Advances In Multidisciplinary & Scientific Research
A Multidisciplinary & Interdisciplinary Journal

Vol. 3 No.2, June , 2017

Simulation Model for Optimization of Call Center

Mughele, E.S.
Department of Computer Science
Delta State School of Marine Technology
Burutu, Delta State Nigeria.
E-mail: prettysophy99@gmail.com

Chiemeke, S.C.
Department of Computer Science
University of Benin
Benin City, Edo State Nigeria.
schiemeke@yahoo.com

Chete, F
Department of Computer Science
Auchi Polytechnic
Auchi, Nigeria
chetefrancis2015@gmail.com

ABSTRACT

The queue experienced by customers at call centres is increasingly alarming as many customers are irritated by the long time before their calls are been answered. Agents are trained to handle all entry calls to a call centre but are characterized with different performance level for the call in terms of average call handling time (AHT) and call resolution (CR). In this work, we analyze various call routing rules for determining which calls should be handled by which call centre agents. We attempt to evaluate and determine an optimal routing rule of low handling time and high call resolution rate. We conducted interviews at the customer service call centre Global Communications in Nigeria, to investigate its operations and to obtain data from its automated call logging system (Database). We tested several routing rules using data obtained from the call centre. Java simulation programs were developed for each existing rule for both CR and wait-time routing rule since their procedure vary from one another. The programs were tested with data collected from the call center. The results allowed us to explain overall performance in terms of average speed of answer and overall call resolution rate. An optimal routing rule was proposed having identified optimal rules for both wait time and CR oriented routing rules. We developed a system model that enhances call resolution rate and reduces waiting time on the queue.

Keywords: Call Center, Simulation, Routing Rule, Average Speed of Answer, Call Resolution

Aims Research Journal Reference Format:
Mughele, E.S., Chiemeke, S.C. & Chete, F. (2017): Simulation Model for Optimization of Call Center.
Advances in Multidisciplinary & Scientific Research Journal. Vol. 3. No.2, Pp 103-124

1. INTRODUCTION

Businesses create value through product offerings or service delivery to their customers. For customers to have access to these products and services sometimes they spend long time waiting in the queue before the service is delivered, in some other instances the customer is abandoned on the queue without accessing the desired services. It is therefore the concern of every company management to render prompt delivery of service, eliminate waiting queue and give value for money so as to ensure customer satisfaction and loyalty. Call centre can be defined as any group whose business is talking to customers or prospective customers through the telephone. According to Brizola et al (2001), a call centre is a system that offers complete management of all communication channels between a business and its customers, optimizing polices, eliminating duplicated work and making better use of time. The call centre service has grown a great deal with its application in all sectors of the economy. It serves as a primary contact between businesses and clients. But in recent times, customers waiting for so long in order to lodge a complaint or make an enquiry have become a worrisome phenomenon in the call centres especially in telecommunications. The Nigerian telecommunications industry is a rapidly growing sector with subscriber base running into millions, and the existence of waiting queue is a common feature of call centres.

Consequently, customers experience unpleasant situation waiting in queue develop a negative perception and attitude to a firm's service which may affect the long-term success or prosperity of such firms. In contemporary society, satisfying customers need has become a phenomenon seen to be highly inevitable for business that wants to survive in this era of high competition amidst the global financial crisis. A customer's experience during a service encounter consist of two parts namely: the time spent waiting for the service and the service itself. Call centres give priority to the two criteria with emphasis on one more than the other. Those that place more emphasis on time spent waiting for the service are more concerned with reducing the average time involved in handling a call while those that are concerned with the service itself aims at effective resolution of customer issues.

A customer's experience during a service encounter consist of two parts namely: the time spent waiting for the service and the service itself, waiting has to do with the queue while the service is a function of the resolution status. Call centres give priority to the two criteria with emphasis on one more than the other. Those that place more emphasis on time spent waiting for the service are more concerned with reducing the average time involved in handling a call while those that are concerned with the service itself aims at effective resolution of customer issues.
Armony (2005) says for a call centre to reduce waiting lines with emphasis on the reduction of time spent, its best to route calls to agents who can handle customer issues the fastest, sometimes even holding a call in queue to wait for that agent than routing the call to a slower agent. This might lead to further increase in congestion, repeat calls from unreceptive issues and undue burden on some agents.

Vericourt et al. (2005), states that for a call centre to reduce waiting lines, emphasis should be on the service itself that is; call resolution. Its best to route calls to agents who resolve customer issues, sometimes holding a call in queue to wait for such agent. This might also lead to increase in congestion and undue burden on some agents. After a customer has received service from a call centre agent on a particular issue, a subsequent call from that customer about the same issue is a clear sign that the issue had not been resolved during the previous service encounter, and this lack of resolution is a strong sign of customer dissatisfaction. In this work, we explore strategies for routing multiple types of calls to a large group of agents, where these assignments are made dynamically based on the specific attributes of the agents and/or the current state of the system. We believe that this study will make several important contributions to the call centre operations/management regarding reduction of queues and enhanced call resolution.

2. LITERATURE REVIEW

2.1 The Theory of Queues

Enyioko, (2016), defined a queueing system is a birth-death process with a population consisting of customers either waiting for services or currently in service. A birth occurs when a customer arrives at the service facilities. A death occurs when a customer departs from the facility. The state of the system is the number of customers in the facilities.
The theory of queues (derived from the Latin *cauda*) was initially the idea of Erlang, who published his first article on the subject in 1909 and is considered the founder of Telephone Traffic Theory and of the Theory of Queues. After the Second World War, interest was solidified with formal operational research and since then much work has been published on the subject (Cooper 1997).

A queue is a situation whereby customers wait in line to be attended to. Sharma (2009) defines it as any place where a customer (human beings or physical entities) that requires service is made to wait due to the fact that the number of customers exceeds the number of service facilities or when service facilities do not work efficiently and take more time than prescribed to serve a customer. Brizola et al (2001), defined a call centre as a system that offers complete management of all communication channels between a business and its customers, optimizing process, eliminating duplicated work and making better use of time. Avramidis et al (2004) noted that it is a set of resources (communication equipment, employees, computers etc.) which enable the delivery of services via the telephone. From the above, it can be seen that call centres are limits that manages an organization communication system. Call centres are known by a variety of names namely: contact centre, customer service centre, customer interaction centre, customer service point etc.

2.1 Routing Techniques in a Call Centre

Call routing is the sequence of path taken to convey a customer's call to a service agent. Call routing also known as call distribution relates to a set of rules which are applied to isolate the most appropriate resource for a specific call. Call routing is experience by the customer as being guided through a decision tree. By progressing through that tree the system provides information to and collects user inputs from the caller. The corresponding realization is often referred to as routing path. However having reached the leaf of the decision tree, the collected information is considered as being sufficiently complete and call distribution takes over to determine the most appropriate agent based on agent properties, user input and system load to route the call.

All routing techniques or algorithms used in call distribution follows a baseline routing rule which serves as a benchmark for routing cells (Mehrotra, 2009). The benchmark routing rule usually followed is the first-come, first serve or longest wait rule. Here the rule states that the first customer to arrive on a queue or the customer that has waited the longest on the queue and it follows the sequence until all calls are attended to. Customer service call centers have obviously become a very integral part of many organisations' business operations today, inbound call centers employ millions of agents across the globe and serve as a primary customer-facing channel for many different industries. There has also been a great deal of research interest in call center operations management, with the extensive and evolving literature thoroughly analysed (Mehrotra et al, 2009). This study determines whether average handling time and call resolution are true determinants of operational success of a call centre to reduce waiting queue. It also examine whether emphasis should be on reducing handling time or effective call resolution.

Aksin et al (2007) noted that the operational challenges from call centers provide a perspective on both traditional and emerging call center management challenges and the associated academic research associated. The researchers deployed literature review method and identified a handful of broad themes for future investigation while also pointing out several very specific research opportunities. Moreover, the work only discussed what others have done. Given the size of the call center industry and the complexity associated with its operations, call centers have emerged as a fertile ground for academic research. Hart et al. (2006) provides a complete review of articles on FCR while also pointing out the importance of measuring and using FCR. Resolving customer queries the first time around is a commonly shared goal. A company's business context, human resources strategy, supporting technology and budget constraints influences this Key Performance Indicator (KPI) in many ways, and makes First Call Resolution (FCR) a difficult measure to benchmark. The study established the differing views on the value and measurement of FCR, identifies the main factors affecting FCR and the relationships among these factors, and relates results in a South African context to academic and practitioner.

Operations management researchers have paid comparatively little attention to models and methods for managing routing. However, there are many published papers that describe call routing and resource allocation rules for call centers. Armory and Maglaras (2004) observed that customers in a call service center experiences real time delay as a result of queue and call back delay. This metrics affect customer's perception of the product or service and this impact on customer's loyalty. Probabilistic choice model was deployed, and the dynamics of the system are modeled as an *M/M/N* multiclass system. The study justifies that as the number of agents increases, the system's load approaches its maximum processing capacity but did not consider the Average handling Time in relation to customer decision, routing rules and system design.

Zhan and Ward (2006), noted that the challenge in call center operation is how to determine the relevant control in call routing; that is, the decision concerning which agent should handle an arriving call when more than one agent is available. An inverted-V model setting was designed, and they formulated an optimization problem with the dual performance objective of minimizing average customer waiting time and maximizing the call resolution. They also noted that focusing on minimizing average waiting time as the sole performance objective may not deliver the best customer experience. However, how does agents make such decisions that are relevant to the call center environment (trade off) was not considered. Véricourt and Zhou (2005) also discovered that traditional research on routing in queuing systems usually ignores service quality related factors. Customers call back when their problems are not completely resolved by the customer service representatives. They used a Markov decision process formulation to obtain analytical results and insights about the optimal routing policy that minimizes the average total time of call resolution, including callbacks. They establish the fact that: for each call, both the call resolution probability (P) and average service time $(^{1}/m)$ are customer service dependent.

Garcia. et al (2012), noted that as time spent on queue at the call centers increases, it becomes unacceptable for customers, and this affect their satisfaction level. A survey research was conducted using Univariate Analysis of Variance (ANOVA) to determine customer's perception of their wait experience at call centers. From their result the researchers argued that though the time spent on the queue waiting can lead to customer's dissatisfaction. Nevertheless, it is not as important as the agent's ability. More so, the concept of routing rules to be deployed for efficient call resolution rate was not emphasized. Dabrowski (2013) observed that the key performance indicators to measure call center performance are not effectively maximized. Metrics such as average speed of answer, cost per call, agent utilization rate, first contract resolution rate, customer satisfaction and aggregate call center performance. The researcher used CallLogic system to improve the fundamental call routing logic of the Northeast Utilities call centers. Although the findings of the CallLogic system lead to discoveries and ideas on how to improve the fundamental call routing logic of the Northeast Utilities call centers, the CallLogic project achieved high success in the average call handling time. The study only made mention of call Resolution rate and its impact on operational success.

The quality of service accessibility and customer waiting time are dominant performance measures (Vericourt and Zhou 2005). Hence capacity planning and call routing software system strive to minimize cost while achieving self imposed service level constraints, hence considering low average time waiting in queue, these approach do not consider the quality of service rendered to customers (Vericourt and Zhou 2005). Low quality of service has significant impact on the call center operations; this operational impact of service failure is often ignored by call center capacity planning and call routing management system. Their work was motivated by the fact that a major European telecommunications service provider discovered that customers needed to talk to more than three different agents before their problems are resolved. Read (2002), also observed that when using routing rules that emphases on reducing queues, calls are quickly routed to agents, without considering the root of the problem being fixed, and avoid the reoccurrence of such calls. Garcia et al (2012) also noted that call center managers and decision makers tends to only look for information that simply confirms existing beliefs and often disregard all other information, the authors believes that these will enable such call center operators implement a convenient routing rule even if it is not the optimal rule.

Gans et al (2003) and Aksin et al (2007), conducted study on the concept of customer waiting time on the queue, these researchers focused on queues, staffing and performance analysis which are input into personal scheduling and rostering models. Gans et al (2010), empirically study the agent's heterogeneity in Average Handling Time (AHT) not on CR routing rules. Majority of the researches conducted in the domain of call center management were focused on reducing waiting time on the queue and how it impacts on customer's satisfaction and loyalty. In a related study by Gong et al (2015), they modeled with repeat and impatient customer behaviors. Their model has two sectors, representing the feedbacks of repeat behavior of customer and abandonment rate. The performance metric of abandonment is the loss of customers based on waiting time; the researchers further explained that the metric of satisfaction with waiting experiences is used to build a link between staffing costs and call center customer revenues. They considered a call center model with a single class of customers and homogeneous and parallel agents, the analysis of Process-Related Metrics of Call Center. The model of the abandonment behavior was developed by the extension of the Erlang-A formula, which can be viewed as an M/M/s+M queuing system with feedback. Let T denote the random variable measuring patience times with rate γ. The queuing discipline was a First-Come-First-Served (FCFS) approach.

Mehrotra et al (2012), maximizes CR routing rules as one the metrices for call center operational performance. They modelled an optimization problem that focus on call resolution only, taking into cognisance the work of L'Ecuyer (2006) and Gans et al (2010), that minimized wait time routing rule. However, the limitation observed from Mehrotra et al is that their proposed minimized waiting time and maximized call resolution, was implemented using FCFS and RP for their optimization, which did not consider optimal solution in each category.

3. RESEARCH METHODOLOGY

The research was conducted using Global communications as a case for the study. A structured Interview was carried out at Global Communications call centre, Lagos in Nigeria. This was to determine mode of operation and possible routing rules been adopted by the call centre. Three (3) personnel were interviewed at the call centre i.e. the database administrator, a call centre agent and a call centre Supervisor. They were interviewed because of the nature of their job description in the organisation and also interviewing and extracting information from them will be relevant to the study.

Haven understood the call centre operations from field study, a further request was made for call centre data from its automated data logging system comprising of agent identity, calls attended to, call handling time, call status, etc. These data were used to test each of the seven routing rules to determine their performance. A JAVA simulation program was designed for each of the routing rules using the data collected from the call center. The result from the simulation gave the optimal rules for both wait-times oriented and Call Resolution oriented routing rules.

The data collected from Global communication call center was limited to eight categories of call centre agents including:

1. 121 call Agents
2. General call Agents
3. Pidgin call Agents
4. Igbo call Agents
5. Hausa call Agents
6. Premium call Agents
7. Yoruba call Agents
8. Sim registration call Agents

3.1 Model Approach

In this model, we consider multiple call types (indexed by i = 1, 2 ...y) and multiple agent groups (indexed by j = 1, 2 ...z). Calls of type i arrive at a rate of $\wedge_{\mathbf{i}}$. There are n_j agents in group j, with $n_j\varepsilon Z^+$ and each agent in group j serves call type i with rate μ_{ij}. Here we allow agents to handle only a subset of all the call types. If agent group j is not capable of handling call type i then $\mu_{ij} = 0$. When $\mu_{ij}> 0$ we say there is a "match" between call type i and agent group j. In addition, we assume independence of past history each agent of group j has a resolution probability for each call of type i of $p_{ij}\varepsilon$ [0, 1].

In the routing rules, $Q_i(t)$ represents the number of type i customers waiting for service at time t and $f_j(t)$ be the number of available agents of type j who are free at time t, where $0 \le f_j (t) \le n_j$, for all j, t.
Formally, we use the term "routing rule" to mean both the logic that determines to which agent group an arriving call is assigned if there are no calls in queue and agents from multiple groups are free as well as the logic that determines which call an agent is assigned to handle when he/she becomes free when calls from more than one type are in queue waiting for service.

3.2 Models for Existing Routing Rules

As adapted from Mehrotra et al. (2012), "the benchmark routing rule will be the First-Come-First-Served/Longest-Wait (FCFS/LW) rule", because this the routing rule deployed in Global communications, MTN call centers call and in majority of other call centers, which we specify with the rules as follows.

First Come First Serve/ Longest Waiting (FCFS/LW)
When a call arrives and finds no calls of that type in queue and agents of one or more matching group available assigns that call to the agent who has been free the longest, regardless of his/her group.

Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and
Let $f_j(t)$ be the number of available agents of type j who are free at time t,
Where $0 \le f_j (t) \le n_j$, for all j, t.
Let Multiple call types be indexed by i = 1, 2 ...I and
Let Multiple agent groups be indexed by j = 1, 2 ...J.
Calls of type i arrive at a rate of $\wedge_{\mathbf{i}}$.
There are n_j agents in group j, with $n_j\varepsilon Z^+$
Each agent in group j serves call type i with rate μ_{ij}
/Here we allow agents to be trained to handle only a subset of all the call types/
If agent group j is not capable of handling call type I then $\mu_{ij} = 0$
When $\mu_{ij}> 0$ we say there is a "match" between call type i and agent group

In addition, we assume independent of past history each agent of group j has a resolution probability for each call of type i of $p_{ij}\varepsilon$[0, 1].

When an agent of group j becomes free, assign that agent to the call that, among all matching call types, has been waiting the longest regardless of its type.

Similarly, if a call arrives and finds no calls of that type in queue and agents of one or more matching group available assigns that call to the agent who has been free the longest, regardless of his/her group.

Below, we introduce several other routing rules whose performance we will compare to that of FCFS/LW.

Waiting-Time Routing Rules
When the system is in a state with multiple routing options – more than one idle server available from the point of view of an arriving customer, or more than one waiting customer available to be served from the point of view of a ready agent the calls are routed such that no call will go unanswered if there are matching agents available (Mehrotra et al 2012).

Fastest Call First Rule (FCF)

A call of a particular type that arrives when agents of multiple matching groups are free will be routed to a matching agent group that has the highest service rate for that call type.

“Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and
Let $f_j(t)$ be the number of available agents of type j who are free at time t,
Where $0 \leq f_j (t) \leq n_j$, for all j, t.
Let Multiple call types be indexed by i = 1, 2 ...I and
Let Multiple agent groups be indexed by j = 1, 2 ...J.
Calls of type i arrive at a rate of ^i.
There are n_j agents in group j, with $n_j \varepsilon Z^+$
Each agent in group j serves call type i with rate μ_{ij}
/Here we allow agents to be trained to handle only a subset of all the call types/
If agent group j is not capable of handling call type I then $\mu_{ij}= 0$
When $\mu_{ij}> 0$ we say there is a “match” between call type i and agent group
In addition, we assume independent of past history each agent of group j has a resolution probability for each call of type i of $p_{ij}\varepsilon[0, 1]$.
When an agent of group j becomes free, select a call of type i,
Where i = $\text{argmax}_{i:Qi(t)>0}\{\mu_{ij} \,|\mu_{ij}> 0\}$;

/therefore an agent coming free will choose the matching call type for which he/she has the highest service rate/

If an arriving call of type i find no calls of that type waiting for service and agents of one or more matching group available select an agent of group j
Where j = $\text{argmax}_{j:fj(t)>0}\{\mu ij \,|\mu ij> 0\}$;

/that is, a call of a particular type that arrives when agents of multiple matching groups are free will be routed to a matching agent group that has the highest service rate for that call type

Shortest Service Time First (SSTF)

A call of a particular type that arrives when agents of multiple matching groups are free will be routed to a matching agent group that has the relatives Shortest Service Time for that call type.
Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and
Let $f_j(t)$ be the number of available agents of type j who are free at time t,

Where $0 \leq f_j (t) \leq n_j$, for all j, t.
Let Multiple call types be indexed by i = 1, 2 ...I and
Let Multiple agent groups be indexed by j = 1, 2 ...J.
Calls of type i arrive at a rate of ^i.
There are n_j agents in group j, with $n_j \varepsilon Z^+$
Each agent in group j serves call type i with rate μ_{ij}
/Here we allow agents to be trained to handle only a subset of all the call types/
If agent group j is not capable of handling call type I then $\mu_{ij} = 0$
When $\mu_{ij}> 0$ we say there is a “match” between call type i and agent group
In addition, we assume independent of past history each agent of group j has a resolution probability for each call of type i of $p_{ij}\varepsilon$ [0, 1].

When an agent of group j becomes free, select $\text{argmax}_{i:Qi(t)>0}\{\mu_{ij} - \max_{k\neq j}\mu_{ik} \,|\mu_{ij}> 0\}$

/that is, an agent coming free will choose the matching call type for which she has the highest relative service rate/

Similarly, if an arriving call of type i finds no calls of that type waiting for service and agents of one or more matching groups available, select an agent of group j,
Where j = $\text{argmax}_{j:fj(t)>0}\{\mu_{ij} - \max_{k\neq j}\mu_{ik} \,|\mu_{ij}> 0\}$

/that is, a call of a particular type that arrives when agents of multiple matching groups are free will be routed to a matching agent group that has the highest relative service rate for that call type/

Highest Service Time First (HSTF)

A call of a particular type that arrives when agents of multiple matching groups are free will be routed to a matching agent group that has the highest Service Time for that call type.

Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and

Let $f_j(t)$ be the number of available agents of type j who are free at time t,

Where $0 \leq f_j\ (t) \leq n_j$, for all j, t.

Let Multiple call types be indexed by i = 1, 2 ...I and

Let Multiple agent groups be indexed by j = 1, 2 ...J.

Calls of type i arrive at a rate of $\wedge_{\mathbf{i}}$.

There are n_j agents in group j, with $n_j \varepsilon Z^+$

Each agent in group j serves call type i with rate μ_{ij}

/Here we allow agents to be trained to handle only a subset of all the call types/

If agent group j is not capable of handling call type I then $\mu_{ij}= 0$

When $\mu_{ij}> 0$ we say there is a “match” between call type i and agent group

/In addition, we assume independent of past history/

Each agent of group j has a resolution probability for each call of type i of $p_{ij}\varepsilon[0, 1]$.

When an agent of group j becomes free, select a call of matching type i,

Where i = $\text{argmax}_{i:Qi(t)>0}\{p_{ij}\mu_{ij}\ |\mu_{ij}> 0\}$;

/that is, an agent coming free will choose the matching call type for which she has the highest effective service rate/

Similarly, if an arriving call of type i find no calls of that type waiting for service and agents of one or more matching groups available select a matching agent group j

Where j = $\text{argmax}_{j:fj(t)>0}\{p_{ij}\mu_{ij}\ |\mu_{ij}> 0\}$

/that is, a call of a particular type that arrives when agents of multiple matching groups are free will be routed to a matching agent group that has the highest effective service rate for that call type/

Resolution Probabilistic Routing Rules

While the rules in the previous section are focused on minimizing the expected time spent waiting per customer, some call centres may place a much higher priority on CR rates. Thus, in this section we describe routing rules that explicitly emphasize CR rates. (Garcia et al 2012, Aksin et al 2007, and Vericourt and Zhou 2005)

Shortest Queue Routing (SQR)

A call of a particular type that arrives when multiple agents are free will be routed to an agent from the group that has the shortest queue for that call type.

Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and

Let $f_j(t)$ be the number of available agents of type j who are free at time t,

Where $0 \leq f_j\ (t) \leq n_j$, for all j, t.

Let Multiple call types be indexed by i = 1, 2 ...I and

Let Multiple agent groups be indexed by j = 1, 2 ...J.

Calls of type i arrive at a rate of $\wedge_{\mathbf{i}}$.

There are n_j agents in group j, with $n_j \varepsilon Z^+$

Each agent in group j serves call type i with rate μ_{ij}

/Here we allow agents to be trained to handle only a subset of all the call types/

If agent group j is not capable of handling call type I then $\mu_{ij} = 0$

When $\mu_{ij}> 0$ we say there is a “match” between call type i and agent group

/In addition, we assume independent of past history/

Each agent of group j has a resolution probability for each call of type i of $p_{ij}\varepsilon$ [0, 1].

When an agent of group j becomes free,

Select $\text{argmax}_{i:Qi(t)>0}\{p_{ij}\mu_{ij} – \text{max}_{k\neq j}p_{ik}\mu_{ik}\ |\mu_{ij}> 0\}$

/that is, an agent coming free will choose the matching call type for which she has the highest relative effective service rate/

Similarly, if an arriving call of type i finds no calls of that type waiting for service and agents of one or more matching group available, select a matching agent group j

Where j = $\text{argmax}_{j:fj\ (t)>0}\{p_{ij}\mu_{ij} – \text{max}_{k\neq j}p_{ik}\mu_{ik}\ |\mu ij> 0\}$

/that is, a call of a particular type that arrives when multiple matching agents are free will be routed to an agent from the matching group that has the highest relative effective service rate for that call type also referred to as the shortest queue for that call type/

Probabilistic Routing (PR): A call of a particular type that arrives when multiple agents are free will be routed to an agent from the group that has the highest resolution probability for that call type.
Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and
Let $f_j(t)$ be the number of available agents of type j who are free at time t,
Where $0 \leq f_j (t) \leq n_j$, for all j, t.
Let Multiple call types be indexed by i = 1, 2 ...I and
Let Multiple agent groups be indexed by j = 1, 2 ...J.
Calls of type i arrive at a rate of λ_i.
There are n_j agents in group j, with $n_j \varepsilon Z^+$
Each agent in group j serves call type i with rate μ_{ij}
/Here we allow agents to be trained to handle only a subset of all the call types/
If agent group j is not capable of handling call type I then $\mu_{ij} = 0$
When $\mu_{ij} > 0$ we say there is a "match" between call type i and agent group
/In addition, we assume independent of past history/
Each agent of group j has a resolution probability for each call of type i of $p_{ij} \varepsilon [0, 1]$.
When agent j becomes free, select $\text{argmax}_{i:Qi(t)>0}\{p_{ij} \mid \mu_{ij} > 0\}$
/that is, that agent will be assigned a call of the type that she is most likely to resolve, regardless of waiting times and queue lengths/
Similarly, if an arriving call of type i finds no calls of that type waiting for service and agents of one or more group available, assign that call an agent of group j,
Where $j = \text{argmax}_{j:fj(t)>0}\{p_{ij} \mid \mu_{ij} > 0\}$
/that is, a call of a particular type that arrives when multiple agents are free will be routed to an agent from the group that has the highest resolution probability for that call type/ (Mehrotra et al, 2012)

Relative Resolution Probability Routing (RRPR): a call of a particular type that arrives when multiple agents are free will be routed to an agent from the group that has the highest relative resolution probability for that call type.
Let $Q_i(t)$ represents the number of type i customers waiting for service at time t and
Let $f_j(t)$ be the number of available agents of type j who are free at time t,
Where $0 \leq f_j (t) \leq n_j$, for all j, t.
Let Multiple call types be indexed by i = 1, 2 ...I and
Let Multiple agent groups be indexed by j = 1, 2 ...J.
Calls of type i arrive at a rate of λ_i.
There are n_j agents in group j, with $n_j \varepsilon Z^+$
Each agent in group j serves call type i with rate μ_{ij}
/Here we allow agents to be trained to handle only a subset of all the call types/
If agent group j is not capable of handling call type I then $\mu_{ij} = 0$
When $\mu_{ij} > 0$ we say there is a "match" between call type i and agent group
/In addition, we assume independent of past history/*
Each agent of group j has a resolution probability for each call of type i of $p_{ij} \varepsilon [0, 1]$.
When agent j becomes free, select $\text{argmax}_{i:Qi(t)>0}\{p_{ij} - \max_{k \neq j} p_{ik} \mid \mu_{ij} > 0\}$*
/that is, that agent will be assigned a call of the type that she is relatively most likely to resolve/
Similarly, if an arriving call of type i finds no calls of that type waiting for service and agents of one or more group available, assign that call an agent of group j,
Where $j = \text{argmax}_{j:fj(t)>0}\{p_{ij} - \max_{k \neq j} p_{ik} \mid \mu_{ij} > 0\}$
/that is, a call of a particular type that arrives when multiple agents are free will be routed to an agent from the group that has the highest relative resolution probability for that call type/ (Mehrotra et al, 2012)".

3. DESIGN AND IMPLEMENTATION

Simulation Process

This call center simulation is a java based application which uses a java simulation Application programming interface (API) (stochastic simulation library) as its back-bone for its implementation. This documentation gives a detailed process of how it works.

Stochastic simulation Library

SSJ is a Java library for stochastic simulation, developed in the Department d'Informatique et de Recherche Operationnelle (DIRO), at the Universite de Montreal. It provides facilities for generating uniform and non uniform random variates, computing different measures related to probability distributions, performing goodness of fit tests, applying quasi-Monte Carlo methods, collecting statistics (elementary), and programming discrete-event simulations with both events and processes.

Classes

This program consists of several classes and methods which are highlighted below:

Call class

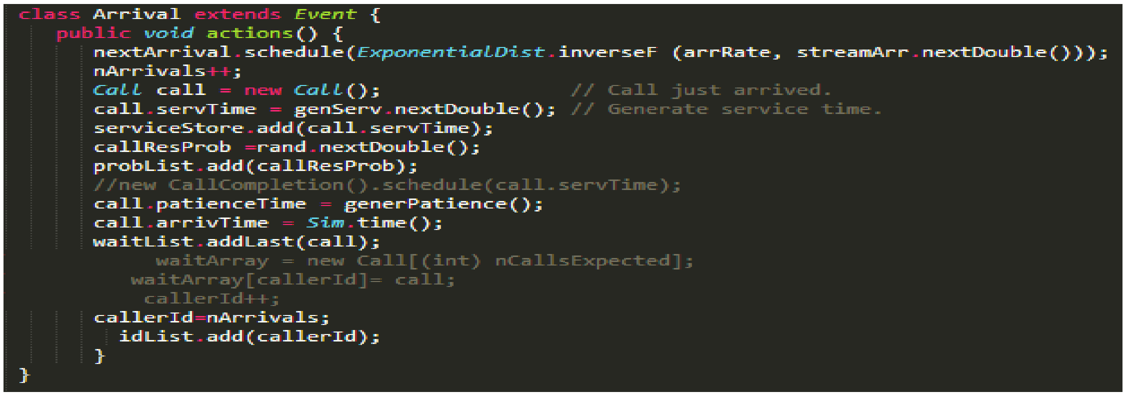
This class tries to simulate the behaviour of a call as it relates to a call center which involves the arrival time of the call, service time of the call and also the waiting time of the call. In this class the constructor is used to determine if there are free agents to handle the call and also assign the call to an agent or put the call in a waiting state if no free agent is available. This call is where the routing rules are implemented using the service time and waiting time of the calls.

Als

o in the call class there is an end wait method which is called when there are free agents to handle calls. The method also determines if a call is abandoned or picked. A call is abandoned if the wait time is greater than the patience time assigned to the call and a call is answered if its wait time is less than its patience time.

Arrival Class

The arrival class determines the arrival time of each calls as they arrive into the system. The class extends the super class of the SSJ library called Event which is an abstract class that provides event scheduling tools. Arrival class contains method action which is used to determine the action to be performed when the call event occur, which in this case is assigning an arrival time to the call, the call class is instantiated in this method which uses the time allocated for handling the call.

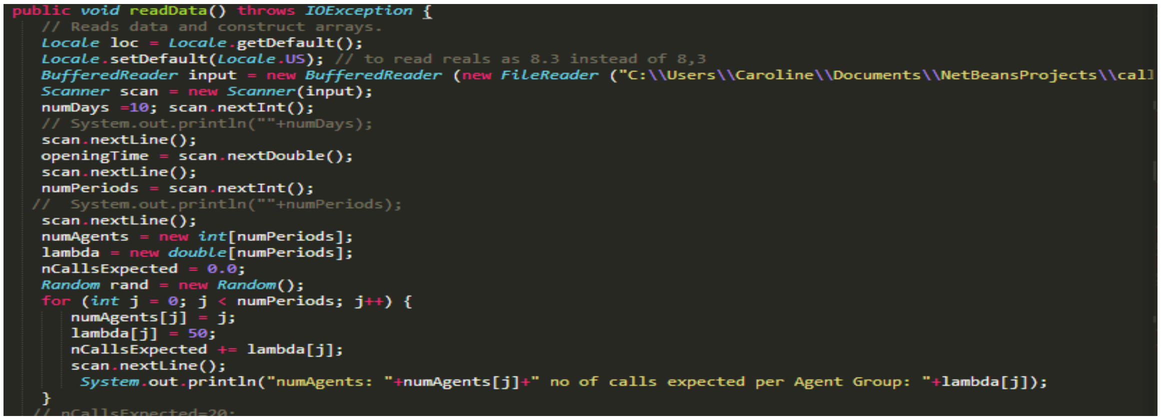


```
class Arrival extends Event {
    public void actions() {
        nextArrival.schedule(ExponentialDist.inverseF (arrRate, streamArr.nextDouble()));
        nArrivals++;
        Call call = new Call();               // Call just arrived.
        call.servTime = genServ.nextDouble(); // Generate service time.
        serviceStore.add(call.servTime);
        callResProb =rand.nextDouble();
        probList.add(callResProb);
        //new CallCompletion().schedule(call.servTime);
        call.patienceTime = generPatience();
        call.arrivTime = Sim.time();
        waitList.addLast(call);
             waitArray = new Call[(int) nCallsExpected];
           waitArray[callerId]= call;
            callerId++;
        callerId=nArrivals;
          idList.add(callerId);
        }
}
```

Figure 1: The Arrival Class

readData() Method

This method takes a string value as argument, which can be the name of the file or the absolute path of the file. This readData() method reads the content of the data file using the Buffered Reader, Scanner and a File reader class of java, the method extracts the number of calls expected, number of agents in the center, simulation start time and number of periods. The file used in this case is a “DAT” type file with file name “call.dat”.

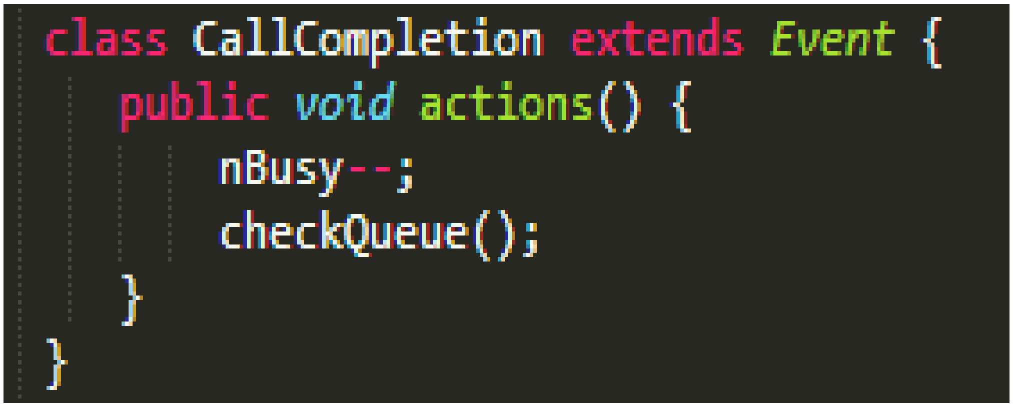


```
public void readData() throws IOException {
   // Reads data and construct arrays.
   Locale loc = Locale.getDefault();
   Locale.setDefault(Locale.US); // to read reals as 8.3 instead of 8,3
   BufferedReader input = new BufferedReader (new FileReader ("C:\\Users\\Caroline\\Documents\\NetBeansProjects\\call
   Scanner scan = new Scanner(input);
   numDays =10; scan.nextInt();
   // System.out.println(""+numDays);
   scan.nextLine();
   openingTime = scan.nextDouble();
   scan.nextLine();
   numPeriods = scan.nextInt();
  //  System.out.println(""+numPeriods);
   scan.nextLine();
   numAgents = new int[numPeriods];
   lambda = new double[numPeriods];
   nCallsExpected = 0.0;
   Random rand = new Random();
   for (int j = 0; j < numPeriods; j++) {
      numAgents[j] = j;
      lambda[j] = 50;
      nCallsExpected += lambda[j];
      scan.nextLine();
       System.out.println("numAgents: "+numAgents[j]+" no of calls expected per Agent Group: "+lambda[j]);
   }
  // nCallsExpected=20;
```

Figure 2: readData() method

Call Completion class

The call completion class also extends the event class of SSJ, this class simply handles the termination of a call after it has been handled by a call center agent. This class has an action() method that handles the actions to be executed after a call completes and is terminated, actions like reducing the number of busy agents and calling the checkQueue() method that fetches calls in the waiting list queue.

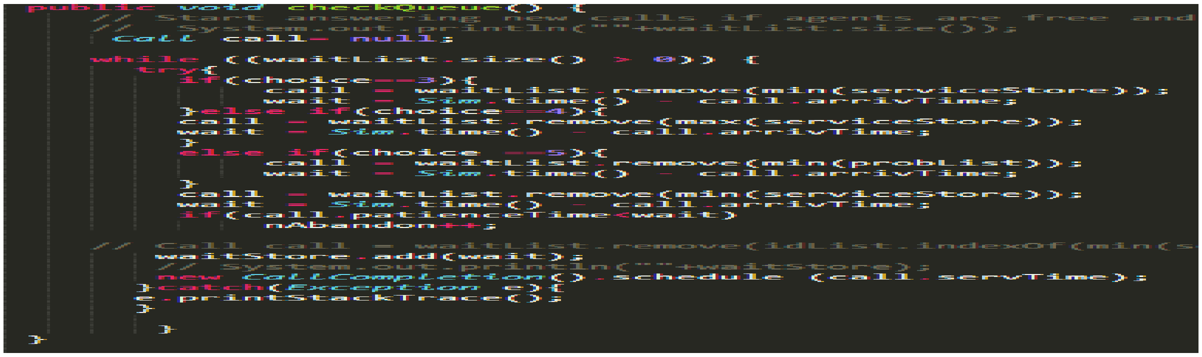


```
class CallCompletion extends Event {
   public void actions() {
      nBusy--;
      checkQueue();
   }
}
```

Figure 3: The Call Completion Class

Check Queue() method

This method checks if the waiting list is not empty and if there are free agents before it retrieves a call from the waiting list Queue and in the process it ends the waits of the call.

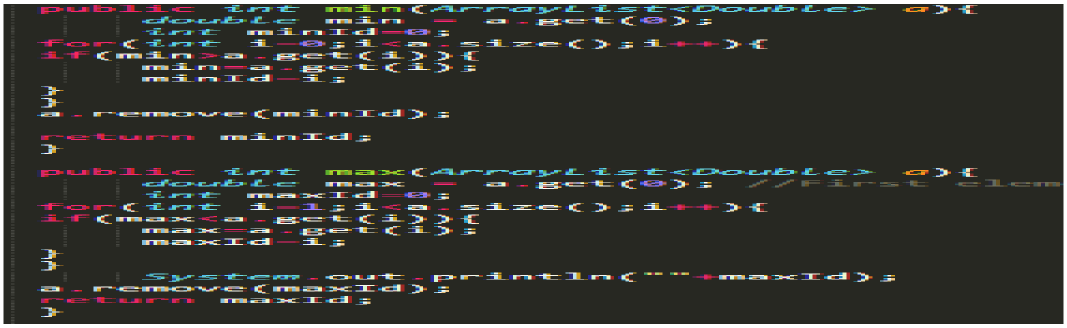


```
public void checkQueue() {
      // Start answering new calls if agents are free and
      // System.out.println(""+waitList.size());
      Call call= null;

      while ((waitList.size() > 0)) {
         try{
            if(choice==3){
                  call = waitList.remove(min(serviceStore));
                  wait = Sim.time() - call.arrivTime;
            }else if(choice==4){
            call = waitList.remove(max(serviceStore));
            wait = Sim.time() - call.arrivTime;
            }
            else if(choice ==5){
                  call = waitList.remove(min(probList));
                  wait = Sim.time() - call.arrivTime;
            }
            call = waitList.remove(min(serviceStore));
            wait = Sim.time() - call.arrivTime;
            if(call.patienceTime<wait)
                  nAbandon++;

      // Call call = waitList.remove(idList.indexOf(min(s
         waitStore.add(wait);
         // System.out.println(""+waitStore);
         new CallCompletion().schedule (call.servTime);
        }catch(Exception e){
        e.printStackTrace();
        }

         }
 }
```

Figure 4: check Queue() method

Max() and min() methods

This method calculates the maximum value of the array passed as argument into it. This is used to calculate the highest resolution, lowest service time etc depending on the routing rule used to route the call.

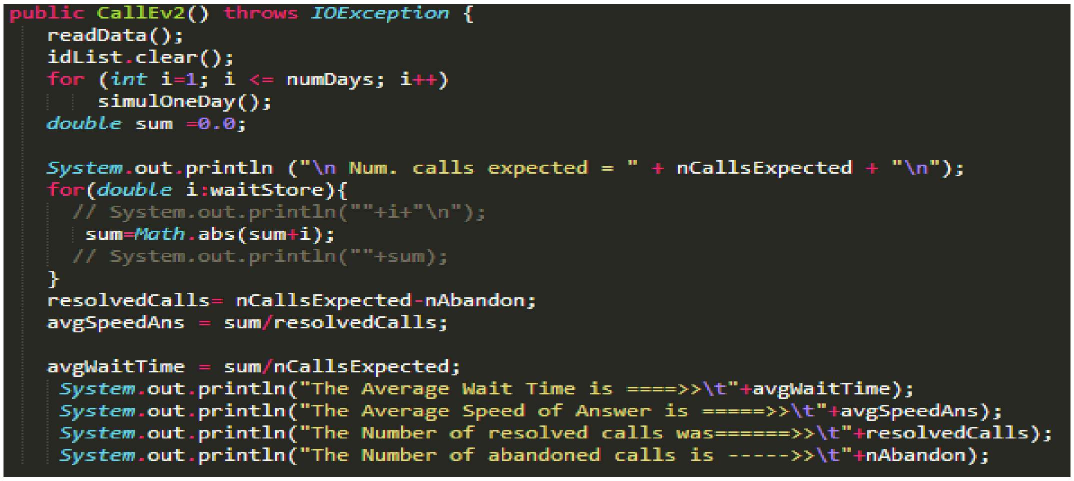


```
public int min(ArrayList<Double> a){
        double min = a.get(0);
        int minId=0;
for(int i=0;i<a.size();i++){
if(min>a.get(i)){
        min=a.get(i);
        minId=i;
}
}
a.remove(minId);
return minId;
}
public int max(ArrayList<Double> a){
        double max = a.get(0); //First elem
        int maxId=0;
for(int i=1;i<a.size();i++){
if(max<a.get(i)){
        max=a.get(i);
        maxId=i;
}
}
        System.out.println(""+maxId);
a.remove(maxId);
return maxId;
}
```

Figure 5: max() and min() methods

The callEV() Constructor

This is the constructor which has the same name as the java file associated with the program. This constructor is used to initialize the class once instantiated in any part of the program (in this case the main method). In this constructor the "Average Waiting Time", "Average Speed of Answer", "Number of Resolved calls" and "Number of Abandoned Calls" was calculated and outputted into the console screen.



```
public CallEv2() throws IOException {
   readData();
   idList.clear();
   for (int i=1; i <= numDays; i++)
       simulOneDay();
   double sum =0.0;

   System.out.println ("\n Num. calls expected = " + nCallsExpected + "\n");
   for(double i:waitStore){
     // System.out.println(""+i+"\n");
      sum=Math.abs(sum+i);
     // System.out.println(""+sum);
   }
   resolvedCalls= nCallsExpected-nAbandon;
   avgSpeedAns = sum/resolvedCalls;

   avgWaitTime = sum/nCallsExpected;
    System.out.println("The Average Wait Time is ====>>\t"+avgWaitTime);
    System.out.println("The Average Speed of Answer is =====>>\t"+avgSpeedAns);
    System.out.println("The Number of resolved calls was======>>\t"+resolvedCalls);
    System.out.println("The Number of abandoned calls is ----->>\t"+nAbandon);
```

Figure 6: The CallEv() Constructor

3.1 Simulation Procedure

Having presented a diverse set of routing rules, we therefore, determine how well each of these routing rules performs. In particular, we define the performance of these routing rules in terms of the two key performance metrics of overall average speed of answer (ASA) and aggregate call resolution (CR) rate. For the call center simulation process, we conducted an extensive simulation study based on data obtained from Global communications call customer service call centre. Below we describe the operational input data, the simulation modelling platform, the program structure and then present and discuss the results from the simulation.

The simulation contained as inputs the date and time of the call, the unique ID number for the agent who handled the call, the Call Type for that call, the time spent by the agent on the phone handling the call, or Handle Time (HT) and the resolution status of the call. We used only a subset of the call types and agents to ensure that the run times for our simulations were fast enough to conduct extensive numerical experiments.

The process of selecting and preparing the data to support our numerical experiments during implementation included the following:

1. **Selection of Call Types**: The number of call types is a significant driver of simulation times; hence we considered the largest call types.
2. **Selection of Agents:** We restricted the number of agents in our model to include only those agents who can handle a certain amount of calls.
3. **Agent clustering:** There are a total of 175 agents, and they are clustered into 8 groups. The numbers of agents in each group are given in Table 1.

Table 1: Total number of Agents in Agent groups

Agent Group	1	2	3	4	5	6	7	8
Number in Group	35	35	15	10	10	25	30	15

4. **Arrival Rate Selection:** For our numerical experiments, we chose arrival rates for each of the call types to maintain the same relative proportion of expected calls of call found in the database. The total new arrival (not including call backs of unresolved calls) is set at 2000calls/hour. This defines the proportion of the calls that goes into queues.

Each routing rule was used independently with the collected data to simulate the call centre operation. Simulation was carried out by using the data gathered from the above call centre to estimate parameters needed to characterize the model. At the end of each simulation analysis, it is important to note that the performance of this system is defined in terms of the ASA and the CR rate, and that these output metrics depend not only on the actual numerical values of the input parameters but also on choice of the routing rule that is used to determine which call types are handled by which agents under what conditions. The overall ASA and CR for each routing rule is the weighted average over all agent groups and call types. For example, Table 3 shows a sample result of our simulation analysis presented using Microsoft excel.

3.2 Simulation Platform

The simulation platform consists of a collection of programs that invoke the simulation library. The library contains all the functionality required to run complex discrete-event simulations of contact centre. Following every service event, the program generates a uniform random variable and compares it to the agent's resolution probability to determine if a callback event occurs.

- Simulation run length: For each of the rules described, we simulated for 2000 calls for a period of one (1) hour for some realizations.
- Simulating call-backs: For all rules, we assume that unresolved calls result in immediate call-backs into the call centre.

3.3 Data used for Simulation

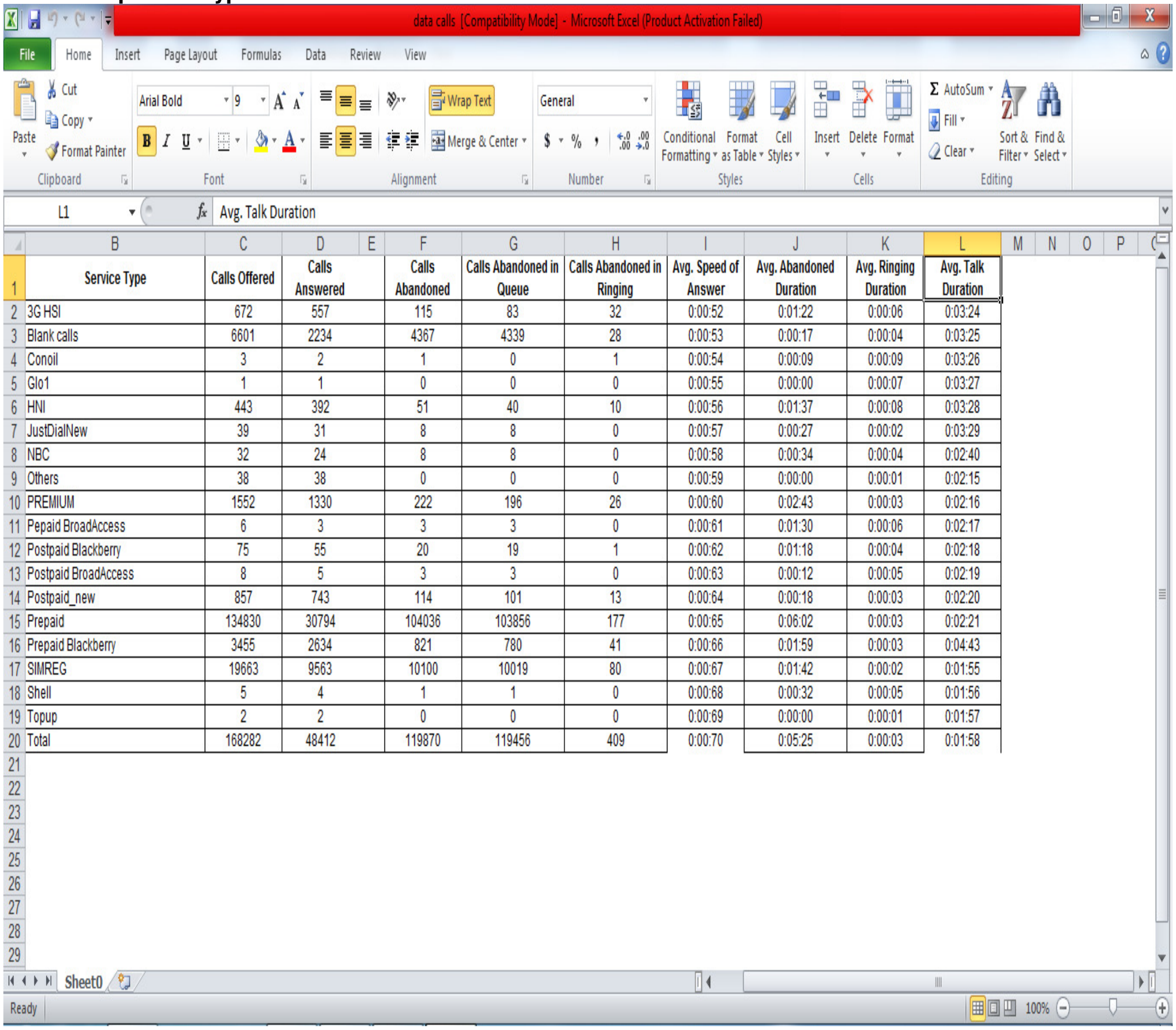
The tools used for the simulation is a collection of Java simulation libraries programs. We also used Microsoft Excel to do some basic data analysis and graphical presentation of results. The data sets used for conducting the simulation for each of the routing rule are obtained from a telecommunications call center.

1. We considered the largest call types
2. We also restricted the number of agents in our model to include only those agents who can handle a certain amount of calls i.e. a minimum of 100 calls.
3. The database used for simulation included records for incoming phone calls. Specifically, each record in our database contained the following fields:
   i. The date and time of the call.
   ii. The unique ID number for the agent who handled the call.
   iii. The Call Type for that call.
   iv. The time spent by the agent on the phone handling the call, or Handle Time (HT).
   v. The resolution status of the call.

We used only a subset of the call types and agents to ensure that the run times for our simulations were fast enough to conduct extensive numerical experiments. The same interface was used for the implementation of the simulation, as well as the input data which were also the same for all routing rule.

The input data in Table 2, shows the various service type, number of call offered, analysis of the number of calls answered, abandoned, average speed of answer, average talk duration and other report from the calls offered.

Table 2: Input call type data for simulation



L1 Avg. Talk Duration

Service Type	Calls Offered	Calls Answered	Calls Abandoned	Calls Abandoned in Queue	Calls Abandoned in Ringing	Avg. Speed of Answer	Avg. Abandoned Duration	Avg. Ringing Duration	Avg. Talk Duration
3G HSI	672	557	115	83	32	0:00:52	0:01:22	0:00:06	0:03:24
Blank calls	6601	2234	4367	4339	28	0:00:53	0:00:17	0:00:04	0:03:25
Conoil	3	2	1	0	1	0:00:54	0:00:09	0:00:09	0:03:26
Glo1	1	1	0	0	0	0:00:55	0:00:00	0:00:07	0:03:27
HNI	443	392	51	40	10	0:00:56	0:01:37	0:00:08	0:03:28
JustDialNew	39	31	8	8	0	0:00:57	0:00:27	0:00:02	0:03:29
NBC	32	24	8	8	0	0:00:58	0:00:34	0:00:04	0:02:40
Others	38	38	0	0	0	0:00:59	0:00:00	0:00:01	0:02:15
PREMIUM	1552	1330	222	196	26	0:00:60	0:02:43	0:00:03	0:02:16
Pepaid BroadAccess	6	3	3	3	0	0:00:61	0:01:30	0:00:06	0:02:17
Postpaid Blackberry	75	55	20	19	1	0:00:62	0:01:18	0:00:04	0:02:18
Postpaid BroadAccess	8	5	3	3	0	0:00:63	0:00:12	0:00:05	0:02:19
Postpaid_new	857	743	114	101	13	0:00:64	0:00:18	0:00:03	0:02:20
Prepaid	134830	30794	104036	103856	177	0:00:65	0:06:02	0:00:03	0:02:21
Prepaid Blackberry	3455	2634	821	780	41	0:00:66	0:01:59	0:00:03	0:04:43
SIMREG	19663	9563	10100	10019	80	0:00:67	0:01:42	0:00:02	0:01:55
Shell	5	4	1	1	0	0:00:68	0:00:32	0:00:05	0:01:56
Topup	2	2	0	0	0	0:00:69	0:00:00	0:00:01	0:01:57
Total	168282	48412	119870	119456	409	0:00:70	0:05:25	0:00:03	0:01:58

3.4 Implementation Interface

The application is a standalone application. On executing the program, the screenshots showing the simulation processes are shown in Figures 7-13 in Appendix 1.

3.5 Performance Criteria

To determine how well each of these rules performs, in particular, our proposed system, we defined the performance of these routing rules in terms of the two key performance metrics of overall Average Speed of Answer and aggregate Call Resolution rate. These output metrics depend on the actual numerical values of the input parameters (arrival rates, service rates and call resolution probabilities) and also on the chosen routing rule that is used to determine which call types are handled by which agents under what conditions. The performance criterion was to determine the optimal routing rule for waiting time and CR routing rules.

4. DISCUSSION OF RESULTS

This section deals with discussion of the results from the simulation process.

Table 3 shows the call resolution and average speed of answer for each of the routing.

Table 3: Weighted Average Results for evaluation obtained from simulation Analysis

RULE	CR	ASA (seconds)	Non CR	RESOLVED CALLS	CALL BACKS	% resolved calls	% Call backs
FCFS/LW	1552	47	448	0.431111111	0.124444444	71.85185185	20.74074074
FCF	1683	36	317	0.4675	0.088055556	77.91666667	14.67592593
SSTF	1935	28	65	0.5375	0.018055556	89.58333333	3.009259259
HSTF	1268	95	732	0.352222222	0.203333333	58.7037037	33.88888889
SQR	1795	34	205	0.498611111	0.056944444	83.10185185	9.490740741
PR	1775	39	225	0.493055556	0.0625	82.17592593	10.41666667
RRPR	1685	78	315	0.423611111	0.071944444	77.9480110	14.5519850

The **SSTF** rule features the lowest ASA, it also results in a higher CR rate than **SQR**, which suffers only slightly higher ASA values. Taken together, these results clearly demonstrate that optimal routing decisions can have a significant positive impact on operational performance. Figure14, shows the weighted average for the speed to answer for waiting time routing rules, from the graph, SSTF has an ASA of 28 seconds, FCF has 36 seconds, HSTF has 47 seconds and FCFS/LW has 95 secnods. This clearly demostrated that SSTF has the lowest ASA amongst the waiting time routing rules, and it is also the optimal routing rule. Figures 15-19 in Appendix 2 shows the entire results generated from the simulation.

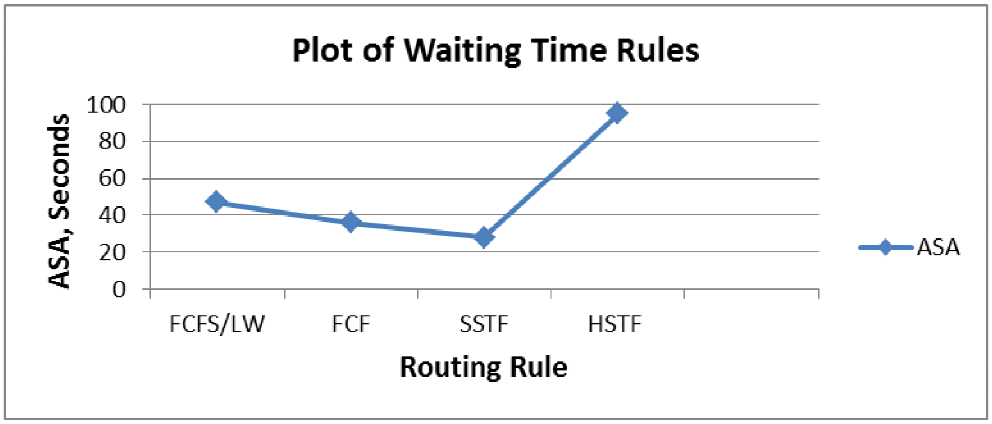


Figure 7: ASA for waiting time routing rules

Table 3, shows that SSTF and SQR has high call CR rate and the callback rate is less than 10% for both of the rules. This justifies that SSTF and SQR are the optimal both for wait-time routing and CR routing rule respectively, from literature and practically from simulation result. The evaluation results to further justify that SSTF and SQR are the optimal routing rules are displayed graphically in the Appendix

Table 4: System CR and callback rates of considered routing rules

RULE	CR (%)	CALL BACKS RATES (%)
SSTF	**89.58333**	**3.009259**
SQR	**83.10185**	**9.490741**

4.1 Proposed System

From our simulation result, we further developed a system model that will be subject to further justification.The model in Figure 19 is the overall system approach which depicts how call centre agents are saddled with the responsibility of attending to customer issues. Due to the volume of customer calls, most call centres employs multiple agents to attend to customer issues. Our simulation model retrieves data from the data logging system, and the data is used to conduct simulation for each of the seven routing rules. The evaluator further evaluates the simulation result to determine the optimal routing rules for both waiting time and CR routing rules respectively. By a comparative data analysis the optimal routing rules are evaluated, hence the result of the comparative analysis is high/enhanced CR and low waiting time (ASA). This implies minimised waiting time and maximised CR, which is expressed in Figure 8.

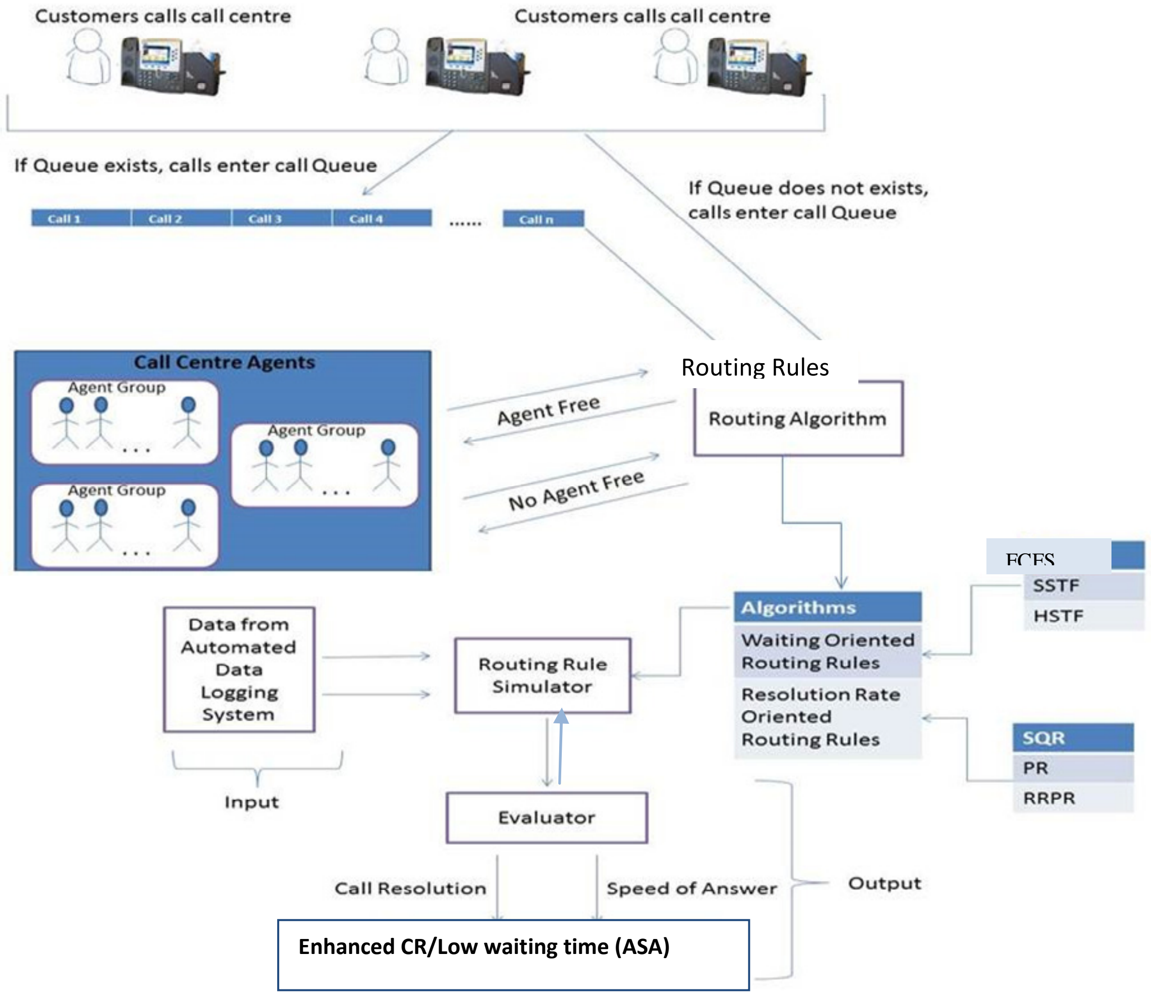

Customers calls call centre
Customers calls call centre
If Queue exists, calls enter call Queue
If Queue does not exists, calls enter call Queue
Call 1
Call 2
Call 3
Call 4
......
Call n
Call Centre Agents
Agent Group
Agent Group
Agent Group
Routing Rules
Routing Algorithm
Agent Free
No Agent Free
FCFS
SSTF
HSTF
Algorithms
Waiting Oriented Routing Rules
Resolution Rate Oriented Routing Rules
SQR
PR
RRPR
Data from Automated Data Logging System
Input
Routing Rule Simulator
Evaluator
Call Resolution
Speed of Answer
Output
Enhanced CR/Low waiting time (ASA)

Figure 8: Proposed System Model

5. CONCLUSION

The study vividly shows that the SSTF and SQR performed optimally both for waiting time and CR routing rules respectively. Our result also shows that for effective operation of call center, the two performance metrics must be integrated for every call initiated. This implies that low wait time (queue reduction) and enhanced and effective CR. Hence this research proposes a hybrid routing rule that will integrate both performance metrics, considering the optimal rule in each category.

We have taken the arrival rates as inputs to our model as time-independent inputs, though in practice all call centers experience different arrival rates at different times of day, which means that the distribution of delay times prior to callbacks can have a significant impact on operational performance. Similarly, we have also taken the number of agents of each group as a time-independent input into our model, though in practice these staffing levels are a function of an underlying scheduling model. Thus, another important related research area is incorporating RP (as inputs) and CR rates (as outputs) into call forecasting and agent scheduling models.

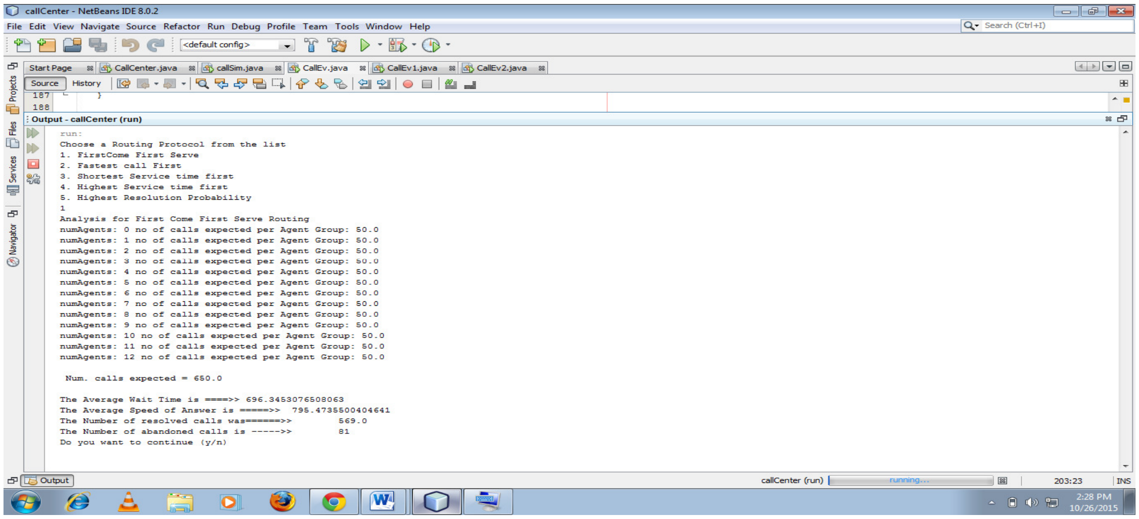
REFERENCES

1. Aksin, Z. Armony, M. and Mehrotra. V. (2007) The modern call-centre: A multi-disciplinary perspective on operations management research. Production and Operations Management, 16(6):665–688, November–December Available at http://www.stern.nyu.edu/om/ faculty/armony/research/CallCentreSurvey.pdf.
2. Armony, M. and Maglaras. C. (2004) On Customer Contact Centers with a Call-Back Option: Customer Decisions, Routing Rules, and System Design. OPERATIONS RESEARCH Vol. 52, No. 2, March–April 2004, pp. 271–292 ISSN 0030-364X _ EISSN 1526-5463 _ 04 _5202 _ 0271
3. Armony .M (2005), Dynamic routing in large-scale service systems with heterogeneous servers. Queueing Systems, 51(3-4):287–329, December 2005.
4. Avramidis, A. N., A. Deslauriers, P. L'Ecuyer. (2004). Modeling daily arrivals to a telephone all center. *Management Science* 50(7) 896–908.
5. Brizola, N, Costa .S, Pazeto .T, and Freitas P. (2001). Planejamento de Capacidade de Call Center. In : ICIE, Flo-rianópolis
6. Cooper. B. (1997). Introduction to Queueing Theory. 2 ed. North Holland, New York.
7. Dabrowski .M. (2013), Business Intelligence In Call Centers. International Journal of Issue Computer and Information Technology (ISSN: 2279 – 0764) Volume 02–02, March 2013. www.ijcit.com
8. Enyioko, N. C. (2016), Relevance of the Queueing Theory to Serviced- Based– Organisations
9. SSRN eLibrary Search Results Service Management eJournal, Medonice Consultiong and Research Institute *Date Posted:* April 01, 2016 Working Paper Series, | Link to this page | Subschttp://papers.ssrn.com/sol3/JELJOUR_Results.cfm?form_name=journalbrowse&journal_id=992385
10. Gans, G. Koole, and A. Mandelbaum. (2003), Telephone call centres: Tutorial, review, and research prospects. Manufacturing & Service Operations Management, 5(2):79–141, Spring 2003.
11. Gans, N., N. Liu, A. Mandelbaum, H. Shen, H. Ye. (2010). Service times in call centers: Agent heterogeneity and learning with Powering Applications—A Festschrift for Lawrence D. Brown, IMS Collections, Vol. 6. Institute of Mathematical Statistics, Beachwood, OH, 99–123. some operational consequences. Borrowing Strength: Theory Powering Applications—A Festschrift for Lawrence D. Brown, IMS Collections, Vol. 6. Institute of Mathematical Statistics, Beachwood, OH, 99–123.
12. Garcia. D, Archer .T, Moradi .S, and Ghiabi .B (2012), Waiting in Vain: Managing Time and Customer Satisfaction at Call Centers. Science Research, http://dx.doi.org/10.4236/psych.2012.32030. Psychology 2012. Vol.3, No.2, 213-216 Published Online February 2012 in SciRes http://www.SciRP.org/journal/psych)
13. Gong J, Yu .M, Tang .J, and Li .M (2015), Staffing to Maximize Profit for Call Centers with Impatient and Repeat-Calling Customers. Mathematical Problems in Engineering Volume 2015, Article ID 926504, 10 pages. Hindawi Publishing Corporation. http://dx.doi.org/10.1155/2015/926504.
14. Hart .M, Fichtner .B, Fjalestad .E, and Langley S. (2006) Contact centre performance: In pursuit of first call resolution. Management Dynamics, 15(4):17–28,
15. L'Ecuyer. P. (2006), Modelling and optimization problems in contact centres. Proceedings of the Third International Conference on the Quantitative Evaluation of Systems - (QEST'06), pages 145–154, 2006.
16. Mehrotra .V, Ross .K, Ryder .G and Zhou .Y (2009), Routing to Manage Resolution and Waiting Time in Call Centers with Heterogeneous Servers. Manufacturing & Service Operations Management Vol. 9, N0. 4, pp. 167-181, ISSN 1523-4614 j EISSN 1526-5498

17. Mehrotra .V, Ross .K, Ryder .G and Zhou .Y (2012), Routing to Manage Resolution and Waiting Time in Call Centers with Heterogeneous Servers. MANUFACTURING & SERVICE OPERATIONS MANAGEMENT Vol. 14, No. 1, Winter 2012, pp. 66–81 ISSN 1523-4614 (print) . ISSN 1526-5498 (online) http://dx.doi.org/10.1287/msom.1110.0349 ©2012 INFORMS
18. Read, B. (2003). Call center checkup. Call Center Magazine (June 1), http://www.icmi.com/Resources/Articles/2003/June/Call-Center -Checkup.
19. Sharma (2010) "Operation Research, theory and application" 4th edition Macmillian publishers.
20. Véricourt, F and. Zhou, P (2005). A routing problem for call centers with customer callbacks after service failure. Operations. Research. 53(6) 968–981.
21. V´ericourt and Zhou .Y (2005). Managing Response Time in a Call-Routing Problem with Service Failure. OPERATIONS RESEARCH INFORMS Vol. 53, No. 6, November–December 2005, pp. 268–281 issn 0030-364X _ issn 1526-5463 _ 05 _ 5306 _0968
22. Zhan .D and Ward .A (2006) Threshold Routing to Trade-off Waiting and Call. Resolution in Call Centers MANUFACTURING & SERVICE OPERATIONS MANAGEMENT Vol. 12, No. 2, pp. 316–323. ISSN 1523-4614 EISSN 1526- *www-bcf.usc.edu/~amyward/ZhWa_7_24_2013MSOM_Body.pd*
23. Véricourt, F and. Zhou, P (2005a). A routing problem for call centers with customer callbacks after service failure. Operations. Research. 53(6) 968–981.
24. V´ericourt and Zhou .Y (2005b). Managing Response Time in a Call-Routing Problem with Service Failure. OPERATIONS RESEARCH INFORMS Vol. 53, No. 6, November–December 2005, pp. 268–281 issn 0030-364X _ issn 1526-5463 _ 05 _ 5306 _0968

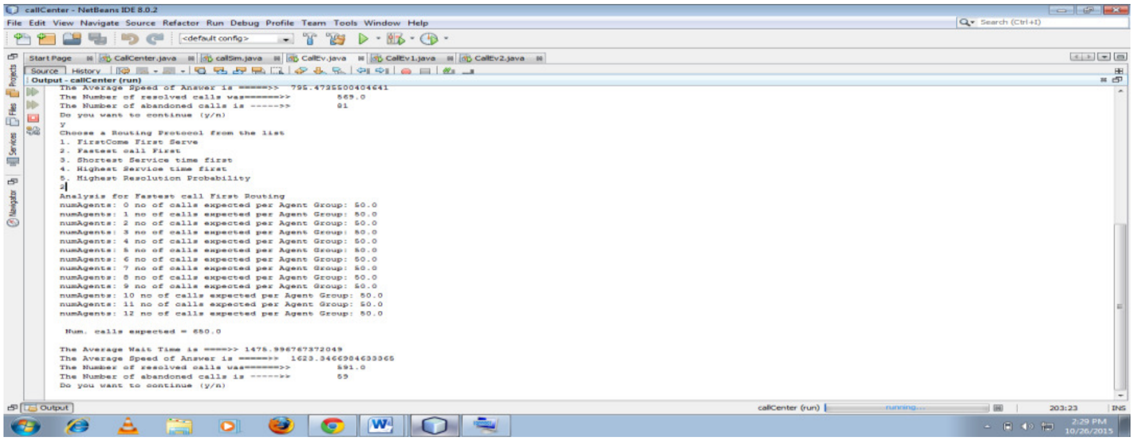
APPENDIX 1

SCREENSHOTS OF SIMULATION PROCESS



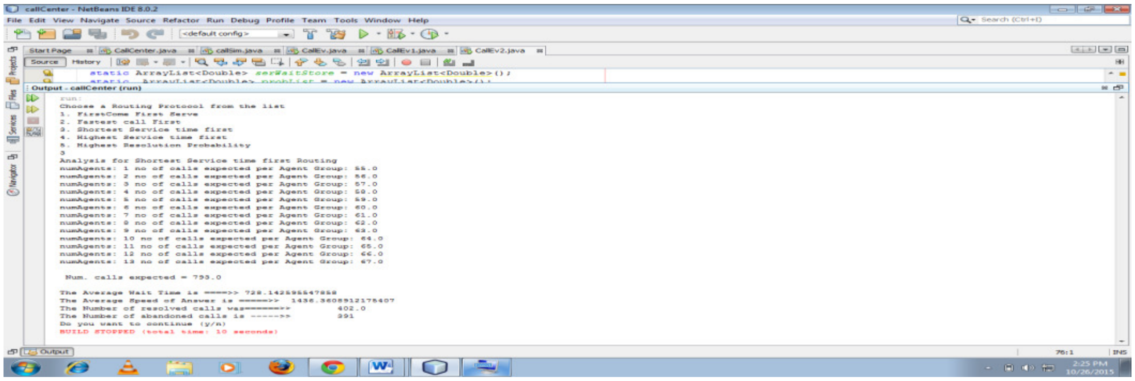
```
run:
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
1
Analysis for First Come First Serve Routing
numAgents: 0 no of calls expected per Agent Group: 50.0
numAgents: 1 no of calls expected per Agent Group: 50.0
numAgents: 2 no of calls expected per Agent Group: 50.0
numAgents: 3 no of calls expected per Agent Group: 50.0
numAgents: 4 no of calls expected per Agent Group: 50.0
numAgents: 5 no of calls expected per Agent Group: 50.0
numAgents: 6 no of calls expected per Agent Group: 50.0
numAgents: 7 no of calls expected per Agent Group: 50.0
numAgents: 8 no of calls expected per Agent Group: 50.0
numAgents: 9 no of calls expected per Agent Group: 50.0
numAgents: 10 no of calls expected per Agent Group: 50.0
numAgents: 11 no of calls expected per Agent Group: 50.0
numAgents: 12 no of calls expected per Agent Group: 50.0

 Num. calls expected = 650.0

The Average Wait Time is ====>> 696.3453076508063
The Average Speed of Answer is =====>>  795.4735500404641
The Number of resolved calls was======>>        569.0
The Number of abandoned calls is ----->>        81
Do you want to continue (y/n)
```

Screen shot of simulation analysis using: First come First Serve Routing Rule



```
The Average Speed of Answer is =====>>  795.4735500404641
The Number of resolved calls was======>>        569.0
The Number of abandoned calls is ----->>        81
Do you want to continue (y/n)
y
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
2
Analysis for Fastest call First Routing
numAgents: 0 no of calls expected per Agent Group: 50.0
numAgents: 1 no of calls expected per Agent Group: 50.0
numAgents: 2 no of calls expected per Agent Group: 50.0
numAgents: 3 no of calls expected per Agent Group: 50.0
numAgents: 4 no of calls expected per Agent Group: 50.0
numAgents: 5 no of calls expected per Agent Group: 50.0
numAgents: 6 no of calls expected per Agent Group: 50.0
numAgents: 7 no of calls expected per Agent Group: 50.0
numAgents: 8 no of calls expected per Agent Group: 50.0
numAgents: 9 no of calls expected per Agent Group: 50.0
numAgents: 10 no of calls expected per Agent Group: 50.0
numAgents: 11 no of calls expected per Agent Group: 50.0
numAgents: 12 no of calls expected per Agent Group: 50.0

 Num. calls expected = 650.0

The Average Wait Time is ====>> 1475.996767372049
The Average Speed of Answer is =====>>  1623.3466984633365
The Number of resolved calls was======>>        591.0
The Number of abandoned calls is ----->>        59
Do you want to continue (y/n)
```

Screen shot of simulation analysis using Fastest Call First Routing

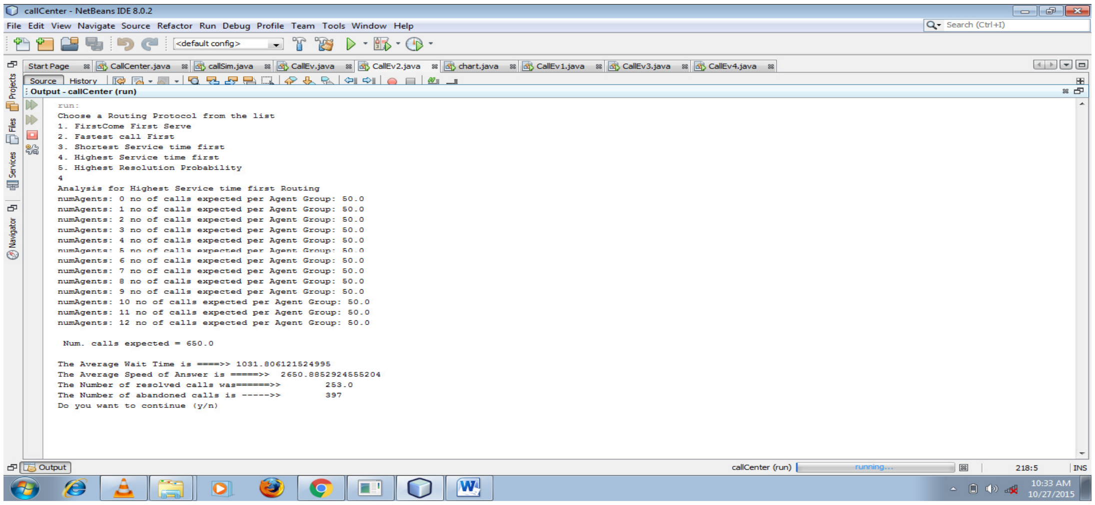


```
static ArrayList<Double> serWaitStore = new ArrayList<Double>();
```

```
run:
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
3
Analysis for Shortest Service time first Routing
numAgents: 1 no of calls expected per Agent Group: 55.0
numAgents: 2 no of calls expected per Agent Group: 56.0
numAgents: 3 no of calls expected per Agent Group: 57.0
numAgents: 4 no of calls expected per Agent Group: 58.0
numAgents: 5 no of calls expected per Agent Group: 59.0
numAgents: 6 no of calls expected per Agent Group: 60.0
numAgents: 7 no of calls expected per Agent Group: 61.0
numAgents: 8 no of calls expected per Agent Group: 62.0
numAgents: 9 no of calls expected per Agent Group: 63.0
numAgents: 10 no of calls expected per Agent Group: 64.0
numAgents: 11 no of calls expected per Agent Group: 65.0
numAgents: 12 no of calls expected per Agent Group: 66.0
numAgents: 13 no of calls expected per Agent Group: 67.0

 Num. calls expected = 793.0

The Average Wait Time is ====>> 728.142595547858
The Average Speed of Answer is =====>>  1436.3608912178407
The Number of resolved calls was======>>        402.0
The Number of abandoned calls is ----->>        391
Do you want to continue (y/n)
BUILD STOPPED (total time: 10 seconds)
```

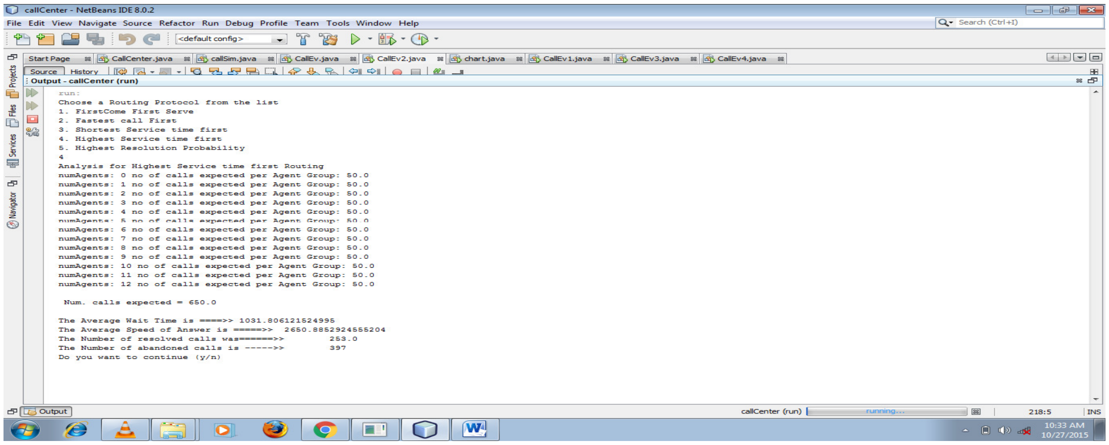
callCenter - NetBeans IDE 8.0.2
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
<default config>
Start Page
CallCenter.java
callSim.java
CallEv.java
CallEv1.java
CallEv2.java
Source
History
Output - callCenter (run)
Output
76:1
INS
2:25 PM
10/26/2015

Screen shot of simulation analysis using shortest service time routing



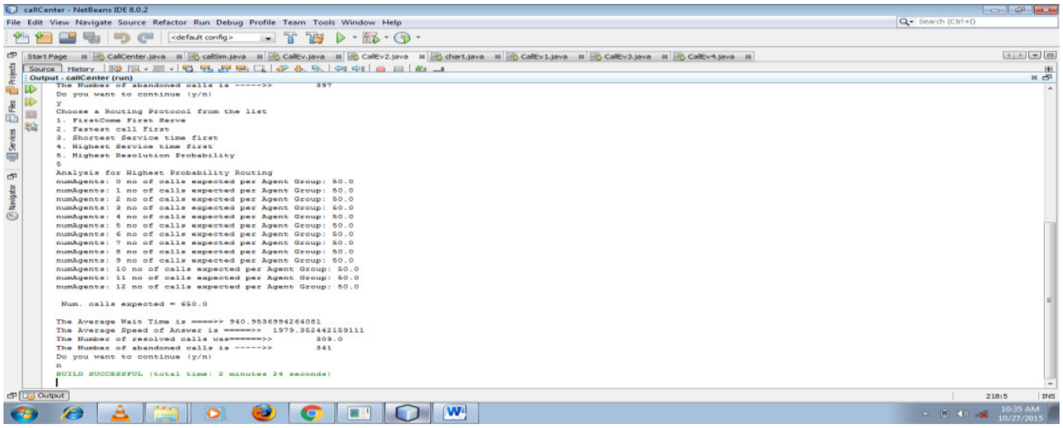
```
run:
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
4
Analysis for Highest Service time first Routing
numAgents: 0 no of calls expected per Agent Group: 50.0
numAgents: 1 no of calls expected per Agent Group: 50.0
numAgents: 2 no of calls expected per Agent Group: 50.0
numAgents: 3 no of calls expected per Agent Group: 50.0
numAgents: 4 no of calls expected per Agent Group: 50.0
numAgents: 5 no of calls expected per Agent Group: 50.0
numAgents: 6 no of calls expected per Agent Group: 50.0
numAgents: 7 no of calls expected per Agent Group: 50.0
numAgents: 8 no of calls expected per Agent Group: 50.0
numAgents: 9 no of calls expected per Agent Group: 50.0
numAgents: 10 no of calls expected per Agent Group: 50.0
numAgents: 11 no of calls expected per Agent Group: 50.0
numAgents: 12 no of calls expected per Agent Group: 50.0

 Num. calls expected = 650.0

The Average Wait Time is ====>> 1031.806121524995
The Average Speed of Answer is =====>>  2650.8852924555204
The Number of resolved calls was======>>        253.0
The Number of abandoned calls is ----->>        397
Do you want to continue (y/n)
```

Screen shot of simulation analysis using Highest Service Time First Routing



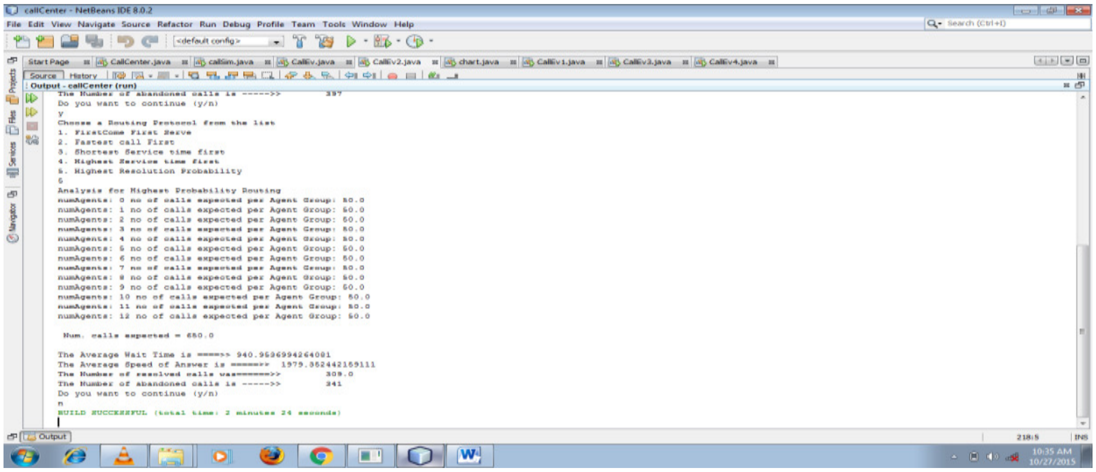
```
run:
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
4
Analysis for Highest Service time first Routing
numAgents: 0 no of calls expected per Agent Group: 50.0
numAgents: 1 no of calls expected per Agent Group: 50.0
numAgents: 2 no of calls expected per Agent Group: 50.0
numAgents: 3 no of calls expected per Agent Group: 50.0
numAgents: 4 no of calls expected per Agent Group: 50.0
numAgents: 5 no of calls expected per Agent Group: 50.0
numAgents: 6 no of calls expected per Agent Group: 50.0
numAgents: 7 no of calls expected per Agent Group: 50.0
numAgents: 8 no of calls expected per Agent Group: 50.0
numAgents: 9 no of calls expected per Agent Group: 50.0
numAgents: 10 no of calls expected per Agent Group: 50.0
numAgents: 11 no of calls expected per Agent Group: 50.0
numAgents: 12 no of calls expected per Agent Group: 50.0

 Num. calls expected = 650.0

The Average Wait Time is ====>> 1031.806121524995
The Average Speed of Answer is =====>>  2650.8852924555204
The Number of resolved calls was======>>        253.0
The Number of abandoned calls is ----->>        397
Do you want to continue (y/n)
```

Screen shot of Simulation using Shortest Queue Routing (SQR)



```
The Number of abandoned calls is ----->>        397
Do you want to continue (y/n)
y
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
5
Analysis for Highest Probability Routing
numAgents: 0 no of calls expected per Agent Group: 50.0
numAgents: 1 no of calls expected per Agent Group: 50.0
numAgents: 2 no of calls expected per Agent Group: 50.0
numAgents: 3 no of calls expected per Agent Group: 50.0
numAgents: 4 no of calls expected per Agent Group: 50.0
numAgents: 5 no of calls expected per Agent Group: 50.0
numAgents: 6 no of calls expected per Agent Group: 50.0
numAgents: 7 no of calls expected per Agent Group: 50.0
numAgents: 8 no of calls expected per Agent Group: 50.0
numAgents: 9 no of calls expected per Agent Group: 50.0
numAgents: 10 no of calls expected per Agent Group: 50.0
numAgents: 11 no of calls expected per Agent Group: 50.0
numAgents: 12 no of calls expected per Agent Group: 50.0

 Num. calls expected = 650.0

The Average Wait Time is ====>> 940.9536594264081
The Average Speed of Answer is =====>>  1979.352442159111
The Number of resolved calls was======>>        309.0
The Number of abandoned calls is ----->>        341
Do you want to continue (y/n)
n
BUILD SUCCESSFUL (total time: 2 minutes 24 seconds)
```

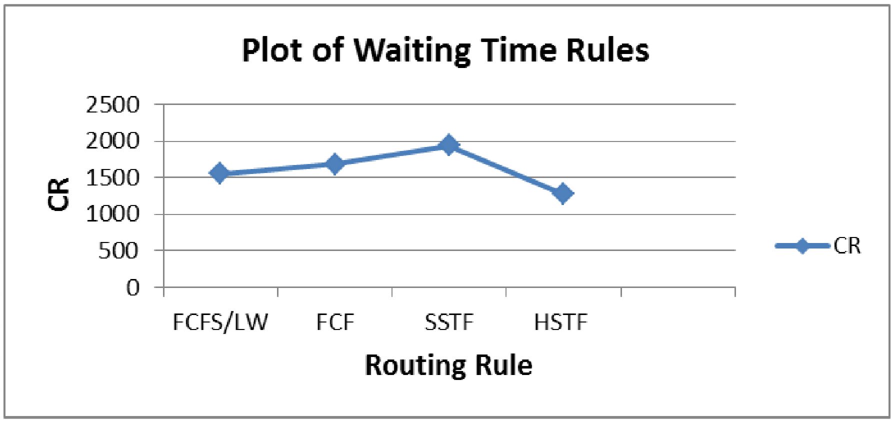
Screen shot of simulation analysis using Highest Resolution probability routing



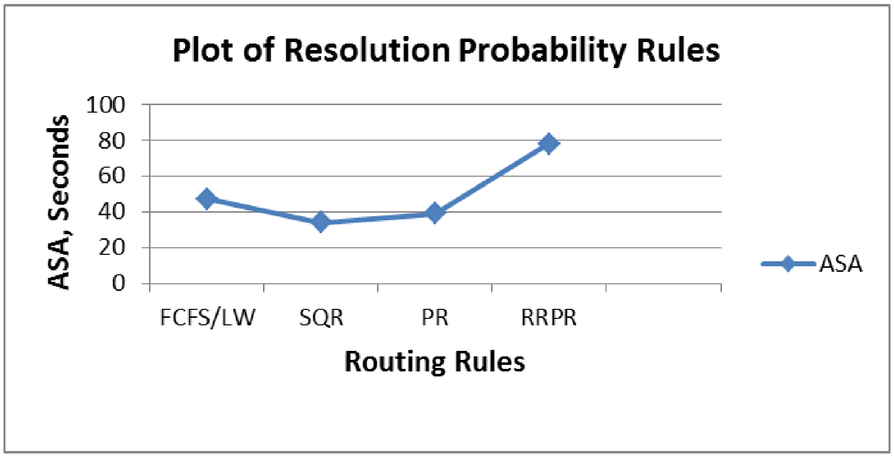
```
The Number of abandoned calls is ----->>        397
Do you want to continue (y/n)
y
Choose a Routing Protocol from the list
1. FirstCome First Serve
2. Fastest call First
3. Shortest Service time first
4. Highest Service time first
5. Highest Resolution Probability
5
Analysis for Highest Probability Routing
numAgents: 0 no of calls expected per Agent Group: 50.0
numAgents: 1 no of calls expected per Agent Group: 50.0
numAgents: 2 no of calls expected per Agent Group: 50.0
numAgents: 3 no of calls expected per Agent Group: 50.0
numAgents: 4 no of calls expected per Agent Group: 50.0
numAgents: 5 no of calls expected per Agent Group: 50.0
numAgents: 6 no of calls expected per Agent Group: 50.0
numAgents: 7 no of calls expected per Agent Group: 50.0
numAgents: 8 no of calls expected per Agent Group: 50.0
numAgents: 9 no of calls expected per Agent Group: 50.0
numAgents: 10 no of calls expected per Agent Group: 50.0
numAgents: 11 no of calls expected per Agent Group: 50.0
numAgents: 12 no of calls expected per Agent Group: 50.0

 Num. calls expected = 650.0

The Average Wait Time is =====>> 940.9596994264081
The Average Speed of Answer is =======>>  1979.352442169111
The Number of resolved calls was=======>>       309.0
The Number of abandoned calls is ----->>        341
Do you want to continue (y/n)
n
BUILD SUCCESSFUL (total time: 2 minutes 24 seconds)
```

Screen shot of Simulation using Relative Resolution Probability Routing (RRPR)

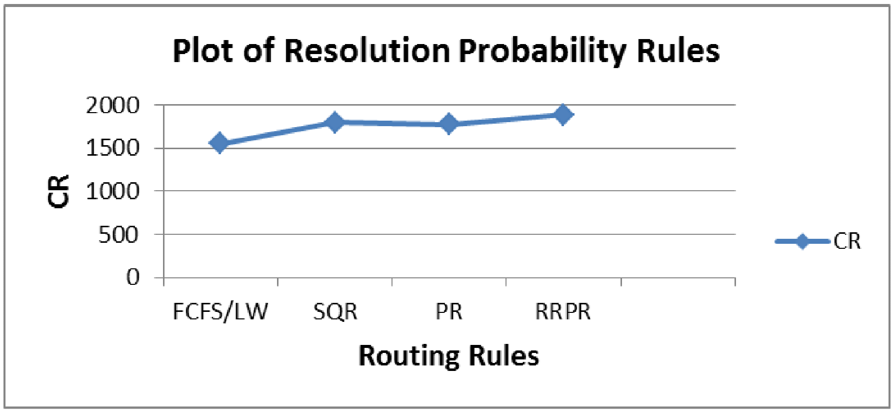
APPENDIX 2
SAMPLE GRAPHS SHOWING THE EVALUATION



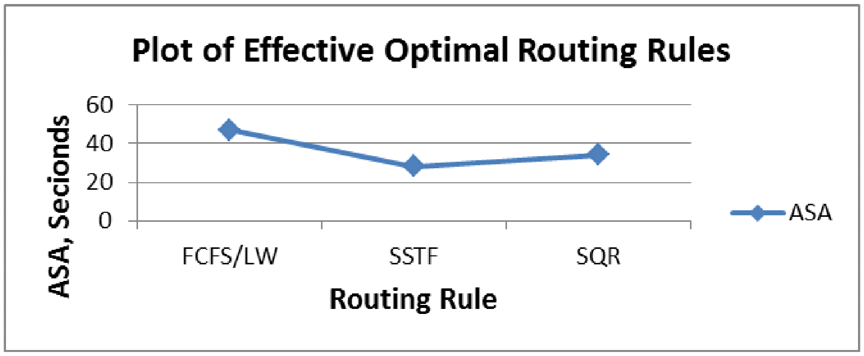
Evaluation of waiting time routing rule for CR



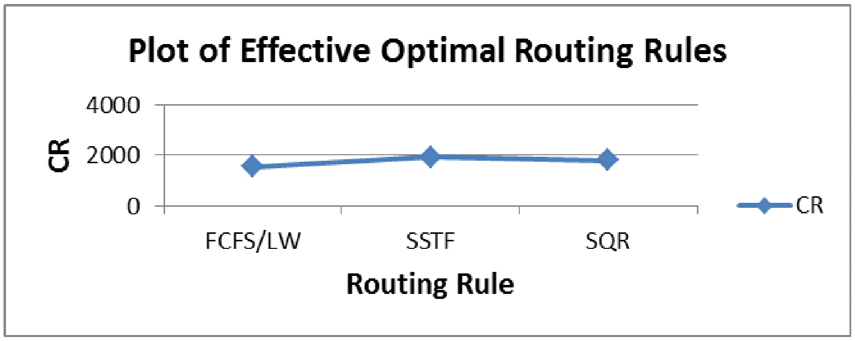
Evaluation of CR routing rule for ASA in seconds



Evaluation of CR routing rule for CR



Evaluation of SSTF, SQR and FCFS/LW routing rule for ASA



Evaluation of SSTF, SQR and FCFS/LW routing rule for CR