Chronicles of Jockeying in Queuing Systems

ANTHONY KIGGUNDU, German Research Center for Artificial Intelligence, Germany
BIN HAN*, University of Kaiserslautern (RPTU), Germany
DENNIS KRUMMACKER, German Research Center for Artificial Intelligence, Germany
HANS D. SCHOTTEN, University of Kaiserslautern (RPTU), German Research Center for Artificial Intelligence, Germany

The relevance of studies in queuing theory in social systems has inspired its adoption in other mainstream technologies with its application in distributed and communication systems becoming an intense research domain. Considerable work has been done regarding the application of the impatient queuing phenomenon in distributed computing to achieve optimal resource sharing and allocation for performance improvement. Generally, there are two types of common impatient queuing behaviour that have been well studied, namely balking and reneging. In this survey, we are interested in the third type of impatience: jockeying, a phenomenon that draws origins from impatient customers switching from one queue to another.

This survey chronicles classical and latest efforts that seek to model jockeying behavior in queuing systems with a focus on those findings related to information and communication systems, especially in the context of Multi-Access Edge Computing. We comparatively summarize the reviewed literature regarding their methodologies, invoked models and use cases.

CCS Concepts: • **Computing Methodologies → Parallel computing methodologies**; *Queuing Theory*; Jockeying.

Additional Key Words and Phrases: Queuing theory, impatience, jockeying, MEC, Markov decision process.

ACM Reference Format:
Anthony Kiggundu, Bin Han, Dennis Krummacker, and Hans D. Schotten. 2023. Chronicles of Jockeying in Queuing Systems. *ACM Comput. Surv.* XXX, XXX, Article XXX (January 2023), 35 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Over the recent past, the latest developments in Information and Communications Technology (ICT) have enabled numerous new use cases. Some of them, such as remote control, industrial automation [8, 182] and autonomous driving [63, 174, 189] are challenging the legacy solutions to manage queues of data packets or computation tasks in ICT systems with their unprecedented stringent latency constraints and traffic dynamics [27]. It is therefore worthwhile, to adopt concepts from impatient queuing to model and assess their applicability to resource allocation paradigms in communication or cloud computing systems. One of these concepts is jockeying, where consumers can choose from different resource pools amongst the available ones to ensure parallelism for maximum utilization of each service line, consequently enabling performance optimization [85, 177]. This preference was earlier on

[*] B. Han is the corresponding author

Authors' addresses: Anthony Kiggundu, German Research Center for Artificial Intelligence, Trippstadter Strasse 122, 67663, Kaiserslautern, Germany, anthony.kiggundu@dfki.de; Bin Han, University of Kaiserslautern (RPTU), Gottlieb-Daimler-Straße 47, 67663, Kaiserslautern, Germany, bin.han@rptu.de; Dennis Krummacker, German Research Center for Artificial Intelligence, Kaiserslautern, Germany, dennis.krummacker@dfki.de; Hans D. Schotten, University of Kaiserslautern (RPTU), and German Research Center for Artificial Intelligence, Kaiserslautern, Germany, hans_dieter.schotten@dfki.de;schotten@eit.uni-kl.de.

0360-0300/2023/1-ARTXXX $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

hindered by the homogeneous nature of preexisting communication setups where consumption was limited to specific vendor implementations given the prevalent lack of flexibility in the systems architectures. Pioneer efforts by the Open Radio Access Network (O-RAN) community [143] for Fifth Generation (5G) networks and Beyond suggest the introduction of programmable interfaces, virtualization technologies [9, 143] and intelligence to radio resource management allowing for fusion of functionalities of the control and distributed units from different vendors [24, 187]. It is suggested that the interleaving in the architectural design can be achieved from the abstraction of previously hardware-embedded control plane functions as software components (virtual network functions) that can be ported to distributed cloud computing platforms such that hardware resources can be provisioned to scale with usage or demand [1]. The decoupling of this functionality from the hardware is envisaged to allow for the adoption of implementations that seek to address foreseeable shortcomings associated with real-time morphing of the Radio Access Network (RAN) [58, 183] in a multi-vendor environment to meet ever changing consumer or application requirements.

The disintegration of the Baseband Unit (BBU) and Remote Radio Unit (RRU) to hierarchically form layers of specialized functionality is expected to revolutionize definitions for optimal resource consumption [108, 140]. This split in the lower and higher layers (fronthaul to backhaul) into independent units (like the Distributed Unit (DU), Control Unit (CU)) constituting other sub-units (like the Control Plane (CP), User Plane (UP)) that are abstracted as network functions with interfaces to one another can be assembled to form a chain of distributed software components that can characterize a slice in the network [16, 135]. An instantiation of the network slice then organizes into a logical end-to-end connectivity with specialised capabilities [50]; a technique that deviates from the one-size-fits-all approach to instead dynamically organize resource consumption based on the underlying application scenarios and Service Level Agreements (SLA) [137, 156]. It is this heterogeneity therefore, that Engineering enthusiasts envisage will bridge the gap between the dynamic behaviour of an impatient consumer and the challenges forthwith such that continuous evaluations of the resource usage profiles will be requisite for preference over slice configurations listed by different vendors. Depending on the resource requirements of any running service instance, the consumer will in essence weigh the choice for a resource pool in real-time whether to switch slice decks or not with the objective of minimizing costs and latency measures while maintaining optimal resources usage to suit the defined Quality of Service (QoS) requirements [162, 176].

It is worth mentioning that concepts from queuing theory, such as jockeying, are highly relevant. This relevance is evident from the extensive coverage in the literature. Such modeling is applied to diverse domains, ranging from manufacturing [62, 159] to biological processes [13, 38]. Existing empirical studies provide strong plausibility for the adoption of these concepts in communication systems, especially in the context of resource sharing in cloud computing [97, 152]. For a deeper understanding of these concepts, a technical brief that chronicles this impatience behavior then becomes handy and our contributions in this survey can be summarized as below:

- Because the consumption of existing resources usually surpasses the provisioned capacity in communication systems, buffering or scheduling algorithms in queues help mitigate the impacts of degradation in system service quality. As the interactivity between multiple interfaces from distributed components continue to place strict response time requirements on the existing infrastructure, competitiveness for the available pools leads to switching from one pool to another as defined by the components' needs. A retrospective chronology of literature that coherently adumbrates the attempts to model for these preferences is what this manuscript embodies and to the best of our knowledge no such repository exists thus far.
- Moreover, the complexity introduced by the integral application landscape ignites the need for assessment of more agile modeling techniques that can encompass all modalities arising from uncertain behavior of the queue occupants in Multi-Access Edge Computing (MEC) setups. Centralized control of uncertain behavior in most buffering approaches has been the norm with disregard for delegation of routing decisions to the resource tenants for more decentralized control. It could be argued however, that because most

studied queueing models adopt statistical techniques, decentralized control would be complex to model given the state-space curse suffered by these techniques. We therefore propose other methodologies like behavioral modeling to characterize for the inherent dynamics in such queueing systems. This manuscript also highlights existing open issues and promising approaches worth further discourse, specifically with regards to practicability of jockeying in next generation communication systems.

We continue our story line in the next section with some general description of concepts about jockeying in queues and some classification of existing approaches in jockeying studies. Then in the subsequent section, based on the groupings of these approaches, we delve into the specifics of each individual finding and results. We however try to refrain from most of mathematical proof of the resultant equations employed when expressing for the underlying problem but where necessary highlight some theorems or lemmas followed to ascertain the final solution. And finally in our discussion section, we shed more light about open issues, promising trends or approaches worth further studies specifically in the context of applications of impatience for performance optimization in next generation communication networks.

2 JOCKEYING IN QUEUES

2.1 Definitions and Concepts

From the behavioral perspective, studies have categorized impatient consumers as those that observe queue status and refrain from joining, a manner termed as balking [12, 139]. Then there are those that join a queue and abandon it when the accumulated delay is more than expected (renege) [70, 166] and another queue setup where consumers join any queue with the option to switch to a supposedly more optimal alternative queue (jockey). For the tenant allowed to jockey from one buffer to another, profiled setups have been characterized by rule definitions that associate costs and classes to each buffer. The classes define the heterogeneity of the system [67] such that some buffers have more processing capacity (priority queues) than others. Supposedly, it is this heterogeneity that triggers preference of one buffer over another, but is it the only reason for such impatience?

Generally, the reasons that influence impatience among consumers as covered in most literature [126, 167] can be irrational with no consideration for prevailing buffer conditions or rational [29]. However, the most widely studied trigger for impatience in queues has been the queue length difference. That is, length of one queue becoming longer than the other by a preset threshold such that when this threshold is reached, a customer at the end of the longer queue jockeys to the end of the shorter [4, 69, 195]. Other variations in modeling premise this jockeying behavior on a combination of the jockeying threshold and the expected waiting time [53, 194]. The switching can also occur from the shorter to the longer buffer [165] or only jockey when the alternative queue is empty [141, 172]. In more versatile setups though, entities can rationally choose to switch from any position within the queue to the end of an alternative queue [103] or more aggressively intermingle randomly in what is referred to as pre-emptive jockeying [23, 81]. However, in heterogeneous systems like mobile communication or cloud compute, the jockeying threshold becomes inapplicable and instead impatient tenants assess measures like the amount of time required to get serviced [47, 67], the distance over which the workload must be migrated [31, 46] or expected delay [141]. Therefore, the tenant's decision to move around workload factors in such information about changes in the system characteristics [76, 92]. This information in other findings is characterized as costs that define service level agreements, subscription profiles or network traffic classifications [87] to influence the rationality of the jockeying tenants [165].

Predominately, queues are composed of multi-server or single server lines. [96]'s effort to standardize the setups into categories has seen the arrangement of these service lines adopt notations derived from three factors $A/S/c$ (where A, S, c denoted the arrival rate, service interval and the number of channels available for processing respectively). The categorizations formed the origins to acronyms like Markovian/ Markovian/ number of queues (M/M/C), General/ General/ number of queues (G/G/C), $Erlang_k$/ Phase-type Distribution/ number of queues (E/PH/C) etc. The notations have been extended to include other descriptive queue properties like queuing discipline rules and the First Come First Server (FCFS) rule have been mostly adopted [25, 134]. Other service disciplines considered in literature are Last Come First Serve (LCFS) [91, 112] and Serve In Random Order (SIRO) [102]. Priority queuing [180, 181] was also introduced for scheduling time critical processing.

In summary, although studies exist that differentiate between the statistical distribution properties of both arrivals and departures in buffers [36, 55], most findings posit that for system stability, the admission of new entrants to and departures from any buffer line obey a Poisson distribution [142, 178]. Others assume the periodicity as batch arrivals [34, 190] and the effect that this periodicity at which the two events occur has on the impatient consumer has been highlighted by [72]. In generalized processor sharing for example, call admission control (CAC) or leaky bucket admission control, admission and departure strategies are key factors in QoS provisioning for multi-media application [33, 188]. In the next subsection we delve more into such resource sharing use cases where jockeying has found practicability.

2.2 Applications of jockeying

From supermarkets, airports, banks or health facilities to call centers, computer and communication networks, operating systems [151] etc, practitioners always seek mechanisms to optimize the sharing of these queues with the objective of improving overall utilization to enhance system performance and to minimize costs [69, 101]. In the realm of cloud computing, geographically distributed data-centers run jobs on servers in parallel while obscuring the sparseness in the distribution of these servers. The inter-connectivity takes shapes of clusters or organized as multi-server queues [51] for the seamless provision of services or caching content at the network edge. In some more complex scenarios, computing resources can even be deployed over different architectural layers with diverse latency profiles [22]. With such heterogeneity in the underlying support infrastructure [68], processing computationally intensive workload involves queueing and the stochastic nature of the service lines motivates studies that suggest the deployment of concepts from jockeying for resource sharing policies to reduces the under-utilization or over-utilization of one resource pool over another. Recently, collaborative approaches like task offloading [74] have received coverage in literature with objectives ranging from optimizing the sharing of the buffers [116] or figuring out the best task offloading strategy to minimizing processing delays [111, 117] for performance improvement [196] and energy reduction [32, 152].

In packet switched networks or next generation communication systems like 5G and beyond, concept of jockeying has found use in different layers in the RAN as packet scheduling and routing algorithms [60] or implemented as compute protocols (for example OpenFlow) in switches or routers, to in minimizing overall packet delays during transmission [193]. Jockeying has also found relevance in network slice controllers for Software Defined Networks (SDN) in rapidly evolving constellations of terrestrial and non-terrestrial elements to balance the load [2]. With frequency sub-bands modelled as queues, where duty cycling has proven to reduce collision rates and meet latency or QoS requirements, researchers have been interested in impatient behavior like jockeying to model preference of one sub-band over another in LoRaWANs protocols [168].

At a more granular level, processors are intrinsically designed with multi-threading capabilities to realize the requisite concurrency in instruction execution. These executions share writable data culminating into dependencies between the threads that execute given program code. Embedding such concurrent behavior then necessitates process synchronization and exchange of data between these processes [48]. As a buffer management technique for shared memory where processors are abstracted as buffers alongside cache, jockeying behavior is allowed within memory blocks and the requests to the buffers are composed of application code instructions that seek access to memory addresses. Analysis of the jockeying behavior as multiprocessor sharing schemes in distributed servers has been documented with varying objectives not limited to performance evaluation [23, 158] or load balancing [131] but also energy efficiency [59].

So, depending on the application use case, switching workloads from one buffer to another benefits either the end user or the system but can also lead to performance bottlenecks. With specific attention to MEC, we highlight some of the benefits that are associated to this jockeying phenomenon in the paragraphs.

2.3 Benefits of jockeying

While flow control ensures that a fast transmitter does not swamp the slow receiver, congestion control on the other hand in infrastructural abstractions like buffers is aimed at developing optimal job routing procedures that regulate the bulk of data pushed into the network [86, 164]. Usually the packets are buffered with minimum delay until the finite capacity limit is hit, then any incoming packets are dropped based on a defined criteria (random early detection, weighted tail drops etc) [41, 114]. However, individual hosts can also request for certain capacity to be allocated for a flow and the router allocates enough resources (buffers and/or percentage of the link's bandwidth) to satisfy this request. Alternatively, more aggressive hosts can transmit without prior reservations and adjust their transmission rates based on router responses to the traffic swamp. More proactive suggestions integrate jockeying in networks of dynamic access points (parked vehicles) that reorganise as packet relays to mitigate congestion by provisioning task dependent resources organized as slices in 5G and Beyond [11, 71].

In real-time load balancing, jockeying studies embed job migration heuristics to balance the utilization of servers for the (re)distribution of tasks in distributed systems [53, 141]. Other jockeying or job migration studies use information about traffic load in SDN controllers to guide application specific resource allocation to meet the expected QoS [115, 169]. Allocation can also be based on the application's average rate at which data is generated. The objective is to achieve a balance between throughput and delay to optimize the performance of the system. That is, minimizing the growth in router buffer size since it is common in practice that the size of the buffers is finite, which leads to packet drops or increase in delays if the buffer size is infinite.

To sustain the expected QoS especially for critical ultra-low latency connections, like in smart grid systems [120], prioritizing some traffic or consumers over others to give the network operator more leverage for control over queueing occupancy becomes inevitable. The applicability of the impatience in queues is therefore encapsulated in traffic scheduling algorithms during Slice-as-a-Service (SLAAS) operations [71] or channel selections [160]. The preference for one buffer over another (low to high priority or vice-versa) could include paying more in form of subscription fees or more pricey options to meet application specific resource requirements [26]. The problem with pegging a priority tag to each packet is that it becomes hard to differentiate traffic sources or arrange packets according to the flow to which they belong. To cover up for these loopholes, fair queuing (FQ) algorithms which maintain a separate queue with guaranteed minimum share of bandwidth for each flow in the router have been suggested [28, 45, 138]. Or Weighted Fair Queuing (WFQ) which associates each flow (queue) with weights that dictate how many bits to transmit for a given queue have been deployed to manage the link's bandwidth that a flow gets [19]. Measuring how fair a system is when allocating resources was evaluated in [148, 150]s' findings.

2.4 Classifications of Jockeying Models

Different approaches have been adopted to characterize equilibrium conditions and optimize queue descriptors for performance enhancement. These approaches model the dynamics associated with impatient queueing.

2.4.1 Stochastic Modeling. Most literature model the impatient queueing as Markov Decision Processes (MDP), descriptive of the series of events that occur in chain to constitute finite state spaces in discrete or continuous time. For an agile agent to orchestrate a discrete plan that achieves a predefined goal in a dynamic environment, MDPs formally aid the decision making process [3, 130]. The plan basically searches for a deterministic optimal policy (a set of decision rules) such that the best action returns the highest utility in the next system states. In multi-agent settings, generalizations of MDP that premise their action plans on evaluating acquired environment state information or likelihoods have been defined as Partially Observable Markov Decision Processes (POMDP)

[128, 169] or Decentralized Partially Observable Markov Decision Processes (Dec-POMDP) for decentralized agents [88, 136]. Generally, because of the state space exploration-exploitation (trial and error to get the optimal policy) trade-off [155], markovian process modeling suffers from getting stuck in local or global optima and evaluations of these approaches reveal computational complexity issues relating to optimal policy searches [66].

Another class of statistical techniques are Nash Equilibrium approaches, which seek optimal policies that guide selfish participants in non-cooperative interactions independent of other participants' behavioral activity [98, 129]. Findings about this game theoretic technique curtail the participants' behavior under varying strategies like how much prior information each participant has about the environment or the effect of the behavior of some players on the others [76, 77]. The strategies have been termed as either pure (a player is constrained to single pre-selected action or game plan, sticks to it because change in plan does not yield any more reward) [79, 107] or mixed (game plans are randomly selected based on a statistical distribution with options of combining actions) [90, 192]. One of the main shortcomings of nash equilibrium in non-cooperative formations is the impractical assumptions about players prior knowledge of other players strategies and desirable outcomes from given plans. Arguments have also been made about the time these models take to converge to equilibrium [30, 40] and susceptibility to saddle-point problems especially in sequential gaming setups [82].

Analogous to continuous fluid flow, fluid modeling has been deployed in a class of infinite size queueing systems models [175]. Some findings characterize for selected performance measures like loss rate, maximum buffer size, etc under different setup conditions of traffic burst [52, 61] while other studies seek to analyse the busy periods, asymptotic and/or equilibrium conditions using different approaches like matrix-geometric techniques [10]. Other modeling variations rewire queue length status in feedback fluid queues such that the transition rate matrix and drift vector depend on this buffer information [154, 186] or fluid queues with Brownian motion [95]. Due to uncertainty and variation in the input space fluid models are inherently non-stationary in nature given the heterogeneity in the flows. These techniques also exhibit non-linearity profiles yielding transient equations that are difficult to solve analytically or numerically. And not forgetting the computational resources needed for assessment of parametric sensitivity, optimization and validation of the models [109].

2.4.2 Analytic Modeling. Matrix geometric approaches have proven relevant for formulating the steady-state probabilities of Quasi-Birth-Death (QBD) and continuous markovian chains with infinite state spaces where the regularity of new events does not follow exponential distributions [125, 132]. The approach depends on identifying the irregular (initial) and regular (repeating) portions of the representational generator matrices that encode the various states a system (re)visits [78, 93]. [132] shows that from the mean drift condition, sufficient ergodicity conditions follow from sub-division of the state space as encoded in the infinitesimal generator matrix into sub-matrices each characterized by transition rates. The partitioning of this state space defines the system state evolution. The sub-division is also fundamental for the iterative computation of the eigenvalues, eigenvectors of the R matrix (a rate matrix defining the rate a state is visited) [94] and the corresponding equilibrium probabilities. Concerns about using cyclic or logarithmic reduction techniques to solve the R matrix render the approach computationally complex. The solution for this rate matrix depends on the partitioning scheme between the state sub-levels to compose for initial and boundary states, making the unique solution more intractable.

2.4.3 Behavioral Modeling. As an emerging trend, the methodology is motivated by propositions for introducing decentralized control of the impatience behavior in queueing systems in deviation from canonical techniques that assume central control of this behavior [73, 97]. Another methodology in this class uses artificial neural network, a connectionist approach originating from cognitive studies that posit mathematical encapsulation of the human mental processes. Neural networks define for ways to obviate the high dimensionality curse suffered by statistical models of complex systems. They can implicitly identify complex nonlinear relationships between dependent and independent variables, hence their deployment in the quantitative prediction with of queueing descriptors [14, 99]. Encapsulating complexity in a "black box" however makes it difficult to understand how the predictions are made. Other disadvantages of these approaches are the computational burden, proneness to under or over-fitting, immature convergence, hyper-parameterization [21, 185], stopping rules, etc. [119, 144]

3 STOCHASTIC MODELS

3.1 Statistical Models

Preliminary findings to model for the behaviour of impatient consumers when queuing up for resources can be traced back to a paper by [69], in which a simple setup with only two service lines (one "near" and the other "far"). Whether "near" or "far" queue was defined by the queue sizes $X(t), X^{'}(t)$ at the time t of an admission respectively. Such that, the inequality $X(t) \leq X^{'}(t)$ meant new customers preferred to join the shorter buffer ("near") line and were allowed to switch to the alternative one when deemed beneficial. The author analysed cases two of customers arriving to stations following a Poisson distribution at rate λ. The customer had the choice to stay in a given queue operating at Poisson distributed service rate μ_i $(i = 1, 2)$ or jockey to another. The objective was to formulate expressions for steady state conditions in infinite time. The authors first considered for the scenario when no jockeying was allowed such that for each change in state given any action (like a new arrival, exit or both actions happening concurrently in either service lines at a given time), expressions for the rate of change in the queue sizes $\frac{\delta pxy(t)}{\delta t}$ were first derived. The formulation then expressed for $p_{xy}(t)$ as $t \longrightarrow \infty$ $(where\ p_{xy}(t) = Pr\{X(t) = x, X^{'}(t) = y\})$ to yield the bi-variate generating representation as a product over state space changes $s^{x}s^{'y}$ and a subsequent summation over this product to denote for the generating function Eq. (1). Partial derivations of this generating function were then evaluated under variations in queue states $((s = s^{'} = 0), (s = 1, s^{'} = 0), (s = 0, s^{'} = 1), (s = s^{'} = 1))$ for each of these activities.

$$\Phi(s, s^{'}) = \sum_{x} \sum_{y} P_{xy} s^{x} s^{'y} \tag{1}$$

where s and $s^{'}$ denoted the states of the "near" and "far" queues in terms of their queue lengths respectively.

The distributions of the sizes for each queue and the overall system occupancy were also formulated for from the representational difference equations. The paper concluded with the proof for stability conditions of the queue lengths when customers needed to switch from one line to another. The switching happened when the queue sizes varied by one (hence the states $x - 1, x$ or x, x or $x, x - 1$) and this was always from the longer to the shorter line. The solutions for the equilibrium conditions in each of the states were expressed for in terms of probability π that both queues were not occupied.

[103] would later on conduct more concrete studies that compared heterogeneous queue setups and rules where tenants could instantaneously jockey given some threshold on queue length difference or jockeying based on some probability computations. The decision to move from one service line to another either was made immediately when the size of the adjacent queue was shorter by one or given a certain probability based on how the differences in sizes of the queue changed. Figure 1 was a depiction of the heterogeneity in setup (also referred to as strategies), each associated with a set of behavioural rules for the customers. Under the "Tellers' Windows with Jockeying" strategy (Figure 1C), new arrivals were queued to the end of the shorter line and could move to the other queue when the difference between the two queues exceeded one (instantaneous strategy) or based on the rate $(k(w_i - w_j))$ *$(w_i, w_j$ as queue sizes)* at which the queues varied in size (probability strategy). In the case of probabilistic jockeying, the structure of the generator function $g_r(\xi)$ and the evaluations for the steady state conditions revealed that the behaviour of the system was the same as the setup where customers simply joined the shortest queue and never left until the end of service ("Teller's Window"). For the instantaneous jockeying, equations were derived under the three presumed occupancy state categories (n, n), $(n + 1, n)$ and $(n, n + 1)$ that restricted jockeying within the system to the shorter line always (*where n was the average number of occupants in a given queue*). More complexity was introduced in the "Lane Changing" strategy (setup) where

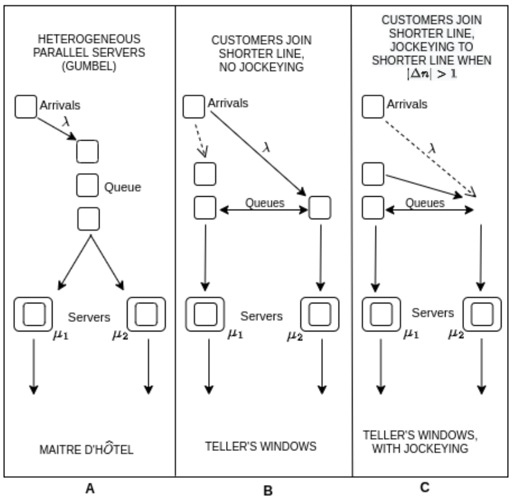
HETEROGENEOUS PARALLEL SERVERS (GUMBEL)
Arrivals
λ
Queue
Servers
μ1
μ2
MAITRE D'HÔTEL
A
CUSTOMERS JOIN SHORTER LINE, NO JOCKEYING
Arrivals
λ
Queues
Servers
μ1
μ2
TELLER'S WINDOWS
B
CUSTOMERS JOIN SHORTER LINE, JOCKEYING TO SHORTER LINE WHEN |Δn| > 1
Arrivals
λ
Queues
Servers
μ1
μ2
TELLER'S WINDOWS, WITH JOCKEYING
C

Fig. 1. Jockeying strategies: In the left-most (A) of the illustration is the Maitre d'Hotel queueing strategy where customers waited in a single line and got served when one of the available stations was empty. In the middle (B) is the Tellers' Window strategy where customers joined and waited in the shorter of the two queues and no switching lines was permitted thereafter. In the right-most (C) was the Tellers' Window with Jockeying, a behaviour where despite a new customer having joined the shorter of the two queues, switching to an alternative one was permitted later given a deviation in the sizes by one.

instead of jockeying to the shorter queue given the threshold, switching was based on rate at which the two service lines differed with probability $(1 - e^{(-k_i(w_i - w_j)t)})$. It was however observed that under this setup, the number of customers that wanted to jockey grew exponentially and that equilibrium conditions were a factor of only λ, $(\lambda = \lambda_1 + \lambda_2)$ and not $\lambda_i, i = 1, 2$. In addition to that, it was noted that some of these strategies led to states where one service line was empty while the other had customers queuing up, states in which queue utilization was compromised. Another interesting customer flow "Route changing" was presented where jockeying was probabilistic, depending on how dissatisfied a customer was being at a certain position in the queue. This was given the fact that customer had no access to queue status information, and the probability that a customer moved from move from queue i to j at time t was computed as $P_{\overrightarrow{ij}}(t) = \begin{cases} 0, & n_i = 0 \\ 1 - e^{-k_i(n_i-1)t}, & n_i \geq 1 \end{cases}$ *where* n_i *was the number of users in queue i and j was the preferred queue.*

It was proven that expressions for equilibrium conditions for the average queue occupancy, the expected number of customers processed and the probability that the a queue was not occupied could be formulated from the transition equations. The work finally presented results from the numerical studies that involved experimentation with varying queue parameters under the aforementioned strategies (setups) and shared insights into the queues' performance and quantitative measures on certain queue descriptors.

It would later be revealed that using the generating function to formulate the steady-state solution to the behaviour of instantaneous switching of service lines was not the only approach to the problem. [46] showed that a closed form solution for some queue descriptors like queue length could be derived from inversion operations on the representational state space matrix. The authors differed in methodology to reason that the underlying Markov process could be modelled using a transition diagram to capture the state changes. It was proposed that, the equivalent coefficient matrix to the equilibrium equations that defined the transition state space was constituent of the regularity property and it was from this property that the solution evolved. Customers that joined the M/M/C queue system were governed by different rules both at admission (like which queue to join based on probabilities or queue size) and in cases of switching queues (say $n_1 - n_2 \geq 2$). The formulation for the proof followed from the definitions of matrices that underpinned the transitions in state (Λ_0, Λ_n). The constituent sub-matrices were then partitioned ($\Lambda_{0i}, \Lambda_{ni}, i = 0, 1...C$) to characterize for the coefficient matrix Λ (*where P was a column vector that denoted probabilities that a queue was in a given state*). Then this coefficient matrix formed the basis for derivation of expressions for the equilibrium conditions ($\Lambda P = 0$) of the queue sizes and it was therefore argued that the solution depended on choosing the proper sub-divisions. Eq. (2) defined for such a coefficient matrix (Λ) constituent of sub-matrices in the case of $C = 2$ (under the assertion that the number of servers did not affect the formation of the coefficient matrix Λ):

$$\Lambda = \begin{bmatrix} \lambda_{01} & \lambda_{02} & 0 & 0 & 0 & 0 & .... \\ 0 & \lambda_{11} & \lambda_{12} & 0 & 0 & 0 & .... \\ 0 & 0 & \lambda_{21} & \lambda_{22} & 0 & 0 & .... \\ 0 & 0 & 0 & \lambda_{31} & \lambda_{32} & 0 & ..... \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{2}$$

The matrices of varying dimensions were defined and elements partitioned into vectors to characterize specific queue state probabilities. This iterative technique was followed by an inversion of each matrix *(Λ_{n2} and Λ_{02})* to yield expressions for the closed form solution of all state probabilities. It was shown how the preference for a certain buffer when both were equal in size then became be a factor of the distance ($n_i - n_k \geq 2$; *i, k were queues*) the jockey candidate had to traverse to the alternative queue. It was also shown how a closed form solution could exist in scenarios where the coefficient matrix was dependent on the number of customers in a queue.

Findings from [46] inspired further exploratory work where [51] not only sought to formulate a general solution for the stability conditions of certain queue descriptors but also modelling how queue parameters like time-until-service and how often newcomers arrived in the system influenced the jockeying behaviour. Here, to switch to any shorter line was preconditioned on the size of any two of $M/M/C$ finite size queues differing by two and a different technique for the partitioning of the state space transitions into sub-matrices was utilized. The first model considered the case when no jockeying was permitted, expressions that characterized for the stability conditions and transition probabilities for an $M/M/3$ setup from the representational differential equations (which were simply probabilities that a given queue was in a certain state) were proven. Analytically, these differential equations were generalised as $AP = 0$ such that A denoted an $n(C^2 - 1) + 1$ sized matrix whose elements corresponded to the measure (coefficient) of the state probabilities, vector P denoted the actual probabilities that a given queue was in a certain state and 0 was a vector that constituted non-zero elements. It was this matrix A that was sub-divided into a series of other column vectors and sub-matrices of varying dimensions and by applying boundary constraints to the generalised equation, expressions that evaluated for the probabilities of the queue being a given state were formulated. The second model then analysed the queue dynamics under the jockeying behaviour where the queue parameters like the arrival λ, the service μ rates and system utilization ρ were a factor of the state of the a given queue; hence the variation in the representative matrix A and magnitudes of the state probabilities. The performance results revealed that the time a customer had to wait before being processed was greater when jockeying was not allowed compared to when the behaviour was permitted reaffirming [46]'s findings. Also, there was a higher probability of the queues staying idle under the non-jockeying setup. Other evaluations for performance measures like the predicted system occupancy under varying degrees of the system utilization or the state dependent queue parameters over time were documented.

[195] caused case for contention about the approaches used by researchers in most of the above reviewed findings, arguing that besides the solutions being repetitive, they yielded no equations that evaluated for the boundary probabilities. In their $M/M/C$ model, the researchers' findings were aimed at unravelling some hidden dependencies between the transition rate matrix R and the overall queue utilization. Arrivals tended to a Poisson distribution

at rate λ were queued to the shorter of $n \geq 2$ servers. Each server processed jobs at Identical and Independently Distributed (IID) rates $\mu_1, \mu_2, \ldots, \mu_n$. The resultant stochastic process $(\{(X_1(t)), X_2(t)), \ldots, X_n(t)), t \geq 0\})$ encapsulated the changes ($S = \{ \overrightarrow{i} = (i_1, \ldots, i_n | i_i \geq 0)$ for $1 \leq j \leq n\}$ and $|i_k - i_l| \leq 1$) in server sizes over infinite time as a factor of the traffic intensity ($\rho = \frac{\lambda}{\mu} < 1$). And stable conditions for this process were defined as $p_{\overrightarrow{i}} = \lim_{n \to \infty} P_{\overrightarrow{i}}(t), \overrightarrow{i} \in S$, ($k$ and l denoting the number of customers in given states). The generator matrix Q and its partitions (sub-matrices A_{ij} with varying dimensions) resulted from arranging the state space (the ordering of state transitions i and j) based on a function that defined which state came prior to or after another. The $\overrightarrow{p}_i, i \geq 1$ distributions were said to each constitute $2^n - 1$ elements as states that formed a block i with $P_0, \overrightarrow{p}_1$ as the bounds. The proof for the solution adopted mathematical techniques (separation of variables plus differential equations calculus)[124][17] for the derivation of the equilibrium probabilities as a factor of the traffic intensity ρ. Taking the case of $i \geq 2$ (therefore only interested in the $2^n - 2$ probabilities in $\overrightarrow{p}_2$ since the bounds were known), it was shown that a solution only existed under steady conditions defined by $\overrightarrow{p}_{i+1} = \overrightarrow{p}_2 \omega^{i-1}, i \geq 1$ only when $\det(A_0 + A_1\omega + A_2\omega^2) = 0, 0 < |\omega| < 1$ *(where* $\omega = \rho^n$*)*. And it was proven through a couple of theorems (which theorized that the eigenvalue(s) $r_2^n - 1$ of the rate matrix R as the lone zero(s) in the determinant) that ρ^n was the only quantity that held true under this conditionality. These theorems were a basis for evaluating for the boundary and equilibrium probabilities of $p(i_1, ..., i_n)$, $\overrightarrow{p}_{i+1}$(for block *i+1*) and $\overrightarrow{p}_2$ to yield a relation between the rate matrix and the equilibrium probabilities. Then taking the case of n servers, to show that p^n was the lone zero in the determinant above, a new chain for the stochastic process ($\bar{S} = \{(i_1, i_2, ..., i_n) | i_t = 0$ or $1, 1 \leq t \leq n\} \bigcup \{m | m > n\}$) consisting of 2^n states was defined and the associated transition state infinitesimal generator matrix $\bar{Q}$ constructed. Most $(2^n - 1)$ of the columns in both generator matrices (Q and $\bar{Q}$) were similar except for some few states (for which a specific generator matrix was formulated). The proof showed that for any server k in the server deck n, $(\frac{p^n}{\omega})_i = constant$, $i \geq 1$ if and only if $p^n = \omega$ (ω as the only real solution of the determinant). Expressions that characterized for measures in the average system occupancy and the time jobs spent being serviced to completion followed from these derivations and the application of Little's Law. The paper then shared numerical results under varying configurations of the system descriptors to verify the formulations.

Instead of single job transfers, [122] provided expressions for policies that could guided bulk workload migration from unreliable or broken servers to available ones with the object of minimizing the accumulated costs. Because no queue status information existed at arrival, a new job was tagged with a time threshold (given the arrival timestamp) within which it had to be processed. In the event that this time threshold expired, the workload was transferred to any available server. Selecting the right server on which to migrate the workload was dictated by the transfer policy deployed. Different policies under different job expiry thresholds were compared to keep the resultant costs (holding and transfer costs) charged during such migrations as low as possible. The costs depended on which position the migrated job landed at and how many job migrations occurred at a given point in time. Initial experiments aimed at obtaining a solution for a single server setup, such that when the server was unavailable, job processing was simply abandoned (reneging). Jobs arriving at each service line following a Poisson distribution λ_i and the processing rates μ_i at each server i ($i = 1, 2, 3, \ldots$) were exponentially distributed. The performance of the model was evaluated based on how optimal configurations for the job expiry periods were, the workload transfer or reneges/transfers ($\beta_{i,j}$) under steady conditions for the queue sizes (L_i say for queue i) etc, while keeping costs low. For the case of the single queue, the stationary probabilities ($p_{i,j}$) of the queue being in a specific state i, j, (where i was the server status of up or down and j as the number of jobs) were ascertained for a few special scenarios. The authors then experimented with the case of N service lines under similar dynamics. From the transitions within the systems emanated a matrix constituent of job migration possibilities $Q = q_{i,j}{}_{i,j=1}^{N}$ that influenced which job(s) was migrated to which queue (transfer policy) with which probability $q_{i,j}$. How often workload was migrated from one queue to another (transfer rate) was then defined by $\beta_i q_{i,j}$ *(i and j as queues).*

To quantitatively obtain the transfer rates, an iterative Poisson approximation was followed to estimate these rates and the total cost given the workload on the server. The process would terminate when two successive rates did not differ beyond a certain margin. However, choosing a job migration rule Q that ensures optimal performance requires exploring the parameter space for each service line. As the number of servers increased, the migration probabilities explodes exponentially making the search harder. It was observed that the Poisson Approximation technique was less viable for the embedded dynamics in such N server setups. The authors proposed some existing heuristic rules (round-robin, fastest other, etc.) that could lead to optimal solutions for the transfer policy under assumptions of consistency in some queue parameters and comparisons in performance of the above mentioned heuristic policies under varying server configurations were also benchmarked.

More like an extension to his [172] earlier work, [171] investigated the evolution in the transition state space of an asymmetric $M/M/2$ setup of finite capacity L using two techniques (randomisation theorem and Runge-Kutta) to formulate statistical equations that characterized the dynamics in the state space. Jockeying was only allowed from the longer line when any of the other service line was empty and the behaviour was captured as a Markovian process. The probability of having a certain number of customers at a time t ($p_{i,j} = Pr(N_1(t) = i, N_2(t) = j)$) in either service lines was defined in a series of difference equations ([171], Equations 1 - 21). The formulation of the model stemmed from the definition of the state probability vectors $P_k(t)$ (and their derivatives $P_k^{'}(t)$, $k = 0, 1, 2, 3, \ldots, L$) for all state transitions over L at given time t. The difference equations were then re-defined in terms of these state probability vectors to evolve into a generalized block-matrix formation $P^{'}(t) = QP(t)$ *(where* $P(t) = (P_0(t), P_1(t), P_2(t), \ldots, P_L(t))^T$*)* constituting sub-matrices (A, B, C). The sub-matrices denoted state space partitions to compose the generator matrix

$$Q = \begin{bmatrix} B_0 & C_1 & 0 & \ldots & & 0 \\ A_0 & B_1 & C_2 & 0 & 0 & 0 \\ & A_1 & B_2 & C_3 & 0 & 0 \\ & & & & \ldots & \\ 0 & .. & & 0 & A_{L-1} & B_L \end{bmatrix} \tag{3}$$

It was argued therein that the expressions for the rate matrices ($R_L = B_L^{-1}$, R_k) could be ascertained from iterations of computations, a method that deviated from the usual technique of calculating for the eigenvalues and eigenvectors of the rate matrix. The modified vector-geometric solution for equilibrium probabilities of the Markov chain emerged therefore from the evaluations of the block-matrix equations in relation to the author's method of expressing for the rate matrix, given $P^{'}(t) = 0$. The resolution for the transition probabilities ($P_{i,j}(t)$) in finite state space followed from the randomization theorem [171, Theorem 3.1], which was more statistically efficient when computing for probabilities in state changes. From applying this theorem to the difference equations emanated recurrence expressions whose properties were used to calculate for the distribution of state changes. It was also shown how the Runge-Kutta method could be manipulated to provide an equivalent evaluation for the transition state distribution. The work then provided numerical and comparative analyses (with [37]) of the results from both methods when used for the calculation of the probabilities in state changes, probabilities that the queue was empty, distributions about the entire system occupancy etc. The author was also interested in tests that sought to understand how probabilities and capacities of the queues were influenced by the overall system utilization ρ under variations in queue parameters. Also, the impact of switching queues on the average processing times and sizes of the queues was assessed versus when no switching queues was possible.

[194] studied the jockeying behaviour and its applicability to multi-beam satellite systems with earth-stations ordered as disjoint zones to form up-link and down-link connections. The sequence of the incoming packets was defined by an independent distribution function such that they were appended to the end of any of the shortest buffers at the satellites and then processed at varying Markovian service rates following the FCFS rule. The vector-geometric solution was based on the assumption that the underlying process that characterized

for such behaviour was non-Markovian. Therefore, the process was segmented to first define an "imbedded Markov chain" ($\{\overrightarrow{X}_l = (X_1(t_l), X_2(t_l), ..., X_c(t_l))l = 1, 2, ...\}$) for which expressions for the probabilities of the buffers' capacities were ascertained under ergodic conditions. Then the solution for stable conditions of the buffers arose from dividing the state spaces into groups $B_{<r}, B_m$ (maximum size of the buffers and maximum deviation between smallest and largest buffer respectively) given the probability (ω) that there existed only a one-to-one relationship between any two states (with the exception of the boundary states) and that new packets did not necessarily visit all states. Following the sub-division of the state transition matrix was the formulation of the equilibrium equations (both for non-boundary and the boundaries states) that characterized for these sub-divisions. The solution for the queue equations borrowed from earlier findings ([133], Lemmas 1.2.4) that there existed an eigenvalue $\omega = \omega_0; \quad 0 < \omega_0 < 1$ of the transition rate matrix R and its determinant $\det(\omega I - \sum_{k=0}^{\infty} \omega^k A_k)$ was 0. *where I was an Identity matrix, A_k was a square sub-matrix block of states when there were k packets in the system.* This same Lemma was adopted for the proof whether the vector-geometric solution was valid for the boundary equations by taking the probabilities of any of the boundary states ($p_{r,r,...,r}$) in $\overrightarrow{p}_r$ and evaluating for redundancy of each of the resultant substitutions of the vector-geometric expressions in the equilibrium equations. The proof concluded by showing that for the stationary probabilities of this "imbedded Markov chain" there existed a geometric parameter $\omega = \sigma$ that defined the uniqueness of the solution. It was also proven that, the stationary probabilities vector ($\pi_{\overrightarrow{i}} = \lim_{t\to\infty} P\{\overrightarrow{X}(t) = \overrightarrow{i}\}, \overrightarrow{i} \in S, S$ denoted the state space) of the buffer capacities was also a modified vector-geometric solution governed by similar uniqueness constraints. This followed by the definition of another "imbedded semi-Markov chain" ($X_i^*(t)$, $i = 1, 2, \ldots, c$) that represented the buffer capacities prior to any last packet at any time. Then, the fact that the stationary probabilities for this kind of process were similar to those of the "imbedded markov chain", it was theorized that from the sub-division of the $\overrightarrow{\pi} = (\overrightarrow{\pi}_{<r}, \overrightarrow{\pi}_r, \overrightarrow{\pi}_{r+1}, \ldots\ldots)$, the process assumed a modified vector-geometric solution too. Numerical experiments were conducted with the purpose of evaluating the system. The performance when jockeying was permitted versus when this behaviour was prohibited were documented and performance results showed how the time the packets spent in the buffers before service varied relative to the processing times, justifying the positive effects of this impatience behavior.

[184] derived for optimal rule-sets that controlled packets in an $M/M/2$ setup of infinite capacity buffers of multi-beam satellite stations for cost effectiveness. Every packet that joined the server incurred holding costs and moving (instantaneous) a packet from one lane to another generated a jockeying cost but no preemption was allowed. The control of packets behavior (admission or jockeying) was also conditioned on the state of current service station in terms of how much load and the monotonic properties of the function F_{ij} (which defined the optimal rule-set as a factor of the expected accumulation of costs under discounted or non-reduced service costs and long-run average costs). Therefore, a packet would only be routed to or migrated to another station if that station was in a valid state (x_1, x_2) at time t. In the model, the system changed states ($X(t) = (X_1(t), X_2(t))$ (state for lanes 1 and 2 respectively) then actions were taken when a new packet arrived or left any of the lanes. Expressions for the expected reduction in costs ($V_t(x_1, x_2)$ over time t and a predicted mean costs over longer time-span under steady conditions were documented. The proof proposed theorems for the existence of a function $F(x)$ that defined when it was okay for a new packet to be routed, when to move packets and when not to. And the functions were adopted for the characterization of the optimal rule-sets for the different behaviour. Explicitly, a new packet was routed to lane 2 only if $x_2 \leq F(x_1)$ was true. That is, if F_{12} and F_{21} were true then it was okay to move a packet from lane 1 to 2 ($x_2 \leq F_{12}(x_1)$) or from lane 2 to lane 1 ($x_2 \geq F_{21}(x_1)$). Migrations of packets was not optimal when $F_{12}(x_1) < x_1 < F_{21}(x_1)$ and when $F_{12}(x_1) < x_2 < F_{21}(x_1)$ One restrictive characteristic of the control functions was that the decision to move from a station with lower costs than its alternative was only plausible if the alternative station was idle. This characteristic was evaluated by taking a state of the system when the alternative station was not empty and validating it under the jockeying control that underpinned the

best-fit rule-set. The proof for existence of other asymptotic characteristics in control functions that defined the optimal rule-set F under discounted costs were also provided and it was shown that the same applied to the non-discounted mean costing over time. It was ascertained though that, the control functions under both costing environments only converged under specific conditions of the both the jockeying and service costs.

Inspired by the notion of vendors offering varying pricing for their services to give the user more preference, according to [165], it did not matter whether the shorter or longer queue was joined or jockeyed to as long as the choice for either yielded costs below a preset limit c. These costs were a factor of both the size of the queue (N_i) and processing fee $\beta_i > 0$, ($i \in \{1, 2\}$, i denoted a queue), such that new customers (with arrival rate λ as a Poisson distribution) were added to a queue depending on the magnitude of $\alpha_i N_i + \beta_i \leq c$ (where $\alpha_i = 1$ was a weighting measure on queue) of either queue. And staying away completely from the services when the overall service costs would turn up being too high was equally an option. In a setup with finite capacity buffers of maximum length defined by $K_i = [c - \beta_i + 1]$, the representational Markov chain $X(t) = (X_1(t), X_2(t))$ of the changing queue sizes was irreducible and a factor of the underlying costs. Formulations for the steady-state distribution deriving from theoretical comparisons between selected queue descriptors (K_i, β_i, c) under specific quantities of the system utilization ($\rho = \frac{\lambda}{2\mu}, \rho \neq 1, \rho \neq \frac{1}{2}$) were defined and for each comparison, expressions that represented the state changes (as a factor of β_1, β_2 and c) obtained. The distinction line of the three cases compared was based on the magnitude of $K_1 - K_2$ (to determined whether a new client had to join or jockeying to either server1 or server2); That is, $\beta_2 - \beta_1 - 1 < K_1 - K_2 < \beta_2 - \beta_1$, $\beta_2 - \beta_1 < K_1 - K_2 < \beta_2 - \beta_1 + 1$ and $K_1 - K_2 = \beta_2 - \beta_1$ such that all possible states reachable for each of the three cases were defined. For each comparison case, balance equations which characterized for the influence of leaving or entering the queue in a given state and under what circumstances a specific state was reachable formed the basis for the proof for equilibrium probabilities ($\pi_{i,j}$) of the system. The solutions for these balance equations evolved from them being re-written as difference equations in relation to the state sequence $s_i, i \in \mathbb{Z}_+$. It was shown from induction principles how $\pi(i, j)$ could be derived for from relations between $\pi(1, 0)$ and $\pi(0, 0)$.

The discussion about admission control in queueing systems was taken further by [113] building on earlier findings from [42]. Here, new arrivals to a two service (M/M/2) line queuing systems were managed using an admission controller that distributed tasks to either lines based on Bernoulli computed probabilities. The customers in the queue were charged a cost for staying in the system and since the customers kept getting knowledge about the status of the queue, jockeying from from a line to the tail-end of another was also associated to a cost. The study was interested in the behavior of a tenant who from the time of entering the queue continuously received updates about the status of the system and had the option to use this knowledge to make the decision whether to stay in the current line or to move. With close similarities to [47], the authors provided limit policies that guided the behaviour of customers joining a service line or moving from one line to another while ensuring that the least costs were accumulated over time in service. And the optimal policy therefore was one that prioritized reducing the expected cumulative costs. The work proved that there existed limits on a current user's position in the a service line and on the number of customers in the alternative line, above or below which the user could make the decision to move to another service line or stay put. It was also hypothesized and justified therein that there existed a maximum queue length that would lead to a new arrival having preference for a specific service line. To model for the continuous-time Markov process, a vector denoting the queue status ($x = (q_1, q_2, l) \in S$; q_i as service lines $i = 1, 2$) at a time t was defined. Then for any of the service lines in a state $q_1 q_2 l \in S$ with a customer at a given position taking a set of actions a, $a \in A(A = \{0, 1\})$, functions for the total anticipated reduction in costs (V_n for server 1, $V_n^{'}$ for and server2) after a finite number of times n, were defined by $V_n(q_1, q_2, l) = \min_{a \in A} V_n[(q_1, q_2, l), a])$ and $V_n^{'}(q_1, q_2, l) = \min_{a \in A} V_n^{'}[(q_1, q_2, l), a])$. The derivations followed from two theorems (one for the jockeying behavior and another for the control of new arrivals) that sought to ascertain whether the optimal rule-sets (that dictate customer behavior) monotonically increased or decreased with limits on the queue characteristics. In the case of the jockeying policy, the proof adopted defined inequalities (given the state of the queues) for which the cost functions held true for the optimal rule-set to maintain monotonicity over a finite period of time. It was therefore shown that, there existed a limit $q * (q_1, l)$ on the current queue length and the number of customers ahead of a current customer such that if the number of customers in the alternative queue was less than this limit, then it was okay for a customer to jockey and this limit was non-decreasing in l. Also, for the sizes of any two servers, if the number of customers ahead l of the current customer was greater than the limit $l * (q_1, q_2)$, it made sense to jockey and this limit decreased monotonically in q_2. The proof for an optimal rule-set that dictated the behavior of new arrivals on the other hand followed from the theorem that, given prior knowledge about the queue status, there was a limit $\epsilon(q_1)$ on the length of a given queue for which if the size of the alternative queue (q_2) was greater than this limit, then a new customer should join q_1 and the limit was monotonically non-decreasing in q_1. And that these two theorems held sufficient for conditions when no discount was given on the overall costing in infinite time.

For jockeying control, [147]'s work concentrated on obtaining analytic expressions that evaluated for the number of jockeys made from one service station to another before getting served given that each jockey accumulated a cost. New arrivals (as a Poisson distribution) were pushed to the shorter station or either station with equal probability and the jockeying for a tail-end customer from the more occupied station to the end of the less occupied station was permitted when the difference between the length of any of $k \geq 2$ servers (each serving customers at IID processing times) equalled a threshold d. The solution involved splitting the transition state space based on queue length statistics like customers ahead (f) or behind (b) a given customer ($(f, b) \implies b \geq f \geq 1$) in the service line and whether the next move by that customer was a jockey or a forward in the same queue. The proof that was adopted from generating function theory, sought to provide mappings for a customer's state or position in the queue to the possible number of jockeys ($Y_{f,b}$) that the customer would make before being processed. The evaluation of the generating function therefore was characterized by the iterative formulation of the relationships ($\Phi_{f,b}(s)\ 0 \leq s \leq 0$) between these state actions. This required initial statistical computation for the chance ($P_f(j|b), 1 \leq f \leq \min b, j$) that the customer's next state ($(f-1, f)$) followed a jockey to the alternative queue or a forward move in the same queue to end up in state $(f-1, f) f \leq \min j, b$.

The expression for the chance that a customer would move to the alternative station (in state $(f-1, j), f \leq j \leq b-1$ or $j \leq b$) were premised on the prevailing conditions in the current queue. That is, provided that $b > f - 1$ and that the variation between the number of clients behind the current customer that left that station versus the number that wanted to join any of the queues was $b - j$. From the total probability [147, Equation 1] principles, the formulation of the relations (given the probabilities of being in a state) evolved into Eq. (4) which defined for the generator function ($\Phi_Y(S)$). This function mapped the expected number of jockeys ($Y_{f,b}$) from one queue to another before the customer was serviced and it was a prediction dependent on the f (the number of people ahead of the current customer being served) and the arrival rate.

$$\Phi_Y(S) = \frac{2-\rho}{2+\rho}\left[\frac{2+\rho}{2} + 2B_0(S) + \frac{4}{\rho}B_1(S)\right] \tag{4}$$

where B_0 and B_1 were linear expressions that denoted aggregations of state relations and service line utilization over n customers in a queue. And $0 \leq s \leq 1$.

Additionally, a generating function ($\Psi_K(\theta)$) that yielded the random distribution of how many customers K ahead of the current customer left the queue was defined. This followed the analysis of the process representative of either actions (joining or leaving) over time, therein referred to as a *"difference random walk" (DRW).*

The authors in [113] aggregated static and dynamic controls from individual controllers like routing, admission, service and jockeying to ascertain a best fit (hedge point) solution for equilibrium conditions. New arrivals did not get to choose which queue they joined but went through an admission controller [42] which managed the allocation to service lines each operating at varying exponentially distributed processing times. Over time, the costs accumulated due to holding, processing or jockeying influenced the required measure for a selected rule-set given the existence of a costing threshold. The jockeying control was triggered on service completion in one service line and workload here was either accepted, rejected or migrated. An admission rule (u such that $u = \{u_0, u_1, u_2, \ldots\} \in U$ *U as the collection of all possible admission rules*) as a set of functions, on the other hand matched a given state (x) to possible actions based on the requirements of the decision and the associated forecast cost discount ($J_u(x)$). Therefore, the optimal admission rule (value function $V(x) = \max_{u \in U} J_u(x)$) was one that maximized this discount. Behind every action was an operator, therefore a function that mapped the transition to the next state as a conditional probability based on the admission rule and the current state was defined. To prove the presence of an optimal control rule-set required verification of each operator to be characteristic of the specific structural (sub-modularity and concavity convergence) properties. The emergent resolution involved repetitively trying different operations (using the Bellman operator T) against initial value function V_0 in infinite time such that the optimal value function set V_s included only those functions that obeyed the properties for all states. To evaluate for the value function that maximized the discounted costs, the structure of each function $V \in V_s$ was validated to ascertain whether the magnitude of the expected discounted costs (related to taking an action that led to the next state) were non-decreasing or otherwise. The optimal policy adopted a form therefore, that was determined by switching functions (S_1, S_2, L_i and G_i where $i = 1, 2, 3, 4$) which depended on the operation in a given state. Some functions managed assignments of new arrivals to queues and which paths they took while some functions managed completed services and migrations from one queue to another. This approach led to the division of the state space along a decision making (hedging) point. The numerical evaluation included two symmetric servers (charging equal holding and jockeying costs as well as operating at equal service rates and arrival rates) and for the resultant stochastic process, the evaluation for the best approximation of the optimal value function $V(x)$ was shown. This followed a couple of repetitive application of the operator T to the value function using value iteration techniques and results showing the optimal behaviour of the value function with regards to the decision controls for server actions (admission, routing, jockeying, stopping etc) were documented.

[89] is built on earlier findings [75, 127], but with specific interest in the boundary asymptotic formation of the probability distributions of the queue sizes in an $M/M/2$ system (one special buffer and the other normal). The growth in size of normal line was continuously monitored after an exponential time interval such that workload was transferred ($L - K$ where L was the length of the normal line) when the size of the line exceeded a preset threshold K). For the continuous Markov chain $\{(L_1(t), L_2(t)), t \geq 0\}$ *(L being the lengths of each service line)*, steady-state conditions of the queue sizes were first derived for as $\lambda q < \mu_2$ and $\lambda < \mu_1 + \mu_2$ *(where $\lambda < \mu_1, \lambda < \mu_2$ were arrival rates and q the probability that new arrivals joined the special line).* The proof developed from adoption of the Foster-Lyapunov[56][123] condition for stability to show that the process states were revisited in finite time (positive recurrence). For uniformity, transitions probability matrix $P = I + Q$ (*where Q represented that rates at which the queue varied in size and I an Identity matrix*) was subsequently divided into sub-matrices based on levels. By exploiting the special structural properties of the irreducible sub-matrix ($D(\sigma) = A + B\sigma + C\sigma^2$ *- where σ defined by Eq.* (5) *was the decay rate; A,B and C as sub-matrices depicting state space splits*), the authors showed that the boundary limits of the probability distribution of the normal queue size decreased at some constant ratio sequentially (geometrically). That is, the number of customers in the normal line increased based on some geometric limits (tail asymptotes) in infinite time vis-a-vis the number of customers in the special line staying constant. A further split of the sub-matrix D along the boundaries evolved into the formulation for the proof. The characterization followed from the evaluations for convergence norms $\overrightarrow{\sigma}, 0 < \sigma < 1$, verification for the existence of the $\frac{1}{\sigma} - invariant$ measures and vectors of the sub-matrices [89, Lemmas 4.1-4.2].

$$\sigma = \frac{(\lambda p + \mu_1 + \eta) - \sqrt{(\lambda p + \mu_1 + \eta)^2 - 4\lambda p \mu_1}}{2\mu_1} \tag{5}$$

where p was the probability of joining the main queue, η was a distribution of the server polling intervals. Especially, when $\lambda q < \mu_2$ and $\lambda < \mu_1 + \mu_2$, it was shown that the boundary limits for joint stationary distribution $\pi_{i,j}$ as approximated from the decay rate decreased geometrically and the decay rate varied with the individual queue sizes. Numerical analysis of the these formulations were performed, experimenting with different queue parameter settings for the arrival or processing rates in each queue. At different monitoring intervals, results were compared under changing values of the arrival rate and it was observable that the increase in the decaying rate was linearly proportionally to the rate at which customers joined the queue. It was interesting to also observe how the decaying rate responded to increasing quantities of the processing rates and the results surprisingly suggested that the decaying rate decreased linearly with respect to an increase in the service rates of the queues.

As a guide for efficient use of energy when allocating jobs in a multi-server processor sharing setup, [153] proposed a an alternative energy-efficient heuristic and evaluated its performance as better than the popular slowest-server-first (SSF) policies. Assuming a heterogeneous infrastructure with finite buffer sizes, insensitivity to job sizes or relationships between system descriptors, the findings aimed at improving the energy efficiency when apportioning workload while maximizing the throughput. The jobs streams followed a Poisson distribution at rate λ to land on any of $j \leq n$ servers for processing at exponentially distributed rates. Each server consumed energy (ε) at a rate $\varepsilon(\mu) = \mu^3$ that was monotonically decreasing with the service rate. The baseline heuristic was the insensitive jockeying policy, where jockeyed jobs displaced existing jobs backward or forward and position allocations were defined with equal probability to the departures to characterize for the server state space evolution. The energy efficiency of the servers was then collectively calculated as the ratio of the summed long-run mean throughput T and the expected consumed power E as $\frac{T}{E}$. The proposed energy-efficient (EE) rule-set on the other hand allocated tasks to the first $\hat{n} \leq n$ set of busy servers such that the tasks were routed to the least occupied or empty buffers. Then the next $\frac{s}{b}$ (b as the defined finite size of a particular queue and state space $s \in S$ denoting the number of jobs in the queue) servers were selected for task processing if all instances in the $\hat{n}$ set were occupied. It was revealed that this server cascading (in states $\hat{n} < s \leq \hat{n}b$, $s \in S$) yielded relatively lower task processing times to minimize the overall load and balance server utilization. Formulation for comparative analysis of the two heuristics followed from the theoretical propositions that showed conditions for $\hat{n}, b, \eta$ and μ_j ($\hat{n} \geq 1$) under which the energy-efficient policy was more optimal than the SSF policy. Qualitative measures were then defined by the relative error between the two rule-sets which was computed as $\Delta\hat{E}^{SSF}_{EE_{\hat{n}}} = \Delta\left(\frac{T_{EE_{\hat{n}}}}{E_{EE_{\hat{n}}}}, \frac{T_{SSF}}{E_{SSF}}\right)$ Numerical evaluations of this error under selected quantities of η and μ in a series of experiments were performed to find optimal $\hat{n}$ that maximized the servers' energy use ($\frac{T}{E}$) in the entire set. It was shown that, a certain measure of $\hat{n}_*$ provided a ceiling such that the mean aggregated processing rates should not exceed the mean arrival rates and under preset configurations of b, a_i (where a_i was a measure of how two cascaded server groups differed in processing capacity), the EE policy was more optimal energy-wise that the SSF. However, this efficiency decreased with server utilization or traffic intensity, that is, EE was only more optimal under limited traffic conditions. Under similar settings though, no significant improvements were recorded in the throughput of the EE ruleset.

3.2 Nash Equilibrium based Models

Nash-equilibrium rules for an $M/M/2$ system with threshold jockeying permitted were studied by [76] to understand the value of prior purchased queue status information to a customer. The experiments were inspired by the notion that such information underpinned optimal usage of the service line to consequently minimize waiting times and that for new arrivals, preference for which queue to join was influenced by the precedent customer having bought similar information (externalities). Analogous to a cost benefit model, the authors deployed Nash-equilibrium strategies that put a value on the purchased status information by evaluating how much benefit a client got from it. The expected benefit $g(p)$ meant knowing how much less time the consumer would be waiting to get serviced given the charges for that information. Two strategies arose here $p, 0 \leq p \leq 1$

which denoted the probability whether the information was purchased or not respectively. That is, a pure and a mixed strategy. But the strategies ($g(p) = C, g(p) \leq C, g(p) \geq C$) were a factor of the relation between the benefit of the acquired information $g(p)$ and the costs C. Basically, on arrival a customer purchased a probability value that abstracted the potential benefits given the current state of the queue. It was also relevant for the investigation to ascertain whether this information from prior clients impacted the subsequent client positively or negatively. Following a partitioning of the state space, the authors used the matrix-geometric method [133] here to obtain the stationary probabilities $\pi_{i,j}$ *(i or j being sizes of either queues)* of each queue size given the jockeying threshold $N = 3(3 \leq N \leq \infty)$. The stationary probabilities were evaluated from the eigenvalues/eigenvectors and spectral properties of the rate matrix R. For each assumed position a customer took, a function $g(p)$ for ascertaining the benefit (expected waiting time) at that position under a given Nash-Equilibrium strategy was formulated from difference equations. The numerical results showed comparisons between the benefit from purchases depending on the jockeying threshold N and the service line utilisation $\rho = \frac{\lambda}{2\mu}$ (*where λ, μ were the arrival and service rate respectively*). The study conclusively expressed for the magnitude of influence (be it negative or positive) of the actions of prior customers on new arriving customers when they purchased knowledge and when $N = 3$. The effect was considered positive if the acquired knowledge helped the consumers optimally use the service line and negative if the customer ended up waiting longer than expected. Evaluations for the average sojourn time under varying measures in system occupancy N revealed that as N grew larger than 4, the benefits of purchasing knowledge were negligible.

3.3 Fluid Theory based Models

Studies in the applications of consumer behavioural control would further penetrate in queuing systems to realise distribution of load in [47]'s findings where they ventured into expressing for the steady and non-steady conditions of service lines using fluid modelling. Exponentially distributed batches of clients chose between a low-cost and high-cost service line in an $M/M/2$ setup with an embedded controller that moved the clients from either service lines at fixed or variable costs depending on the bulk of client transfers involved. The authors here extended [105]'s earlier work to derive for the class of policies that optimised the average long-run costs accumulated with client transfers, without factoring in the deviations (jockeying threshold) in the sizes of the service lines. A policy or rule-set $\pi \in \Pi$ basically encapsulated the magnitude of bulks transfers to move, how long any of the clients to migrate had been waiting plus any other queue status knowledge relevant for the Markov chain. Besides the holding costs $h_i n_i$ (at server i), the decision $\mathbb{D}$ (given a service completion or new admission to the service line) to transfer was associated with fixed K and varying costs mj. Eq. (6) characterized for the average predicted cost of a client starting in a state x given rule-set π as $g^{\pi^*}(x)$, the optimal rule-set π^* was conditioned on $g^{\pi^*}(x) \leq g^{\pi}(x)$ for all states.

$$g^{\pi}(x) = \lim_{n \to \infty} sup \frac{\mathbb{E}_x^{\pi}\left\{ \sum_{i=0}^{n} [k(X_i, Y_i) + \int_{\rho_i}^{\rho_i+1} c(X_i, Y_i)\delta t] \right\}}{\mathbb{E}_x^{\pi}\{\rho_n\}} \tag{6}$$

where X_i was the state at the i^{th} decision step and $\mathbb{E}_x^n$ the expectation (given policy π) of taking action Y_i would accumulate $k(.)$ as the overall cost at a cost rate $c(.)$.

The fluid model was said to be in equilibrium when all incoming clients had been processed ($\bar{M}(t) = 0$, $t \geq t_0$) by the system (service rate greater than the arrival rate) and otherwise ($\bar{M}(\delta) \neq 0; \delta > 0$) if at a certain time-frame $\delta > 0$ the queues were still occupied. The control rule-set therefore ensured that transfer of workload within the system led to non-busy service lines in the shortest time possible while keeping the accumulated costs as low as possible (that is, moving workload to the costly service line was only if it was aimed to making sure that the line was not idle). The derivation (based on the sample path argument) emanated from the comparison of processes that mimicked clients joining the queues, guided by two policies (π - from high cost queue to low cost queue and $\pi^{'}$ otherwise) but each dedicated to moving clients from h_1 to h_2 and vice-versa. Function definitions that expressed for the differences in costs were formulated for each rule-set under varying queue sizes to maintain some level of uniform distribution when moving around clients while making sure that $\pi^{'} < \pi$. These expressions formed the basis for determining the optimal rule-set that ensured transfer of workload from either queue only when not occupied (non-idling policy), single client transfers or bulk transfers under symmetry conditions of the queue. The performance of the rule-sets was numerically analysed in a setup of six asymmetric queues (varied costs and arrival times) while maintaining a consistent service rate μ and each queue was polled randomly for workload to transfer after a set time T. The results suggested direct proportionality between the overall costs under given optimal rule-sets. It was also observed that there existed relationships between the polling time T and the performance or system load.

In [44]'s studies, fluid theory was the basis for formulating steady-state expressions for a cluster of cascading servers that continuously co-operated to share task executions. Specific classes of customers with IID arrival times were allocated positions in specific queues. But these class specific queues served (IID processing times) customers from other classes only when they were idle. Two variations were presented therein; one of the class of *X-models* as a setup with two servers (each representing a class of customers) and the other, coined the term "tree-cascade system" that included three servers (classes). It was argued that to compute the steady-state conditions for such a networked system, one needed to prove the stability of the underlying fluid limit model, which required that the system was in equilibrium if the emergent Markov process X had a non-repeating consistent statistical value [39]. It was theorized (from [43]) therefore that, assuming two service line scenarios, where one had a lower processing rate ($r_1, r_2 \geq 1$ or $r_1, r_2 < 1$) than the other (given $r_1 = \frac{\mu_1}{\mu_{2,1}}$ and $r_2 = \frac{\mu_2}{\mu_{1,2}}$), then stability for fluid limit based models could only exist under specific conditions of comparative variations in the arrival (λ) and service rate (μ). These conditions were specified by Eq. (7), such that the aggregations over the quantities $\bar{Q}(0), \bar{U}(0), \bar{V}(0)$ of the queue network equations at a time $t \geq 0$ equalled unit, $\bar{Q}(t) = 0,\ t \geq t_1$.

$$\begin{cases} (A_1) & \lambda_1 - \mu_1 + \frac{\lambda_2 - \mu_2}{r_2} < 0, \\ (A_2) & \frac{\lambda_1 - \mu_1}{r_1} + \lambda_2 - \mu_2 < 0. \end{cases} \tag{7}$$

where $U(t)$ = the time before a new arrival seeks to join the server, $V(t)$ = time left to service end for customers and $Q(t)$ = size of the buffer at any time.

The proof for the stability of the X-model then derived from the adoption of the Lyapunov function $f(t)$ [123] which related the sizes of the queues ($\bar{Q}(1), \bar{Q}(2)$) over time $t \geq 0$ and it was necessary to show that $f(t) \leq -C$ (constant C>0) under varying comparisons of the arrival (λ_i) and service rates (μ_1, $i = 1, 2$). The studies were extended to the tree-cascade setup under work-conserving rules where jockeying was allowed to any of the servers that were free although the third station was dedicated to supporting the other two stations. The corresponding Markov process X (renewal arrivals and jockeying when one queue was empty) was similar to the one for the two server setup except with higher dimensionality in the state space and with slight differences in the system equations that expressed for the interactions amongst the service stations. The stability of the fluid model in this setup was also premised on ratios of processing times ($r_{1,3} \leq r_{1,2}, r_{2,3}$) and the rate at which customers sought to join a given buffer. Similarly, adopting the Lyapunov function $f(t)$ [123] and because the function was inherent of similar differentials at any time t with the server sizes, the proof for the equilibrium conditions gathered from evaluation of the inequality $f(t) \leq -C < 0; C := \min\{C_1, C_2, C_3\}$ to hold under varying sizes of the queues ($\bar{Q}_i(t), i = 1, 2, 3$). It was shown how the fluid limits on the servers' occupancy hold stable in infinite time, hence that with such server conditions the Markov process revisited specific states in finite time (positive Harris recurrent).

4 ANALYTIC MODELS

4.1 Matrix-geometric models

Pioneer modeling of the shortest queue problem using matrix geometric approaches by [133] was foundational to [64]'s findings that expressed for the stationary probability vectors of queuing systems. The study simulated an $M/M/2$ airport setup with planes that accessed runways (as queues) following a Poisson distribution at rate λ and exponentially distributed service times μ for each runway. The continuous-time Markov chain representation of the state transitions for all the dynamics within such a system took the form of an infinitesimal generator matrix Q (Eq. (8)) with sub-matrices (A_0, A_1, A_2 and B_0) that inherently encoded state transitions.

$$Q = \begin{bmatrix} B_0 & A_0 & 0 & 0 & 0 & .... \\ A_2 & A_1 & A_0 & 0 & .... & \\ 0 & A_2 & A_1 & A_0 & 0 & .... \\ 0 & 0 & A_2 & A_1 & A_0 & 0..... \\ . & & & & & \end{bmatrix} \tag{8}$$

Ideally, it was suggested that completing a service in either queue and a jockey to a shorter queue summed to a total transition rate of 2μ. And this also altered the dimensions of the sub-matrices that composed the generator matrix Q. For that, a non-negative rate matrix R that represented the rates at which states changed was defined and subsequent evaluations led to the solution for the stationary probability vectors $\pi = (\pi_1, ...\pi_n)$ being formulated. It was then shown how the proof for the necessary steady state conditions of the stochastic process existed only if $\lambda < 2\mu$. This was premised on the proposition ([64], proposition 1) that the underlying process to such dynamics inherently possessed conditions or there was a probability that certain queue states (i, j) in the process were revisited in finite time (positive recurrence [133]). And the stationary probability vectors were verified to exist under the steady conditions $\lambda < 2\mu$. The author provided results from some numerical experiments with variations in parameter settings for how busy the queue was ρ and $n(n = 5, n = 10, n = 15 \ldots)$. It was conclusively suggested that as the threshold n increased $(10 < n > \infty)$, differences in stationary probability vectors became negligible. The experiments also provided computations for some queue statistics like the means of individual queue lengths, mean waiting time and reaffirmed findings that systems with jockeying permitted performed better than those where the behaviour was prohibited.

Using the matrix-geometric approach to evaluate for stationary probability vectors for the stochastic chain of events was extended to an M/M/C queue by [93]. It was shown that from manipulating the structure of the generator matrix emanated a more reliable approach for ascertaining the stationary probability vectors. This followed the author's counter arguments about the methods used when partitioning the state space by [51], hence suggestions for an alternative solution. The setup assumed multiple servers with an infinite number of tenants that arrived following a Poisson distribution λ and each server processed tenants at rates μ that were exponentially distributed. New arrivals joined the shortest queue but if queues were equal in length they joined either queues with the same probability while jockeying was allowed only when the difference in queue sizes was two (jockeying threshold $n = 2$). The representation of such a QBD process was an infinitesimal generator matrix Q composed of sub-matrices to denote the state transitions of the queue lengths but introduced slight modifications to the structure of the stationary probability vectors. It was proposed that the boundary conditions required a special extension of the vectors, hence a new expression for the rate matrix. The proof applied previous theorems (theorem 3 [65]) such that the iterative evaluations culminated into an expression for the rate matrix R. Extensions were also done to [146]'s work, introducing the argument that because the structure of the infinitesimal generator matrix exhibited likelihood of boundaries state being revisited, there was need for new evaluations of the average size L of the queue. Premised on the existence of an absorbing state θ (a state when a new arrival got processed immediately) as part of the state space of the underlying QBD, expressions for the stationary waiting time probabilities were also derived. The re-partitioning of the representative infinitesimal generator matrix followed by the relevant proofs morphed into formulations for $W(.)$ as the Laplance Transform $w(s)$ and the corresponding closed-form solution for the average time taken before being processed. The authors additionally showed how application of randomization methods could yield for the characterization of the stationary waiting time probabilities $W(t)$ for QBD processes. The proof assumed that given a statistical initial state denoted as a vector z there existed an n^{th} transition step of the stochastic process such that the queue was idle (state θ). The numerical analysis initialized a few system parameters to compute for some properties of rate matrix, the average queue occupancy L and Little's Law as the basis for the arithmetic computation of other descriptors.

As they attracted more recognition, matrix-geometric techniques to formulate solutions for steady state condition in a "join the shorter" queue M/M/C setup with threshold jockeying permitted were also the subject of a documentation by [4]. To evaluate for the queue size equilibrium probabilities would then necessitate the partitioning of the state space using the transition rate matrix. In contrast to earlier suggestions [64] [194] that a solution for the steady state could not be achieved when the number of queues C exceeded 2, it was proven by the author that the solution lay within the state transition space sub-division method used. It was argued therein that, although as earlier presumed by studies that ergodic conditions could be derived from the sub-division of these state spaces, solutions then did not evaluate for the rate matrix hence the need for revisiting the state space splitting approach. For the proof therefore, sets of sub-levels were defined to map to sets of states and splitting was based on these sub-levels. The splitting was a factor of whether a sub-level's behavior was regular ($l = T, T+1, \ldots$) or otherwise $(1, \ldots, T-1)$, l being a collection of state and T the jockeying threshold. The authors showed that the condition for ergodicity was only possible when the system utilisation $\rho < 1$. Then by applying an earlier theorem [133](1.7.1) to the resultant Markovian generator matrix Q, statistical equilibrium was possible. After categorizing based on sub-levels, the generator matrix Q took a different form that was irreducible but this was solved with the same theorem [133](1.7.1). The stationary vectors were similarly split along categories with respect to the sub-levels to obtain a mapping from stationary probability vectors to sub-levels using the Eqs. (9) and (10).

$$p_l = P_T R^{l-T}, (l < T) \tag{9}$$

$$D_0 + RD_1 + R^2 D_2 = 0 \tag{10}$$

where D_0,D_1 and D_2 denoted square sub-matrices constituting states at all sub-levels $\geq T$ and p_l or p_T as stable probability vectors corresponding to the level l (bound on T) that resulted from the split of the stable probability vector p

Because one of the sub-matrices in the generator Q took different dimensions due to the constituent states in the sub-level set, it was shown how given this difference in structure, a solution for the rate matrix (R) could be expressed. The evaluation derived its reference from suggestions by [146] that by ascertaining the maximum eigenvalue of rate matrix, one could easily express for the solution of R using Eq. (11).

$$R = \begin{pmatrix} 0 \\ w \end{pmatrix} \tag{11}$$

where $w = -v(D_1 + \rho^c D_2)^{-1}$, *($w = w_0, ..., w_{m-1}$), m being number of states at a given level that defined the dimension of the square sub-matrices. c was the number of queues available*

Contrary to the mostly homogeneous setups studied, [6] analysed the behaviour of two heterogeneous queues ($M/M/2$) each with a different service rate to derive expressions for sizes of the queues under stable conditions. New customers joined the shorter of either queues (or based on a probability if the sizes of the two queues were equal) and it was allowed to switch from a longer to shorter one given some jockeying threshold T. The authors proved that when the size of the larger line exceeded the jockeying threshold T, because of the unique formation at the boundaries, the formulation of the solution was the product of the stationary probabilities of each service line. Hence that given the appropriate sub-division of the state space, the matrix-geometric method yielded the same solution. The proof built on the authors' earlier findings [5, 7] where it was shown that with no jockeying allowed between the queues, the equilibrium probabilities ($p_{m,n}$, m, n as queue sizes) of the queue sizes conformed to product-form solutions in infinite time. Therefore, numerical evaluations under set parameter configurations were done to validate whether the same was true when jockeying was permitted. It was found that the evaluation held true only for a defined portion Q ($max(m,n) > T$ and (T,T)) of states. Then observing the rates of change in state of the stochastic process for only this portion of states conclusively categorized the process as irreducible. This necessitated the analysis of this portion of states as a separate process (with distribution $q_{m,n}$) with a relation to the main process denoted by $p_{m,n} = q_{m,n}P(Q)$ (where $P(Q)$ was the probability that the portion Q included the main process). The product-form solution was derived from the notion of defining a set of metrics that were a factor of the arrival rate, the service rate of the shorter queue and the queue admission probability (specifically in case the two queues were of the same size) for which $l > T$ (l as the length of the longer line). The general purpose principle was the basis for the derivation of equations that resolved for these metrics and this system of equations characterized the steady-state and ergodicity conditions for the portion of states Q. The work sought to also draw comparisons between the product-form solution and the one ascertained using the matrix-geometric technique. For the geometric solution, the generator matrix emanated from sub-division of the steady-state probability vector into sub-levels that grouped states depending on the size l of the longer queue ($l < T$ and $l \geq T$) and associating each of these sub-levels to a steady-state probability vector $\overrightarrow{p_l}$. Since some of sub-matrices (A_0, A_1, A_2) in the generator matrix G were irreducible, using existing theorems for ergodicity conditions [133] (Theorem 1.7.1, 1.7.11), $\overrightarrow{p}_l$ when $l > T$ was ascertained from $\overrightarrow{p}_l = \overrightarrow{p}_T R^{l-T}$ (by taking the unique structural properties of A_0 to compute for R from $A_0 + RA_1 + R^2A_2 = 0$ given knowledge of R's maximum eigenvalue). The the matrix-geometric formulation was found to bear similarities to the product-form solution.

More work on a finite capacity $M/M/2$ queuing system was done by [172] where, customers had the chance to change from one queue to another. The arrivals were simulated to imitate a Poisson distribution with rate λ, choosing the shorter of the two service lines each operating with exponentially distributed service rates (μ_1 and μ_2). If none of the service lines was shorter than the other, preference for one would follow probabilities (α or β, $\alpha + \beta = 1$) while instantaneous jockeying was possible when one of the service line was empty. For the transitions in queue lengths sizes that formed a Markovian chain, the author defined a sequence of difference equations that characterized the probability ($P_{i,j}$) of queues (i,j) being in equilibrium state. The compressed form of the difference equations ($A_{k-1}P_{k-1} + B_kP_k + C_{k+1}P_{k+1} = 0, k = 2, 3, 4, ..., L-1$ and $A_{L-1}P_{L-1} + B_LP_L = 0, k = L$) were first re-arranged to compose for a matrix block ($A_1P_1 + B_2P_2 + C_3P_3 = 0$) of sub-matrices ($A, B$ and C) that generalized the system dynamics and traffic intensities. Then the evaluations for the sub-matrices (A, B, C) were computed from the definition of column vectors that encoded state transition probabilities over all positions in both queues. Theoretically, the proof required that the sub-matrix B_k was invertible and this meant computing the inverse of the sub-matrix and its determinants. There existed an inverse of this sub-matrix only if its determinant did not evaluate to 0 for all values of the traffic intensity ρ. Based on P_k^+ being the theoretical solution for stability conditions, $A_{L-1}P_{L-1} + B_LP_L = 0, k = L$ expressed for this solution as the probability that the system was occupied to maximum capacity ($P_L = -R_LA_{L-1}P_{L-1}$, $k = L$) given that $R_L = B_L^{-1}$. Iteration of computations for R evolved into solutions for the differential equations expressed in terms of $P_{0,0}$ (probability that both queues were idle). The proof by induction on k was required to further verify the relation $g_k = \rho^{k-2}g_2$ for $k = 2, 3, ..., 2L$ (given $g_k = Pr(N = k)$; $N = N_1 + N_2$ and N_i, $i = 1, 2$ *were the number of customers in either queue*) that formulated for the equilibrium probabilities as the M/M/2 system capacity was doubled (*2L*) or as $L \implies \infty$. A numerical analysis included setting the arrival and service rates to different values and comparisons made with earlier results from the Conolly's model[37]. Performance evaluations for the effect of the system utilization on the equilibrium probability g_n under different queue setups were illustrated to show how the author's model was quantitatively better.

Bulk workload migration received further attention in a setup of two $M/M/1$ queuing servers when [80] considered the variation in the sizes of the two service lines (q_1, q_2) hitting a preset threshold L as the trigger for moving some workload $K(0 < K < L)$ to the shorter queue. New Poisson distributed arrivals (at rate λ_i) joined service queues that run at exponentially distributed processing rates (μ_i; $i = 1, 2$) to yield a QBD process $((q_1(t), q_2(t)), t \geq 0)$ with state space $(n_1, n_2) : n_1 \geq 0$, $|n_1 - n_2| < L$ ($n_i, i = 1, 2$ as the number of customers in a given queue) Then based on the inherent recurrence (the probability that the system will revisit an earlier state in finite time) properties of such processes, it was theorized and proven [80, 101] that steady-state conditions could only exist under specific conditions of the traffic intensity ($\rho < 1$). The derivation adopted findings from [56] where a state mapping $f(n_1, n_2)$ (Lyapunov function) related the state of each queue; this mapping was then the basis for the computation of the mean drift as an aggregation of the process generator matrices Q in state space followed by a proof for necessity. Although the structure of the stochastic process made it hard to analyse, the fact that the process constituted special properties ($q_1 - q_2 \ngtr L - 1$) allowed for the sub-division of the state space $\{(q_1(t), (q_2(t)), t \geq 0\}$ to re-organise into a valid QBD to which matrix-geometric techniques could be applied for a solution to the equilibrium probabilities. This state space was re-organised based on the system occupancy $q(t) = q_1 + q_2$ and the deviation between the sizes of the queues $(j(t) = q_1 - q_2)$ at any time t. This iteration was done until the resultant stochastic process $(X(t), J(t), t \geq 0)$ (with its corresponding generator matrix Q_2) was irreducible and not dependent on the number of customers (system occupancy). The equilibrium probabilities for this process were then formulated for using matrix-geometric methods based on earlier solutions ($A_0 + RA_1 + R^2A_2 = 0$) [133],[110] that entailed resolving for the eigenvalues (with the largest modulus - Perron-Frobesius eigenvalue) of the rate matrices R at the different levels (L) under the assumption that the process was constituent of recurrence properties. And these probabilities formed the basis for proof for stability conditions for system descriptors like likelihood that a queue was idle, number of serviced units, estimates on the number of consumers in a queue under steady conditions, how often workload was transferred ($T_{R,1->2}, T_{R,2->1}$) within the system, etc. Expressions for the rate at which the distribution of system occupancy diminished were formulated and it was shown that this decay rate was neither affected by the workload transfer threshold L nor the number of workload migrations K under varying inequalities of the traffic intensity ρ. Then based on theoretical assumptions on the structural (like spectral radius) properties of the non-negative matrices like R, an expression for the rate matrix and eigenvectors for the corresponding eigenvalues were also derived. The analytic solution was validated by numerical evaluations that involved experimentation with variations in queue design parameters. For example, the case of the first queue with $\lambda_1 = 1, \lambda_2 = 2, \mu_1 + \mu_2 = 1 = 4, L = 5, K = 3$, it was observed that as the service rate increased, the rate at which customers were transferred to the second queue decreased and vice-versa. This revealed the inherent convexity properties of the relationship between the service and transfer rates. It was conclusively suggested that, choosing the right processing rate for each queue was requisite for even load distribution ($\rho_1 \approx \rho_2$) so as to keep transfers within the system at reasonable minimum quantities and that the parameter that exerted much influence on these decisions was the deviation in queue utilization (traffic intensity). This was given the fact that the transfer rates exhibited monotonicity properties in the processing rate.

5 BEHAVIORAL MODELS

5.1 Modeling based on the Value of Information

Time critical applications whose output depends highly on freshness of information in a MEC 5G network environment motivated [73] to study the impatience of customers who could either renege or balk (with some semblance to jockeying) from an $G/G/1/\infty$ FCFS queue system. The customers had the option to continuously

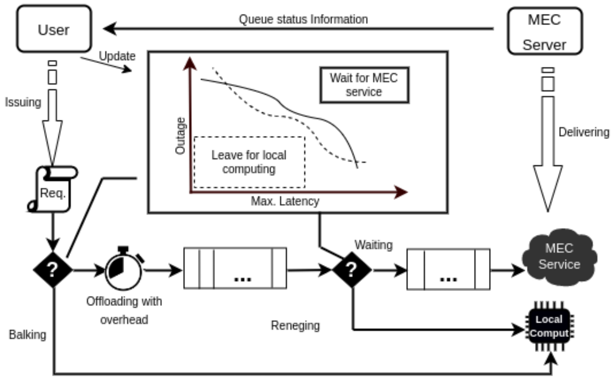

Fig. 2. An schematic depiction of the impatient customer that had the option to either process a task on a MEC server or locally depending on the latency critical requirements of the underlying application

weigh the risk related to either process jobs locally (on their devices) or forward them to a cloud server as depicted in Figure 2. Depending on the latency requirements of the requesting application, the approach referenced information like expected cloud server response times τ_c in terms of the task forwarding overhead (τ_s), the predicted waiting time ($\tau_{w,k}$) of a user landing in position k at joining time $t > 0$ etc to influence the decision as to which platform the computations of the task at hand took place. A utility model was first developed for the latency-critical tasks where customers got rewards from the tasks that were successfully processed and the reward (u) decreased to null with the total delay Δt. This delay varied depending on the choice for platform and the user got access to this kind of information for a given task beforehand. While at the MEC server, the user then had the ability to assess the risk (P_o) associated with continuing with the cloud-based processing given the ever changing network/compute resource delays so as to make decisions that maximized the expected reward and minimize the risks. It was shown in this case that a user that submitted a task to a queue in the cloud did not retrieve the task from the queue when provided with reliable and flawless channel information. The proof followed from the fact ([73], Lemmas 1-2) that there existed a time, given the task forwarding overhead to be incurred, that the risk P_o associated to the predicted cloud delay was less than that associated with remaining local delay. This implied that the customer could not leave the queue immediately after joining it (t_1) and because the predicted remaining latency diminished as more customers that were infront in the queue got processed, it was argued that the customer could not also retrieve the task at time t_2. It was also necessary to understand customers' reaction when availed with flawed information about the state of the channels under the hypothesis that may be the customers could develop their own knowledge from experience and use this knowledge to make the necessary choices for a compute platform. The error arising from this knowledge validations and estimation for the rates was computed. And it was argued that the margin of this error directly affected the predictions for the risk associated with preferences for processing platform such that customers would end up either being reluctant or hasty. A numerical study of the impatient customer's regretting behaviour under correct or partially correct system status information to evaluate the predicted loss in reward, learning gain etc using a risk analysis algorithm was performed. The results provided interesting insights into the influence of queue status knowledge on a customer's decision for preference of a processing platform and on the overall system performance in terms of the response times under changing conditions.

Extended analysis of the value of information and its applicability in balancing the load within distributed compute systems was the subject of [141]'s work. Here, the state of a node's knowledge about prevailing queue setup was fundamental for admitting tasks to the queues and transferring them around to alternative queues. The nodes cooperated on task executions by migrating excess load (total system load minus load on node) to other nodes. Eq. (12) (the case of $n \geq 3$ for example) was definitive of the excess load partitioning and distribution to $n-1$ nodes. And transfer of this excess load depended on the size of the partition such that the least loaded node received the bigger partition.

$$p_{ij} = \begin{cases} \frac{1}{n-2}\left(1 - \frac{\lambda_{d_i}^{-1} Q_i(t-\eta_{ji})}{\sum_{l\neq j}\lambda_{d_l}^{-1} Q_l(t-\eta_{jl})}\right), & \sum_{l\neq j} Q_l(t-\eta_{jl}) > 0 \\ \frac{\lambda_{d_i}}{\sum_{k\neq j}\lambda_{d_k}}, & otherwise, \end{cases} \tag{12}$$

where η_{jl} was the expected lag when node l and node j communicated, λ_{d_i} was the rate at which a countable number of tasks departed the queue i. Also λ_{d_k} denoted the departure rate at preset values of the k^{th} load balancing instant. And $Q_l(t-\eta_{jl})$ was node j's assumption about the number of tasks running on node l which depended on the communication lag not exceeding time t.

The state of the knowledge local to nodes then required prior broadcasting of buffer sizes by all nodes within the cluster. And the load balancing algorithm resident on each node was executed before accepting any collaboration in the task processing. Using the regeneration approach, a random variable τ was defined (Eq. (**??**)) as a regeneration time to denote events such as the time a task was completed. That, the recurrence of similar events should characterize for stochastic behavior similar to previous events though under different environment settings. Two load balancing policies were evaluated, i.e. centralized one-shot and dynamic load balancing. For the centralized one-shots rule-set, the proof evolved from adoption of principles of conditional expectation and regeneration-event decomposition to express for the average overall completion time (AOCT) using the resultant difference equations. The centralized one-shots rule-set was extended for distributed environments as a sender-initiated dynamic load balancing (DLB) algorithm that morphed to meet the dynamic processing speeds and delay of the infrastructure. Also, each node embedded an optimal load-balancing instant and gain measure (unlike in the one-shot policy where all nodes received the same) such that load was redistributed to achieve system wide reduction in completion time based on the up-to-date status information. Therefore, the DLB heuristic's objective functions sought to minimize the overall processing time with respect to the knowledge states, load balancing instant and gain measures. For the one-shot centralized policy, experiments (two servers) to optimize the overall time to complete (AOCT) tasks showed that load balancing actions taken when t_b increased beyond one second evolved into the slower node carrying more load. Hence the larger measures in the AOCT due to delayed updates to the knowledge state coupled with keeping the faster server idle for some time. The performance of the DLB policy on the other hand was evaluated in terms of the time it took to complete a given task (ACTT as a combination of processing, queueing and transfer time) within a defined time frame. In conclusion, analysis of the two qualitative measures system processing rate (SPR) and mean task completion time (ACTT) for both policies under difference configurations in parameter K led to the generalization that in either policies, improvements in SPR were recorded under lower measures in K (but more transfer activity in higher measures of k or excessive load migration delays that led to higher ACTT) for the static load balancing policy (one-short) while the DLB policy yielded lower queueing transfer delays to reduce the ACTT. Further benchmarking of the proposed policies relative to classical DLB policies like Shortest-Expected-Delay (SED) and Never-Queue (NQ) revealed measurable improvements in the ACTT with the DLB over NQ and SED.

In a shift from centralized to decentralized control in multi-tenancy MEC environments, [97]'s pioneering work in behavioral modeling of impatience in queueing systems sought to understand the benefits switching queues brings to the impatient tenant. It is argued that the state space curse in stochastic models of jockeying coupled with the centralized control of the behavior might not be practical given the dynamics inherent in next generation communication systems. That for decentralized management, the rationale to move workload from one queue to another should be made by the individual tenants after assessing the up-to-date availed information about the expected waiting time. The Monte Carlo findings assumed a setup of network slices arranged as queues ($M/M/C, C = 2$), such that arrivals that obeyed a Poisson distribution (with rate λ) joined the shorter of the two heterogeneous buffer lines given prior knowledge about their lengths. Tasks were processed at exponentially distributed service times and at each completion of a given task in either queue, tenants evaluated whether to stay in their current position(s) k or to jockey to the alternative buffer line. The jockeying here disregarded whether jockeying was to the shorter queue but the rational was premised on the expected waiting time that the jockeyed task would take. However, the position of the jockey in the alternate queue was a factor of what portion of the new arrivals $\beta \leq \lambda$ would prefer the same queue as the jockey and to obscure this competitiveness, the jockey's final position followed a shuffle operation with the portion β of new arrivals. The work assumed at least a single departure $N \geq 1$ occurred such that if $\beta \leq \lambda$ sought to join the preferred queue at that point in time, switching buffers was only if the expected waiting time in the preferred queue (at position τ) was less than when the tenant stayed put (at position k). (13) was definitive of these sojourn time-position dependencies.

$$F^{\overrightarrow{ij}}_{T_w|\tau}(t_w|\tau) = \begin{cases} T_w \quad | \quad P^{\beta}_{Q_{i,j}}(t+1) & \text{if } \beta, N \geq 1 \\ T_w \quad | \quad P(N \geq 1) & \text{if } \beta = 0 \\ T_w & \text{if } \beta, n = 0 \end{cases} \tag{13}$$

where T_w was the expected waiting time and $P^{\beta}_{Q_{i,j}}(t+1)$ denoted the probability that β new arrivals joined the preferred queue at $t + 1$ when the jockey decision was to be taken.

Formulations for the number of times that tenants switched from one queue to another then followed from the adoption of principles of conditional probability theory (Baye's theorem) to resolve for the dependencies between the expected waiting time, departures and new arrivals. It was shown therein that, from the Monte Carlo simulations, the sensitivity of the impatient tenant to perturbations in the system descriptors could first be assessed to extract correlations or dependencies that could guide the rationale to jockey. The results from the numerical evaluations of the model were a further revelation about the positive impact of switching buffers such that, the tenants that jockeyed more than once ended up waiting less until service in comparison to tenants that did not jockey at all.

5.2 Artificial Neural Networks modeling

The application of jockeying in Facility location problems (FLP) problems was investigated in [31] studied a cooperative multi-layered setup of queues where jobs were distributed to empty service lines within the hierarchical layers based on certain rules that considered job priorities. Jockeying had mostly been discouraged given the presumable associated costs and complexity that the behavior presented forthwith. The authors were interested in dynamically distributing the facilities in each layer effectively so as to meet the emerging demand in small and large systems. A job was processed through each layer by those facilities in closest proximity to the job's location and the optimal solution was determining the set of facilities that could partake in the servicing of the job request. The demand for service queues followed a Poisson distribution, the service rates at each facility were exponentially distributed and each facility in a given layer participated in the processing of a jockeyed applicant. Starting with a solution for small-scale systems, expressions that characterized for the objective functions (reduce jockeying plus mean waiting times and keeping all facilities busy) were formulated. The applicability of the augmented ϵ-constraint method when evaluating for global solutions (Pareto-Optimal) to multi-objective non-linear models was restricted to small scale scenarios. While for medium to large scale service systems, an Non-Dominated Sorting Genetic Algorithm (NSGA)-II was deployed and a technique called non-dominated sorting used to rank each entity. This evolutionary algorithm included a sequence of operators in an iterative process that evaluated each chromosome for eligibility to participate in parenting the next generation of off-springs. The Taguchi scheme was found relevant for the adjustment of input parameters used for the initialization of the population given the effect these parameters had on the genetic algorithms. The characteristic behaviour of each entity in the population and system components like layers or facilities was abstracted as chromosomes to which constraints ($g(x)$ where x was a chromosome) and cost functions were pegged. This aided the evaluations for how worth a chromosome was for parenting the next generation by associating the chromosome to penalties defined by $p(x) = U * Max\{0, \frac{g(x)}{b} - 1\}$ *such that* U *was a constant,* $g(x)$ *the constraint and* $p(x)$ *the punishment on chromosome* x

The final mating population evolved from the iterative application of conventional genetic algorithm operations (like selection, cross-over, mutation and offspring evaluation) until the fitness function quantitatively yielded no better results for a series of selection runs. The authors concluded the investigations with a numerical analysis of the model by setting up a manufacturing system and results were documented when jockeying was permitted from one facility to another within the the layers versus when applicants were processed by a single facility. The performance measurements confirmed that not only did allowing switching from one facility to another reduce the waiting and idle time of the facilities but the behaviour also ensured that the job transfer overhead was minimized given that only facilities within the proximity of a potential jockey were those considered to participate in a job's sub-processing.

6 DISCUSSION, CONCLUSION AND FUTURE WORK

6.1 Discussion and Conclusion

In the context of resource allocation in MEC or 5G and Beyond communication systems, where it has been proposed that application requirements will be mapped categorically to specific network slice configurations, jockeying is one those queuing theory concept that could accelerate flexibility for multi-vendor resource sharing [57]. And because different vendors will provide varying costings for the services they provide, giving the consumer the ability to select from the pool of available resources will mitigate the expected impact of diminishing availability while ensuring optimal usage of the communication channels. From these preferences of one queue over another emerge new problems that relate to modeling for the dynamics introduced by the tenants' impatience. It is common practice to generalize these dynamics as stochastic processes in nature with definitive assumptions that bound the measures in buffer descriptors. One of the main assumptions adopted in most Markovian models is that the new arrivals benefit from prior access to some information [18, 134] and the strategy is to join the shorter queue [6, 147]. It has been however argued that this strategy might not be optimal given the existential differences in queue capacities and workload [179]. However, technical mechanisms for providing tenants with access to this buffer status information have inspired proposals for broadcasting of or subscription to this information in dedicated or shared communication channels [73].

The second dominant assumption about the service discipline queues is the FCFS. However, practitioners argue that mission critical implementations like rescue (where tasks urgently need to be prioritized by jockeying) render the FCFS approach less viable given the strict latency constraints in such operations [104, 106].

Another assumption is modeling the setups as homogeneous queueing systems which is a limitation to there applicability in communication systems. Yet, heterogeneity in these kind of systems is limited by the use of preset difference in buffer size based jockeying thresholds as the criteria for the task migration behavior. Moreover, limiting the jockeying to occur from the shorter buffer to the longer one raises concerns about such thresholds as the optimal trigger for switching buffers. To achieve some balance, some MEC setups embed traffic classification routines [163] to prioritize packet flows but limited information exposed to the flows still keeps the behavioral control mechanisms centralized, which further limits optimizing the impatience behavior. So literally, best engineering practice would require decentralizing the decision making process [157] such that the rational consumer has access to up-to-date queue descriptor information like waiting time [97] or billing, subscription costs metrics from the Network Slice Selection Function (NSSF) in the network core, etc. Earlier attempts that close this information gap propose techniques like active queue management [35, 83, 100] or information signaling in value of information setups [76, 191]. Other studies incorporate neural modeling as input to guide the queue selection process and jockeying behaviour [145, 170].

6.2 Future Work

Decentralized control versus centralized control: The irrationality of the impatient consumer in distributed compute environments has rendered less viable canonical statistical approaches which assume centralized control of the routing in communication networks. The diverse state space introduced by the underlying uncertainty and complexity of these approaches has inspired recent efforts towards decentralized control (i.e. moving this control to the consumer), a technique referred to as behavioral modeling [157]. Proponents for this methodology suggest that emphasis be placed on the rationality of the impatient tenant for autonomous and proactive decision making in dynamic systems like next generation networks. This autonomy should however be underpinned by access to up-to-date information about the system [15, 97, 161]; Technically, broadcasting this information versus subscription to it (as an incentive for premium lines) in a kind of publish-subscribe mechanism becomes the question [73, 173]. For example, in new paradigms like network slicing as a service (SlaaS) [72], the performance metrics from network core's virtual network functions (like NSSF) are expected to be a source of vital knowledge to guide the slice admission or switching heuristics. However, disseminating up-to-date information in congested environments introduces extra communication overhead. This implies, to achieve leverage under such impatience manifestations, it is important to evaluate what value broadcasting queue status information [15, 84, 161] brings. Therefore, future work should provide extensive evaluation of the bulk of overhead introduced by this information exchange or task offloading [141] and the resultant effect on system performance.

Worth more studying too is the effect of setting low quantities of the jockeying threshold as this could evolve into a Ping-Pong scenario where impatient tenants continuously switch from one queue to another. This kind of jockeying behavior can be hard to differentiate from security breaches like denial of service attacks. A trusted and authenticated tenant could also be the source of corrupted queue status information for it's selfish benefit.

When designing the buffer preference policies especially for task migration in distributed systems, the time it takes to move a task (inter-transfer time) is dependent on the task size, underlying infrastructure conditions, the number of network hops that exist between any two or more collaborating devices [61, 121]. The effect of such dynamics on the routing decisions appears an open area for research studies.

The Value of Information: Trending in most scientific domains is the adoption of artificial neural networks for the computational modeling of complex phenomena [118]. Their learning prowess has motivated studies that instead of statistically quantifying for selected queueing descriptors, these data-driven methods make deductive inferences that underpin automation processes [14, 145]. For example, in routing and scheduling algorithms as packet transmission rules for congestion control in edge compute systems [49, 54]. Besides their predictive capabilities, neural networks also define for ways to obviate the high dimensionality curse suffered by stochastic models and have been deployed as viable sources of valuable queue status information to guide queue operations. Hence, the neural networks as a source of valuable information in behavioral modeling then becomes a worthwhile subject.

In conclusion, the selfish drive for maximizing profit or minimizing time to task completion arouses exhaustive scrutiny of the behavior of these impatient tenants. That is, how fair the adopted jockeying policies are to other buffer occupants. For organizations, prioritization versus fairness (fairness takes into account both the arrival time, how long the job has been queued and the time the job takes to process) [149, 150] considerations are important such that workload can be distributed depending on the application use case. There is therefore need for further evaluation of metrics like RAQFM (Resource Allocation Queueing Fairness Measure) or Credit Based Shaper (CBS) [20] which provide measurable statistics about the fairness of the servers when processing jobs of varying sizes under different buffer sharing policies [148]. Lastly, to counteract unnecessary impatience in these systems, stochastic optimization techniques could provide definitive limits on the number of times tasks are jockeyed such that under certain system conditions agents are full aware of the level of risk associated to the jockeying behavior.

ACKNOWLEDGMENTS

This work is supported in part by the German Federal Ministry of Education and Research (BMBF) within the project **Open6GHub** under grant numbers 16KISK003K and 16KISK004.

REFERENCES

[1] 3GPP. 2016. *System architecture for the 5G System (5GS).* Technical Specification (TS) 23.501-15. 3rd Generation Partnership Project (3GPP).

[2] Md. Abu Baker Siddiki Abir, Mostafa Zaman Chowdhury, and Yeong Min Jang. 2023. A Software-Defined UAV Network Using Queueing Model. *IEEE Access* 11 (2023), 91423–91440. https://doi.org/10.1109/ACCESS.2023.3281421

[3] Mohammad Abu Alsheikh, Dinh Thai Hoang, Dusit Niyato, Hwee-Pink Tan, and Shaowei Lin. 2015. Markov Decision Processes With Applications in Wireless Sensor Networks: A Survey. *IEEE Communications Surveys and Tutorials* 17, 3 (2015), 1239–1267. https://doi.org/10.1109/COMST.2015.2420686

[4] Ivo Adan, J. Wessels, and W. Henk M. Zijm. 1993. Matrix-geometric analysis of the shortest queue problem with threshold jockeying. *Operations Research Letters* 13 (03 1993), 107–112. https://doi.org/10.1016/0167-6377(93)90037-H

[5] Ivo J. B. F. Adan, Wessels Jaap., and W. Henk M. Zijm. 1991. Analysis of the Asymmetric Shortest Queue Problem. *Queueing Syst. Theory Appl.* 8, 1 (2 1991), 1–58. https://doi.org/10.1007/BF02412240

[6] Ivo J. B. F. Adan, Jaap Wessels, and W. Henk M. Zijm. 1991. Analysis of the asymmetric shortest queue problem with threshold jockeying. *Communications in Statistics. Stochastic Models* 7, 4 (1991), 615–627. https://doi.org/10.1080/15326349108807209

[7] J. B. F. Ivo Adan, Wessels Jaap, and W. Henk M. Zijm. 1990. Analysis of the symmetric shortest queue problem. *Communications in Statistics. Part C, Stochastic Models* 6 (1990), 691–713. https://doi.org/10.1080/15326349908807169

[8] Anders Ahlen, Johan Akerberg, Markus Eriksson, Alf J. Isaksson, Takuya Iwaki, Karl Henrik Johansson, Steffi Knorn, Thomas Lindh, and Henrik Sandberg. 2019. Toward Wireless Control in Industrial Process Automation: A Case Study at a Paper Mill. *IEEE Control Systems Magazine* 39, 5 (2019), 36–57. https://doi.org/10.1109/MCS.2019.2925226

[9] Sassan Ahmadi. 2019. *5G NR: Architecture, technology, implementation, and operation of 3GPP new radio standards.* Academic Press. 1–194 pages.

[10] Soohan Ahn and V. Ramaswami. 2006. Matrix-geometric algorithms for stochastic fluid flows. In *Proceeding from the 2006 Workshop on Tools for Solving Structured Markov Chains (SMCtools '06).* Association for Computing Machinery, 11–es. https://doi.org/10.1145/1190366.1190376

[11] Moayad Aloqaily, Venkatraman Balasubramanian, Faisal Zaman, Ismaeel Al Ridhawi, and Yaser Jararweh. 2018. Congestion Mitigation in Densely Crowded Environments for Augmenting QoS in Vehicular Clouds. In *Proceedings of the 8th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet'18).* Association for Computing Machinery, 49–56. https://doi.org/10.1145/3272036.3272038

[12] Clinton J. Ancker and A. V. Gafarian. 1963. Some Queuing Problems with Balking and Reneging. II. *Operations Research* 11, 6 (1963), 928–937. https://doi.org/10.1287/opre.11.6.928

[13] Arnon Arazi, Eshel Ben-Jacob, and Uri Yechiali. 2004. Bridging genetic networks and queueing theory. *Physica A: Statistical Mechanics and its Applications* 332 (2004), 585–616.

[14] Alex Aussem, Antoine Mahul, and Raymond Marie. 2000. Queueing network modelling with distributed neural networks for service quality estimation in B-ISDN networks. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*, Vol. 5. IEEE, 392–397.

[15] A. Aussem, A. Mahul, and R. Marie. 2000. Queueing network modelling with distributed neural networks for service quality estimation in B-ISDN networks. 5 (2000), 392–397 vol.5. https://doi.org/10.1109/IJCNN.2000.861501

[16] Alcardo Alex Barakabitze, Arslan Ahmad, Rashid Mijumbi, and Andrew Hines. 2020. 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. *Computer Networks* 167 (2020), 106984.

[17] D. Y. Barrer. 1959. Queues, Inventories and Maintenance (Philip M. Morse). *SIAM Rev.* 1, 2 (1959), 186–187. https://doi.org/10.1137/1001042

[18] Ward Whitt Benjamin Melamed. 1990. On Arrivals That See Time Averages. *Operations Research* 38 (1990), 156–172. Issue 1. https://doi.org/10.1287/opre.38.1.156

[19] Jon C. R. Bennett and Hui Zhang. 1996. WF/sup 2/Q: worst-case fair weighted fair queueing. *Proceedings of IEEE INFOCOM '96. Conference on Computer Communications* 1 (1996), 120–128 vol.1.

[20] B. Bensaou, D.H.K. Tsang, and King Tung Chan. 2001. Credit-based fair queueing (CBFQ): a simple service-scheduling algorithm for packet-switched networks. *IEEE/ACM Transactions on Networking* 9, 5 (2001), 591–604. https://doi.org/10.1109/90.958328

[21] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems* 24 (2011).

[22] Jyoti Bisht and Venkata Subrahmanyam Vampugani. 2022. Load and cost-aware min-min workflow scheduling algorithm for heterogeneous resources in fog, cloud, and edge scenarios. *International Journal of Cloud Applications and Computing (IJCAC)* 12, 1 (2022), 1–20.

[23] B. L. Bodnar and A. C. Liu. 1989. Modeling and Performance Analysis of Single-Bus Tightly-Coupled Multiprocessors. *IEEE Trans. Comput.* 38, 3 (mar 1989), 464–470. https://doi.org/10.1109/12.21134

[24] Christos Bouras, Anastasia Kollia, and Andreas Papazois. 2017. SDN and NFV in 5G: Advancements and challenges. In *2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN).* 107–111. https://doi.org/10.1109/ICIN.2017.7899398

[25] Alexandre Brandwajn and Thomas Begin. 2019. First-come-first-served queues with multiple servers and customer classes. *Perform. Eval.* 130, C (apr 2019), 51–63. https://doi.org/10.1016/j.peva.2018.11.001

[26] James Broberg, Zahir Tari, and Panlop Zeephongsekul. 2004. Task Assignment Based on Prioritising Traffic Flows. *Proceedings of the Eighth International Conference on Principles of Distributed Systems* 3544 (12 2004), 415–430. https://doi.org/10.1007/11516798_30

[27] Mark Campbell, Magnus Egerstedt, Jonathan P How, and Richard M Murray. 2010. Autonomous driving in urban environments: approaches, lessons and challenges. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 368, 1928 (2010), 4649–4672.

[28] Zhiruo Cao, Zheng Wang, and Ellen Zegura. 2000. Rainbow fair queueing: Fair bandwidth sharing without per-flow state. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, Vol. 2. IEEE, 922–931.

[29] Delphine Caruelle, Line Lervik-Olsen, and Anders Gustafsson. 2023. The clock is ticking—Or is it? Customer satisfaction response to waiting shorter vs. longer than expected during a service encounter. *Journal of Retailing* 99, 2 (2023), 247–264. https://doi.org/10.1016/j.jretai.2023.03.003

[30] Sofia Ceppi, Nicola Gatti, Giorgio Patrini, and Marco Rocco. 2010. Local Search Methods for Finding a Nash Equilibrium in Two-Player Games. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Vol. 2. 335–342. https://doi.org/10.1109/WI-IAT.2010.57

[31] Amir Eshaghi Chaleshtori, Hamed Jahani, and Abdollah Aghaie. 2020. Bi-objective optimization approach to a multi-layer location–allocation problem with jockeying. *Computers and Industrial Engineering* 149 (2020), 106740. https://doi.org/10.1016/j.cie.2020.106740

[32] Ying Chen, Ning Zhang, Yongchao Zhang, Xin Chen, Wen Wu, and Xuemin Sherman Shen. 2021. TOFFEE: Task Offloading and Frequency Scaling for Energy Efficiency of Mobile Devices in Mobile Edge Computing. *IEEE Transactions on Cloud Computing* 9, 4 (2021), 1634–1644. https://doi.org/10.1109/TCC.2019.2923692

[33] J. Choi, M. Naghshineh, Y. Choi, and T. Kwon. 2000. Call Admission Control for Multimedia Services in Mobile Cellular Networks: A Markov Decision Approach. In *Proceedings of 5th IEEE Symposium on Computer and Communications (ISCC 2000).* IEEE Computer Society, 594. https://doi.org/10.1109/ISCC.2000.860701

[34] Gautam Choudhury and Kailash C Madan. 2004. A two phase batch arrival queueing system with a vacation time under Bernoulli schedule. *Appl. Math. Comput.* 149, 2 (2004), 337–349.

[35] Jae Chung and M. Claypool. 2003. Analysis of active queue management. In *Second IEEE International Symposium on Network Computing and Applications, 2003. NCA 2003.* 359–366. https://doi.org/10.1109/NCA.2003.1201176

[36] T. Collings and C. Stoneman. 1976. The $M/M/\infty$ Queue with Varying Arrival and Departure Rates. *Operations Research* 24, 4 (1976), 760–773.

[37] Brian W. Conolly. 1984. "The Autostrada Queueing Problem." Journal of Applied Probability. *JSTOR* 21, no. 2 (1984), 394–403. https://doi.org/10.2307/3213648

[38] Natalie A Cookson, William H Mather, Tal Danino, Octavio Mondragón-Palomino, Ruth J Williams, Lev S Tsimring, and Jeff Hasty. 2011. Queueing up for enzymatic processing: correlated signaling through coupled degradation. *Molecular systems biology* 7, 1 (2011), 561.

[39] Jim G. Dai. 1995. On Positive Harris Recurrence of Multiclass Queueing Networks: A Unified Approach Via Fluid Limit Models. *The Annals of Applied Probability* 5, 1 (1995), 49–77.

[40] Constantinos Daskalakis, Paul Goldberg, and Christos Papadimitriou. 2006. The complexity of computing a Nash equilibrium. *SIAM J. Comput.* 39, 71–78. https://doi.org/10.1145/1461928.1461951

[41] Wladimir Gonçalves de Morais, Carlos Eduardo Maffini Santos, and Carlos Marcelo Pedroso. 2022. Application of active queue management for real-time adaptive video streaming. *Telecommunication Systems* 79 (02 2022), 261–270. Issue 2. https://doi.org/10.

1007/s11235-021-00848-0

[42] Amin Dehghanian, Jeffrey P. Kharoufeh, and Mohammad Modarres. 2016. Strategic dynamic jockeying between two parallel queues. *Probability in the Engineering and Informational Sciences* 30, 1 (2016), 41–60. https://doi.org/10.1017/S0269964815000273

[43] Rosario Delgado and Evsey Morozov. 2014. Stability analysis of cascade networks via fluid models. *Performance Evaluation* 82 (2014), 39–54. https://doi.org/10.1016/j.peva.2014.10.001

[44] Rosario Delgado and Evsey V. Morozov. 2014. Stability Analysis of Some Networks with Interacting Servers. *International Conference on Analytical and Stochastic Modeling Techniques and Applications* ASMTA 2014 (2014), 1–15. https://doi.org/10.1007/978-3-319-08219-6_1

[45] A. Demers, S. Keshav, and S. Shenker. 1989. Analysis and simulation of a fair queueing algorithm. *SIGCOMM Comput. Commun. Rev.* 19, 4 (09 1989), 1–12. https://doi.org/10.1145/75247.75248

[46] William E. DISNEY, Ralph L; MITCHELL. 1970. A Solution for Queues with Instantaneous Jockeying and Other Customer Selection Rules. *Naval Research Logistics* 17 (9 1970), 315–325. https://doi.org/10.1002/nav.3800170308

[47] Douglas G. Down and Mark E. Lewis. 2006. Dynamic load balancing in parallel queueing systems: Stability and optimal control. *European Journal of Operational Research* 168, 2 (2006), 509–519. https://doi.org/10.1016/j.ejor.2004.04.041 Feature Cluster on Mathematical Finance and Risk Management.

[48] M. Dubois and C. Scheurich. 1990. Memory access dependencies in shared-memory multiprocessors. *IEEE Transactions on Software Engineering* 16, 6 (1990), 660–673. https://doi.org/10.1109/32.55094

[49] Dmitry Efrosinin, Vladimir M. Vishnevsky, and Natalia V. Stepanova. 2023. Optimal Scheduling in General Multi-Queue System by Combining Simulation and Neural Network Techniques. *Sensors (Basel, Switzerland)* 23 (2023).

[50] Salah Eddine Elayoubi, Sana Ben Jemaa, Zwi Altman, and Ana Galindo-Serrano. 2019. 5G RAN slicing for verticals: Enablers and challenges. *IEEE Communications Magazine* 57, 1 (2019), 28–34.

[51] Elsayed A. Elsayed and Ali S. Bastani. 1985. General solutions of the jockeying problem. *European Journal of Operational Research* 22, 3 (1985), 387–396. https://doi.org/10.1016/0377-2217(85)90258-9

[52] Anwar I. Elwalid and Debasis Mitra. 1991. Analysis and design of rate-based congestion control of high speed networks, I: stochastic fluid models, access regulation. *Queueing Systems* 9, 1 (03 1991), 29–63. https://doi.org/10.1007/BF01158791

[53] D.J Evans and W.U.N Butt. 1993. Dynamic load balancing using task-transfer probabilities. *Parallel Comput.* 19, 8 (1993), 897–916. https://doi.org/10.1016/0167-8191(93)90073-T

[54] Hassan Fawaz, Djamal Zeghlache, Quang Tran Anh Pham, Leguay Jérémie, and Medagliani Paolo. 2021. Deep reinforcement learning for smart queue management. In *NETSYS 2021: Conference on Networked Systems 2021 (Conference on Networked Systems 2021 (NetSys 2021), Vol. 80)*. TU Berlin, Universitätsbibliothek TU Berlin, 1–14. https://doi.org/10.14279/tuj.eceasst.80.1139

[55] Huei-Wen Ferng. 2001. Departure processes of BMAP/G/1 queues. *Queueing Systems* 39 (10 2001), 109–135. https://doi.org/10.1023/A:1012786932415

[56] Robert D. Foley and David R. McDonald. 2001. Join the Shortest Queue: Stability and Exact Asymptotics. *The Annals of Applied Probability* 11, 3 (2001), 569–607.

[57] X. Foukas, A. Elmokashfi, G Patounas, and MK Marina. 2017. Network Slicing in 5G: Survey and Challenges. *IEEE Communications Magazine* 55, 5 (2017), 94–100. https://doi.org/10.1109/MCOM.2017.1600951

[58] Hasna Fourati, Rihab Maaloul, Lamia Chaari, and Mohamed Jmaiel. 2021. Comprehensive survey on self-organizing cellular network approaches applied to 5G networks. *Computer Networks* 199 (2021), 108435.

[59] Jing Fu, Jun Guo, Eric W. M. Wong, and Moshe Zukerman. 2015. Energy-efficient heuristics for job assignment in processor-sharing server farms. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. 882–890. https://doi.org/10.1109/INFOCOM.2015.7218459

[60] Tom Z.J. Fu, Jianbing Ding, Richard T.B. Ma, Marianne Winslett, Yin Yang, and Zhenjie Zhang. 2015. DRS: Dynamic Resource Scheduling for Real-Time Analytics over Fast Streams. (2015), 411–420. https://doi.org/10.1109/ICDCS.2015.49

[61] R. Gaeta, M. Gribaudo, D. Manini, and M. Sereno. 2006. Analysis of resource transfers in peer-to-peer file sharing applications using fluid models. *Performance Evaluation* 63, 3 (2006), 149–174. https://doi.org/10.1016/j.peva.2005.01.001 P2P Computing Systems.

[62] Sixiao Gao, Jose I. U. Rubrico, Toshimitsu Higashi, Toyokazu Kobayashi, Kosuke Taneda, and Jun Ota. 2019. Efficient Throughput Analysis of Production Lines Based on Modular Queues. *IEEE Access* 7 (2019), 95314–95326. https://doi.org/10.1109/ACCESS.2019.2928309

[63] David Garcia-Roger, Edgar E González, David Martín-Sacristán, and Jose F Monserrat. 2020. V2X support in 3GPP specifications: From 4G to 5G and beyond. *IEEE access* 8 (2020), 190946–190963.

[64] Ilya B. Gertsbakh. 1984. The shorter queue problem: A numerical study using the matrix-geometric solution. *European Journal of Operational Research* 15 (1984), 374–381.

[65] Fablenne Gillent and Guy Latouche. 1983. Semi-explicit solutions for M/PH/1-like queuing systems. *European Journal of Operational Research* 13, 2 (1983), 151–160. https://doi.org/10.1016/0377-2217(83)90077-2

[66] Judy Goldsmith and Martin Mundhenk. 1998. Complexity Issues in Markov Decision Processes.. In *CCC*. 272–280.

[67] Harold Gumbel. 1960. Waiting Lines with Heterogeneous Servers. *Operations Research* 8, 4 (1960), 504–511.

[68] Mian Guo, Quansheng Guan, Weiqi Chen, Fei Ji, and Zhiping Peng. 2022. Delay-Optimal Scheduling of VMs in a Queueing Cloud Computing System with Heterogeneous Workloads. *IEEE Transactions on Services Computing* 15, 1 (2022), 110–123. https://doi.org/10.

1109/TSC.2019.2920954

[69] Frank A. Haight. 1958. Two Queues in Parallel. *Biometrika* 45, 3/4 (1958), 401–410.

[70] Frank A. Haight. 1959. Queueing with reneging. *Metrika* 2, 1 (12 1959), 186–197. https://doi.org/10.1007/BF02613734

[71] Bin Han, Vincenzo Sciancalepore, Xavier Costa-Pérez, Di Feng, and Hans D. Schotten. 2020. Multiservice-Based Network Slicing Orchestration With Impatient Tenants. *IEEE Transactions on Wireless Communications* 19, 7 (2020), 5010–5024. https://doi.org/10.1109/TWC.2020.2988644

[72] Bin Han, Vincenzo Sciancalepore, Di Feng, Xavier Costa-Perez, and Hans D. Schotten. 2019. A Utility-Driven Multi-Queue Admission Control Solution for Network Slicing. (2019), 55–63. https://doi.org/10.1109/INFOCOM.2019.8737517

[73] Bin Han, Vincenzo Sciancalepore, Yihua Xu, Di Feng, and Hans D. Schotten. 2023. Impatient Queuing for Intelligent Task Offloading in Multiaccess Edge Computing. *IEEE Transactions on Wireless Communications* 22, 1 (2023), 59–72. https://doi.org/10.1109/TWC.2022.3191287

[74] Yinghua Han, Qinqin Xu, Qiang Zhao, and Fangyuan Si. 2023. Queue-aware computation offloading for UAV-assisted edge computing in wind farm routine inspection. *Journal of Renewable and Sustainable Energy* 15, 6 (2023).

[75] Lani Haque, Yiqiang Q. Zhao, and Liming Liu. 2005. Sufficient conditions for a geometric tail in a QBD process with many countable levels and phases. *Stochastic Models* 21, 1 (2005), 77–99. https://doi.org/10.1081/STM-200046489

[76] Refael Hassin and Moshe Haviv. 1994. Equilibrium strategies and the value of information in a two line queuing system with threshold jockeying. *Stochastic Models - STOCH MODELS* 10 (01 1994), 415–435. https://doi.org/10.1080/15326349408807302

[77] Refael Hassin and Moshe Haviv. 1997. Equilibrium Threshold Strategies: The Case of Queues with Priorities. *Operations Research* 45, 6 (1997), 966–973.

[78] B.R. Haverkort. 1995. Matrix-geometric solution of infinite stochastic Petri nets. In *Proceedings of 1995 IEEE International Computer Performance and Dependability Symposium*. 72–81. https://doi.org/10.1109/IPDS.1995.395815

[79] Moshe Haviv and Liron Ravner. 2015. Strategic bidding in an accumulating priority queue: equilibrium analysis. *Annals of Operations Research* 244 (2015), 505–523.

[80] Qiming He and Marcel F. Neuts. 2002. Two M/M/1 Queues with Transfers of Customers. *Queueing Systems* 42 (2002), 377–400.

[81] Qi-Ming He. 2013. Analysis of queueing systems with customer interjections. *Queueing Systems* 73 (01 2013), 79–104.

[82] Samid Hoda, Andrew Gilpin, Javier Peña, and Tuomas Sandholm. 2010. Smoothing Techniques for Computing Nash Equilibria of Sequential Games. *Mathematics of Operations Research* 35, 2 (2010), 494–512.

[83] Toke Hoeiland-Joergensen, Paul McKenney, Dave Taht, Jim Gettys, and Eric Dumazet. 2018. *The flow queue codel packet scheduler and active queue management algorithm*. Technical Report.

[84] Longbo Huang. 2017. The Value-of-Information in Matching With Queues. *IEEE/ACM Transactions on Networking* 25, 1 (2017), 29–42. https://doi.org/10.1109/TNET.2016.2564700

[85] Einollah Jafarnejad Ghomi, Amir Masoud Rahmani, and Nooruldeen Nasih Qader. 2019. Applying queue theory for modeling of cloud computing: A systematic review. *Concurrency and Computation: Practice and Experience* 31, 17 (2019), e5186.

[86] R. Jain. 1986. A Timeout-Based Congestion Control Scheme for Window Flow-Controlled Networks. *IEEE Journal on Selected Areas in Communications* 4, 7 (1986), 1162–1167. https://doi.org/10.1109/JSAC.1986.1146431

[87] Xingguo Ji and Qingmin Meng. 2020. Traffic Classification Based on Graph Convolutional Network. In *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications(AEECA)*. 596–601. https://doi.org/10.1109/AEECA49918.2020.9213630

[88] Jiekai Jia, Anam Tahir, and Heinz Koeppl. 2022. Decentralized Coordination in Partially Observable Queueing Networks. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. 1491–1496. https://doi.org/10.1109/GLOBECOM48099.2022.10001584

[89] Tao Jiang and Li-Wei Liu. 2016. Tail Asymptotics of Two Parallel Queues with Transfers of Customers. *Journal of the Operations Research Society of China* 4 (04 2016). https://doi.org/10.1007/s40305-016-0127-1

[90] J.S. Jordan. 1993. Three Problems in Learning Mixed-Strategy Nash Equilibria. *Games and Economic Behavior* 5, 3 (1993), 368–386. https://doi.org/10.1006/game.1993.1022

[91] Michelle H. Kallmes, D. Towsley, and Christos G. Cassandras. 1989. Optimality of the last-in-first-out (LIFO) service discipline in queuing systems with real-time constraints. *Proceedings of the 28th IEEE Conference on Decision and Control,* 2, 0 (1989), 1073–1074. https://doi.org/10.1109/CDC.1989.70296

[92] Clement Kam, Joseph P. Molnar, and Sastry Kompella. 2018. Age of Information for Queues in Tandem. In *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)*. 1–6. https://doi.org/10.1109/MILCOM.2018.8599728

[93] Edward P.C. Kao and Chiunsin Lin. 1990. A matrix-geometric solution of the jockeying problem. *European Journal of Operational Research* 44, 1 (1990), 67–74. https://doi.org/10.1016/0377-2217(90)90315-3

[94] Stella Kapodistria and Zbigniew Palmowski. 2017. Matrix geometric approach for random walks: Stability condition and equilibrium distribution. *Stochastic Models* 33, 4 (2017), 572–597. https://doi.org/10.1080/15326349.2017.1359096

[95] Rajeeva L. Karandikar and Vidyadhar G. Kulkarni. 1995. Second-Order Fluid Flow Models: Reflected Brownian Motion in a Random Environment. *Operations Research* 43, 1 (1995), 77–88.

[96] David G. Kendall. 1953. Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain. *The Annals of Mathematical Statistics* 24, 3 (1953), 338–354.

[97] Anthony Kiggundu, Bin Han, Dennis Krummacker, and Hans D. Schotten. 2024. Resource Allocation in Mobile Networks: A Decision Model Of Jockeying in Queues. arXiv:2402.11054 [cs.NI]

[98] Bara Kim, Jeongsim Kim, and Ole Bueker. 2023. Equilibrium analysis of a partially observable priority queue. *Computers and Industrial Engineering* 182 (2023), 109434. https://doi.org/10.1016/j.cie.2023.109434

[99] Minsu Kim, Muhammad Jaseemuddin, and Alagan Anpalagan. 2021. Deep Reinforcement Learning Based Active Queue Management for IoT Networks. *J. Netw. Syst. Manage.* 29, 3 (jul 2021), 28 pages. https://doi.org/10.1007/s10922-021-09603-x

[100] Minsu Kim, Muhammad Jaseemuddin, and Alagan Anpalagan. 2021. Deep Reinforcement Learning Based Active Queue Management for IoT Networks. *J. Netw. Syst. Manage.* 29, 3 (jul 2021). https://doi.org/10.1007/s10922-021-09603-x

[101] John Frank C. Kingman. 1961. Two Similar Queues in Parallel. *The Annals of Mathematical Statistics* 32, 4 (1961), 1314–1323.

[102] John Frank C. Kingman. 1962. On queues in which customers are served in random order. *Mathematical Proceedings of the Cambridge Philosophical Society* 58 (1962), 79 – 91.

[103] Ernest Koenigsberg. 1966. On Jockeying in Queues. *Management Science* 12 (1966), 412–436.

[104] Yaakov Kogan, Yonatan Levy, and Rodolfo A Milito. 1997. Call routing to distributed queues: Is FIFO really better than MED? *Telecommunication Systems* 7 (06 1997), 299–312.

[105] Ger Koole. 1996. On the Pathwise Optimal Bernoulli Routing Policy for Homogeneous Parallel Servers. *Mathematics of Operations Research* 21, 2 (1996), 469–476.

[106] V. Kumar and N.S. Upadhye. 2022. On first-come, first-served queues with three classes of impatient customers. *International Journal of Advances in Engineering Sciences and Applied Mathematics* 13 (03 2022), 368–382.

[107] Corine M. Laan, Judith Timmer, and Richard J. Boucherie. 2021. Non-cooperative queueing games on a network of single server queues. *Queueing Systems* 97, 3 (4 2021), 279–301. https://doi.org/10.1007/s11134-020-09681-9

[108] Line M. P. Larsen, Aleksandra Checko, and Henrik L. Christiansen. 2019. A Survey of the Functional Splits Proposed for 5G Mobile Crosshaul Networks. *IEEE Communications Surveys and Tutorials* 21, 1 (2019), 146–172. https://doi.org/10.1109/COMST.2018.2868805

[109] Toni Lassila, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. 2014. *Model Order Reduction in Fluid Dynamics: Challenges and Perspectives.* Springer International Publishing, 235–273. https://doi.org/10.1007/978-3-319-02090-7_9

[110] Guy Latouche and V. Ramaswami. 1987. *Introduction to Matrix Analytic Methods in Stochastic Modelling.* Society for Industrial and Applied Mathematics, Philadelphia.

[111] Lei Li, Quansheng Guan, Lianwen Jin, and Mian Guo. 2019. Resource Allocation and Task Offloading for Heterogeneous Real-Time Tasks With Uncertain Duration Time in a Fog Queueing System. *IEEE Access* 7 (2019), 9912–9925. https://doi.org/10.1109/ACCESS.2019.2891130

[112] Vlada Limic. 2001. A LIFO Queue in Heavy Traffic. *The Annals of Applied Probability* 11, 2 (2001), 301–331.

[113] Bing Lin, Yuchen Lin, and Rohit Bhatnagar. 2022. Optimal Policy for Controlling Two-Server Queueing Systems with Jockeying. *Journal of Systems Engineering and Electronics* 33, 1 (2022), 144–155. https://doi.org/10.23919/JSEE.2022.000015

[114] Dong Lin and Robert Morris. 1997. Dynamics of random early detection. 27, 4 (oct 1997), 127–137. https://doi.org/10.1145/263109.263154

[115] Frank Po-Chen Lin and Zsehong Tsai. 2020. Hierarchical Edge-Cloud SDN Controller System With Optimal Adaptive Resource Allocation for Load-Balancing. *IEEE Systems Journal* 14, 1 (2020), 265–276. https://doi.org/10.1109/JSYST.2019.2894689

[116] Chen-Feng Liu, Mehdi Bennis, Mérouane Debbah, and H. Vincent Poor. 2019. Dynamic Task Offloading and Resource Allocation for Ultra-Reliable Low-Latency Edge Computing. *IEEE Transactions on Communications* 67, 6 (2019), 4132–4150. https://doi.org/10.1109/TCOMM.2019.2898573

[117] Chen-Feng Liu, Mehdi Bennis, and H. Vincent Poor. 2017. Latency and Reliability-Aware Task Offloading and Resource Allocation for Mobile Edge Computing. (2017), 1–7. https://doi.org/10.1109/GLOCOMW.2017.8269175

[118] Yuanhe Liu and Ruiming Quan. 2023. Research and Analysis on the Interaction between Queuing Theory and Artificial Intelligence. (2023), 391–401. https://doi.org/10.2991/978-94-6463-300-9_40

[119] Aleksander Lodwich, Yves Rangoni, and Thomas Breuel. 2009. Evaluation of robustness and performance of Early Stopping Rules with Multi Layer Perceptrons. In *2009 International Joint Conference on Neural Networks.* 1877–1884. https://doi.org/10.1109/IJCNN.2009.5178626

[120] Lingling Lv, Zongyu Wu, Lei Zhang, Brij B. Gupta, and Zhihong Tian. 2022. An Edge-AI Based Forecasting Approach for Improving Smart Microgrid Efficiency. *IEEE Transactions on Industrial Informatics* 18, 11 (2022), 7946–7954. https://doi.org/10.1109/TII.2022.3163137

[121] Mamoru Maekawa. 1977. Queueing Models for Computer Systems Connected by a Communication Line. *J. ACM* 24, 4 (oct 1977), 566–582.

[122] Simon P. Martin and Isi Mitrani. 2008. Analysis of job transfer policies in systems with unreliable servers. *Annals of Operations Research* 162 (2008), 127–141.

[123] Sean P. Meyn and Richard L. Tweedie. 1993. Stability of Markovian Processes III: Foster-Lyapunov Criteria for Continuous-Time Processes. *Advances in Applied Probability* 25, 3 (1993), 518–548.

[124] Ronald E. Mickens. 2015. *Difference Equations: Theory, Applications and Advanced Topics*. Vol. Third Edition. Chapman and Hall/CRC., London. https://doi.org/10.1201/b18186

[125] Isi Mitrani and Ram Chakka. 1995. Spectral expansion solution for a class of Markov models: application and comparison with the matrix-geometric method. *Performance Evaluation* 23, 3 (1995), 241–260. https://doi.org/10.1016/0166-5316(94)00025-F

[126] Banwari Mittal. 2016. Retrospective: why do customers switch? The dynamics of satisfaction versus loyalty. *Journal of Services Marketing* 30 (09 2016), 569–575. https://doi.org/10.1108/JSM-07-2016-0277

[127] Masakiyo Miyazawa and Yiqiang Q. Zhao. 2004. The Stationary Tail Asymptotics in the GI/G/1-Type Queue with Countably Many Background States. *Advances in Applied Probability* 36, 4 (2004), 1231–1251.

[128] George E. Monahan. 1982. State of the Art—A Survey of Partially Observable Markov Decision Processes: Theory, Models, and Algorithms. *Management Science* 28, 1 (1982), 1–16. https://doi.org/10.1287/mnsc.28.1.1

[129] John F. Nash. 1950. Equilibrium Points in n-Person Games. *Proceedings of the National Academy of Science* 36, 1 (1 1950), 48–49. https://doi.org/10.1073/pnas.36.1.48

[130] M. Natarajan and A. Kolobov. [n. d.]. *Planning with Markov Decision Processes: An AI Perspective*. Springer International Publishing.

[131] Randolph D Nelson and Thomas K Philips. 1993. An approximation for the mean response time for shortest queue routing with general interarrival and service times. *Performance evaluation* 17, 2 (1993), 123–139.

[132] Marcel F. Neuts. 1986. The caudal characteristic curve of queues. *Advances in Applied Probability* 18, 1 (1986), 221–254. https://doi.org/10.2307/1427244

[133] Marcel Fernand Neuts. 1994. *Matrix-geometric Solutions in Stochastic Models: An Algorithmic Approach*. Dover Publications, Newyork.

[134] David M. Nicol. 1988. Parallel discrete-event simulation of FCFS stochastic queueing networks. *SIGPLAN Not.* 23, 9 (jan 1988), 124–137. https://doi.org/10.1145/62116.62128

[135] Jose Ordonez-Lucena, Pablo Ameigeiras, Diego Lopez, Juan J Ramos-Munoz, Javier Lorca, and Jesus Folgueira. 2017. Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges. *IEEE Communications Magazine* 55, 5 (2017), 80–87.

[136] Joni Pajarinen, Ari Hottinen, and Jaakko Peltonen. 2013. Optimizing spatial and temporal reuse in wireless networks by decentralized partially observable Markov decision processes. *IEEE Transactions on Mobile Computing* 13, 4 (2013), 866–879.

[137] Apostolos Papageorgiou, Adriana Fernández-Fernández, Leonardo Ochoa-Aday, Miguel Silva Peláez, and Muhammad Shuaib Siddiqui. 2020. SLA Management Procedures in 5G Slicing-based Systems. (2020), 7–11. https://doi.org/10.1109/EuCNC48522.2020.9200904

[138] Kon Papazis, Naveen K. Chilamkurti, and Ben Soh. 2004. A New Receiver-Based Layered-Rate Estimator Algorithm for Fair Bandwidth Distribution. In *Proceedings of the 28th Annual International Computer Software and Applications Conference - Volume 01 (COMPSAC '04)*. IEEE Computer Society, 560–565.

[139] Amit I. Pazgal and Sonja Radas. 2008. Comparison of customer balking and reneging behavior to queueing theory predictions: An experimental study. *Comput. Oper. Res.* 35, 8 (8 2008), 2537–2548.

[140] Jordi Pérez-Romero, Oriol Sallent, Antoni Gelonch, Xavier Gelabert, Bleron Klaiqi, Marcus Kahn, and David Campoy. 2023. A Tutorial on the Characterisation and Modelling of Low Layer Functional Splits for Flexible Radio Access Networks in 5G and Beyond. *IEEE Communications Surveys and Tutorials* 25, 4 (2023), 2791–2833. https://doi.org/10.1109/COMST.2023.3296821

[141] J. E. Pezoa, M. M. Hayat, D. A. Bader, C. Yang, and S. Dhakal. 2007. Dynamic Load Balancing in Distributed Systems in the Presence of Delays: A Regeneration-Theory Approach. *IEEE Transactions on Parallel and Distributed Systems* 18, 04 (04 2007), 485–497. https://doi.org/10.1109/TPDS.2007.1009

[142] S.D. Poisson and S.D. Poisson. 1837. *Recherches sur la probabilité des jugements en matière criminelle et en matière civile: précédées des règles générales du calcul des probabilités*. Bachelier, online.

[143] Michele Polese, Leonardo Bonati, Salvatore D'oro, Stefano Basagni, and Tommaso Melodia. 2023. Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges. *IEEE Communications Surveys and Tutorials* 25, 2 (2023), 1376–1411.

[144] Lutz Prechelt. 1998. Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks* 11, 4 (1998), 761–767. https://doi.org/10.1016/S0893-6080(98)00010-0

[145] Rezaur Rahman and Samiul Hasan. 2021. Real-time signal queue length prediction using long short-term memory neural network. *Neural Comput. Appl.* 33, 8 (apr 2021), 3311–3324.

[146] V. Ramswami and Guy Latouche. 1986. A general class of Markov processes with explicit matrix-geometric solutions. *Operations-Research-Spektrum* 8 (1986), 209–218.

[147] Rachel Ravid. 2021. A new look on the shortest queue system with jockeying. *Probability in the Engineering and Informational Sciences* 35, 3 (2021), 557–564. https://doi.org/10.1017/S0269964819000469

[148] David Raz, Benjamin Avi-Itzhak, and Hanoch Levy. 2005. Fair operation of multi-server and multi-queue systems. In *Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '05)*. Association for Computing Machinery, 382–383. https://doi.org/10.1145/1064212.1064265

[149] David Raz, Benjamin Avi-Itzhak, and Hanoch Levy. 2006. Fairness considerations of scheduling in multi-server and multi-queue systems. In *Proceedings of the 1st International Conference on Performance Evaluation Methodolgies and Tools (valuetools '06)*. Association for Computing Machinery, New York, NY, USA, 39–es. https://doi.org/10.1145/1190095.1190145

[150] David Raz, Hanoch Levy, and Benjamin Avi-Itzhak. 2004. A resource-allocation queueing fairness measure. *SIGMETRICS Perform. Eval. Rev.* 32, 1 (jun 2004), 130–141. https://doi.org/10.1145/1012888.1005704

[151] T.G. Robertazzi. 2000. *Computer Networks and Systems: Queueing Theory and Performance Evaluation.* Springer New York.

[152] Zvi Rosberg, Yu Peng, Jing Fu, Jun Guo, Eric W. M. Wong, and Moshe Zukerman. 2014. Insensitive Job Assignment With Throughput and Energy Criteria for Processor-Sharing Server Farms. *IEEE/ACM Transactions on Networking* 22, 4 (2014), 1257–1270. https://doi.org/10.1109/TNET.2013.2276427

[153] Zvi Rosberg, Yu Peng, Jing Fu, Jun Guo, Eric W. M. Wong, and Moshe Zukerman. 2014. Insensitive Job Assignment With Throughput and Energy Criteria for Processor-Sharing Server Farms. *IEEE/ACM Transactions on Networking* 22, 4 (2014), 1257–1270. https://doi.org/10.1109/TNET.2013.2276427

[154] Werner Scheinhardt, Nicky van Foreest, and Michel Mandjes. 2005. Continuous feedback fluid queues. *Operations Research Letters* 33, 6 (2005), 551–559. https://doi.org/10.1016/j.orl.2004.11.008

[155] William T. Scherer, Stephen Adams, and Peter A. Beling. 2018. On the Practical Art of State Definitions for Markov Decision Process Construction. *IEEE Access* 6 (2018), 21115–21128. https://doi.org/10.1109/ACCESS.2018.2819940

[156] Vincenzo Sciancalepore, Konstantinos Samdanis, Xavier Costa-Perez, Dario Bega, Marco Gramaglia, and Albert Banchs. 2017. Mobile traffic forecasting for maximizing 5G network slicing resource utilization. (2017), 1–9. https://doi.org/10.1109/INFOCOM.2017.8057230

[157] Flore Sentenac, Etienne Boursier, and Vianney Perchet. 2021. Decentralized Learning in Online Queuing Systems. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 18501–18512.

[158] S.K. Setia, M.S. Squillante, and S.K. Tripathi. 1994. Analysis of processor allocation in multiprogrammed, distributed-memory parallel processing systems. *IEEE Transactions on Parallel and Distributed Systems* 5, 4 (1994), 401–420. https://doi.org/10.1109/71.273047

[159] J George Shanthikumar, Shengwei Ding, and Mike Tao Zhang. 2007. Queueing theory for semiconductor manufacturing systems: A survey and open problems. *IEEE Transactions on Automation Science and Engineering* 4, 4 (2007), 513–522.

[160] Hsien-Po Shiang and Mihaela van der Schaar. 2008. Queuing-Based Dynamic Channel Selection for Heterogeneous Multimedia Applications Over Cognitive Radio Networks. *IEEE Transactions on Multimedia* 10, 5 (2008), 896–909. https://doi.org/10.1109/TMM.2008.922851

[161] Hsien-Po Shiang and Mihaela van der Schaar. 2008. Queuing-Based Dynamic Channel Selection for Heterogeneous Multimedia Applications Over Cognitive Radio Networks. *IEEE Transactions on Multimedia* 10, 5 (2008), 896–909. https://doi.org/10.1109/TMM.2008.922851

[162] Zhaogang Shu and Tarik Taleb. 2020. A novel QoS framework for network slicing in 5G and beyond networks based on SDN and NFV. *IEEE Network* 34, 3 (2020), 256–263.

[163] Arunan Sivanathan, Daniel Sherratt, Hassan Habibi Gharakheili, Adam Radford, Chamith Wijenayake, Arun Vishwanath, and Vijay Sivaraman. 2017. Characterizing and classifying IoT traffic in smart cities and campuses. In *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 559–564.

[164] Li song Shao, He ying Zhang, and Wen hua Dou. 2006. General window-based congestion control: buffer occupancy, network efficiency and packet loss. In *4th Annual Communication Networks and Services Research Conference (CNSR'06)*. 7 pp.–195. https://doi.org/10.1109/CNSR.2006.29

[165] Wolfgang Stadje. 2009. A Queueing System with Two Parallel Lines, Cost-Conscious Customers, and Jockeying. *Communications in Statistics - Theory and Methods* 38:16-17 (2009), 3158–3169. https://doi.org/10.1080/03610920902947618

[166] Robert E. Stanford. 1979. Reneging Phenomena in Single Channel Queues. *Mathematics of Operations Research* 4, 2 (05 1979), 162–178.

[167] Oded Stark, Wiktor Budzinski, and Grzegorz Kosiorowski. 2019. Switching queues, cultural conventions, and social welfare. *European Journal of Operational Research* 278, 3 (2019), 837–844. https://doi.org/10.1016/j.ejor.2019.02.053

[168] René Brandborg Sørensen, Dong Min Kim, Jimmy Jessen Nielsen, and Petar Popovski. 2017. Analysis of Latency and MAC-Layer Performance for Class A LoRaWAN. *IEEE Wireless Communications Letters* 6, 5 (2017), 566–569. https://doi.org/10.1109/LWC.2017.2716932

[169] Anam Tahir, Bastian Alt, Amr Rizk, and Heinz Koeppl. 2023. Load Balancing in Compute Clusters With Delayed Feedback. *IEEE Trans. Comput.* 72, 6 (jun 2023), 1610–1622. https://doi.org/10.1109/TC.2022.3215907

[170] Junichi Takinami, Yutaka Matsumoto, and Norio Okino. 1993. Performance Evaluation of Neural Networks Applied to Queueing Allocation Problem. In *Artificial Neural Nets and Genetic Algorithms*. Springer Vienna, 316–323.

[171] Ahmed M.K. Tarabia. 2009. Transient analysis of two queues in parallel with jockeying. *Stochastics* 81, 2 (2009), 129–145. https://doi.org/10.1080/17442500802200658

[172] Ahmed M. K. Tarabia. 2008. Analysis of two queues in parallel with jockeying and restricted capacities. *Applied Mathematical Modelling* 32 (2008), 802–810.

[173] Rafael Tolosana-Calasanz, Javier Diaz-Montes, Omer F. Rana, and Manish Parashar. 2017. Feedback-Control and Queueing Theory-Based Resource Management for Streaming Applications. *IEEE Transactions on Parallel and Distributed Systems* 28, 4 (2017), 1061–1075. https://doi.org/10.1109/TPDS.2016.2603510

[174] Montanaro Umberto, Dixit Shilp, Fallah Saber, Dianati Mehrdad, Stevens Alan, Oxtoby David, and Alexandros Mouzakitis. 2019. Towards connected autonomous driving: review of use-cases. *Vehicle System Dynamics* 57, 6 (2019), 779–814. https://doi.org/10.1080/00423114.2018.1492142

[175] James S. Vandergraft. 1983. A Fluid Flow Model of Networks of Queues. *Management Science* 29, 10 (1983), 1198–1208.

[176] Qi Wang, Jose Alcaraz-Calero, Ruben Ricart-Sanchez, Maria Barros Weiss, Anastasius Gavras, Navid Nikaein, Xenofon Vasilakos, Bernini Giacomo, Giardina Pietro, Mark Roddy, et al. 2019. Enable advanced QoS-aware network slicing in 5G networks for slice-based media use cases. *IEEE transactions on broadcasting* 65, 2 (2019), 444–453.

[177] Shuang Wang, Xiaoping Li, and Ruben Ruiz. 2019. Performance analysis for heterogeneous cloud servers using queueing theory. *IEEE Trans. Comput.* 69, 4 (2019), 563–576.

[178] Ward Whitt. 1984. Departures from a Queue with Many Busy Servers. *Mathematics of Operations Research* 9, 4 (1984), 534–544.

[179] Ward Whitt. 1986. Deciding Which Queue to Join: Some Counterexamples. *Operations Research* 34, 1 (1986), 55–62.

[180] J. W. J. Williams. 1964. Algorithm 232 - Heapsort. *Communications of ACM* 7, 6 (6 1964), 347–348. https://doi.org/10.1145/512274.512284

[181] J. W. J. Williams. 1964. Algorithm 245 - Treesort. *Communications of ACM* 7, 12 (12 1964), 701. https://doi.org/10.1145/355588.365103

[182] Martin Wollschlaeger, Thilo Sauter, and Juergen Jasperneite. 2017. The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0. *IEEE industrial electronics magazine* 11, 1 (2017), 17–27.

[183] Jun Xu, Junmei Yao, Lu Wang, Kaishun Wu, Lei Chen, and Wei Lou. 2018. Revolution of self-organizing network for 5G mmWave small cell management: From reactive to proactive. *IEEE Wireless Communications* 25, 4 (2018), 66–73.

[184] Susan H. Xu and Y. Quennel Zhao. 1996. Dynamic routing and jockeying controls in a two-station queuing system. *Advances in Applied Probability* 28, 4 (1996), 1201–1226. https://doi.org/10.2307/1428170

[185] Li Yang and Abdallah Shami. 2020. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing* 415 (2020), 295–316.

[186] Mehmet Akif Yazıcı and Nail Akar. 2013. Analysis of continuous feedback Markov fluid queues and its applications to modeling Optical Burst Switching. In *Proceedings of the 2013 25th International Teletraffic Congress (ITC)*. 1–8. https://doi.org/10.1109/ITC.2013.6662952

[187] Faqir Zarrar Yousaf, Michael Bredel, Sibylle Schaller, and Fabian Schneider. 2017. NFV and SDN—Key technology enablers for 5G networks. *IEEE Journal on Selected Areas in Communications* 35, 11 (2017), 2468–2478.

[188] Xiang Yu, I. Thng, Yuming Jiang, and Chunming Qiao. 2005. Queueing processes in GPS and PGPS with LRD traffic inputs. *IEEE/ACM Transactions on Networking* 13, 03 (05 2005), 676–689. https://doi.org/10.1109/TNET.2005.850213

[189] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. 2020. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access* 8 (2020), 58443–58469. https://doi.org/10.1109/ACCESS.2020.2983149

[190] A. Badamchi Zadeh. 2015. A batch arrival multi phase queueing system with random feedback in service and single vacation policy. *OPSEARCH* 52, 4 (12 2015), 617–630. https://doi.org/10.1007/s12597-015-0206-9

[191] V. M. Zaiats, O. M. Rybytska, and M. M. Zaiats. 2019. An Approach to Assessment of the Value and Quantity of Information in Queueing Systems Based on Pattern Recognition and Fuzzy Sets Theories. *Cybernetics and Systems Analysis* 55 (07 2019), 638–648. Issue 4. https://doi.org/10.1007/s10559-019-00172-1

[192] Yan Ma Zaiming Liu and Zhe George Zhang. 2015. Equilibrium Mixed Strategies in a Discrete-Time Markovian Queue Under Multiple and Single Vacation Policies. *Quality Technology and Quantitative Management* 12, 3 (2015), 369–382. https://doi.org/10.1080/16843703.2015.11673387

[193] Yuan Zhang, Peng Du, Jiang Wang, Teer Ba, Rui Ding, and Ning Xin. 2019. Resource Scheduling for Delay Minimization in Multi-Server Cellular Edge Computing Systems. *IEEE Access* 7 (2019), 86265–86273. https://doi.org/10.1109/ACCESS.2019.2924032

[194] Y. Zhao and Grassmann K. Winfried. 1995. Queuing Analysis of a Jockeying Model. *Operations Research* 43, 3 (1995), 520–529.

[195] Yiqiang Q. Zhao and Winfried K. Grassmann. 1990. The shortest queue model with jockeying. *Naval Research Logistics* 37 (1990), 773–787.

[196] Jianshan Zhou, Daxin Tian, Zhengguo Sheng, Xuting Duan, and Xuemin Shen. 2021. Distributed Task Offloading Optimization With Queueing Dynamics in Multiagent Mobile-Edge Computing Networks. *IEEE Internet of Things Journal* 8, 15 (2021), 12311–12328. https://doi.org/10.1109/JIOT.2021.3063509

Received 09 January 2023