development of a simulation of the queueing system in question, validating the simulation, and then using the validated simulation to generate system state data for each arriving entity at the time of its arrival. This system state data, along with the actual delay information of the service-seeking entities for which the data has been generated, is used to train statistical/ML methods for the purpose of real-time delay prediction. An example of such a queueing system can be found in Baldwa et al. [12], where the multi-class queueing system represented by the admission, surgery, and recovery stay processes at the neurosurgery ward in a large public tertiary care hospital uses an algorithm based on patient severity to determine admission to the neurosurgery ward. Real-time delay prediction for this queueing system was accomplished by first simulating the admission and the patient stay processes at the neurosurgery ward. Then, the validated simulation was used to generate data to train ML algorithms to predict whether the patient will be admitted or not as a function of the state of the simulation at the time the patient arrives seeking admission to the ward.

The key steps involved in generating real-time delay predictions using our proposed hybrid simulation and machine learning technique are given below.

Develop a DES of the queueing system under consideration.

Use the DES in steady state to record for each of, say, M entities (e.g., the kth entity) the following information:

The state of the simulation at the time the k th entity arrives in the system (denoted by S_{k}).

If there exists a prespecified threshold wait time T_{k} prior to which entity k must receive service before it exits the queue (i.e., a form of reneging, which is common for most healthcare service systems where the patient's condition may deteriorate), record whether the said entity receives service within T_{k} as a binary variable V_{k} ($V_{\mathrm{k}} = 1$ if service is received prior to T_{k} and 0 otherwise).

For entities receiving service prior to T_{k} (where applicable), record the wait time w_{k} before the start of their service.

For queueing systems where the reneging threshold T_{k} is not applicable, record w_{k} for every service-seeking entity k.

Construct training sets (S, V) and/or (S_{w}, w) (note that $S_{\mathrm{w}} \subseteq S$) as applicable using the data recorded for all M entities.

Train and validate ML methods f and f_{w} on (S, V) and (S_{w}, w), respectively.

For each new service-seeking entity $k(\text{new})$, record $S_{k(\text{new})}$ at the time of its arrival to the queueing system and predict $\hat{V}_{k(\text{new})}$ as $f\left(S_{k(\text{new})}\right)$. If $\hat{V}_{k(\text{new})} = 1$, then predict $w_{k(\text{new})}$ as $f_{\mathrm{w}}\left(S_{k(\text{new})}\right)$.

In the above procedure, it is assumed that if an entity does not receive treatment prior to T_{k}, they exit the queueing system (i.e., renege or leave the queue). In the context of the kidney transplantation system modeled in this study, this implies that the patient dies, or their condition deteriorates to the extent that they become ineligible for a transplant.

Recording the system state data S for each service-seeking entity precisely at the time of its arrival to the queueing system may be possible if a sufficiently comprehensive information technology infrastructure is available to capture the required data. For example, in the neurosurgery ward case mentioned above, which had a large number of servers (beds in the ward), key system state variables involved the duration of occupancy of each bed at the time a new patient arrived at the ward seeking admission. This information is likely already tracked by the billing system (using the time of admission for each patient currently occupying a bed) and hence can be leveraged in the deployment (if not development) of the above approach.

8.2.2 Literature Review

We now provide an overview of the literature around real-time delay prediction, beginning with a very brief discussion of the use of DES in modeling healthcare delivery. Subsequently, we briefly discuss HM literature.

DES is one of the most commonly used methods for modeling healthcare delivery operations across the world, and we refer readers to [19, 20] for a comprehensive discussion of the relevant literature. Specifically, since we use the DES of a kidney

transplantation system to illustrate our approach, we refer to [21–23] for examples of the application of simulation and operations' research methods to analyze and optimize different aspects of organ transplantation systems and allocation policies' systems in multiple countries.

Researchers investigated delays using queueing theory, game theory, and data-driven approaches at call centers, airports, construction sites, retail industries, healthcare facilities, and others [24–29]. Primarily, two types of approaches have been adopted: (a) analytical approaches grounded in queueing theory and (b) data-driven statistical learning approaches that are trained on queue log data [17]. With regard to queueing-theoretic approaches, most studies focused on developing system state and/or delay history-based predictors for queuing systems ranging from $M/G/1$ to $M(t)/\mathrm{GI}/s(t) + \mathrm{GI}$ systems. System state-based predictors estimated real-time delays using queue length, elapsed service time, number of servers, or the quantiles of the service time distribution. One of the earliest studies on the application of system state-based delay estimation was conducted by Whitt in 1999 [18], where customers were communicated information on expected delays in single and multi-server queues. Additionally, information about various other system parameters such as the arrival rate, the abandonment rate, and the number of servers were considered by Whitt in [30]. Nakibly studied ways to predict waiting times based on information about the system state upon arrival mainly for queueing models with priority [31]. Fatma and Ramamohan [8, 32] developed novel approximate system state-based delay predictors for simple queueing systems with symmetric and unimodal service time distributions and used the predictions for diverting patients in a healthcare facility network. Delay history information such as the delay of the last entity to receive service, wait time elapsed at the head of the line, etc., were discussed in [33–35]. Ibrahim and Whitt [36–38] highlighted the better performance of queue length-based system state delay prediction methods over the delay history-based estimators in simple and complex queueing systems.

The limitations of queueing-theoretic analyses such as assumptions that are often used to make the analysis mathematically tractable led to recent interest in data-driven methods such as ML algorithms and data mining techniques for complex queuing systems [26, 29, 39, 40]. ML-based predictors, which consist of classification and regression methods trained on queue log data, were discussed in Senderovich et al. and Thiongane et al., respectively [41, 42]. Ang et al. [43] and Arora et al. [29] combined process mining and queueing-theoretic results for predicting waiting times in the emergency departments (EDs) of the healthcare facilities. Baldwa et al. [12] proposed a hybrid simulation and machine learning method for real-time delay prediction where adequate queue log data for training a predictor is not maintained. Further, robust ML-based prediction models were developed for predicting real-time lengths of stay of patients, which is a metric of quality, efficiency, and hospital performance [44, 45]. The effectiveness of predictors was quantified using mean absolute deviation, mean absolute percentage error score, and other metrics via computer simulation models.

With respect to the HM literature, Harper and Mustafee [46] demonstrated the applicability of an HM approach involving DES and time-series forecasting in a

real-life setting to support short-term decision-making in an urgent care network. Similarly, other studies, such as those by Ordu et al. [47] and He et al. [48], developed hybrid frameworks that incorporated DES, system dynamics, and optimisation techniques like integer programming to address both the operational (short-term) and strategic (long-term) objectives of healthcare facilities. However, for delay prediction, except for the study by Baldwa and colleagues [12], there has been limited exploration of hybrid methods.

The majority of the empirical work on delay prediction involved using historical data regarding the queuing system under consideration and training statistical/ML predictors using this data. However, in situations where such data is not available, or sufficient information regarding system state data required for training an accurate prediction is not maintained in the queuing system logs, the data may be generated from a DES model of the system. From our review of the literature, only one previous study [12] has incorporated DES with ML for real-time delay prediction. This is summarized in Table 8.1. An example of such a system is the kidney transplantation system that we consider as a case study to illustrate our proposed approach. In this chapter, we build upon the work by Baldwa et al. [12], which to our knowledge is the only study that uses a DES of the queueing system under consideration for generating the system state data for training data-driven predictors. However, unlike their approach, we do not use predetermined reneging thresholds for service-seeking entities (patients); we instead use patient-specific 'personalised' reneging thresholds that are based on patient characteristics. Our approach may be used by healthcare providers to help advise ESRD patients on the best course of action from the standpoint of obtaining a kidney transplant.

Finally, our approach resembles metamodeling methods to some extent. A metamodel f has an explicit form, deterministic output, and once fitted, is computationally inexpensive to evaluate as they serve as proxies for evaluation, thereby replacing the need for conducting computationally expensive and stochastic simulation runs [49]. A relevant study by Fatma et al. [50] explored the use of stochastic metamodels in developing primary healthcare delivery network systems, resulting in reduced execution times while maintaining comparable results. In this work, similar to metamodeling, we use DES to generate a dataset for training classifiers and regressors. However, in the case of metamodeling, the system simulation (e.g., a DES) is executed multiple times with different sets of input parameters, while we generate a dataset only once with a single set of input parameters for training the delay predictors.

8.3 Case Study

We discuss a case study through which we illustrate our HM approach for real-time delay prediction. The case study involves predicting whether a patient registering on the kidney transplant waitlist—at the time of registration on the waitlist—will receive a kidney transplant or not, and if predicted to receive a transplant, then their wait time to allocation of a kidney is also estimated.

Table 8.1 Studies utilizing machine learning algorithms and simulation methods for real-time delay prediction

Study	Problem description	Methodology	Predictor variables
Baldwa et al. [12]	Prediction of whether a patient seeking admission receives admission within a prespecified duration	Simulation, ML (ensemble bagged trees, gradient boosted trees, neural network, decision tree)	Patient type, waitlist-related features and operational system state features such as number of empty beds
Balakrishna et al. [51]	Estimation of average taxi-out times at airport	Stochastic dynamic programming with reinforcement learning	Features describing the airport and runway state
Arora et al. [29]	Estimation of probability distribution of individual patient wait times	Quantile regression using decision trees	Calendar effects, demographics, staff count, ED workload, severity of patient condition
Ang et al. [43]	Wait-time prediction	Data mining, queuing theory (Q-Lasso technique)	Patient visit data, mode of arrival, triage level
Senderovich et al. [39]	Delay prediction for single-class setting (homogeneous customers) and multi-class setting (different class of customers)	Queue mining, regression-based predictors	Time of event occurrence, instance of service process, service transition, customer class
Arik et al. [40]	Prediction of the time to meet with the first provider at hospital	Supervised learning (congestion graphs of two types—heavy traffic approximations of congested systems and Markovian state representation of queues)	Clinical state of patients and congestion-related features
Chocron et al. [52]	Prediction of the wait time for service at the time of arrival	ML models, queueing theory	Arrival-related features, service-related features, queue-related features, short-term history-related features

8.3.1 Kidney Transplantation System: Problem Introduction

Large urban public tertiary care hospitals in India typically face significantly more demand than their available capacity. Kidney transplantation is the most effective long-term treatment option for ESRD patients undergoing maintenance dialysis. The substantial shortage of donated kidneys in India causes an increasingly long waitlist

of ESRD patients awaiting a transplant. According to the Indian Ministry of Health, the number of Indian ESRD patients who need kidney transplants ranges between 200,000 and 300,000, with only 6000 donors available [53]. A kidney patient who gets on the state government's waiting list typically waits for at least four years to get a cadaveric donor transplant, thereby aggravating uncertainties among patients regarding whether they will receive a transplant before their health deteriorates to a critical level [54]. A first step toward alleviating the uncertainty is to provide ESRD patients and/or their medical care providers with information regarding whether they will receive a transplant or not—at the time of their registration on the kidney transplantation waitlist—and if the patient is predicted to receive a transplant, the wait time of the candidate before receiving the transplant. This will help patients and/or their medical care providers make an informed decision about whether they should wait or seek treatment elsewhere (e.g., seek living donors). We list the specific objectives associated with the case study below.

1. Development of a DES of the patient arrival and registration, organ arrival and organ allocation processes.
2. Classification of waitlisted patients to predict whether they will receive a transplant within a 'personalised' patient characteristic based duration.
3. Regression to estimate the time to allocation of an organ for patients predicted to receive a transplant.

We use publicly available domestic data based on real-world reports from kidney transplantation organizations in Indian states, namely Rajasthan, Kerala, and Tamil Nadu, for developing and parameterizing the DES of the kidney transplantation system in the South Indian state of Kerala. We now describe the development of this DES, beginning with a description of the cadaveric donor kidney allocation process. We note here that considering kidney transplantation from living donors is beyond the scope of this work.

8.3.2 Patient Registration, Organ Arrival, and Organ Allocation Process Simulation Development

The allocation of cadaveric kidneys to transplant candidates is a complex process determined by a variety of organ and patient characteristics, including time spent on the waitlist. As per kidney allocation guidelines published by the Indian government's National Organ and Tissue Transplantation Organization (NOTTO) [55], ESRD patients eligible for registering on the waitlist must be aged less than 75 years at the time of registration, must have undergone regular maintenance dialysis for at least three months, and must be registered in a single approved transplantation center. Upon registration in a transplantation center, the patient is assigned a kidney allocation priority (KAP) score after registering in the respective state and district waitlists that determine the position of the patient on the transplantation waitlist. The KAP score is computed based on a scoring algorithm provided in NOTTO's

kidney allocation guidelines [55]. If a cadaveric kidney is retrieved in a government hospital, then patients registered in the government transplant centers within the state are given higher priority for allocation and patients registered in private hospitals are considered for allocation only if a suitable recipient is not found on the government hospital waitlist. The same recipient selection process is followed if the kidney is retrieved from the deceased donor in a private hospital, but in reverse order. Therefore, whether the patient is registered with a government or a private hospital affects their probability of transplant.

Further, patients registered in transplant centers within the district where the organ was retrieved are given higher priority. In other words, organs are first considered for allocation to patients registered in the same district where the organ is retrieved, and patients registered in other districts in the state are considered for allocation only if a suitable recipient is not found in the district of retrieval.

In the event that the kidney is retrieved from a donor aged less than 18 years, then patients aged less than 18 years are first considered for allocation. Finally, for each 'subwaitlist' (i.e., district and then state waitlist), donor/recipient matching is done on the basis of the blood groups of the donor and the patients. A blood group O (universal donor) kidney is first matched to a recipient with group O, then to the other compatible blood groups—first, it is allocated to group A, then to group B, and finally to a blood group AB (universal recipient) patient. Group A or B kidneys are allocated to patients with the same blood groups; else it is allocated to a group AB patient. An AB group kidney is only allocated to an AB patient. More details regarding the kidney allocation process, including an algorithmic representation of the above process, can be found in Shoaib et al. [56].

The advancement of the DES of the kidney allocation process is dependent on three principal events: patient arrival, removal of patients due to death, and organ arrival. Organ arrival drives removal of patients via organ allocation and transplantation. The mechanisms that determine the removal of patients are as follows: either the patient receives a transplant, or the patient dies, implying that we do not consider balking in our model. We represent districts in each state by their district headquarters, and hence, travel times between districts (for calculating organ transport times) are also calculated between the district headquarters. We now describe the estimation of two primary types of model parameters: (a) those related to patients and (b) those related to organs.

8.3.2.1 Patient-Related Parameters

The patient's position on the waitlist (district as well as state, government as well as private 'subwaitlists') is determined by the KAP score. Thus, a key set of parameters that need to be estimated with respect to patient characteristics is those that constitute the KAP score. These parameters include the following:

1. Time spent on dialysis.

2. Whether the patient has had a previous immunological graft failure, and if so, the number of such failures.
3. Age of the recipient.
4. Patient with all failed arteriovenous (AV) fistula sites.
5. Patient with failed AV graft after all failed AV fistula sites.
6. Panel reactive antibody level.
7. Whether the patient under consideration has previously donated a kidney or not.

The time spent on dialysis, which is a key driver of the KAP score, is estimated from the data available for this parameter from the waitlist data for the state of Rajasthan, because similar data was not available on the waitlist for Kerala. Because this is a clinical parameter, its distribution is assumed to not change substantially across states in India. The exponential distribution was found to fit the time on dialysis data best under a χ-squared goodness-of-fit test, with a p-value of 0.581. Parameter 7 was not considered in our analysis since it was highly unlikely to encounter a patient with this characteristic in the kidney transplantation process, based on discussions with clinicians involved in organ transplantation. Other parameters—i.e., parameters 2 through 6—were estimated from the clinical literature, and the sources, along with the parameter estimates, are given in Table 8.1. The patient interarrival time was estimated from the patient waitlist data available on the state organ transplantation authority website (Kerala Network for Organ Sharing (KNOS), [57]). The interarrival time was found to follow the exponential distribution with a mean of 1.382 days, which was estimated by applying the χ-squared goodness-of-fit test on the KNOS waitlist data, using information regarding the date of patient registration. Patient blood group data was estimated from waitlist data available on the neighboring Tamil Nadu state organ transplantation authority website (TNOS, the Tamil Nadu Network for Organ Sharing) [58]. Indian census data was used to assign the district in which a newly arriving patient was registered, and the transplant center of registration was also assigned based on the transplant hospital set available in the KNOS dataset.

A critical patient parameter in this context is the patient removal time due to death, occurring due to unavailability of a cadaveric kidney. We used survival data of ESRD patients from the clinical literature [56] and the KAP score computation process to estimate this parameter. This computation process is described below.

Algorithm: Computation of patient removal time due to death.

- Input: KAP score data of a waitlisted patient
- Output: Removal time of the patient under consideration
- Generate a large sample of KAP scores by generating multiple random realizations of the KAP score components and combining them according to the KAP score computation algorithm given in the NOTTO kidney allocation guidelines.
- Find the distribution that best fits this sample. This was determined to be a beta distribution with $\alpha = 0.89$; $\beta = 33.99$ (best fit out of alternatives).
- For each new patient registering on the waitlist, using the distribution of the KAP score estimated using steps 1 and 2, do the following:

Compute the patient's KAP score based on their randomly assigned KAP score component values.
Find the percentile of the KAP score for a given patient from its distribution estimated in Steps 1 and 2. Let this percentile be x.
A patient in the x th percentile of KAP scores is likely to be in the $(100 - x)$th percentile of removal times. Thus, the patient's removal time percentile $= 100 - x$.

- Estimate the mean removal time μ_{rt} of the patient by determining the $(100 - x)$th quantile of the removal time distribution, which we assume to a beta distribution with limits $a = 3$ months and $b = 67$ months and mean $= 40.31$ months, yielding $\alpha = 4.38$ and $\beta = 3.51$ for this distribution. The removal time distribution was estimated based on the mean and standard deviation of the mean survival time of patients on hemodialysis as reported in Lakshminarayana et al. [59] (40.31 months and 26.69 months, respectively).
- Using this estimate of the mean removal time, define another beta distribution with limits $a = 0.67 \times \mu_{\text{rt}}$ and $b = 1.33 \times \mu_{\text{rt}}$. α and β for this beta distribution were estimated via the beta-PERT three-point estimation procedure. This was done in order to avoid making the removal time a deterministic function of the KAP score.
- The assigned removal time value for the patient is a random sample from the beta distribution for the removal time estimated in Step 5.

8.3.2.2 Organ-Related Parameters

All organ-related parameters were estimated using data pertaining to kidney transplantation alone. Key organ-related parameters involve the interarrival time of kidneys, the district in which the organ originates, the deceased donor's blood group, and the number of kidneys retrieved from an organ (i.e., 1 or 2). With regard to the organ interarrival time, a parameter critical to the analysis, precise data regarding the dates of arrivals of organs were not publicly available on organ-sharing websites. Hence, we estimated the parameters of the interarrival times of the kidneys from deceased donors using the published annual aggregate organ donation data after assuming it to be exponentially distributed. According to the aggregate organ donation data published on the KNOS website [57], the number of organs donated in the years 2016, 2017, and 2018 were 113, 34, and 14, respectively, and hence, the mean interarrival time in days was estimated as 365 divided by the average of the number of organs arriving in those three years (this average amounts to approximately 33 kidneys being donated every year). Thus, the estimate of the mean interarrival time, assuming an exponential distribution, was 11.17 days per organ. We estimated the other organ-related parameters, such as the donor blood group and age, which are required to determine the kidney allocation, according to the proportions of various blood groups and age ranges in the population of the entire state. We estimated the probabilities of retrieving a kidney in a public or private hospital in a given district

based on the proportion of each type of hospital in each district. We list all the patient-related and the organ-related parameters, along with their distributions and estimates and corresponding sources in Table 8.2.

Table 8.2 Patient and organ-related parameters

Parameter	Distribution	Estimate	Source
Patient-related parameters			
Patient interarrival time	Exponential	Mean = 1.382 days	[57]
District origin (14 districts in the state of Kerala)	Discrete	$P(0) = 0.128$, $P(1) = 0.096$, $P(2) = 0.023$, $P(3) = 0.041$, $P(4) = 0.018$, $P(5) = 0.100$, $P(6) = 0.055$, $P(7) = 0.064$, $P(8) = 0.091$, $P(9) = 0.050$, $P(10) = 0.055$, $P(11) = 0.164$, $P(12) = 0.091$, $P(13) = 0.013$	[57]
Age	Gaussian	$\mu = 49.74$; $\sigma = 7.423$	[57]
Blood group	Discrete	$O = 0.458$, $A = 0.238$, $B = 0.224$, $AB = 0.079$	[58]
Time on dialysis	Exponential	Mean = 260.3 days	[57]
Removal time	Beta	$\mu = 40.31$; $\sigma = 26.69$ Random sampling (beta) $\alpha = 0.66 * \mu$; $\beta = 1.33 * \mu$	[59]
PRA level	Discrete	P(PRA level = 0) = 0.65, P(1–20) = 0.05; P(21–79) = 0.136, P(80–100) = 0.158	[60]
Probability of a previous immunological graft failure within 3 months of transplant With failed all AV fistula sites With failed AV Graft after failed AVF sites	Discrete Discrete Discrete	P(yes) = 0.020, P(no) = 0.980 P(yes) = 0.052, P(no) = 0.948 P(yes) = 0.031, P(no) = 0.968	[61] [62] [62]
Organ-related parameters			
Donor interarrival time	Exponential	Mean = 11.17 days/organ	[63]
Donor blood group	Discrete	P(AB) = 0.069, P(A) = 0.192, P(B) = 0.254, P(O) = 0.485	[58]
District in which organ originates	Discrete	$P(0) = 0.067$, $P(1) = 0.098$, $P(2) = 0.033$, $P(3) = 0.075$, $P(4) = 0.039$, $P(5) = 0.078$, $P(6) = 0.059$, $P(7) = 0.092$, $P(8) = 0.123$, $P(9) = 0.084$, $P(10) = 0.035$, $P(11) = 0.099$, $P(12) = 0.093$, $P(13) = 0.024$	[64]
Number of kidneys retrieved from an organ	Discrete	$P(1) = 0.777$; $P(2) = 0.223$	[63]

μ : mean; σ : standard deviation; PRA: panel reactive antibody, a screening test to identify the immunological sensitization of a transplant recipient for estimating likelihood of finding a compatible donor.

8.3.3 Simulation Model Outcomes

Using the parameters reported in Table 8.1 and the kidney allocation process described earlier, we programmed the simulation on the *Python* computing platform. We ran the simulation on a workstation with an Intel *i*7 10th generation processor system with 16 gigabytes of memory. We used a warm-up period of 12 years of simulation time before collecting the results over a period of 18 years. We performed 20 replications for collecting and reporting the results. The length of the warm-up period was determined by observing when the average simulation outcomes became stable. A data collection period spanning 18 years was selected as it provided enough time to calculate all the outcomes to the necessary precision. The decision to use 20 replications was influenced by the observation that the variances of the outcomes changed minimally when the number of replications was increased beyond 15. These choices were also influenced by the availability of computational resources. The key outputs collected from the simulation include (for all patients, by blood group and by the type of hospital (government or private) in which the patient is registered): (a) the probability of receiving a kidney transplant; (b) the average wait time to allocation for those who received an organ; (c) the average number of deaths; and (d) the average number of allocated organs.

The probability of receiving a transplant, which is possibly the most critical output from a patient and provider standpoint, was estimated as follows. For a set of patients who register in a given year (the registration year), we record the proportion of those patients who receive a transplant over the next year, the second year after the registration year, and so on for a period of five years. We then average these probabilities for patients registering every year after the warm-up period to obtain the within replication estimate of the average probability of receiving a transplant within 1 year, 2 years, and so on up to 5 years. The across replication average values of these probabilities are then calculated by averaging the within replication average probability estimates. These probabilities are depicted in Fig. 8.1a, b.

From Fig. 8.1a, the 5-year probability of receiving a transplant, which is negligibly different from the overall probability of receiving a transplant (before death from ESRD), is approximately 21%. It is also evident that patients with blood group AB and blood group B are the most likely to receive a transplant, with 5-year probabilities exceeding 25% (nearly 40% for blood group AB). Patients with blood group AB have the highest transplant probabilities because they are universal recipients. Patients with blood group O have the lowest probability of transplant likely because of the large volume of patients on the waitlist compared to the lower number of organs of the blood group being donated. This low likelihood reflects the disparity between the organ (including kidneys) donation rate and the number of patients needing

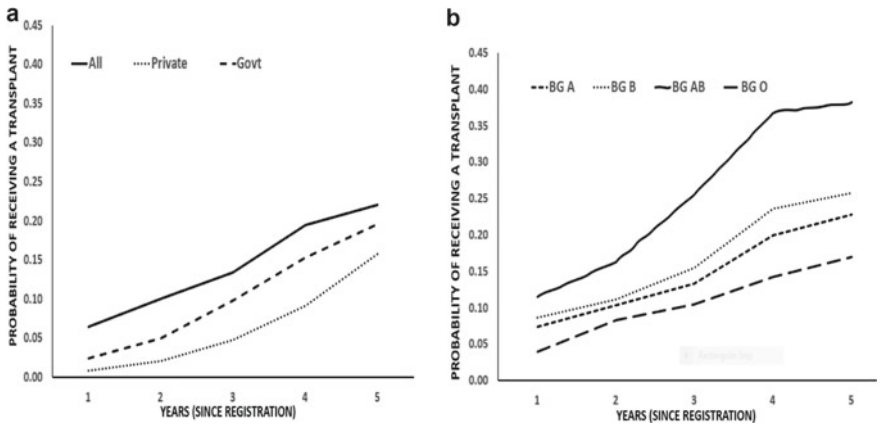

Fig. 8.1 **a** Year-wise probabilities of receiving a transplant for all patients and by the type of hospital where they are registered. **b** Year-wise probabilities of receiving a transplant by patient blood group

transplants. Further, patients registered in a private hospital are significantly less likely to receive a transplant than those registered in a government hospital. This reflects the interaction between the number of government transplant centers versus the number of private transplant centers and the number of government organ retrieval centers versus the number of private retrieval centers.

We report the other outcomes collected from the simulation model in Table 8.3. Average organ transport time is defined as the average time required to transport an organ from the organ retrieval location to a transplant center where the recipient is registered. The average time to transplant was calculated only for patients who received a transplant during the post warm-up (steady-state simulation) period. We calculated the number of deaths by counting those removed from the waitlist without receiving a transplant.

Based on the results in Table 8.3, we observe significant variations in the allocation based on the patient's blood group and the hospital type. This supports the outcomes in Fig. 8.1b. For example, patients with blood group O receive the highest number of allocations compared to those with other blood groups. However, because the number of patients with the blood group O is the highest (it is the most common blood group in India), as evident from the highest probability of a patient being of blood group O, the probability of receiving a transplant for patients with blood group O is the lowest.

We now discuss validation of the model outcomes prior to describing the process of generating real-time delay predictions from the DES. Validation of the DES of the kidney transplantation system in Kerala is challenging due to the lack of data regarding waitlist and transplantation outcomes for the state and in general for the Indian transplantation system. For example, in our knowledge, no data suitable for validating simulation results for outcomes such as time to allocation or the probability

Table 8.3 Simulation outcomes for Kerala

Outcomes (year)	Average (95% CI)
Average number of deaths per year	203.81 (1.57)
Average organs allocated per year	58.31 (1.14)
Average wait to allocation (hours)	19,175.10 (215.43)
Average time to transportation (minutes)	200.73 (4.21)
Average number of allocations in government hospitals	24.45 (0.63)
Average number of allocations in private hospitals	33.68 (0.93)
Average number of allocations to blood group A patients	14.34 (0.46)
Blood group AB allocations	8.17 (0.34)
Blood group B allocations	14.91 (0.49)
Blood group O allocations	20.72 (0.61)
Average number of unallocated organs	0.35 (0.09)

of receiving an organ is available. Since kidney transplantation outcomes data was not available for validation, we decided to validate the patient and organ arrival outcomes against a portion of the publicly available patient arrival data that we reserved for this purpose. We recall here that publicly available waitlist data was available for the years 2016–2019 and that we used data from 2016 to 2018 for calculating patient arrival rates. A similar approach was followed for organ and donor arrivals as well; however, we must recall that organ arrival data was much more limited in comparison to patient arrival data. In fact, for 2016–2018, we only had access to three values; that is, the number of organs donated and donors arriving in each year. Hence, we validated our patient arrival and organ outcomes—i.e., the number of patients arriving in one year against the actual values for 2019. We provide the results of this validation exercise in Table 8.4.

From the results from the above validation exercise, we can see that the patient arrival process is reasonably well represented by our DES. This is particularly the case when patients from all blood groups are considered together, and even when breaking

Table 8.4 Simulation validation outcomes

Parameters	Actual [57]	Simulation (UL, LL)
Number of patients registered in 2019	264	261.61 (265.90, 257.32)
Patient in blood group A	82	61.56 (63.71, 59.40)
Patient in blood group AB	13	21.22 (22.37, 20.08)
Patient in blood group B	51	59.43 (61.29, 57.56)
Patient in blood group O	118	119.41 (122.60, 116.22)
Organ arrived in 2019	32	57.61 (60.45, 54.77)
Donor arrived in 2019 [63]	19	32.70 (34.32, 31.09)

UL is upper limit, *LL* is lower limit

Table 8.5 Input features for the machine learning models

Feature type	Features
Continuous	Clinical: age, KAP score, PRA level, time on dialysis, Waitlist related: position on waitlist, patients above this patient, A patients above, B patients above, O patients above, AB patients above, total patients on the waitlist, total A patients, total B patients, total O patients, total AB patients
Categorical	Operational: district name, hospital name, hospital type Clinical: blood group, PRA type, AVG, AVF, PIGF

AVG arteriovenous grafts, *AVF* arteriovenous fistula, *PIGF* placental growth factor

down arrivals by blood group, we see that for blood groups with larger numbers of patients, such as O and B, the simulation outcomes and validation outcomes are reasonably close. Note that while we report the confidence intervals for the simulation outcomes, formal statistical inference based on these CIs and the actual simulation outcomes may not be advisable because only a single value of the actual number of organs is available for each year for each blood group. In other words, the actual value is also a realization of a random variable (the number of organs arriving in a year), and given the small sample size (3), sufficient data is not available to conduct formal inference—for example, a two-sample nonparametric test for equality of means.

We also perform a comparison between the probabilities of allocation from the simulation versus an approximate value of this outcome calculated from the organ and patient arrival data. The average probability of receiving a transplant calculated across years 1–5 from the simulation data is approximately 14% (ranging from 7 to 21% from 1 to 5 years and taking their average), and the value of this outcome from the validation data is approximately 12.2% (obtained by dividing the yearly organ arrivals with the patient arrivals). We computed the average probability of transplant in this manner because while the information regarding the average number of patients registering is known, it was unclear when the patients who registered in these three years (2016, 2014, and 2013) will undergo the transplant. Therefore, we took the average of the yearly probabilities to make a comparison between the simulated data and the actual data. This indicates that our DES of the kidney transplantation system appears to approximate the actual system to an acceptable level.

Overall, it is clear that the probability of receiving a transplant, even at 5 years on the waitlist, is low. However, it is also clear that there is significant variation based on patient and operational characteristics such as blood group and the type of transplant hospital where the patient is registered. This motivated us to develop a classification model that will predict whether a patient on the waitlist will receive a transplant at the time of registration on the waitlist—i.e., real-time delay prediction of transplant registration outcomes. We discuss this now.

8.3.4 Real-Time Delay Prediction for Waitlisted ESRD Patients: Classification

As described in previous sections, we now use the DES of the ESRD patient registration, waitlisting, and organ allocation process to generate the dataset required to train ML-based methods to predict whether a patient will receive an organ at the time of their registration on the waitlist. At the point in time at which a patient registers on the waitlist, we record three types of features for this patient: clinical, operational, and waitlist-related features. Clinical features primarily include those that determine the patient's KAP score and their blood group, and operational features include those features such as the district of registration, the hospital type, the hospital name, and so on. Waitlist-related features include the total number of patients above the current patient on the waitlist and the numbers of patients of different blood groups that are above the current patient on the waitlist. The label for the classification exercise was whether the patient received a transplant or not before they were removed from the waitlist due to death. Note that data was recorded only for patients who were allocated an organ or those who were removed due to death. There could be patients still in the model with neither of these outcomes at the end of the model time horizon, but the data for such patients are not recorded. For those who did receive an organ, we recorded the time to allocation from their time of registration. This formed the label for the regression exercise. The feature set consisted of both continuous as well as categorical features, and we list them in Table 8.5.

The training dataset thus consisted of a total of 23 features representing the patient characteristics and the queueing system state at the time of registration of the patient and the classification/regression labels. We split the input data into training and test sets with a 75/25 split ratio. The data was scaled—after it was split to prevent data leakage—using the MinMaxScaler function of the *scikit-learn* ML package in the Python programming platform. The input dataset consisted of 929 patients who received an organ out of a total of 3945 patients, which indicated a dataset imbalance. Hence, we used the Synthetic Minority Oversample Technique (SMOTE) to balance the dataset so that it does not negatively impact the accuracies of the classification models [65]. SMOTE is an oversampling technique that allowed us to generate synthetic samples for the Label 1 class (minority class, those who received an organ) until the number of samples became equal to the majority class. Note that SMOTE was applied only on the training dataset and the validation (test) dataset was left untouched by the dataset balancing technique.

We trained several ML methods for the classification exercise such as support vector machines, decision tree methods such as bagging and random forest classifiers, and artificial neural networks to find the method that performed best on the dataset. Before training the model, we optimized the hyperparameters of each of these classifiers via the GridSearch hyperparameter tuning method. For example, the gradient boosting classifier implementation contained 100 boosting stages with a learning rate of 0.1, while the bagging classifier implementation contained 1000

estimator trees. We also trained the artificial neural network with two hidden layers using the *adam* optimizer with a batch size of 10 [66].

To measure the performance of the classification models, we used the Receiver Operator Characteristics-Area Under Curve (ROC-AUC) and F1 performance measures. The ROC is a probability curve that plots the true-positive rate against the false-positive rate at various threshold values. The AUC score, which is used as a numerical characterization of the ROC curve, is the measure of the ability of the classifier to distinguish between classes. We compute the $F1$ score instead of using the classifier accuracy alone as it calibrates the trade-off between sensitivity and specificity at the best-chosen threshold. The $F1$ score is the harmonic mean of precision and recall. Precision is the number of correct positive predictions relative to total positive predictions, while recall is the ratio of the number of correct positive predictions relative to the total number of actual positives. The $F1$ score provides a measure of both the Type I (false-positive) and Type II (false-negative) errors in the model. We provide the classification results in Table 8.6, where the mean and the standard deviation of the AUC score of the model along with the precision, recall, and $F1$ scores for both cases where a transplant is received (target $= 1$) and a transplant is not received (target $= 0$) are listed. We also show the ROC curve for the gradient boosted trees classifier in Fig. 8.2.

The mean and the standard deviation of the performance measures were generated by training and validating each model over ten random permutations of the dataset. From these results, we see that the classifiers, especially the decision tree ensembles (bagging and random forest) and gradient boosting techniques, are performing well in classifying the data. We achieve over 80% precision for patients receiving a transplant

Table 8.6 Classification results for status of transplant

Mean	LR	SVM	Bagging classifier	Random forest classifier	ANN	Gradient boosted trees
AUC score	0.91 (0.010)	0.91 (0.010)	0.90 (0.009)	0.90 (0.008)	0.90 (0.018)	0.91 (0.011)
Precision (target = 0)	0.97 (0.007)	0.97 (0.006)	0.96 (0.007)	0.96 (0.005)	0.97 (0.018)	0.96 (0.007)
Recall (target = 0)	0.91 (0.013)	0.91 (0.012)	0.93 (0.011)	0.94 (0.007)	0.92 (0.031)	0.94 (0.011)
F1 score (target = 0)	0.94 (0.007)	0.94 (0.007)	0.95 (0.006)	0.95 (0.005)	0.94 (0.011)	0.95 (0.007)
Precision (target = 1)	0.76 (0.030)	0.76 (0.0280)	0.80 (0.028)	0.82 (0.021)	0.77 (0.063)	0.81 (0.030)
Recall (target = 1)	0.91 (0.019)	0.91 (0.015)	0.87 (0.016)	0.87 (0.012)	0.89 (0.057)	0.87 (0.017)
F1 score (target = 1)	0.83 (0.018)	0.83 (0.018)	0.83 (0.016)	0.84 (0.013)	0.82 (0.021)	0.84 (0.019)

LR logistic regression, *SVM* support vector machine, *ANN* artificial neural network

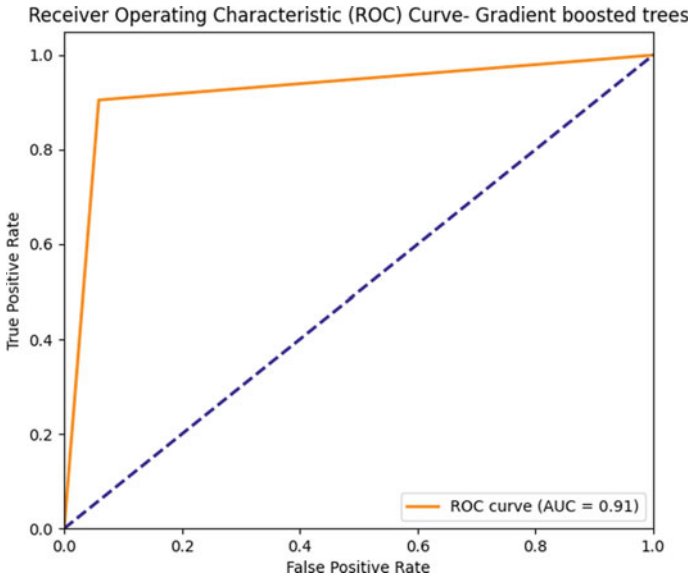

Fig. 8.2 ROC curve for gradient boosted tree classifier

(varied between 76 and 82% for different classifiers) and around 90% recall (varied between 87 and 91% for different classifiers), implying that around 90% of those who receive a transplant are identified correctly.

After generating the above predictions, we performed additional computational experiments around generating similar predictions for: (a) whether patients receive transplants within 2 years or 5 years (referred to subsequently as the 2-year and 5-year analyses) and (b) individual patient blood group-based classification. We also performed feature selection analyses by estimating the variance inflation factor (VIF) of all features and by identifying features with statistically significant association with the classification label via logistic regression models trained on the entire dataset. We estimated the accuracy metrics for successfully receiving transplants in 5 years and 2 years for different blood groups.

The results for the 5-year organ transplant status classification were similar in comparison to the overall organ transplant status results, reflecting a low chance of receiving a transplant in time (before removal) if not received in 5 years. We observed that the precision and recall values for the 2-year analysis crossed 90%, while the AUC score exceeded 95% for patients receiving an organ. Note that the classification label for these analyses was determined by whether a patient received an organ or not within the period of interest. With regard to the classification exercises for datasets with a single blood group, we observed that the AUC score of the blood group AB was the highest, but with the lowest precision, while blood group O had the lowest AUC score with greater precision. This can be attributed to the difference in organ

and patient arrival rates for the blood groups, which affects the size of the training dataset for these blood groups.

With regard to the feature selection analysis, we observed that the precision and recall values for the 2-year analysis crossed 90%, while the AUC score exceeded after removing: (a) the age and KAP score features based on the R^2 estimate of the classification models and (b) the age, KAP score, and hospital name, respectively, for the VIF and logistic regression feature selection analyses. The former result can be attributed to the fact that the KAP score is a function of its components, which are also features in the dataset.

We now discuss the real-time prediction of the time to allocation for patients predicted to receive a transplant.

8.3.5 Real-Time Prediction of Time to Allocation

In order to predict the time to allocation for patients predicted to receive a transplant, we used a subset of the simulation-generated dataset developed for classification, obtained by restricting the dataset to only those cases where a transplant was successful. We used the time to allocation as the label for the same feature set. Before training, we preprocessed and balanced the input data and later conducted hyperparameter tuning to optimize the hyperparameters, similar to the exercise conducted earlier for classification. Once again, we applied several ML methods, including standard support vector regressors, decision tree methods such as bagging and random forests used as regressors, and artificial neural networks, to find the best-performing method. Once a model was trained, we estimated the coefficient of determination (R^2 error), root mean squared error (RMSE), and mean absolute percentage error (MAPE) scores on the validation dataset to compare the relative performance of the regressors. We present the accuracy metrics of the regression models in Table 8.7.

LR logistic regression, *SVM* support vector machine, *ANN* artificial neural network

It is evident that the regression models do not perform well, as evidenced by the high MAPE values (MAPE values up to 20–25% are considered acceptable in the regression literature). Further, the negative R^2 value for the ANN indicates significant overfitting. The ensemble bagging decision tree classifier yielded the best results, without the presence of any negative predictions.

Table 8.7 Regression results for time to allocation

Accuracy metrics	SVM	Bagging	Random forest	ANN
R^2	0.87	0.87	0.84	− 1.42
RMSE	193.29	193.62	208.61	856.4
MAPE	80.31	89.01	190.39	91.49

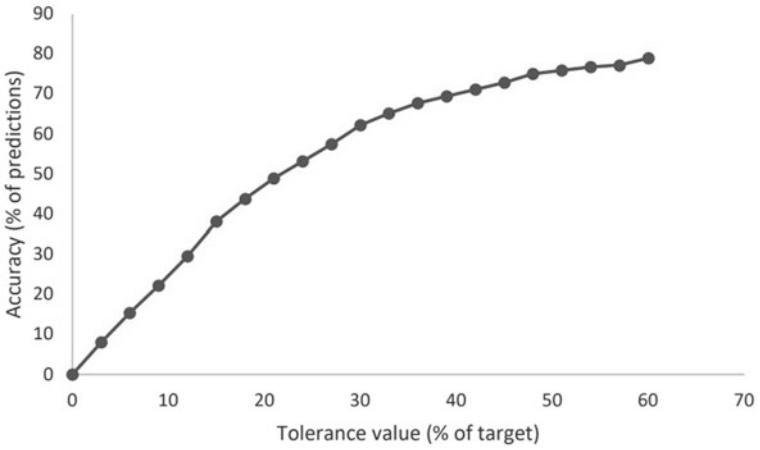

Fig. 8.3 Tolerance range analysis for real-time prediction of time to allocation of an organ

In order to better understand the regression performance, we created tolerance ranges around the prediction label (i.e., the time to allocation for a patient predicted to receive a transplant) to determine the tolerance range at which we observe reasonable performance. For example, a 20% tolerance range implied that we measure whether our predictions are within 20% of the actual time to allocation. This is then treated as a classification problem, wherein a prediction within the tolerance range is treated as 'acceptable' and predictions outside this range are 'unacceptable'. The results of this exercise are depicted in Fig. 8.3.

As expected, larger tolerance values yielded better performance. We see that at a tolerance range of 25%, more than 50% of predictions are acceptable, and at a tolerance range of 40%, more than two-thirds of the predictions are acceptable.

Following this analysis, we performed an outlier detection and elimination exercise on the dataset and retrained the ensemble bagging decision tree classifier on the revised dataset. This retrained model yielded significantly better MAPE scores. Removing these outliers caused the MAPE score to improve to 22%. As part of the outlier detection process, we determined that the worst-performing prediction cases were predominantly due to disproportionately fast transplants—in all such cases, the target time to allocation was orders of magnitude less than the predicted time to allocation. We provide the blood group-wise and total MAPE and MAD scores after outlier removal in Table 8.8.

We describe the above process to highlight the importance of outlier detection in training regression models on the dataset generated by the simulation for real-time delay prediction.

Table 8.8 Blood group-wise and entire dataset regression performance after outlier removal

Blood groups	MAPE	MAD
A	19.830	121.779
AB	29.862	186.055
B	18.049	160.645
O	18.532	162.548
Grand total	22.041	154.409

MAD mean absolute deviation, *MAPE* mean absolute percentage error.

8.4 Discussion and Conclusions

In this study, we present a hybrid modeling approach for real-time delay prediction. This approach is suitable for generating real-time delay predictions for complex queueing systems which are not amenable to analytical approaches and also do not maintain adequate queue log data required for directly training ML/statistical methods for delay prediction. HM involves combining research approaches or methods from other disciplines with one or more stages of the simulation modeling process [16]. Our approach toward real-time delay prediction for complex queueing systems involves developing a DES of the queueing system in question, validating it, and then using this validated DES to generate system state and other relevant data (e.g., characteristics of the service-seeking entity) and train a statistical/ML method on this dataset as a real-time delay predictor. Our approach, given its combination of DES and ML, fits in well within the hybrid modeling paradigm.

A natural question that arises with regard to our approach is this: Given that the ML real-time delay predictor is trained and validated on a synthetic dataset generated by the DES, how does simulation error affect the 'real-world' performance of the real-time delay predictor? Answering this question definitively is beyond the scope of this study; however, our preliminary analytical work on the first question yields the answer that simulation error and ML method error add linearly to form the total error associated with the prediction. Further, we have shown that for a validated simulation wherein 'validated' implies that the expected value of the simulation error is zero, then the expected value of the total error of the real-time delay predictor then depends only on the expected value of the ML method error. Another important question is as follows: While the above result holds for the average total error of prediction (that is, the average total error of the prediction, averaged across the total errors of individual predictions, will tend to zero for validated simulations), how does simulation error affect an individual prediction? These questions and more will need to be answered as this approach matures.

The approach proposed in this study also will need appropriate IT infrastructure for deployment. For example, in the kidney transplantation case, the state of the waitlist will have to be queried each time a new patient registers on the waitlist for generation of the real-time delay prediction. Thus, a suitable software set up for

generating the feature set required for input into the ML real-time delay predictor will be required. The process of setting up the IT infrastructure required to record the input data for each service-seeking entity may lead to the generation of adequate queue log data capturing the system state so that an ML model can directly be trained. In that case, the hybrid DES-based approach may eventually be phased out. This approach will be of use until such queue log data is generated; however, prior to its phasing out, comparing the performances of both approaches may be useful as the hybrid approach may outperform the direct ML approach depending upon the extent of the inaccuracy in the recording of the queue log system state data.

Note that the DES of the queueing system in consideration may not need to be developed specifically for this purpose—the DES may a priori be developed for routine operational analysis of the system and can be repurposed for generating the synthetic dataset. On the other hand, even if it is developed de novo for the real-time delay prediction purpose, it can later be repurposed for routine operational and policy evaluation analyses.

References

1. Sharma R, Prakash A, Chauhan R, Dibhar DP (2021) Overcrowding an encumbrance for an emergency health-care system: a perspective of Health-care providers from tertiary care center in Northern India. J Educ Health Promot 10(January):1–6
2. newslaundary, Queue for queues, packed online slots, and the endless wait to be treated at AIIMS Delhi. https://www.newslaundry.com/2023/04/21/queue-for-queues-packed-online-slots-and-the-endless-wait-to-be-treated-at-aiims-delhi. Accessed 9 July 2023
3. Diwas Singh KC, Scholtes S, Terwiesch C (2020) Empirical research in healthcare operations: past research, present understanding, and future opportunities. Manuf Serv Oper Manag 22(1):73–83
4. Awoke N, Dulo B, Wudneh F (2019) Total delay in treatment of tuberculosis and associated factors among new pulmonary TB patients in selected health facilities of Gedeo Zone, Southern Ethiopia, 2017/18. Interdiscip Perspect Infect Dis 2019
5. Dharmawan Y, Fuady A, Korfage IJ, Richardus JH (2022) Delayed detection of leprosy cases: a systematic review of healthcare-related factors. PLoS Negl Trop Dis 16(9):1–14
6. Woodworth L, Holmes JF (2020) Just a minute: the effect of emergency department wait time on the cost of care. Econ Inq 58(2):698–716
7. Sun Y, Teow KL, Heng BH, Ooi CK, Tay SY (2012) Real-time prediction of waiting time in the emergency department, using quantile regression. Ann Emerg Med 60(3):299–308
8. Fatma N, Ramamohan V (2023) Patient diversion using real-time delay predictions across healthcare facility networks, vol 45, no 2. Springer, Berlin
9. Fatma N, Ramamohan V (2022) Outpatient diversion using real-time length-of-stay predictions. In: Proceedings of the 11th international conference on operations research and enterprise systems (ICORES 2022), 2022, pp 56–66. ISBN: 978-989-758-548-7; ISSN: 2184–4372
10. Xu K, Chan CW (2016) Using future information to reduce waiting times in the emergency department via diversion. Manuf Serv Oper Manag 18(3):314–331
11. Deo S, Gurvich I (2011) Centralised vs. decentralised ambulance diversion: a network perspective. Manage Sci 57(7):1300–1319
12. Baldwa V, Sehgal S, Ramamohan V, Tandon V (2020) A combined simulation and machine learning approach for real-time delay prediction for waitlisted neurosurgery candidates Vaibhav. In: Proceedings of the 2020 winter simulation conference, 2020, vol 21, no 1, pp 956–967

13. Mustafee N, Harper A, Fakhimi M (2022) From conceptualisation of hybrid modelling & simulation to empirical studies in hybrid modelling Navonil. In: Proceedings of the 2022 winter simulation conference, pp 1199–1210
14. Mustafee N, Harper A, Onggo BS (2020) Hybrid modelling and simulation (MS): driving innovation in the theory and practice of MS. In: Proceedings of winter simulation conference, vol 2020-Dec, no September 2023, pp 3140–3151
15. Mustafee N, Powell JH (2018) From hybrid simulation to hybrid systems modelling, pp 1430–1439
16. Tolk A, Harper A, Mustafee N (2021) Hybrid models as transdisciplinary research enablers. Eur J Oper Res 291(3):1075–1090
17. Ibrahim R (2018) Sharing delay information in service systems: a literature survey. Queueing Syst 89(1–2):49–79
18. Whitt W (1999) Predicting queueing delays. Manage Sci 45(6):870–888
19. Jun JB, Jacobson SH, Swisher JR (1999) Application of discrete-event simulation in health care clinics: a survey. J Oper Res Soc 50(2):109–123
20. Vázquez-Serrano JI, Peimbert-García RE, Cárdenas-Barrón LE (2021) Discrete-event simulation modeling in healthcare: a comprehensive review. Int J Environ Res Public Health 18(22)
21. Bertsimas D, Farias VF, Trichakis N (2011) Fairness, efficiency and flexibility in organ allocation for kidney transplantation dimitris bertsimas fairness, efficiency and flexibility in organ allocation for kidney transplantation. Business
22. Yahav I, Shmueli G (2014) Outcomes matter: estimating pre-transplant survival rates of kidney-transplant patients using simulator-based propensity scores. Ann Oper Res 216(1):101–128
23. Cechlárová K, Hančová M, Plačková D, Baltesová T (2021) Stochastic modelling and simulation of a kidney transplant waiting list. Cent Eur J Oper Res 29(3):909–931
24. Jouini O, Akşin Z, Dallery Y (2011) Call centers with delay information: models and insights. Manuf Serv Oper Manag 13(4):534–548
25. Chan CW, Farias VF, Escobar GJ (2017) The impact of delays on service times in the intensive care unit. Manage Sci 63(7):2049–2072
26. Gondia A, Siam A, El-Dakhakhni W, Nassar AH (2020) Machine learning algorithms for construction projects delay risk prediction. J Constr Eng Manag 146(1):1–16
27. Salari N, Liu S, Shen ZJM (2022) Real-time delivery time forecasting and promising in online retailing: when will your package arrive? Manuf Serv Oper Manag 24(3):1421–1436. https://doi.org/10.1287/msom.2022.1081
28. Zhao X, Wang Y, Li L, Delahaye D (2022) A queuing network model of a multi-airport system based on point-wise stationary approximation. Aerospace 9(7):1–14
29. Arora V, Taylor JW, Mak H-Y (2023) Probabilistic forecasting of patient waiting times in an emergency department. Manuf Serv Oper Manag Publ 1–20
30. Whitt W (1999) Improving service by informing customers about anticipated delays. Manage Sci 45(2):192–207
31. Nakibly E (2002) Predicting waiting times in telephone service systems
32. Fatma N, Ramamohan V (2021) Patient diversion using real-time delay predictions across healthcare facility networks
33. Armony M, Shimkin N, Whitt W (2009) The impact of delay announcements in many-server queues with abandonment. Oper Res 57(1):66–81
34. Ibrahim R, Whitt W (2011) Real-time delay estimation based on delay history in many-server service systems with time-varying arrivals. Prod Oper Manag 20(5):654–667
35. Dong J, Yom-Tov E, Yom-Tov GB (2019) The impact of delay announcements on hospital network coordination and waiting times. Manage Sci 65(5):1969–1994
36. Ibrahim R, Whitt W (2009) Real-time delay estimation in overloaded multiserver queues with abandonments. Manage Sci 55(10):1729–1742
37. Ibrahim R, Whitt W (2009) Real-time delay estimation based on delay history. Manuf Serv Oper Manag 11(3):397–415

38. Ibrahim R, Whitt W (2011) Wait-time predictors for customer service systems with time-varying demand and capacity. Oper Res 59(5):1106–1118
39. Senderovich A, Weidlich M, Gal A, Mandelbaum A (2015) Queue mining for delay prediction in multi-class service processes. Inf Syst 53:278–295
40. Arik S, Weidlich M, Gal A (2017) Feature learning for accurate time prediction in congested healthcare systems, pp 189–190
41. Senderovich A, Weidlich M, Gal A, Mandelbaum A (2014) Queue mining—predicting delays in service processes. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinf) 8484:42–57
42. Thiongane M, Chan W, L'Ecuyer P (2020) Delay predictors in multi-skill call centers: an empirical comparison with real data. In: ICORES 2020—proceedings of the 9th international conference on operations research and enterprise systems, no 1999, pp 100–108
43. Ang E, Kwasnick S, Bayati M, Plambeck EL, Aratow M (2016) Manufacturing & service operations management. Manuf Serv Oper Manag 18(1):141–156
44. Baril C, Gascon V, Vadeboncoeur D (2019) Discrete-event simulation and design of experiments to study ambulatory patient waiting time in an emergency department. J Oper Res Soc 70(12):2019–2038
45. Daghistani TA, Elshawi R, Sakr S, Ahmed AM, Al-Thwayee A, Al-Mallah MH (2019) Predictors of in-hospital length of stay among cardiac patients: a machine learning approach. Int J Cardiol 288:140–147
46. Mustafee N, Powell JH, Harper A (2016) RH-RT: a data analytics framework for reducing wait time at emergency departments and centres for urgent care. In: Proceedings—winter simulation conference, 2019, no 2016, pp 100–110
47. Ordu M, Demir E, Tofallis C, Gunal MM (2021) A novel healthcare resource allocation decision support tool: a forecasting-simulation-optimisation approach. J Oper Res Soc 72(3):485–500
48. He Y, Li M, Sala-Diakanda S, Sepulveda J, Bozorgi A, Karwowski W (2013) A hybrid modeling and simulation methodology for formulating overbooking policies. In: Proceedings of the 2013 winter simulation conference, 2013, p 10
49. Barton RR, (2009) Simulation optimisation using metamodels. In: Proceedings of winter simulation conference, pp 230–238
50. Fatma N, Mohd S, Ramamohan V, Mustafee N (2020) Primary healthcare delivery network simulation using stochastic metamodels. In: Proceedings of winter simulation conference, vol 2020-Decem, no June 2021, pp 818–829
51. Balakrishna P, Ganesan R, Sherry L, Levy BS (2008) Estimating taxi-out times with a reinforcement learning algorithm. In: Proceedings of AIAA/IEEE digital avionics systems conference, pp 1–12
52. Chocron E, Cohen I, Feigin P (2022) Delay prediction for managing multiclass service systems: an investigation of queueing theory and machine learning approaches. IEEE Trans Eng Manag 1–11
53. NarayanaHealth (2019) The current scenario of kidney transplants in India. NH Narayana Health: health for all for health. https://www.narayanahealth.org/blog/kidney-transplants-in-india/. Accessed 9 July 2023
54. TimesofIndia, Waiting time for kidney transplant is 4 years in Karnataka; 5,000 on list_Doctor. https://timesofindia.indiatimes.com/city/bengaluru/waiting-time-for-kidney-transplant-is-4-years-in-karnataka-5000-on-list-
55. NOTTO (2018) Allocation criteria for deceased donor kidney transplant (guidelines)
56. Shoaib M, Prabhakar U, Mahlawat S, Ramamohan V (2022) A discrete-event simulation model of the kidney transplantation system in Rajasthan, India. Heal Syst 11(1):30–47
57. KNOS (2018) Kerela Network for Organ Sharing. http://knos.org.in/Aboutus.aspx
58. TNOS (2013) TRANSTAN | Transplant Authority Government of Tamil Nadu, Government of Tamil Nadu | Statistics. https://transtan.tn.gov.in/statistics.php
59. Lakshminarayana GR, Sheetal LG, Mathew A, Rajesh R, Kurian G, Unni VN (2017) Hemodialysis outcomes and practice patterns in end-stage renal disease: experience from a tertiary care hospital in Kerala. Indian J Nephrol 27(1):51–57

60. Cecka JM, Kucheryavaya AY, Reinsmoen NL, Leffell MS (2011) Calculated PRA: initial results show benefits for sensitised patients and a reduction in positive crossmatches. Am J Transplant 11(4):719–724
61. Abraham G, John GT, Sunil S, Fernando EM, Reddy YNV (2010) Evolution of renal transplantation in India over the last four decades. NDT Plus 3(2):203–207
62. Sreeramareddy CT, Qin ZZ, Satyanarayana S, Subbaraman R, Pai M (2014) Delays in diagnosis and treatment of pulmonary tuberculosis in India: a systematic review. Early Hum Dev 18(3):1–24
63. MohanFoundation (2018) Transplant Centres in India. https://www.mohanfoundation.org/transplant-centres/index.asp
64. Census Commissioner of India. Ministry of Home Affairs. Ministry of Home (2011). https://censusindia.gov.in/census.website/
65. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority oversampling technique. J Artif Intell Res 16:321–357
66. Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization. In: 3rd International conference on learning representations, ICLR 2015—conference track proceedings, 2015, pp 1–15