Optimal Spectrum Utilisation in Cognitive Radio Networks Based on Processor Sharing Techniques

Hilary M. Tsimba*[1], B.T. Maharaj[1] and Attahiru S. Alfa[1,2]

[1]Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Pretoria, South Africa
[2]Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, Canada

*Correspondence: Hilary M. Tsimba, Email: hilarytsimba@gmail.com

Summary
Cognitive radio networks have achieved higher efficiency in terms of spectrum usage; however they do not readily solve any competition for access among secondary users. Optimisation is applied to an underlay network to obtain the optimal solution for at least two secondary users operating simultaneously on the same channel. Performance measures are used as the target for optimisation. However, the objective function is difficult to obtain in closed form. For the performance measures, queueing theory, particularly weighted processor sharing techniques are employed to model the system dynamics and behaviour. Transmission power and the interference temperature limit are used to allocate weights to the secondary users. Queue length and waiting time functions obtained from the queuing models are used for optimisation. After establishing that the objective function can be considered to be pseudo-convex, convex programming is then deployed to obtain the optimised solution. The results suggest that there is indeed an improvement in network performance after optimisation. The immediate benefits of such a system are firstly improved spectrum utilisation through adding multiple secondary users and secondly, through optimisation, higher performance that can be achieved by the secondary users.

Keywords:
Cognitive radio networks, Spectrum efficiency, Queueing theory, Optimisation

1. Introduction

Wireless mobile technology has over the past years experienced massive growth in terms of use and innovation. The world population continues to grow and so does the demand for information. Every day many individuals seek to be connected, to communicate and absorb information. People want to do this fast and reliably and also while mobile. Radio frequency spectrum, the medium which carries all wireless communication is finite. The increase in the number of users is leading to congestion and dropped connections in networks.

Research has shown that regardless of the high number of users and services requiring spectrum access but not being allowed, spectrum usage is still under-utilised[1]. This is due to the licensing structure of radio spectrum. Users called primary users (PUs) pay for the right to have exclusive use of spectrum. This means that even though the PUs are not using the particular channel at that time, no-one else is allowed to use it. The result is that there are idle channels while there are users waiting with no access.

Cognitive radio networks (CRN) have been proposed as a solution to the effects of the increasing number of network users[2]. CRNs add a secondary user (SU) to the network that can and may opportunistically access the network when it is not in use.

Therefore, when a PU is not using its licensed channel, an SU may occupy and transmit on that channel until the PU returns. There are various approaches to achieve this. These approaches are generally classified into three types namely underlay, overlay and hybrid.

In the underlay CRN the SU may occupy the spectrum simultaneously with the PU. The PU imposes an interference temperature limit (ITL) on the SUs' transmission power. This ITL is the limit below which the SUs must always transmit to ensure that the PU is unaffected by their activity. The ITL will therefore be known to all SUs in the network. The actual power that the SUs will transmit will be dependent on real channel conditions and on what the SU itself is capable of transmitting[3]. In overlay mode, the SU only transmits when the PU is not occupying the channel. The SUs must thus be able to sense when the PU is not active before proceeding with their own transmission. The SUs must also immediately leave the channel when the PU becomes active. Sensing when the PU becomes active, however, has proven to be a challenge[4]. False and missed detections of PU activity are the primary issues. Hybrid mode is a combination of the two modes. When the PU is active the SU will still be able to transmit below the interference limit. However, on PU vacation the SU will be able to transmit at any power[5]. The PU transmission is also known to the SU and the SU can aid in the transmission. CRNs can allow an SU to transmit in the licensed spectrum but they do not readily solve any competition between SUs. A situation might arise were there are two or more SUs requiring to transmit on the same channel. The network must then distribute resources to these SUs using some fair criteria. Examples of these criteria are cost (a case where SUs can be made to pay for access), importance and type of service.

Every system has a desirable quality or a minimum performance requirement. These requirements can be throughput, minimum power use and minimal total network time[6]. In a network where multiple SUs are competing for access and certain requirements associated with them, there is now a need for the network to provide a reasonable distribution of available resources to allow each SU to meet its particular requirements. This can be solved by finding an optimal point. That is, some network parameters can be adjusted to determine the best environment for all involved. This is usually achieved by determining an objective function to suit all clients. Objective functions, however, may not be straightforward and must be carefully determined to ensure that the solution found is indeed optimal. Determining a fair optimisation point remains a challenge, especially if the requirements per SU are different. However, objective functions can be modelled to meet multiple requirements, given different constraints in the network.

Attributes for optimisation can be maximising SU throughput or data rate and minimising transmission power in the network[7]. Optimisation problems can be classified into two classes, namely linear programming and non-linear programming. Linear programming describes a problem where the objective and the constraint functions are all linear. A number of techniques and methods have been developed to solve both classes, such as Dantzig's simplex method[8]. Non-linear programming means that either the objective or the constraint function is not linear. However, non-linear problems are not straightforward to solve.

A further generalisation of linear optimisation is when the objective function itself is a convex function. This is known as convex optimisation. Should the objective function be convex, any local minimum of the function is also a global function and hence the minimal (optimal) point. Convex optimisation has been used to find the unknown signals in wireless sensor networks where sparse estimation was achieved through optimisation[9]. In order to determine the objective function in a multiple SU scenario, the network layout and behaviour must be defined first. Queueing theory has been used to model networks and from there to determine the performance measure and hence the optimisation targets. Optimisation has not previously been extensively implemented together with queueing theory in CRNs.

In this paper a CRN system with one always present PU and two SUs is considered. An optimal point for the performance is determined based on a model designed using queuing theory. After establishing convexity, convex programming is deployed to maximise the network performance. The effects of optimisation, power allocation ratio and weighting are investigated. Finally, the optimisation result is applied and a comparison with existing solutions is made.

2. Related work

Queueing theory has seen wide use in CRN as a technique to implement and evaluate the performance of many schemes targeted at increasing spectrum utilisation[10]. Queueing is one of the more recent approaches deployed in an attempt to solve the CRN spectrum allocation problem. It has been used to solve and analyse real world situations such as telephone exchange systems, air traffic control, hospital queues, etc[11]. Queue analysis provides useful quality of service (QoS) information, such as waiting times and blocking probability. Knowledge of such parameters in a CRN can be used to provide better service to all users involved, since the network can be optimised according to the analysis. SU packets, for example, can use knowledge of the waiting time to decide whether to join the service queue or not. Performance between SUs has also been investigated under outage probability constraint. However the SUs do not transmit simultaneously over a single channel in the underlay mode[12]. Fairness in underlay cognitive radio networks has also been investigated by considering the throughput of the SUs. The admission control algorithm decides to admit SUs in the network based on max-min fairness criteria. Optimisation is then deployed to solve the problem. However, this is not a queueing problem[13].

By definition a queue comprises items waiting in line to be served. In CRNs packets will wait in line, at their respective SU transmitters, to be transmitted (served) through the channel (server). The most common type of queue is the single-node queue. Here, packets are served at one location and thereafter leave the system. The packets do not proceed to another location for further service, but are allowed to re-enter the system as a new packet by joining the queue again. One variation is that there can be multiple parallel servers serving one queue. Another variation is to have multiple queues at the location, but with only one server. The queues may be served through polling were the single server goes around serving each queue for a certain amount of time. The polling system can also be changed to allow the server to serve all the queues present by dividing its resources equally among the packets waiting to be served. This technique is called processor sharing[14].

In a processor-sharing system, a single server will serve all the packets present in the system at an equal rate. Should another packet join the system, the server will adjust its service to accommodate the new arrival and all packets will be served at the same rate. When the server can no longer reasonably divide its resource, any new packets that arrive to join the system will be turned away. Processor-sharing techniques have traditionally been applied to computer processes where the processing rate (service rate) is fixed and known. However in communication networks the transmission rate is not fixed.

Priority queues have also been used to allocate resources in situations where there are two or more SUs[15]. Here, SUs are divided into two classes and assigned priorities. The SUs with the highest priorities will be served first ahead of any lower class SU. A variation of the processor-sharing scheme is to classify the packets as well. This is called weighted processor sharing whereby packets in the system are weighted according to some criteria and higher weighted packets are given a bigger slice of the processing power. Packets that arrive when the system is at capacity are blocked and are lost[16].

In computer processes, the processing time or service rate is fixed and can thus be divided equally among users[17] [18]. However in transmission networks weighted sharing can be challenging to implement because the service rate (transmission time) is not fixed but is dependent on many factors. Head of line processor sharing, which entails only serving the packet and the front of queues imposed in the network, has been implemented in a Poisson arrival and exponential service environment[19].

Bhoopendra et el[20] compiled a survey on access techniques in CRNs. The survey details how there is a lot of research focusing on hybrid access techniques. That is, using both underlay and overlay techniques are employed in a system at the same time. SUs can switch to overlay mode in the absence of a PU in the system and then to underlay when the PU returns to the system. The survey identifies several aspects that can be exploited to improve efficiency. Among these, power constraint and ITL are of interest. Senthuran et al[21] propose a system where a switching ratio between overlay and underlay modes can be calculated and used to improve the system. Simulations are then done on the system to provide results. The system was implemented using Markov chain analysis.

Zou et al[22] propose a multi-band CRN where multiple SUs transmit via the same relay. The system is a hybrid access technique where the SUs allowed to operate in overlay and underlay mode. Since the SUs are allowed to transmit on a common relay, an auction based scheme where an SU is allocated transmission power based on its payment is proposed. Morte Carlo simulations are run to verify the theoretical analysis.

Wang et al[23] investigate an underlay cognitive radio system that has SUs decide whether to join the queue or balk. The SUs evaluate the trade-off between delay cost and service reward. Two situations are proposed, namely (1) SUs can partially obverse the system and (2) SUs are not able to obverse the system at all. The proposed solution makes use of two strategies for both situations. The first is finding a selfish equilibrium solution and second is finding a social equilibrium solution. The proposed system is evaluated under mean number of packets and expected delay performance measures.

She et al[24] develop a framework for reliable and low-latency transmission in radio access networks. A packet dropping policy that takes into consideration the transmit power is proposed. The overall packet loss probability is then determined using transmission error probability, exceeding set queueing delay probability, probability of packet drop as metrics. Finally the transmit power power is also optimised. Numerical analysis and simulations are used to verify their results. Mannivannan et al[25] propose an overlay cognitive radio model that makes use of spectrum sensing to increase utilisation. A major finding is that sensing and data transmission can take place simultaneously.

Tadayon and Kaddoum[26] utilise queueing to establish packet level cooperative diversity. The authors use BCMP queues to extend on known protocols such as decode-forward,incremental relaying and opportunistic relaying. Performance metrics for the cooperation enabled BCMP network were then derived. Numerical results and are compared to the calculated closed form metrics. The paper mostly deals with a total of two queues. That is to say one primary and one secondary queue although not specifically defined as such. Processor sharing queues are a type of BCMP queues. Xia et al[27] overview cooperative underlay systems proposed to overcome the problem of limited power available for SU transmissions. To extend the transmission range, cooperative amplify and forward techniques are employed on the secondary network. They find that imperfect channel state information (CSI) has a major effect on system performance and many of the works so far do not take into account the worst possible CSI.

Duy and Son[28] propose a multicast underlay CRN that can serve multiple SUs through a secondary base station. Hardware limitations are also factored into system. The system is then investigated on a case by case basis to determine the effect on the system of parameters such as SU to PU distance, channel conditions and number of SUs.

An M/G/1/K queue where the ITL's effect on transmission time has been determined has been studied[29]. The system considers a Nakagami-m fading channel with a single queue for the secondary network. The fading channel follows a general distribution and this extends to the queue service times. The general distribution model allows for the queue to be embedded at the departure point of an SU packet. However when the number of queues operating are more than two, a single departure point cannot be imposed since the queues run simultaneously. An exponential distribution is memoryless and will solve the problem of where to embed while maintaining the concept of the proposed solution.

3. System model

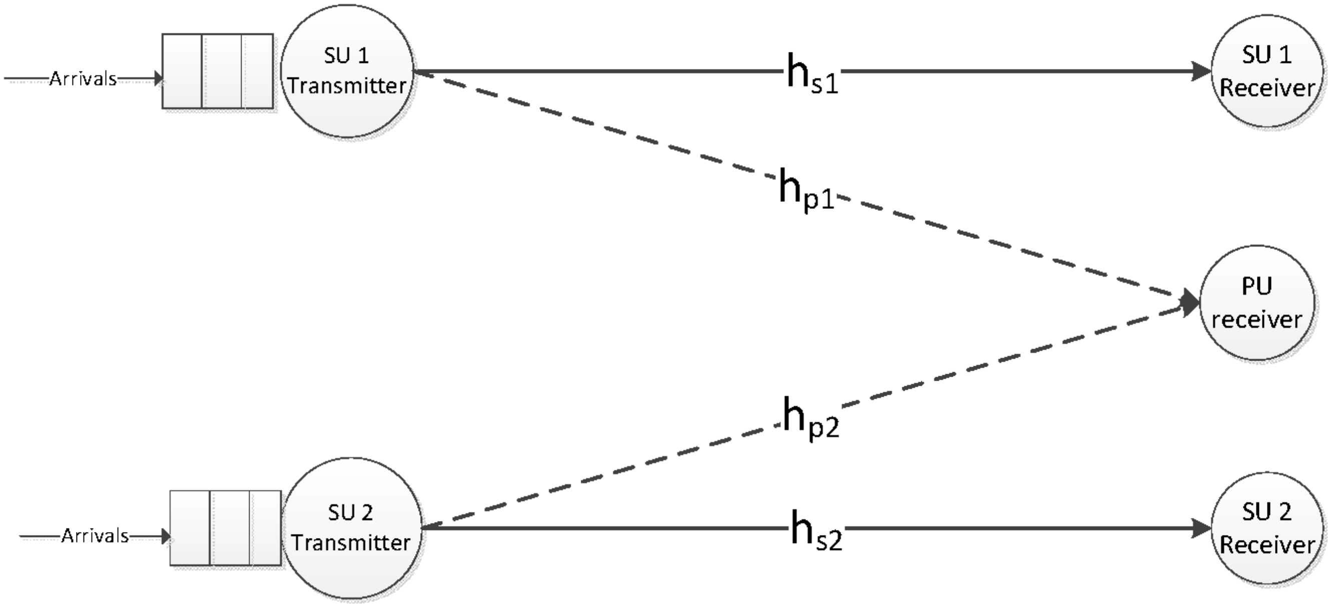

Figure 1. Network layout.

Consider an underlay CRN with multiple SUs competing for access. Figure 1 illustrates the system under consideration. The SUs will have to keep their transmit powers below the ITL imposed by the PU to prevent any harmful signals from reaching the PU. SU transmit powers will have a cumulative effect on the PU receiver and hence the total power from the SUs must remain below the ITL. For increased network efficiency the model proposes that the SUs share the available transmit power. The SUs can thus be weighted according to some network-defined rules allowing some SUs more transmission power than others. The SU weights will be assigned on admission to the network. The allocated transmission power is also limited by the maximum power that particular SU hardware can achieve.

In Figure 1 only two SUs are shown for clarity. In order to maximise efficiency, the transmit power allocations are dynamic. That is, in the absence of the other SUs, an SU can transmit at any power resulting in interference less than or equal to the ITL. The SU will have to adjust its power according to the weighting, should another SU arrive in the system, on the assumption that SUs cooperate in this regard. The implementation of the model can be achieved in two ways, namely pre-emptive and non-pre-emptive approaches.

3.1.Pre-emptive approach

The SUs will share the transmission power according to previously determined weights. In the case where a single SU is transmitting in the system, it is allowed to transmit at any power it needs, provided that the power reaching the PU is below or equal to the given ITL. Transmission power resulting in the interference being at or below the ITL will be known as maximum allowable power. On arrival of another SU the transmit power will immediately be adjusted according to the weights such that the arriving SU can begin transmitting immediately. In short, the arrival of another SU will pre-empt the transmission of the SU already transmitting by immediately reducing its transmit power. All SUs will be pre-empted on arrival of another SU, regardless of weight.

3.2. Non-pre-emptive approach

The SUs will also share the transmission power according to previously determined weights. In a case where a single SU is transmitting at maximum allowable power and another SU arrives, the transmit power will be not be adjusted immediately to accommodate the arriving SU. Instead, the packet that is currently being transmitted will be allowed to continue at maximum allowable power until completion. The arriving SU will have to wait before beginning its own transmission. The transmit power will only be re-allocated according to the weights after the initial packet has completed transmission. In short, the arrival of another SU will not pre-empt the transmission of the SU already transmitting and it will therefore not begin transmitting immediately. All SUs will follow this rule regardless of weight.

Weighting is used to determine the transmission power rather than the order of transmission. The transmission time here is the service rate of the queueing system and is dependent on factors such as the channel conditions, noise, bandwidth etc.

Let:

- h_{pi} be the channel coefficient for the channel SU_i to the PU receiver.
- h_{si} be the channel coefficient for the channels SU_i transmitter to SU_i receiver.
- L be the interference temperature limit imposed by the PU receiver.
- M be the total number of SUs transmitting in the CRN.

Given this information, the total received interference signal is calculated as:

$$y_r = \sum_{i=1}^{M} h_{pi} x_{si} + n_s \tag{1}$$

where x_{si} is the signal from SU_i with transmission power P_{si}. n_s is the additive white Gaussian noise with zero mean and variance N_o.

The transmission signal powers of the SUs are to be constrained such that the total signal power reaching the PU receiver is less than Q. Therefore, each SU will only be allowed to transmit at a power such that its interference power reaching the PU receiver is a fraction of L. Let r_i be the given allocation for SU_i such that:

$$\sum_{i=1}^{M} r_i = 1. \tag{2}$$

Taking into account the channel conditions and the allocation per SU, the transmission power for the SUs must then be capped at:

$$P_{si} \leq \frac{r_i L}{|h_{pi}|^2},\ i \in \{1, 2...M\}. \tag{3}$$

However, there are physical limitations on the transmission power of the SU transmitter. This is the highest amount that the transmitter can output and is denoted by P_{max_i}. It is possible that P_{max_i} can be less than P_{si}, hence the instantaneous signal-to-noise-ratio (SNR) of a secondary user is given by:

$$\gamma_{si} = \min\{\frac{P_{max_i}|h_{si}|^2}{N_o}, \frac{r_i L|h_{si}|^2}{N_o|h_{pi}|^2}\},\ i \in \{1, 2...M\}. \tag{4}$$

Shannon's theorem gives the maximum rate at which information can be transmitted over a channel given the bandwidth and the SNR. Using the theorem, and assuming that the transmission rate is equal to the channel capacity, the transmission time for SU_i is given by:

$$T_i = \frac{N}{B\log_2(1+\gamma_{si})},\ i \in \{1, 2...M\} \tag{5}$$

where B is the bandwidth of the transmission channel and N is the number of bits per packet. This will usually be dictated by an IEEE standard.

3.3. Optimisation Setup

Given the model, an approach that may be utilised for optimisation is focusing on the optimisation of a performance measure such as the time spent in the system. That is, the objective function will be based on trying to ensure that a packet spends as little time as possible in the system. The system constraints will have to be adjustable network parameters such that the network can vary them as required to achieve optimisation. In the model above only the power allocation ratio, r_i, can be adjusted. The other parameters, such as the interference limit and the channel coefficients, are environment-dependent and hence cannot be adjusted by the network. In this case, the optimisation problem is a minimisation problem. By definition, the optimal solution to a minimisation problem is the lowest objective function value in the feasible region[30].

Here an objective function is defined, and a general constraint formulation is given. A general objective function to a resource allocation problem is given below[7]. The objective problem is:

$$\text{minimise } z = f(x, y) \tag{6}$$

and general constraints are:

$$\text{s.t. } g_i(x, y) \le b_i\ i = 1, ...m, \tag{7}$$

$$x_k \ge 0,\ k = 1, 2, ..., m, \tag{8}$$

$$y_j \ge 0,\ j = 1, 2, ..., m. \tag{9}$$

However, for the proposed models above the objective function, z, is not straightforward to derive. The two SUs are dependent on each other regardless of the constraint; this results in difficulty in developing a closed form objective function. In addition, given that the optimisation of a performance measure is more desirable, queueing theory is used to develop and determine the queue behaviour. Queueing will allow performance measures such as waiting time and number of packets in the network to be obtained. These will then become the objective function z. Therefore, queueing theory is used to define and determine $x, y, f(x, y), g_i, b_i$ etc. The definitions are given in equation 11. Optimisation will then be done according to the results of the queueing models. The queueing models have been developed previously and a brief summary will be given here for completeness[31]. The work is now extended by deploying optimisation to the previously developed results in order to ensure that optimum performance is achieved by all SUs.

3.4. Queue models

The M/M/1-PS queue was deployed for the queueing models, that is, Poisson arrival and exponential service with a single server/channel and processor sharing. The memoryless property of the exponential service allows the service mean time to be changed dynamically. The transmission time, T_i, will be the service rate and is assumed to follow an exponential distribution with mean μ_i.

The system will initially be limited to two SUs. That is, $M = 2$. The two SUs are now defined as the higher weighted (HW) and lower weighted (LW) SUs respectively. The number of HW users allowed in the system is infinite. There may be a buffer on the LW users of K. Therefore, given the buffer, there can only be a maximum of $K+1$ LW SU packets at any given time. However, the buffer can be removed when K is infinite. Queue discipline is first come, first served. The model definitions are as follows:

Let:

- λ_1 be the mean arrival rate of the HW queue,
- λ_2 be the mean arrival rate of the LW queue,
- μ_1 be the mean service rate of the HW queue when there are no LW users,
- μ_3 be the mean service rate of the LW queue when there are no HW users,
- μ_2 be the mean service rate of the HW queue when there are LW users in the system,
- μ_4 be the mean service rate of the LW queue when there are HW users in the system, and
- $\mu_b = \mu_2 + \mu_4$.

Therefore: $\lambda = \lambda_1 + \lambda_2$ is the mean arrival rate of the entire system.

3.4.1. Pre-emptive queue model

An arriving SU has the ability to reduce the transmission power (pre-empt) of the SU packet already in service. Therefore, the service of the SU packet already in service will be completed at reduced power. The model is defined as follows:

State space

Figure 2 shows the state space for the pre-emptive queueing model. Let state (h,j) $\{h$ = number of HW users in the system, j = number of LW users in the system$\}$. $h \in 0, 1, 2, 3, ..., j \in 0, 1, 2..., K + 1$. The circles represent the state in which system currently is. The arrows show valid and possible transitions from one state to the other. For example, to move from state $\{2,2\}$ to state $\{2,1\}$ there must be a service completion on the LW queue before any other event. Note that because of the continuous nature of the queue there cannot be simultaneous events. That is, there cannot be an arrival and departure at the same time.

Transition matrix

The transition matrix is a quasi-birth-death (QBD) process. Define Q_{hj} as the rate of transition from state h to state j for the system. Since there are two queues in the system, the elements of Q will be matrices to capture the rates of the LW queue.

$$Q = \begin{bmatrix} B & C & & & \\ E & A_1 & A_0 & & \\ & A_2 & A_1 & A_0 & \\ & & \cdot & \cdot & \cdot \\ & & & \cdot & \cdot \end{bmatrix} \tag{10}$$

The details of the Q matrix are given in the appendix.

3.4.2. Non-pre-emptive queue model

An arriving class will not pre-empt the transmit power of the class already in service. There is no immediate adjustment of transmission power. Power reallocation will only be adjusted once the SU packet that was being transmitted on arrival has been completed and both SUs have packets present in their queues.

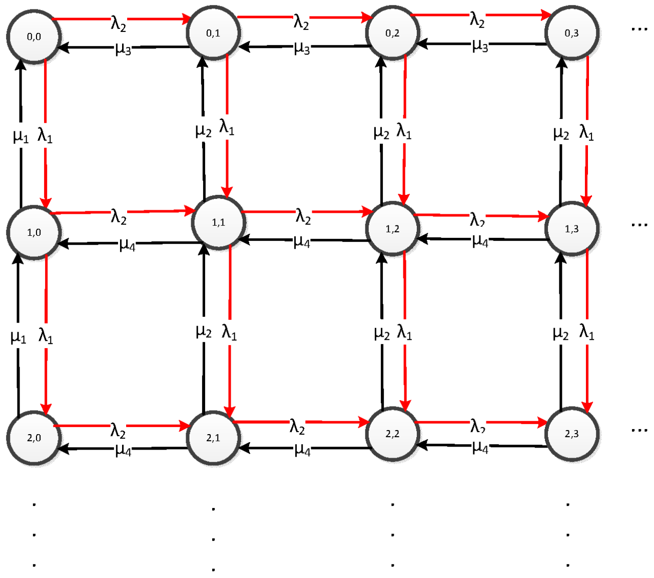

Figure 2. Pre-emptive model state space.

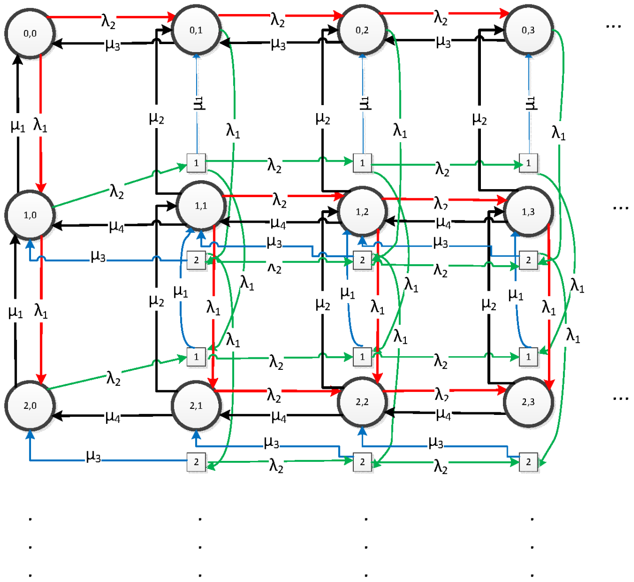

Figure 3. Non-pre-emptive model state space.

State diagram

The state space of the proposed non-pre-emptive queue model is shown in Figure 3 . Let state *h,j,l* be $\{h =$ number of HW users in the system, $j =$ number of LW users in the system, $l =$ the type (HW or LW) of user currently in service$\}$. $h \in 0, 1, 2, 3, ...,$ $j \in 0, 1, 2, ..., K + 1$.

$$l = \begin{cases} 0, & \textit{when both classes are in service.} \\ 1, & \textit{when an HW user is in service.} \\ 2, & \textit{when an LW user is in service.} \end{cases}$$

The circles represent the state in which the system currently is and the squares are dummy states used to determine which SU packet is currently being served. It is necessary to keep track of the current packet, as this packet will need to be served to completion at maximum allowable power before any other packet is served. There may also still be more arrivals at both queues at any time.

Transition matrix

The transition matrix is also a QBD process and is the same as equation 10. However, the non-pre-emption rule will generate dummy states needed to keep track of the SU currently being served. This will cause the elements of the inner matrices to be matrices as well. The details of the Q-matrix are given in the appendix.

3.5. Convex optimisation

Convex programming was found to be adequate and efficient for the given problem. Firstly the objective function and the constraints need to be mapped to the queueing models. Therefore the objective is to minimize the mean number of packets in the system subject to the power allocation ratio. Equation 6 now defines a performance measure and is redefined and given as equation 11. That is, z is now the total average number in the queue for the whole system.

$$\min z = f(x, y) = x + y \tag{11}$$

x and y are now the average number of packets in the HW and LW queues respectively. $g_i(x, y)$ is now the power allocation ratio, r_i, which is the only constraint in the system. Equation 7 now becomes:

$$\text{s.t. } r_1 + r_2 = 1. \tag{12}$$

Equations 8 and 9 remain unchanged because there cannot be fewer than zero packets in the queues at any time. For z to be convex it has to satisfy

$$f(\alpha w_1 + (1-\alpha)w_2) \leq \alpha f(w_1) + (1-\alpha)f(w_2) \tag{13}$$

where $w = (x, y)$ and $0 \leq \alpha \leq 1$.

Also let $f(x)$ and $f(y)$ be equal to x and y respectively. Given that $f(x)$ and $f(y)$ are convex their sum, given weights, will also be convex. Let the weight associated with the HW queue be ϕ and β for the LW queue.

The power allocation ratio, r_i, has a significant effect on the system performance. The ratio affects the service rates, hence the traffic intensity. This leads to the average number and waiting times in the queues being affected, which in turn means the optimisation result is affected. The goal is to determine a power allocation ratio point that after taking all this into account, will allow an optimal solution to be found. This means that weighting will no longer determine the power allocation explicitly. The ratio will be varied by decreasing the allocation to the HW SU while simultaneously increasing the allocation to the LW SU by the same amount. Let θ be a variable such that $r_1 = 1 - \theta$ and $r_2 = \theta$. Algorithm 1 was used to do the convex optimisation.

Algorithm 1 Convex optimisation

```
Require: $r_i$ $i \in \{1, 2\}$
Determine $\mu_1$ and $\mu_3$. Use initial $r_i$ to define the weights allocated to each SU.
Make weights equal to initial $r_i$
$r_1 = 1, r_2 = 0$
while $r_1 > 0.01$ do
    Decrease $r_1$ by 0.01
    Increase $r_2$ by 0.01
    Determine $\mu_2$ and $\mu_4$
    Run queue simulation
end while
Apply weights and sum the two queues
Plot graph of simulation
Obtain $r_i$ for minimum point
```

Figure 4   shows the functions $f(x)$ and $f(y)$ varied across θ. Therefore, $f(x)$ and $f(y)$ can be considered convex.

The system presents a feedback issue whereby the power allocation ratio affects the system performance, which in turn affects the optimisation results. That is to say, adjusting the allocation ratio due to previous optimisation results will lead to change in service times and hence the system performance.

4. Results

4.1. Experimental Setup

The algorithms used to generate the results for the pre-emptive and non-pre-emptive queueing models are shown in algorithms 2 and 3.

4.2. Convexity of results

Figure 4   is the result on the system performance of varying the power allocation ratio through the range of possible values of the allocation ratio. The results do not present as a convex graph at first glance. The results fit a pseudo-convex function. Pseudo-convex functions are sufficient to be deemed convex with regard to determining the minimum point. A function can generally be considered pseudo-convex if it is increasing in a direction with positive directional derivative[32]. However, the results are shown to be sufficient for convexity to be established using a concept in Whitt[33]. The concept involves increasing the number of plot points in order to determine the pattern or shape, given scaling. The scaling can be achieved by adjusting the vertical scales so as

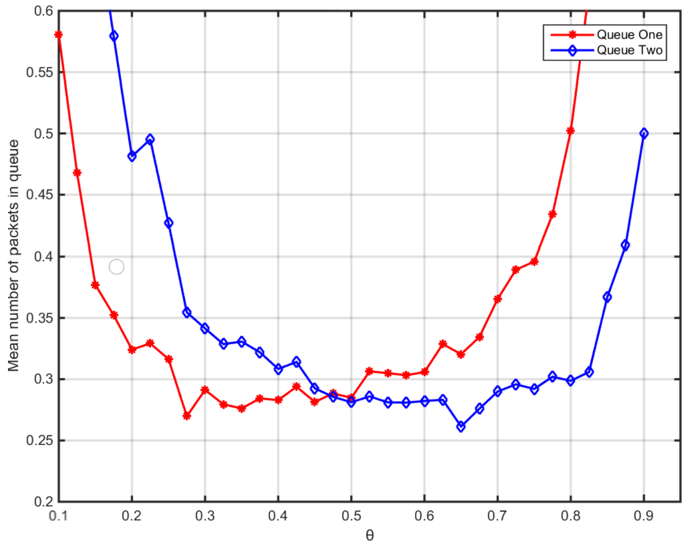

Figure 4. Mean number of packets for queue one and two showing pseudo-convexity.

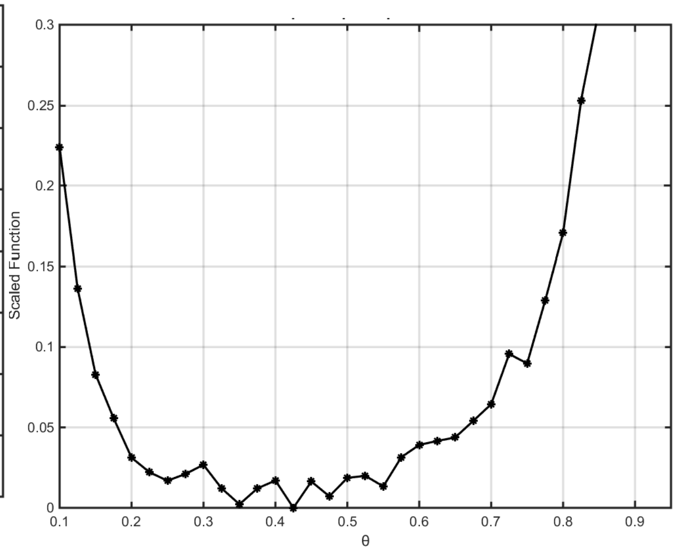

Figure 5. Queue one length vs. allocation power for $n = 40$.

to limit the observable region to only the relevant data. Considering only the HW queue (f_x), the vertical scaling consequently becomes:

$$plot(\{f_x : 0 \leq k \leq n\}) \equiv plot(\{(f_x - min)/range : 0 \leq k \leq n\}), \tag{14}$$

where

$$min \equiv \min f_x : 0 \leq k \leq n$$

and

$$range \equiv \max f_x - \min f_x : 0 \leq k \leq n.$$

n is the number of plot points that also determines the horizontal scaling, since the plot points are spaced by $\frac{1}{n}$.

Simulations are then done to determine if the function f_x (and similarly f_y) is adequate to be determined convex. Figure 5 shows the function f_x with $n = 40$ and Figure 6 shows the same function but with $n = 1600$. From Figure 6 it can be seen that the function appears to be convex.. The results are conclusive enough for convexity to be approximated and consequently for convex optimisation to be carried out. For Figures 4 to 6 the weights for the queues are equal and the traffic intensity is 40%. Hence the optimal solution is an equal power allocation or 50%. Figure 7 shows the shape of the graph from the effects of varying the power allocation ratio for the pre-emptive model.

4.3. System performance

Firstly the simulation results for the unoptimised system under various allocation ratios are presented and compared to evaluate the performance of the proposed system. The parameters are chosen based on the IEEE 802.22 standard[34]. The number of bits is 8184 and the bandwidth is 6 MHz. Figure 8 shows the variation of the mean transmission time based on the standard, different channel coefficients and varying power allocation ratio. Queue two is the LW and SU2. Optimisation is then factored in for the same conditions and the results are compared.

The parameters chosen for the results that follow are $P_{max}/N_o = 25$ dB, $L/N_o = 15$ dB, $h_{s1} = 0.7$, $h_{p1} = 0.8$, $h_{s2} = 0.6$ and $h_{p2} = 0.9$. The buffer size K is infinite. The coefficients are obtained from a snapshot of the system. The results presented are therefore during this snapshot. This is so that the effect of transmit power sharing can be observed with other parameters held constant as much as possible. Service times μ_{1-4} are obtained using equation 5. These are however only initial service times and will change dynamically because of the optimisation feedback system.

Algorithm 2 Pre-emptive model algorithm

```
Require: $r_i$ $i \in \{1,2\}$
 1: Set $h_{p1}, h_{p2}, h_{s1}, h_{s2}, P_{max}, Q, B, N, totalPackets$
 2: Use the values set is step one and $r_i$ to calculate the service times for Queue one and two. $\mu_2$ and $\mu_4$ will be determined using the ratio $r_i$. $\mu_1$ and $\mu_3$ will be determined using a power allocation equal to unity for both. Equation 5 is used here.
 3: Generate arrival times using a random exponential distribution and a mean inter-arrival time such that the traffic intensity of Queue one is 0.2 with $\mu_2$ service rate.
 4: while Queue traffic intensity is less than one do
 5:     Generate arrival times using a random exponential distribution and a mean inter-arrival time such that the traffic intensity of Queue two is 0.05 with $\mu_4$ service rate.
 6:     while Number of packets in system is less than total number of packets do
 7:         Using the simulation time, arrival times and service times determine which event occurs first and adjust the queue length accordingly. % There are four types of events in the simulations, namely Queue one arrival, Queue one departure, Queue two arrival and Queue two departure.
 8:         Any packet, from any queue, that arrives first will enter into service immediately and depending on the queue, will be allocated either $\mu_1$ or $\mu_2$ service rate.
 9:         if A Queue one packet arrives while a Queue one packet is in service then
10:             Increase Queue one size by one and generate next Queue one arrival time.
11:         end if
12:         if A Queue two packet arrives while a Queue two packet is in service then
13:             if Queue two size is less than buffer size then
14:                 Increase Queue two size by one.
15:             end if
16:             Generate next Queue two arrival time.
17:         end if
18:         if A Queue one/two packet arrives while a Queue two/one packet is in service then
19:             Adjust service rates to $\mu_3$ or $\mu_4$ immediately to accommodate arriving packet.
20:         end if
21:         if A Queue one/two packet arrives while both Queue one and two packets are in service then
22:             Increase Queue one size by one.
23:             if Queue two size is less than buffer size then
24:                 Increase Queue two size by one.
25:             end if
26:             Generate next Queue one and two arrival times.
27:         end if
28:         Use time spent in the queue and the simulation time to determine the average number of packets in the queue and the average delays in the queues.
29:         Increase mean inter-arrival time such that traffic intensity of Queue two increases by 0.05.
30:     end while
31:     Increase Queue one traffic intensity by 0.2.
32: end while
33: Plot simulation graph
```

4.3.1. Pre-emptive queue model

Figures 9 to 11 show the expected number of packets in Queue one against the traffic intensity of Queue two under various traffic intensities. The mean number in the queue increases with the traffic intensity because more Queue two packets in the system will result in Queue one transmitting under reduced power more often.

Figures 12 to 14 show the expected delay in Queue one against the traffic intensity of Queue two under various traffic intensities. Again the delay increases with traffic intensity owing to the queues being served at the reduced rate. Should the service

Algorithm 3 Non-pre-emptive model algorithm

```
Require: $r_i$ $i \in \{1, 2\}$
Set $h_{p1}, h_{p2}, h_{s1}, h_{s2}, P_{max}, Q, B, N, totalPackets$
Use the values set is step one and $r_i$ to calculate the service times for Queue one and two. $\mu_2$ and $\mu_4$ will be determined using the ratio $r_i$. $\mu_1$ and $\mu_3$ will be determined using a power allocation equal to unity for both. Equation 5 is used here.
Generate arrival times using a random exponential distribution and a mean inter-arrival time such that the traffic intensity of Queue one is 0.2 for $\mu_2$ service rate.
while Queue traffic intensity is less than one do
    Generate arrival times using a random exponential distribution and a mean inter-arrival time such that the traffic intensity of Queue two is 0.05 for $\mu_4$ service rate.
    while Number of packets in system is less than total number of packets do
        Using the simulation time, arrival times and service times determine which event occurs first and adjust the queue length accordingly. % There are four types of events in the simulations, namely Queue one arrival, Queue one departure, Queue two arrival and Queue two departure.
        The first packet, from any queue, that arrives first will enter into service immediately and depending on the queue, will be allocated either $\mu_1$ or $\mu_2$ service rate.
        if A Queue one packet arrives while a Queue one packet is in service then
            Increase Queue one size by one and generate next Queue one arrival time.
        end if
        if A Queue two packet arrives while a Queue two packet is in service then
            if Queue two size is less than buffer size then
                Increase Queue two size by one.
            end if
            Generate next Queue two arrival time.
        end if
        if A Queue one/two packet arrives while a Queue two/one packet is in service then
            if Queue one then
                Increase Queue one size by one.
            else if Queue two size is less than buffer size then
                Increase Queue two size.
            end if
            Set service rates of next packets in Queue one and two to $\mu_3$ and $\mu_4$ respectively.
        end if
        if A Queue one/two packet arrives during a busy period and there packets in Queue two/one then
            if Queue one then
                Increase Queue one size by one.
            else if Queue two size is less than buffer size then
                Increase Queue two size
            end if
        end if
        if A Queue one/two packet arrives while both Queue one and two packets are in service then
            Increase Queue one size by one.
            if Queue two size is less than buffer size then
                Increase Queue two size by one.
            end if
            Generate next Queue one and two arrival times.
        end if
        Use time spent in the queue and the simulation time to determine the average number of packets in the queue and the average delays in the queues.
        Increase mean inter-arrival time such that traffic intensity of Queue two increases by 0.05.
    end while
    Increase Queue one traffic intensity by 0.2.
end while
Plot simulation graph
```

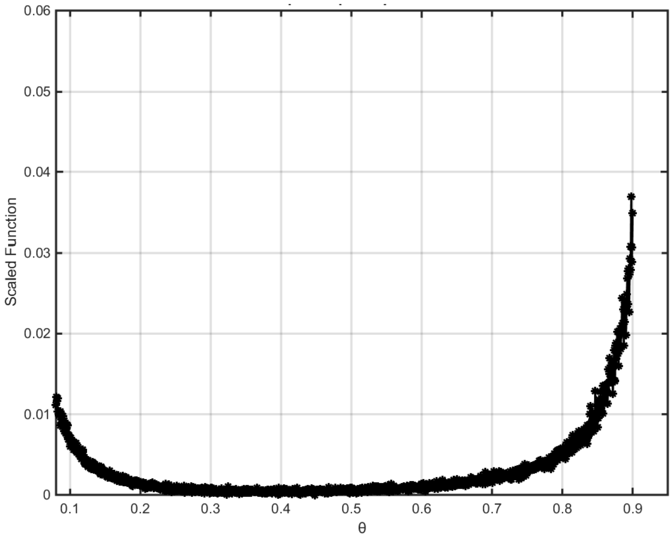

Figure 6. Queue one length Vs. allocation power for $n = 1600$.

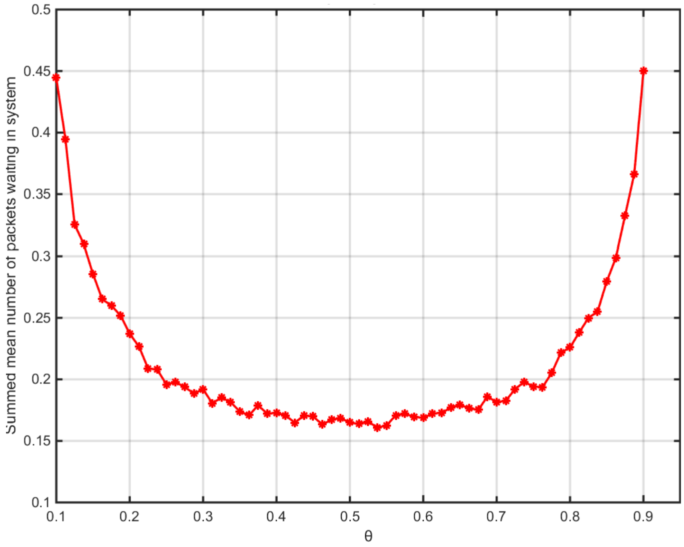

Figure **7.** Mean number of packets for the pre-emptive model with equal weighting.

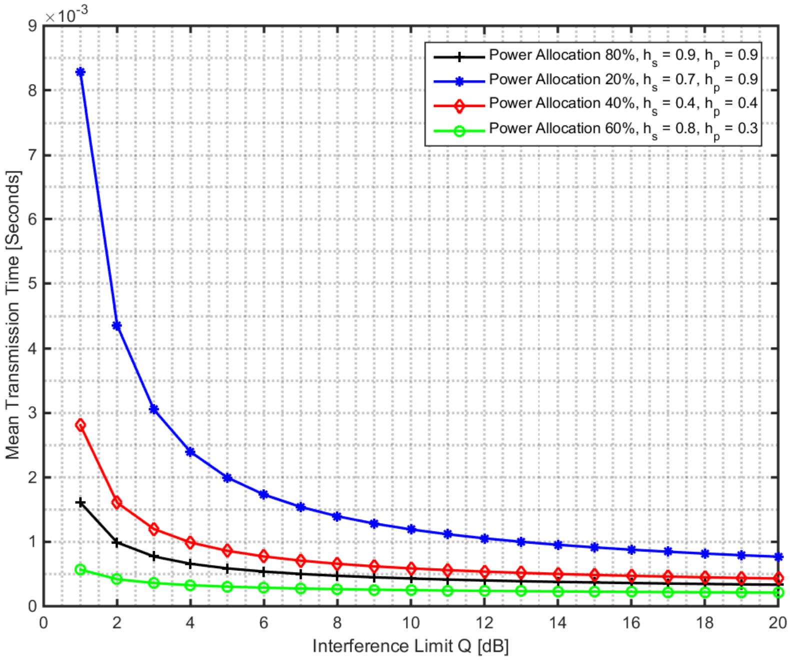

Figure **8.** Mean transmission time vs interference temperature limit L under various scenarios.

rate of Queue two be high enough, then Queue one will be negatively affected by the higher wait times before commencing service.

4.3.2. Non-pre-emptive queue model

Figures 15 to 17 show the expected number of packets in Queue one against the traffic intensity of Queue two under various traffic intensities. The mean number in the queue increases with traffic intensity because more Queue two packets will result in more Queue one packets having to wait in the queue because of the non-pre-emption rule.

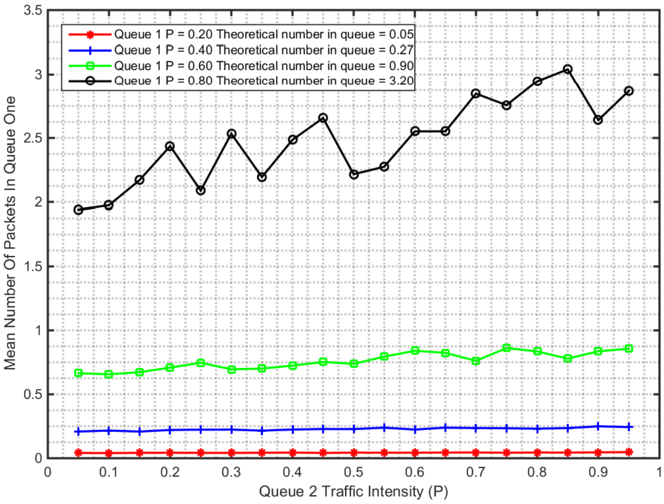

Figure 9. Queue one average number of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00045$ sec/packet and $\mu_4 = 0.00096$ sec/packet.

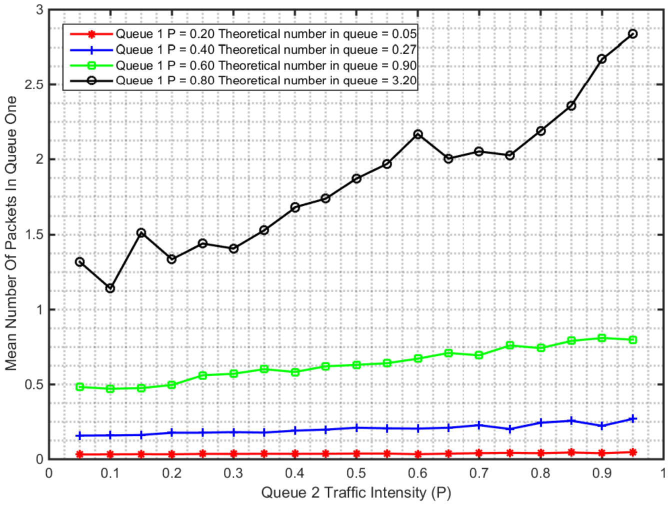

Figure 10. Queue one average number of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00051$ sec/packet and $\mu_4 = 0.00064$ sec/packet.

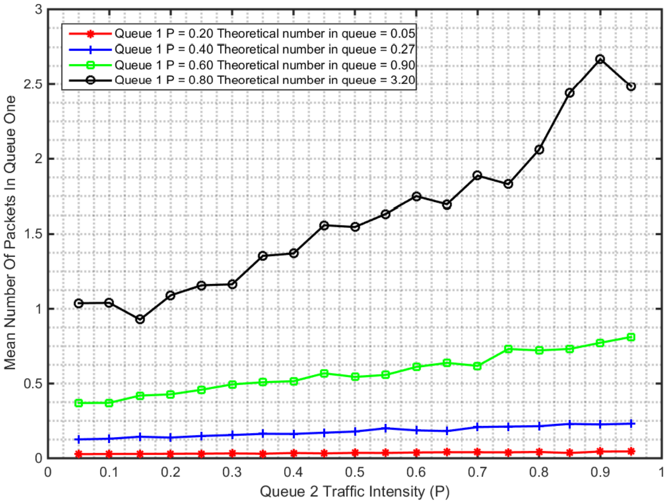

Figure 11. Queue one average number of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00055$ sec/packet and $\mu_4 = 0.00057$ sec/packet.

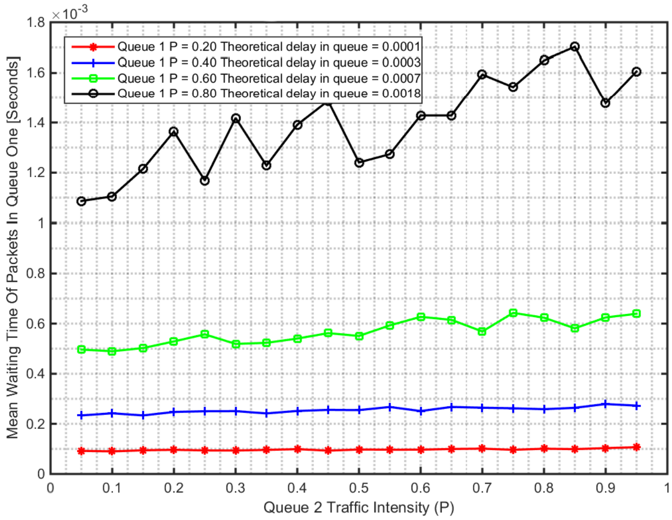

Figure 12. Queue one average waiting time of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00045$ sec/packet and $\mu_4 = 0.00096$ sec/packet.

Figures 18 to 20 show the expected delay in Queue one against the traffic intensity of Queue two under various traffic intensities.

The non-pre-emption rule will result in the packets experiencing higher waiting times as the traffic intensity increases. This is due to fact that an arriving packet is more likely to find a packet already in service and has to wait for the service to complete. Longer service times for any queue will result in higher waiting times for the other queue and vice versa.

In all cases the channel share priority scheme can be seen to be beneficial to the queues. The models allow for the transmission of two classes of packets without much harm to the other queue. Here Queue two is shown to benefit most from the scheme because of the large improvement from its normal rate to the maximum allowable rate. This holds if the traffic intensity for the HW queue remains low. It is important to note that the queue behaviour is identical, that is, if more power is allocated to Queue two than to Queue one then Queue two will yield performance similar to Queue one here and Queue one will then perform the

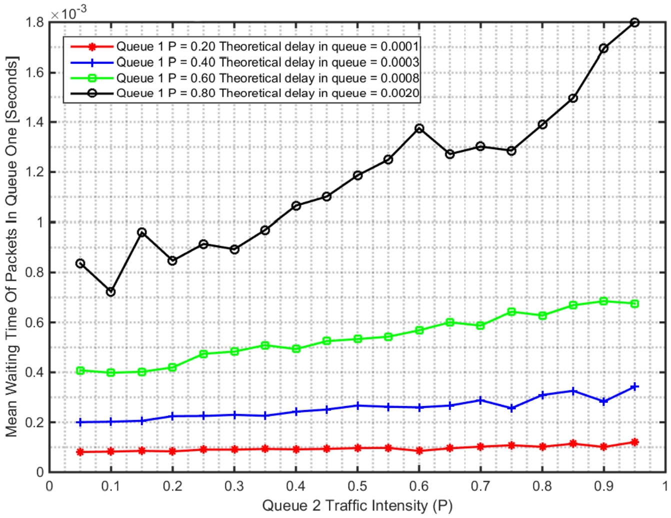

Figure 13. Queue one average waiting time of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00051$ sec/packet and $\mu_4 = 0.00064$ sec/packet.

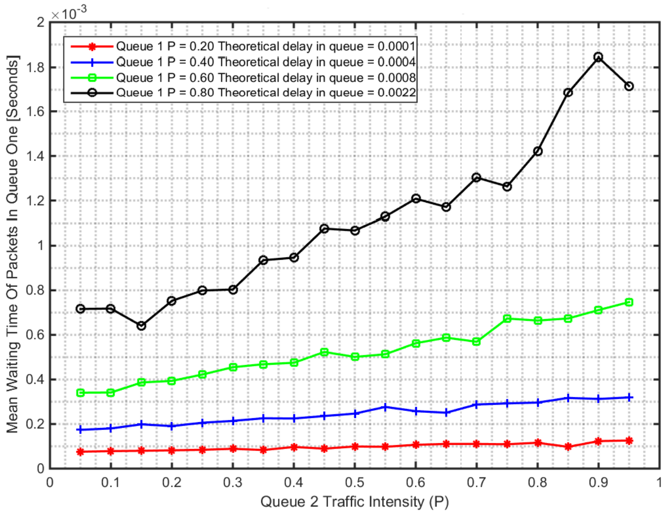

Figure 14. Queue one average waiting time of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00055$ sec/packet and $\mu_4 = 0.00057$ sec/packet.

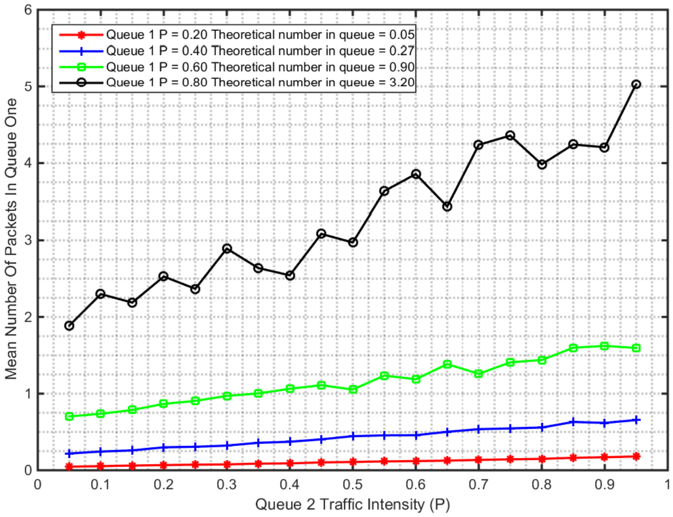

Figure 15. Queue one average number of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00045$ sec/packet and $\mu_4 = 0.00096$ sec/packet.

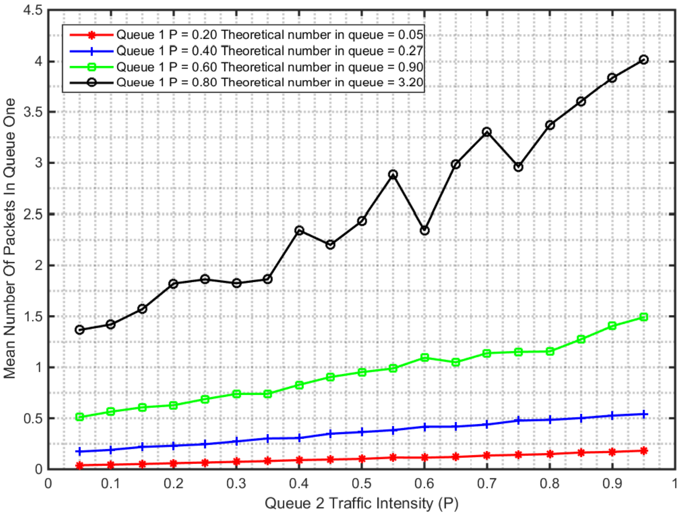

Figure 16. Queue one average number of packets. Here $\mu_1 = 0.00041$ sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00051$ sec/packet and $\mu_4 = 0.00064$ sec/packet.

same as Queue two. From the results of Queue one it can be seen that there is a threshold point where the queue performance will exceed the theoretical value. This is the point where optimisation can be carried out.

4.4. Optimisation Results

Given the above results and models, convex optimisation is deployed in order to maximise queue performance.

Figure 21 shows the minimum point for the graph to be around $\theta = 0.4$. That is to say, the power allocation ratio is 60% to the HW queue for optimum results despite it being weighted at 0.8 initially. Figure 22 , where the weighting is initially 60% to 40%, shows the minimum point for the graph to be around $\theta = 0.35$. The power allocation ration would dedicate 65% of the power to the HW queue for optimum results.

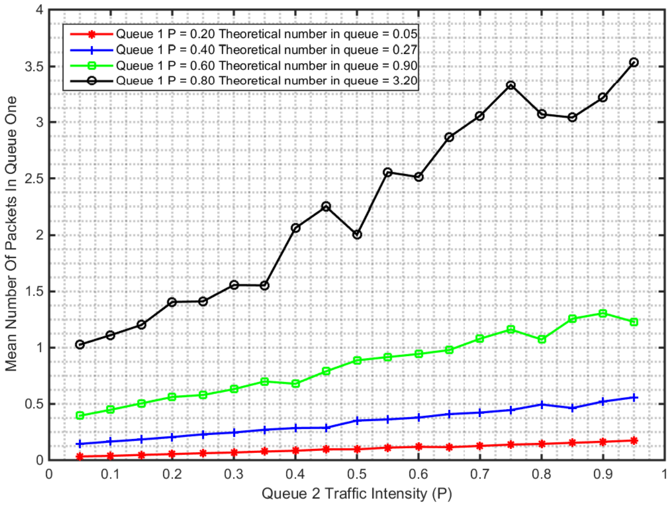

Figure 17. Queue one average number of packets. Here $\mu_1 =$ 0.00041 sec/packet, $\mu_3 = 0.00042$, $\mu_2 = 0.00055$ sec/packet and $\mu_4 = 0.00057$ sec/packet.

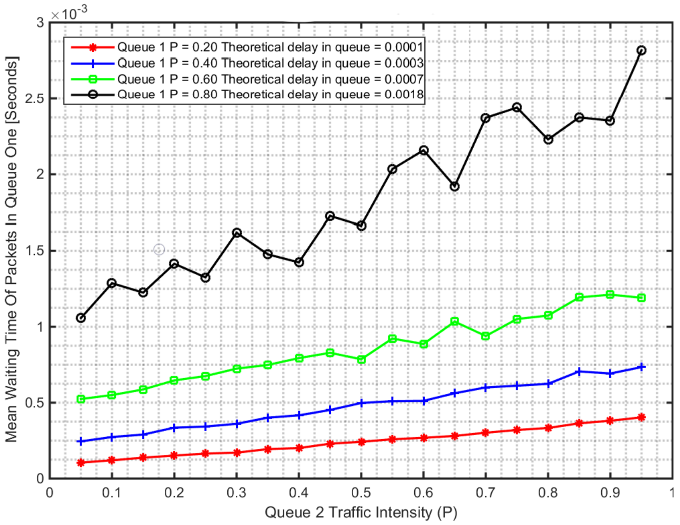

Figure 18. Queue one average waiting time of packets. Here μ_1 = 0.00041 sec/packet, μ_3 = 0.00042, μ_2 = 0.00045 sec/packet and $\mu_4 = 0.00096$ sec/packet.

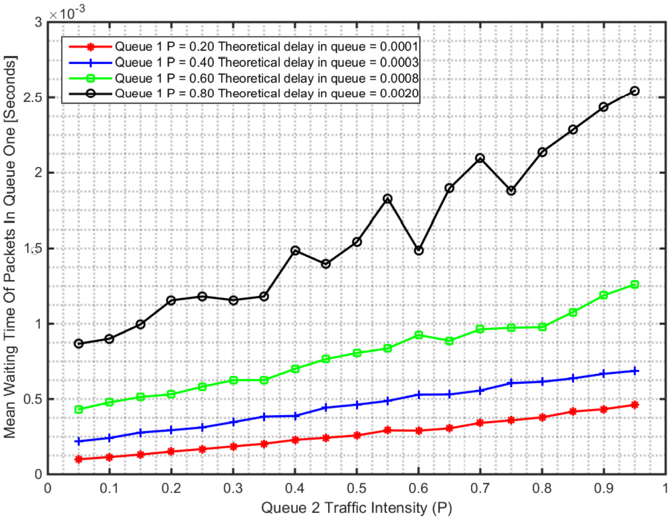

Figure 19. Queue one average waiting time of packets. Here μ_1 = 0.00041 sec/packet, μ_3 = 0.00042, μ_2 = 0.00051 sec/packet and $\mu_4 = 0.00064$ sec/packet.

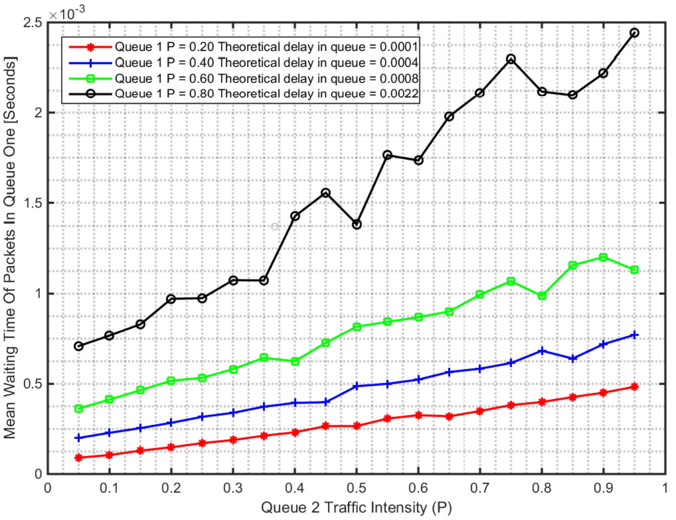

Figure 20. Queue one average waiting time of packets. Here μ_1 = 0.00041 sec/packet, μ_3 = 0.00042, μ_2 = 0.00055 sec/packet and $\mu_4 = 0.00057$ sec/packet.

The results show that the optimal allocation point varies only slightly with large changes in weighting. This may be due to the high dependency of the two queues on each other.

4.5. Comparison

The unoptimised and optimised models are compared to a standard M/M/1 two-class priority system in tables 1 to 4 . A standard priority system completely halts service for the LW queue if there are HW packets to be transmitted. Here the results are measured when Queues one and two are at 80% and 20% traffic intensity respectively. The power ratio is the power allocated to Queue one/power allocated to Queue two. The optimised and unoptimised results are compared to the well known result of standard priority queues by Miller[35].

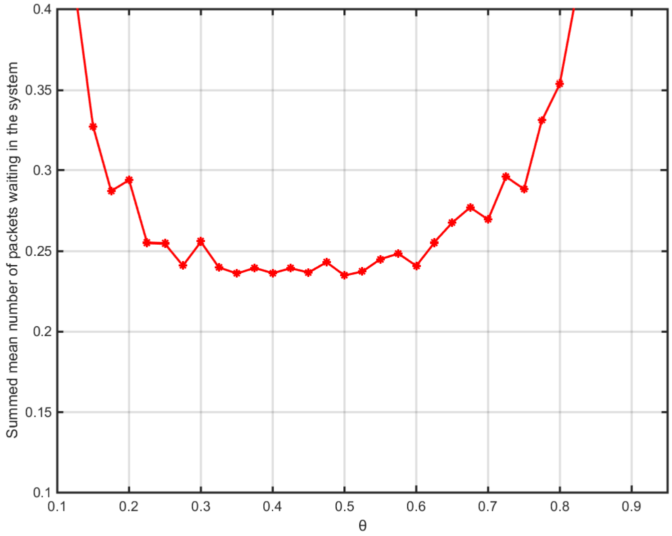

Figure 21. Mean number of packets in queues for the system. $\phi = 0.8$, $\beta = 0.2$

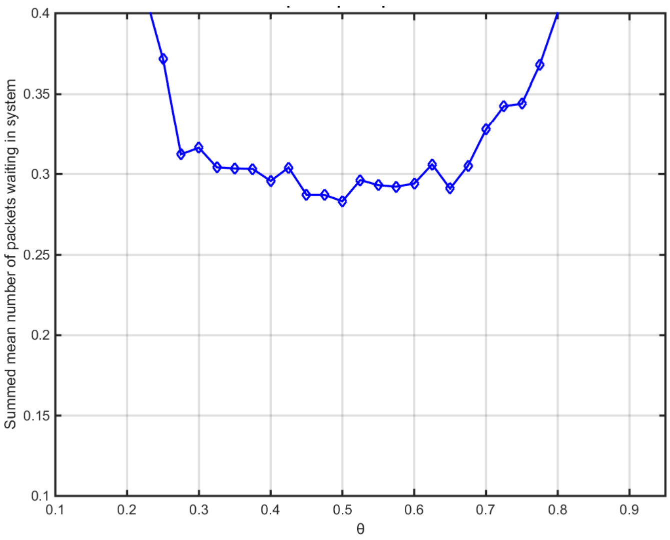

Figure 22. Mean number of packets in queues for the system. $\phi = 0.6$, $\beta = 0.4$

Table 1. Comparison between Processor Sharing and Standard Priority Queue Average Queue Lengths Pre-Emptive Discipline

Model	Queue 1	Queue 2	Total
Processor Sharing Power Ratio 80/20	2.5	0.13	2.6
Miller Standard Priority	0.9	20	20.9
Optimised Processor Sharing Power Ratio 80/20	1.5	0.7	2.2

Table 2. Comparison between Processor Sharing and Standard Priority Queue Average Queue Lengths Non-Pre-Emptive Discipline

Model	Queue 1	Queue 2	Total
Processor Sharing Power Ratio 80/20	2.5	0.7	3.2
Miller Standard Priority	1.4	>20	>20
Optimised Processor Sharing Power Ratio 80/20	2	1	3

The tables show that the proposed sharing scheme is beneficial in terms of the mean number of packets in the system overall. This means the second SU can be reliably served simultaneously with the first SU, thus increasing the number of users making use of the spectrum. Optimisation shows improvement to the network.

Table 3. Comparison between Processor Sharing and Standard Priority Queue Average Waiting Times Pre-Emptive Discipline

Model	Queue 1	Queue 2	Total
Processor Sharing Power Ratio 80/20	1.4 ms	0.1 ms	1.5 ms
Miller Standard Priority	0.5 ms	11 ms	11.5 ms
Optimised Processor Sharing Power Ratio 80/20	0.88 ms	0.4 ms	1.2 ms

Table 4. Comparison between Processor Sharing and Standard Priority Queue Average Waiting Times Non-Pre-Emptive Discipline

Model	Queue 1	Queue 2	Total
Processor Sharing Power Ratio 80/20	1.4 ms	0.5 ms	1.9 ms
Miller Standard Priority	0.5 ms	13 ms	13.5 ms
Optimised Processor Sharing Power Ratio 80/20	1.1 ms	0.6 ms	1.7 ms

5. Conclusion

A cognitive radio network model was used to increase spectrum utilisation by allowing at least two SUs to transmit together in the underlay mode. Queueing theory was used to formulate the objective function and constraints. After establishing convexity of the objective function, convex optimisation was then carried out on the system to determine optimal points for the resource allocation problem. The only constraint for the minimisation problem was the power allocation ratio per SU. ITL and power constraint were used as the basis for quality control. Queue length was the major performance measure used in the analysis of the queues. Future research will take into account more than two SUs.

APPENDIX

A PRE-EMPTIVE MODEL

$$B = \begin{array}{c} \\ 0,0 \\ 0,1 \\ 0,2 \\ \vdots \\ 0,K+1 \end{array} \begin{array}{c} \begin{array}{ccccc} 0,0 & 0,1 & 0,2 & \dots & 0,K+1 \end{array} \\ \begin{pmatrix} -\lambda & \lambda_2 & & & \\ \mu_3 & -(\lambda+\mu_3) & \lambda_2 & & \\ & \mu_3 & -(\lambda+\mu_3) & \lambda_2 & \\ & & \ddots & \ddots & \ddots \\ & & & \mu_3 & -(\lambda_1+\mu_3) \end{pmatrix} \end{array}$$

$$C = \begin{array}{c|ccccc} & 0,0 & 0,1 & 0,2 & \dots & 0,K+1 \\ \hline 0,0 & \lambda_1 & & & & \\ 0,1 & & \lambda_1 & & & \\ 0,2 & & & \lambda_1 & & \\ \vdots & & & & \ddots & \\ 0,K+1 & & & & & \lambda_1 \end{array} \quad A_0 = \begin{array}{c|ccccc} & n,0 & n,1 & n,2 & \dots & n,K+1 \\ \hline n,0 & \lambda_1 & & & & \\ n,1 & & \lambda_1 & & & \\ n,2 & & & \lambda_1 & & \\ \vdots & & & & \ddots & \\ n,K+1 & & & & & \lambda_1 \end{array} n \in 1, 2, 3...$$

$$E = \begin{array}{c} \\ 1{,}0 \\ 1{,}1 \\ 1{,}2 \\ \vdots \\ 1{,}K{+}1 \end{array} \begin{array}{c} \begin{array}{ccccc} 1{,}0 & 1{,}1 & 1{,}2 & \ldots & 1{,}K{+}1 \end{array} \\ \begin{pmatrix} \mu_1 & & & & \\ & \mu_2 & & & \\ & & \mu_2 & & \\ & & & \ddots & \\ & & & & \mu_2 \end{pmatrix} \end{array} \quad A_2 = \begin{array}{c} \\ n{,}0 \\ n{,}1 \\ n{,}2 \\ \vdots \\ n{,}K \end{array} \begin{array}{c} \begin{array}{ccccc} n{,}0 & n{,}1 & n{,}2 & \ldots & n{,}K{+}1 \end{array} \\ \begin{pmatrix} \mu_1 & & & & \\ & \mu_2 & & & \\ & & \mu_2 & & \\ & & & \ddots & \\ & & & & \mu_2 \end{pmatrix} \end{array} n \in 2, 3, 4...$$

$$A_1 = \begin{array}{c|ccccc} & n{,}0 & n{,}1 & n{,}2 & \ldots & n{,}K{+}1 \\ \hline n{,}0 & -(\lambda_1 + \mu_1) & \lambda_2 & & & \\ n{,}1 & \mu_4 & -(\lambda + \mu_b) & \lambda_2 & & \\ n{,}2 & & \mu_4 & -(\lambda + \mu_b) & \lambda_2 & \\ \vdots & & & \ddots & \ddots & \ddots \\ n{,}K{+}1 & & & & \mu_4 & -(\lambda_1 + \mu_b) \end{array}$$

$$n \in 1, 2, 3...$$

A.1 Rate Matrix

$$R = \begin{bmatrix} k_0 & k_1 & k_2 & k_3 & k_4 & k_5 & k_6 & \ldots \\ h_0 & r_0 & r_1 & r_2 & r_3 & r_4 & r_5 & \ldots \\ h_1 & w_0 & r_0 & r_1 & r_2 & r_3 & r_4 & \ldots \\ h_2 & w_1 & w_0 & r_0 & r_1 & r_2 & r_3 & \ldots \\ h_3 & w_2 & w_1 & w_0 & r_0 & r_1 & r_2 & \ldots \\ h_4 & w_3 & w_2 & w_1 & w_0 & r_0 & r_1 & \ldots \\ h_5 & w_4 & w_3 & w_2 & w_1 & w_0 & r_0 & \ldots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix} \tag{A1}$$

B NON-PRE-EMPTIVE MODEL

$$Q = \begin{bmatrix} B & C & & & \\ E & A_1 & A_0 & & \\ & A_2 & A_1 & A_0 & \\ & & \cdot & \cdot & \cdot \\ & & & \cdot & \cdot \end{bmatrix}, \tag{B2}$$

where

$$B = \begin{array}{c|ccccc} & 0{,}0 & 0{,}1 & 0{,}2 & \ldots & 0{,}K{+}1 \\ \hline 0{,}0 & B_1 & B_2 & & & \\ 0{,}1 & B_3 & B_4 & B_2 & & \\ 0{,}2 & & B_6 & B_4 & B_2 & \\ \vdots & & & \ddots & \ddots & \ddots \\ 0{,}K{+}1 & & & & B_6 & B_5 \end{array}$$

$$B_1 = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & -\lambda & & \\ 1 & & & \\ 2 & & & \end{array} \quad B_2 = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & & & \lambda_2 \\ 1 & & & \\ 2 & & & \end{array} \quad B_3 = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & & & \\ 1 & & & \\ 2 & \mu_3 & & \end{array} \quad B_4 = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & & & \\ 1 & & & \\ 2 & & & -(\mu_3 + \lambda) \end{array}$$

$$B_5 = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & & & \\ 1 & & & \\ 2 & & & -(\mu_3 + \lambda_1) \end{array} \quad B_6 = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & & & \\ 1 & & & \\ 2 & & & \mu_3 \end{array}$$

$$C = \begin{array}{cccccc} & 0,0 & 0,1 & 0,2 & \dots & 0,K{+}1 \\ 0,0 & C_1 & & & & \\ 0,1 & & C_2 & & & \\ 0,2 & & & C_2 & & \\ \vdots & & & & \ddots & \\ 0,K{+}1 & & & & & C_2 \end{array} \qquad E = \begin{array}{cccccc} & 1,0 & 1,1 & 1,2 & \dots & 1,K{+}1 \\ 1,0 & E_1 & & & & \\ 1,1 & & E_2 & & & \\ 1,2 & & & E_2 & & \\ \vdots & & & & \ddots & \\ 1,K{+}1 & & & & & E_2 \end{array}$$

$$C_1 = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & & \lambda_1 & \\ 1 & & & \\ 2 & & & \end{array} \quad C_2 = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & & & \\ 1 & & & \\ 2 & & & \lambda_1 \end{array} \quad E_1 = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & & & \\ 1 & \mu_1 & & \\ 2 & & & \end{array} \quad E_2 = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & & & \mu_2 \\ 1 & & & \mu_1 \\ 2 & & & \end{array}$$

$$A_2 = \begin{array}{cccccc} & n,0 & n,1 & n,2 & \dots & n,K{+}1 \\ n,0 & A_{21} & & & & \\ n,1 & & A_{22} & & & \\ n,2 & & & A_{22} & & \\ \vdots & & & & \ddots & \\ n,K{+}1 & & & & & A_{22} \end{array} \quad A_{21} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & \mu_1 & & \\ 1 & & \mu_1 & \\ 2 & & & \mu_1 \end{array} \quad A_{22} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & \mu_2 & & \\ 1 & \mu_1 & & \\ 2 & & & \end{array}$$

$$A_1 = \begin{array}{cccccc} & n,0 & n,1 & n,2 & \dots & n,K{+}1 \\ n,0 & A_{11} & A_{15} & & & \\ n,1 & A_{13} & A_{12} & A_{15} & & \\ n,2 & & A_{13} & A_{12} & A_{15} & \\ \vdots & & & \ddots & \ddots & \ddots \\ n,K{+}1 & & & & A_{13} & A_{14} \end{array}$$

$$n \in 1, 2, 3, \dots$$

$$A_{11} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & & & \\ 1 & & -(\mu_1 + \lambda) & \\ 2 & & & \end{array} \quad A_{13} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & & \mu_4 & \\ 1 & & & \\ 2 & & \mu_3 & \end{array} \quad A_{15} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & \lambda_2 & & \\ 1 & & \lambda_2 & \\ 2 & & & \lambda_2 \end{array}$$

$$A_{12} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & -(\lambda + \mu_b) & & \\ 1 & & -(\lambda + \mu_1) & \\ 2 & & & -(\lambda + \mu_3) \end{array} \quad A_{14} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & -(\lambda_1 + \mu_b) & & \\ 1 & & -(\lambda_1 + \mu_1) & \\ 2 & & & -(\lambda_1 + \mu_3) \end{array}$$

$$A_0 = \begin{array}{cccccc} & n,0 & n,1 & n,2 & \dots & n,K{+}1 \\ n,0 & A_{01} & & & & \\ n,1 & & A_{01} & & & \\ n,2 & & & A_{01} & & \\ \vdots & & & & \ddots & \\ n,K{+}1 & & & & & A_{01} \end{array} \quad A_{01} = \begin{array}{cccc} & 0 & 1 & 2 \\ 0 & \lambda_1 & & \\ 1 & & \lambda_1 & \\ 2 & & & \lambda_1 \end{array}$$

B.1 Rate Matrix

$$R = \begin{bmatrix} K_0 & K_1 & K_2 & K_3 & K_4 & K_5 & K_6 & \dots \\ H_0 & R_0 & R_1 & R_2 & R_3 & R_4 & R_5 & \dots \\ H_1 & W_0 & R_0 & R_1 & R_2 & R_3 & R_4 & \dots \\ H_2 & W_1 & W_0 & R_0 & R_1 & R_2 & R_3 & \dots \\ H_3 & W_2 & W_1 & W_0 & R_0 & R_1 & R_2 & \dots \\ H_4 & W_3 & W_2 & W_1 & W_0 & R_0 & R_1 & \dots \\ H_5 & W_4 & W_3 & W_2 & W_1 & W_0 & R_0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix} \tag{B3}$$

where

$$K_i = \begin{array}{cc} & \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \end{array} & \begin{pmatrix} k_{i1} & p_{i1} & 0 \\ p_{i2} & k_{i2} & 0 \\ p_{i3} & p_{i4} & k_{i3} \end{pmatrix} \end{array} \quad H_i = \begin{array}{cc} & \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \end{array} & \begin{pmatrix} 0 & h_{i1} & 0 \\ 0 & h_{i2} & 0 \\ 0 & h_{i4} & 0 \end{pmatrix} \end{array} \quad R_i = \begin{array}{cc} & \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \end{array} & \begin{pmatrix} r_{i1} & s_{i1} & 0 \\ s_{i2} & r_{i2} & 0 \\ s_{i3} & s_{i4} & r_{i3} \end{pmatrix} \end{array} \quad W_i = \begin{array}{cc} & \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \end{array} & \begin{pmatrix} w_{i1} & t_{i1} & 0 \\ t_{i2} & w_{i2} & 0 \\ t_{i3} & s_{i4} & 0 \end{pmatrix} \end{array}$$

References

1. Stable G, Werbach K. The end of spectrum scarcity. *IEEE Spectr.* 2004;41(3):48–52.
2. Mitola J, Maguire G.Q. Cognitive radio: Making radios more personal. *IEEE Pers Commun.* 1999;6(4):13–18.
3. Pischella J, Le Ruyet D. Cooperative allocation for underlay cognitive radio systems. In: IEEE 14th Workshop on Signal Processing Advances in Wireless Communications:245-249; Jun 16–19, 2013; Darmstadtium.
4. Sun S, Ju Y, Yamao Y. Overlay cognitive radio OFDM system for 4G cellular networks. *Wireless Commun.* 2013;20(2):68–73.
5. Oh J, Choi W. A hybrid cognitive radio system: A combination of underlay and overlay approaches. In: Vehicular Technology Conference Fall; September 1–9, 2010; Ottawa.
6. Akyildiz IF, Lee W, Vuran MC, Mohanty S. A survey on spectrum management in cognitive radio networks. *IEEE Commun Mag.* 2008;46(4):40–48.
7. Alfa AS, Maharaj BT, Lall S, Pal S. Mixed-integer programming based technologies for resource allocation in underlay cognitive radio networks: A survey. *J Commun Netw.* 2016;18(5):744-761.
8. Boyd S, Vandenberghe C. *Convex Programming*. Cambridge: Cambridge University Press; 2004. ISBN 978-0521833783.
9. Zhao H, Chen L, Feng W. A signal detection scheme for wireless sensor networks based on convex optimization. *IEEE SENSORS J.* 2016;77(11):1–3.
10. Chuhan L, Trivedi A. Priority based MAC scheme for cognitive radio networks: A queuing theory modelling. *Wireless Opt Commun Netw.* 2012;188(2):1–5.
11. Gross D, CM Harris. *Fundamentals of Queueing Theory*. New Jersey: John Wiley and Sons Inc; 1985. ISBN 978-0471791270.
12. Farraj AK, Miller SL, Qarage KA. Queue performance measures for cognitive radio networks. In: IEEE International Workshop on Recent Advances in Cognitive Communications and Networking:997–1001; December 2011; Budapest.
13. Le LB, Hossain E. Resource allocation for spectrum underlay in cognitive radio networks. *IEEE Trans Wireless Commun.* 2008;7(12):5306–5315.
14. Veciana G, Kesidis G. Bandwidth allocation for multiple qualities of service using generalized processor sharing. In: Global Telecommunication Conference:1550–1554; December 1994; San Fransisco.
15. Jiang T, Wang H, Vasilakos AV. Schemes for multimedia transmission of priority-based secondary users over cognitive radio networks. *IEEE J Sel Areas Commun.* 2012;30(7):1215–1224.
16. Szabo R, Barta P, Biro J, Nemeth F. Nonrate-proportinal weighting of generalized processor sharing schedulers. In: Global Telecommunication Conference:1334–1339; December 1999; Rio de Janeiro.
17. Zhang B, Zhang G. Design of dual-processor sharing DRAM controller. *Comp Sci Eng.* 2009;2:298–302.
18. Jing F, Guo J, Wong EWM, M Zukerman. Energy-efficient heuristics for insensitive job assignment in processor-sharing server farms. *IEEE J Sel Areas Commun.* 2015;33(12):2878–2891.

19. Morrison JA. Head of the line processor sharing for many symmetric queues with finite capacity. *J App Maths.* 1993;14(1):215–237.

20. Bhoopendra Kumar, Sanjay Kumar Dhurandher, Isaac Woungang. A survey of overlay and underlay paradigms in cognitive radio networks. *International Journal of Communication Systems.* 2017;31(2):e3443. e3443 dac.3443.

21. Senthuran S., Anpalagan A., Das O.. Throughput Analysis of Opportunistic Access Strategies in Hybrid Underlay-Overlay Cognitive Radio Networks. *IEEE Transactions on Wireless Communications.* 2012;11(6):2024-2035.

22. Zou J., Xiong H., Wang D., Chen C. W.. Optimal Power Allocation for Hybrid Overlay/Underlay Spectrum Sharing in Multiband Cognitive Radio Networks. *IEEE Transactions on Vehicular Technology.* 2013;62(4):1827-1837.

23. Wang J., Zhang Y., Li W. W.. Strategic Joining and Optimal Pricing in the Cognitive Radio System With Delay-Sensitive Secondary Users. *IEEE Transactions on Cognitive Communications and Networking.* 2017;3(3):298-312.

24. She C., Yang C., Quek T. Q. S.. Cross-Layer Optimization for Ultra-Reliable and Low-Latency Radio Access Networks. *IEEE Transactions on Wireless Communications.* 2018;17(1):127-141.

25. K. Manivannan C.G Ravichandran, Durai B. Sakthi Karthi. A Dynamic QoS Model for improving the throuput of wideband spectrum sharing in cognitive radio networks. *KSII Transactions on Internet and Information Systems.* 2014;8(11):3731-3749.

26. Tadayon N., Kaddoum G.. Packet-Level Modeling of Cooperative Diversity: A Queueing Network Approach. *IEEE Access.* 2018;:http://dx.doi.org/10.1109/ACCESS.2018.2832130.

27. Xia M., Aissa S.. Underlay cooperative af relaying in cellular networks: performance and challenges. *IEEE Communications Magazine.* 2013;51(12):170-176.

28. Duy Tran Trung, Son Pham Ngoc. A novel adaptive spectrum access protocol in cognitive radio with primary multicast network, secondary user selection and hardware impairments. *Telecommunication Systems.* 2017;65(3):525–538.

29. Chu TMC, Phan H, Zepernick H. On the performance of underlay cognitive radio networks using M/G/1/K queueing model. *IEEE Commun Lett.* 2013;17(5):876–879.

30. Winston WL, Venkataramana M. *Introduction to Mathematical Programming*. Boston: Thomson Learning; 2002. ISBN 978-0534359645.

31. Tsimba HM, Maharaj BT, Alfa AS. Increased spectrum utilisation in a cognitive radio network: An M/M/1-PS queue approach. In: IEEE Wireless Communications and Networking Conference:1–6; March 19-23, 2017; San Fransisco.

32. Mangasarian OL. Pseudo-convex functions. *J Soc Ind Appl Math Series A Control.* 1965;3(2):281–290.

33. Whiit W. *Stochastic-Process Limits: An Introduction to Stochastic-Process Limits and their Application to Queues*. New York: Springer Series in Operations Research and Financial Engineering; 2002. ISBN 978-0-387-21748-2.

34. IEEE . *Functional requirements for the 802.22 WRAN standard.* IEEE 802.22-05/0007r46: IEEE; 2005.

35. Miller D. R.. Computation of steady-state probabilities for M/M/1 priority queues. *Journal of Applied Probability.* 1981;29(5):945-958.