A Bounded Multi-Vacation Queue Model for Multi-Stage Sleep Control*

Jie Chen[a]
Imperial College London, Exhibition Road, Exhibition Road, SW7 2AZ, London, U.K.

Keywords: Queuing Analysis, Performance Evaluation, Optimisation, Communication System Performance Control.

Abstract: To evaluate the performance of multi-stage sleep telecommunication systems, this paper presents a bounded multi-vacation queue model. The energy consumption predicted by this model, shows an average error rate of 0.0177 and the delay (predicted by the same model) shows an average error rate of 0.0655. Both error rates were calculated over 99 instances.A general algorithmic method integrating the analytical model further demonstrates the model's accuracy.

1 INTRODUCTION

Bounded multi-stage sleep mode control has emerged as an implementation feature for energy efficient telecommunication networks.

In this scheme, mobile devices go into hibernation gradually from light to deep sleep through a limited number of discrete stages. They will then resume to work when either a new workload arrives or the hibernation is finished.To analyse this scheme, this paper draws upon queuing theory and proposes the BMV (Bounded Multi-Vacation) policy that is generalised from the scheme. The vacation queue system has been in discussion in the literature for a long time. These systems work upon the policy of determining whether or not the number of packets in a queue has reached a threshold ($N > 0$) or the vacation time has exceeded a certain amount (T policy) (B.T.Doshi, 1986). In the following subsections, the feasibility of this new policy and its performance against other options are being discussed.

1.1 The Merits of BMV-Policy over Other Policies

Investigations have demonstrated the convincing results that BMV policy can beat N-policy and T-policy in terms of system performance and reliability. As N-policy has only adjustable parameters of K (the system buffer maximum quota), it has a bounded energy consumption rate and delay. Whereas BMV-policy can tune its N_v (vacation amount limit) and L_v (vacation length mean) across much wider ranges to guarantee an improved solution. Similarly T-policy can be treated as a single vacation(SV) policy. Given a fixed T, results have shown that if being broken into multiple equally weighted vacations to make a BMV-policy, the system would achieve a much smaller delay. Figure 1 is in agreement with paper (J.Chen et al., 2018) that power consumption level fluctuates and delay increases with the increasing N. In this particular simulation, $\lambda = 550$, $\mu = 1000$, $K = 50$, $power_{on} = 130$, $power_{off} = 75$. Results show that though $\rho = 0.55$ and $ratio_{power} = 0.5769$, the normalised energy consumption per bit for N-policy regardless of which N is selected, goes much higher than 0.6. The system has not been saturated in terms of energy conservation efficiency. As delay is a traditional QoS metric of a network system, we might also want the new scheme would outperform N-policy in terms of processing speed. Given a bounded delay for N-policy as $[\, D_{min}, D_{max}\,]$,solutions with BMV-policy can be easily founded that match the design criteria that consumes less energy while falls within the delay bounds. Two of them are depicted on the figure as examples.

Suppose in T-policy, the vacation length is L_v and in BMV-policy, with the increase of n (maximum number of vacations), $L_v^{BMV} = \frac{L_v}{n}$, where n is the maximum number of vacations. The cases where $n \in [1\ 7]$ are executed and evaluated.Figure 2, shows that with the increasing n, the delay decreases whilst the energy level fluctuates.Based on the limited tested cases, $NE_{BMV} > NE_T$ where NE stands for normalised en-

[a] https://orcid.org/0000-0002-7147-6570

*An abstract presentation has been conducted in EURO 2022 Aalto (J.Chen,).

Chen, J.
A Bounded Multi-Vacation Queue Model for Multi-Stage Sleep Control.
DOI: 10.5220/0012325200003639

In *Proceedings of the 13th International Conference on Operations Research and Enterprise Systems (ICORES 2024)*, pages 245-252
ISBN: 978-989-758-681-1; ISSN: 2184-4372

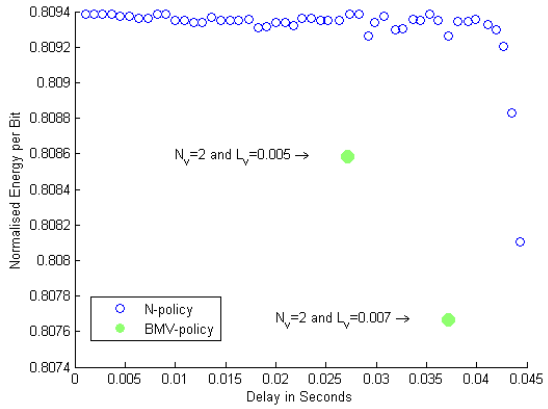


Figure 1: BMV-policy vs N-policy (Simulation).

ergy whilst $D_{BMV} < NE_T$. Given a fixed delay bound $[\ D_{min}\ D_{max}\ ]$ imposed by N-policy, results show that BMV-policy can produce feasible solutions with higher energy savings.

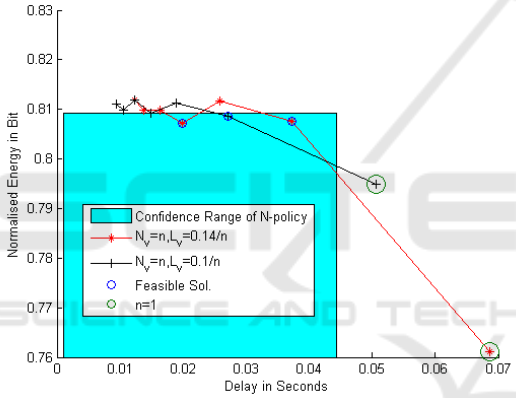


Figure 2: BMV-policy vs T-policy (Simulation).

1.2 Contribution

- The following work is the first to theoretically discuss multi-stage sleep control in the current state of art telecommunication system.
- This work bridges the mathematically theoretical analysis (rigorous mathematical derivation and proof) and the practical engineering problem by the validation using software simulation. Previous works in theoretical queuing analysis rarely endeavor to go through thorough experimental tests. Works in mobile engineering have also not yet developed analysis from generalised problem settings. Analysis from a more generalised abstract level can contribute to theoretical queuing analysis development.
- This work treats the system design problem as an optimal control problem considering the trade-off between delay and energy consumption and provides sound analysis against both of these system metrics. Most of the new queuing analysis are solely devised to evaluate the delay metrics. These analyses have additionally evaluated the cost metrics such as power consumption in a typical telecommunication system.This work is the first to propose a validated model for accurate future prediction of those cost metrics.

2 RELATED WORKS

Works on energy efficient network optimisation fall into two categories: the first is using analytical derivation (computational intelligence) to attain a precise solution and the second is using artificial intelligence to converge into an approximately accurate solution. In general, though AI algorithm has the advantage to handle larger data set,it seeks a solution that is considered empirically close to the ground truth without knowing what the ground truth is. And research works in AI application in energy efficient network design emerges in recent years.For example,paper(Q.Wu et al.,) proposed an AI methodology based framework to predict the dynamic of the traffic and henceforth to control the base station activities. Somehow, the base station has only on and off two states and the operation time span is half an hour. It means, the base station once set to be off will stay asleep for half an hour regardless of any emergency traffic abnormal to the training data set.It is not realistic in practical hardware design. As mentioned in paper(M.Feng et al., 2017), especially for small scale base station, on site cooling is not possible and it is critical to take into account the detection power.The sleep mode is classified into 4 stages: ON,OFF,Standby,Sleep.Furthermore,the author in paper (Q.Wu et al.,) has not demonstrated the effectiveness of the scheme in delay cutoff.According to the optimisation problem formulation, the delay penalty is expected to be as minute as possible and there exists a trade-off between the power consumption and the delay.In traffic prediction, the author utilises 20 days' traffic data set to train and evaluate the scheme using the last 10 days' traffic data. The first category of analytical approach, though relatively conventional, has been still robust and reliable in network design in general. In the speciality area of energy efficient network design, vacation queue is applied. N-policy is of most interest as it relates much to the buffer size in terms of data packet which has been conceptualised concretely in network software design. Paper(P.Badian-Pessot et al., 2016) provides

a theoretical proof of the existence of an optimal work conserving policy by utilising continuous-time markov chain theory and analysing the average cost optimality equations (ACOE) for the problem. It also provides the experimental verification of the proof such that scenario where the server always works at the highest service rate and where the server turns off when the queue is empty are being used as the benchmark policies. It shows that the on policy where the server turns on when the system has N packets in queue and turn off otherwise outperforms these two benchmark policies.Paper(J.Wu et al., 2020)applies the N-policy in that it proposes three schemes to achieve the system performance goal in terms of energy consumption,delay and blocking probability : the first scheme is literally N-policy queue, the second scheme is cooperative where the residual traffic from sleepy BS can be diverted to active BS and the third scheme is hybrid. An analytical model is proposed for the N-policy queue. Furthermore, the authors also discusses the feasibility to accommodate different service rate distributions and Markov Modulated Poisson Process as input rate distribution. Regarding the cooperative scheme, the author utilises IESA (Information Exchange Surrogate Approximation) to best estimate the performance parameters.

Apart from just applying queuing policies, queueing analysis techinques have been borrowed to evaluate more complex scenario such as cognitive radio where users are priotorized to access the spectrum.The authors in paper(J.Liu et al., 2019) treat the system state as a tuple size of three, each representing the overall number of secondary users in the system, the number of secondary user packets to be served by the channel and the number of primary user packets to be served by the channel respectively.Then they derive the probability transition matrix and from the matrix, attain the stable queue length distribution. The authors also propose how to measure the latency,throughput,energy saving rate, etc and form the cost function as the weighted sum of these system parameters.The analytical results are consistent with the simulation results. The below works also employ other theorem and techniques to facilitate the queuing analysis.Paper(T.Phung-Duc, 2020) provides a precise analysis of the waiting time and queue length probability distribution. In doing that, the author applies Rouche's Theorem to gain a closed form solution to the generating function of the probability distribution and proposed a recursive algorithm to attain the queue length probability distribution. The author considers the set up time and treated the system as one with no abandonment.The short paper (Yazici.M.A and T.Phung-Duc, 2020) is tightly written. It applies fluid analysis to attain the workload distribution of the system, evaluates the cost function borrowed and provides results for power consumption and system waiting time tradeoff based on the analysis. Paper(J.Pender and T.Phung-Duc, 2016) contributed by the same author continues to use fluid limit theorem to predict the queue length.

3 PROBLEM FORMULATION

In paper (J.Chen et al., 2018), the system is perceived to rotate between sleep mode and working mode. Following this approach, the system performance is evaluated such that instead of focusing on an equilibrium long term, with the system running a countably infinite time frame by simulation, the system running thread is composed of multiple running cycles. In this paper, by averaging over these running cycles, a particular uniform cycle is inspected, that consists of a sleeping sub-frame and an active sub-frame, the statistically distributed measurements such as power consumption and delay are calculated and equated to those sub-frames in the longer term.

3.1 System Description

The queuing system consists of an intelligent server that can vacate whenever the queue is empty. The vacation duration is adjusted based on two parameter configurations. These parameters are, the maximum vacation number N_v and the average vacation period L_v. To be more specific, the queue, once in vacation mode will return to the workstation whenever a vacation period expires. If the queue is still empty, it will continue to next vacation period until the maximum vacation number is reached. Otherwise it will resume to work upon its return to the workstation.Please refer to Figure 3 for further illustration.

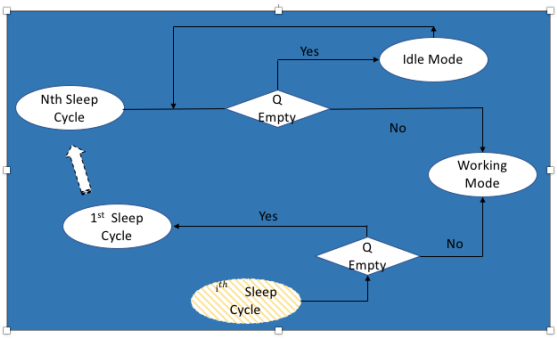


Figure 3: System Work Flow.

According to 3, the server transits from the working mode to the sleep mode whenever the queue is

Table 1: The list of symbols used in this paper.

Symbol	Definition
N_v	maximum vacation number
L_v	average vacation period
λ	input poisson traffic rate
μ	exponential distributed service rate
p_s	the vacation power
p_a	the working power
L_s	average sleeping sub-frame length
L_b	average working sub-frame length
$P_{L_s}(i)$	the probability that packets start to arrive during i^{th} sleeping period
P_{init}	the probability distribution of queue length upon the working period
p_k	the probability that the arrival packet number is equal to k
$P_{\text{other}}(n)$	the queue length distribution at n^{th} departure epoch with the zero queue length probability value set as 0
$P_{\text{zero}}(n)$	the probability that the server encounters an empty queue at n^{th} departure epoch
P_0	initial queue length probability distribution
$\bar{P}_0$	the conditional queue length distribution that the server doesn't see an empty queue at the initial departure epoch
P_{sum_k}	the probability that the server stops before k^{th} departure epoch
$E[L_i]$	average idle period length
L_a	inter-arrival time for the first packet in the idle mode
N_{L_s}	number of arrivals within period L_s
E_i	normalised energy per bit
$E[NE]$	average normalised energy per bit
W	average waiting time
$\gamma(t)$	packet time summation up to time instance t
$\alpha(t)$	in-queue packet number summed up to time instance t
$\triangle\gamma(t)$	the overall packet-in queue time for an averaged running cycle
$\triangle\alpha(t)$	the overall in queue packet number for an averaged running cycle
ρ	load
K	the queue buffer limit
$L_Q{}^{init}$	the conditional queue length when the system resumed to work
N_a	number of arrivals
P_{ql_k}	the conditional queue length when the system is active at departure epoch k

empty.It transits from the sleep mode to the working mode whenever the queue is not empty. If the server has waited for the maximum sleep cycle length and there is still no packet coming, the server will enters the idle mode. From idle mode, the server will decide whether it enters the working mode depending on whether the queue is empty or not. The input traffic model follows a Poisson distribution with an average rate of λ and the service pattern follows an exponential distribution with an average rate of μ. Currently, the power that the sever uses at vacation is p_s, the vacation length is uniform over all stages and the power that the server uses at work is p_a.

3.2 Power Consumption Analysis

In this paper, it is assumed that the sleeping sub-frame has a length of L_s and the working sub-frame has a length of L_b. By design it is also assumed that the system starts with sleeping mode.

As there are at most N_v sleeping periods, for the system to enter working mode after the first period, there must be at least 1 arrival during the first period. For the system to enter working mode after the n-th period with $n \leq N_v$, there needs to be at least 1 arrival during the previous $(n-1)th$ period but none happens during the previous $(n-2)$ periods.

Let A be the scenario where there is at least 1 arrival within time frame iL_v; let B be the scenario where there is at least 1 arrival within time frame $(i-1)L_v$; let C be the scenario where there is at least 1 arrival within time frame $iL_v - (i-1)L_v$ only, which is equivalent to the phenomena where the packets start to arrive during i^{th} sleeping period. Henceforth $C = A - B$. Further, $P(C) = P(A) - P(B)$. The formula is constructed as:

$$P_{L_s}(i) = e^{(-\lambda(i-1)L_v)} - e^{(-\lambda(i)L_v)}$$

Upon entering the working period, the starting probability distribution of the queue length of the system which is $P_{init} = [p_k] \times K$ with K is the maximum queue size. $p_k = e^{(-\lambda L_v)} \frac{(\lambda L_v)^k}{k!}$

Theorem 1. *As the system probabilistically evolves from the initial distribution towards an approximately zero position dominated distribution,such that the $P_{\text{other}}(n)$ is approximately 0, the summation of the $P_{\text{zero}}(n)$ across all the stopping point is approximately to 1 as much as possible, $\forall\varepsilon$, $\exists N$, when $n > N$, $1 - \sum_k^n P_{\text{zero}}(k) \leq \varepsilon$.*

Proof. Assume the initial probability distribution has $P_0 = [P_0(0)\ P_{\text{other}}(0)]$ and $\overline{P}_0 = [0\ P_{\text{other}}(0)]$, $[P_0(1)\ P_{\text{other}}(1)] = \overline{P}_0 * P_{\text{tran}}$.

$$\begin{aligned} P\text{sum}_1 &= \sum_{i=0}^{1} P_0(i) = P_0(0) + P_0(1) \\ &= \sum_{i=1}^{K} P^i_{\text{other}}(0) - \sum_{i=1}^{K} P^i_{\text{other}}(1) + P_0(0) \\ &= 1 - \sum_{i=1}^{K} P^i_{\text{other}}(1) \leq 1 \end{aligned}$$

$$\begin{aligned} P\text{sum}_2 &= \sum_{i=0}^{2} P_0(i) = P_0(0) + P_0(1) + P_0(2) \\ &= 1 - \sum_{i=1}^{K} \overline{P1}_i + \sum_{i=1}^{K} \overline{P1}_i - \sum_{i=1}^{K} \overline{P2}_i \\ &= 1 - \sum_{i=1}^{K} \overline{P2}_i \end{aligned}$$

It can be intuitively derived that

$$\begin{aligned} P\text{sum}_n &= 1 - \sum_{i=1}^{K} \overline{P(n)} \\ &= 1 - P\text{sum}_{\text{other}}(n) \\ \sum_{i=0}^{K} \overline{P_i(n)} &= \sum_{i=0}^{K} \sum_{j=1}^{K} \overline{P_j(n-1)} * P_{\text{tran}}(j,i) \\ &= 1 - \sum_{i=0}^{K} \overline{P_0(n-1)} * P_{\text{tran}}(0,i) \\ &= \sum_{i=1}^{K} \overline{P_i(n-1)} \\ &= \sum_{i=0}^{K} \overline{P_i(n-1)} - \varepsilon_n \end{aligned} \tag{1}$$

where $\varepsilon_n = \overline{P_0(n-1)}$. Hence, the following equation can be justified, $P\text{sum}_{\text{other}}(n) < P\text{sum}_{\text{other}}(n-1) \leq 1$. $P\text{sum}_{\text{other}}(i)$ is thus a monotonically decreasing sequence while $P\text{sum}_n$ is a monotonically increasing sequence within the frame $[0,1]$. Hence,given an ε as small as possible, there always exist an N that $N = {}_i P\text{sum}_{\text{other}}(i) > \varepsilon$, for $n > N$, $1 - P\text{sum}_n = P\text{sum}_{\text{other}}(n) < \varepsilon$. □ □

Assuming the transition matrix is P_{tran},the formation of P_{tran} for the current $M/M/1/K$ queue system can be extended from paper (J.Chen et al., 2018).

$$P_{\text{tran}} = [p_{i,j}] * [K \times K]$$

$$p_{i,j} = \begin{cases} 0 \\ \text{if } j < i-1 \\ \int_0^\infty \mu e^{(-(\lambda+\mu)t)} \frac{(t\lambda)^{(j-i+1)}}{(j-i+1)!} \\ \text{if } (i-1) <= j < K \\ \sum_{j=K}^{\infty} \int_0^\infty \mu e^{(-(\lambda+\mu)t)} \frac{(t\lambda)^{(j-i+1)}}{(j-i+1)!} \\ \text{if } j = K \end{cases} \tag{2}$$

$$[P_{\text{zero}}(k)\ P_{\text{other}}(k)] = [0\ P_{\text{other}}(k-1)] * P_{\text{tran}}$$

$$\begin{aligned} E[L_b] &= E[E[L_b|l_k = \sum_{i}^{k} x_i] \\ &= \sum_{k}^{n>N} P_{\text{zero}}(k) E[l_k]] \\ &= \sum_{k}^{n>N} P_{\text{zero}}(k) k E[x_i] \\ &= \sum_{k}^{n>N} P_{\text{zero}}(k) k \frac{1}{\mu} \end{aligned} \tag{3}$$

The special scenario in which there is no arrival within the maximum number of vacation periods is analysed as below: The period between the end of the overall sleeping sub-frame and the beginning of server running period is labeled as *ilen* - the idle length. As the Poisson Distribution follows an individually independent Markovian pattern, *ilen* is perceived as the inter-arrival time between the zeroth arrival and the first arrival minus the maximum overall sleeping sub-frame.

$$\begin{aligned} E[L_i] &= E[L_a|N_{L_s} = 0] \\ &\quad - L_v N_v \\ &= \int_{L_v N_v}^{\infty} \frac{t\lambda e^{(-t\lambda)}}{e^{(-(L_v N_v)\lambda)}} dt \\ &\quad - L_v N_v \\ &= \frac{e^{L_v N_v \lambda}}{\lambda} \Gamma(2, L_v N_v \lambda) - L_v N_v \end{aligned} \tag{4}$$

L_a is the inter-arrival time for the first packet in the idle mode and N_{L_s} is the number of arrivals within period L_s.

At the end of this inter-arrival time, the queue length probability distribution can be written as P_{init} is $[0] \times K$ and $P_{\text{init}}[1] = 1$

Let the ratio $r = \frac{L_s}{L_b + L_s}$. Normalised energy per bit can be derived from $E_i = 1 - r + r * \frac{p_s}{p_a}$. $i <= N_v$ are the events where the server resumes to work within the maximum amount of sleep frames. In these cases $L_s = i * L_v$ and $L_b + L_s = L_b + i * L_v$. Then $i = N_v + 1$ is the event where the server has an idle stage between

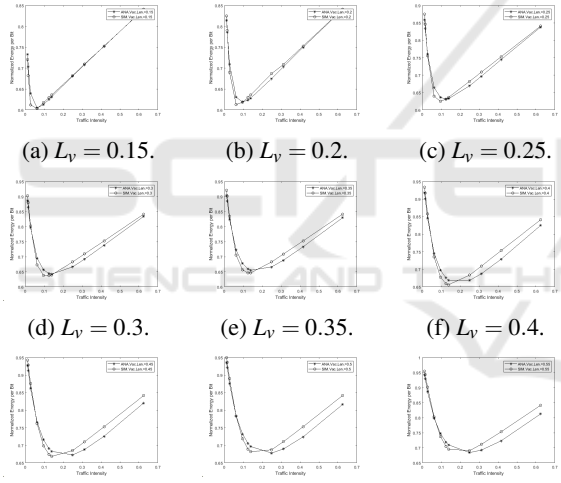
the sleeping sub-frame and the working sub-frame. In this case $L_s = N_v * L_v$ and $L_s + L_b = L'_b + ilen + N_v * L_v$. Lastly the event for $i > N_v + 1$ doesn't exist. Let NE be the acronym for normalised energy per bit,

$$E[NE] = \sum_{i}^{N_v} E_i * P_{L_s(i)} + E_{N_{(v+1)}} e^{(-\lambda N_v L_v)} \quad (5)$$

where the probability of first arrival within i^{th} vacation period $P_{L_s(i)} = e^{-\lambda(i-1)L_v} - e^{-\lambda i L_v}$.

3.2.1 Case Study

$\mu = 0.8$,$N_v = 4$,$L_v = \{a | a = \frac{1}{L_v} = 0.1 + 0.05 * i\, i \in [\,1,9\,]$and $i \in \mathbb{Z}\}$ From Figure 4, it can be noticed that the analytical plots based from the above procedure have a similar curve as the simulation plots and the numerical values are pretty close to each other.The average error rate is 0.0177 and deviation is 0.0102 over 99 data instances.



(a) $L_v = 0.15$. (b) $L_v = 0.2$. (c) $L_v = 0.25$.

(d) $L_v = 0.3$. (e) $L_v = 0.35$. (f) $L_v = 0.4$.

(g) $L_v = 0.45$. (h) $L_v = 0.5$. (i) $L_v = 0.55$.

Figure 4: Normalised Power Analysis Validation.

3.3 Waiting Time Analysis

Waiting time analysis borrows the basic idea from Little's Theorem.The analysis is performed as decomposing the long term waiting time average for the system into two event cases: A no arrival within the limited vacation time; B no less than 1 arrival within the limited vacation time. It is easy to conclude that $P(A) = e^{(-\lambda(N_v L_v))}$ and $P(B) = 1 - P(A)$.

Theorem 2. *For the event that no arrival within the limited vacation time, the system is working as a $M/M/1/K$ system without any policy.*[1]

[1]the theory has been similarly mentioned in literature already (B.T.Doshi, 1985). Here a more intuitive and alternative approach is presented.

Lemma 3. *The waiting time for the vacation queuing system in general is equivalent to the waiting time for packets in an averaged running cycle.*

Proof. By Little's theorem, in the long term, the overall packet-in-queue time summation divided by the number of overall in queue packets is the waiting time. This can be written as, $W = \lim_{t\to\infty} \frac{\gamma(t)}{\alpha(t)}$, where $\gamma(t)$ is the packet time summation up to time instance t and $\alpha(t)$ is the in-queue packet number summation up to time instance t. The overall system time consists of an infinite number of running cycles. Suppose for an averaged running cycle, the overall packet-in queue time summation is $\triangle\gamma_k$ and the overall in queue packet number is $\triangle\alpha_k$.

$$\begin{aligned} W &= \lim_{n\to\infty} \frac{\sum_{i=1}^{n} \triangle\gamma_i}{\sum_{i=1}^{n} \triangle\alpha_i} \\ &= \lim_{n\to\infty} \frac{n\triangle\gamma_i}{n\triangle\alpha_i} \\ &= \frac{\triangle\gamma_k}{\triangle\alpha_k} \end{aligned}$$

□ □

Lemma 4. *In a no policy $M/M/1/K$ system, the timer of arrival process and departure process are synchronised. The Markov transition diagram can be drawn time-invariably and subsequently the classical equilibrium probability formula can be derived. The scenario where no arrival within vacation time falls into the category because the arrival process and departure process are synchronised.In this case, the Markov transition diagram starts when the server finishes vacation and embarks on idle period. Hence, the waiting time is $W = (\rho * (1 + K * \rho^{K+1} - (K+1) * \rho^K)/((1-\rho) * (1-\rho^{K+1}))) * \lambda^{-1}$ with $\rho = \frac{\lambda}{\mu}$ and K is the queue limit.*

For event B, following **Theorem 2**, $W = \frac{\triangle\gamma_B}{\triangle\alpha_B}$. Here α_B is the overall packet in queue number and is equivalent to the number of packets that have been departed during an averaged running cycle (as the running cycle only stops when all the packets in queue are out of the system). This can be written as, $\triangle\gamma_B = A_s + A_b$ where A_s is the packet in-queue time summation during an averaged vacation cycle and A_b is the packet in-queue time summation during an averaged busy cycle.

The conditional queue length when the system resumes to work is $L_Q^{init} = E(L_Q | N_a > 0) = \frac{\sum_{i=1}^{K} iPinit(i)}{\sum_{i=1}^{K} Pinit(i)}$, where N_a is the number of arrival.

Algorithm 1: Calculation of A_s.

procedure CALCAS(λ, L_Q^{init})
  $A_s = 0$, $i = 0$ and $res = L_Q^{init}$
  if $L_Q^{init} < 1$ **then**
    $A_s = \frac{1}{lambda} * res$
  else
    while $i \leq L_Q^{init}$ **do**
      if $res < 1$ **then**
        $A_s = A_s + i * \frac{1}{\lambda} * res$
      else
        $A_s = A_s + i * \frac{1}{\lambda}$
      $i = i + 1$
      $res = L_Q^{init} - i$

The conditional queue length when the system is active at departure epoch k is $P_{ql_k} = \frac{\sum_i P^k_{\text{other}}(i) i}{\sum_i P^k_{\text{other}}(i)}$ when $k = 0$, $Pqlen = L_Q^{init}$,

$$A_b = \sum_{i=0}^{K \to \infty} P_0(i) A_b^i \tag{6}$$

A_b^i is the packet in-queue time summation when the queue becomes empty at the ith departure epoch for an averaged busy cycle.

$$A_b^i = \sum_{k=0}^{i} 0.5 * ((P_{ql_k} + P_{ql_k} + \frac{\lambda}{\mu})/\mu) \tag{7}$$

3.3.1 Case Study

The parameters are set in accordance with Section B - Case Study. The analytical plots in figure 5 have some discernible discrepancies from the simulation results, esp. for $L_v = 16.66667$ when high chance of multiple arrivals within the first single sleep vacation exists. It is not an ideally targeted situation for this bounded multi-vacation policy. The average error rate $\frac{|VAL_{ana} - VAL_{sim}|}{VAL_{sim}}$ over all the 99 instances is 0.0655 and standard deviation is 0.0483.

3.4 Optimisation Scheme

The goal of the optimisation is to select an ideal (L_v, N_v) pair from a feasible pool for a given input traffic rate λ, given a fixed service rate μ.

3.4.1 Case Study

With a specific pool of (L_v, N_v), $L_v = [\,0.2\ 0.5\ 0.8\ 1.1\ 1.6\ 2.1\ 3\ 4\ 6\,]$ and $N_v = [\,1\ 2\ 3\ 4\ 5\ 6\,]$, the analytical results are plotted as below in Figure 6. $\mu = 0.8$ and $\lambda = 0.3$.

Figure 5: Waiting Time Analysis Validation.

Algorithm 2: Search for Optimal Vacation Period and Vacation Maximum Number.

procedure OPTSEARCH($\lambda, \mu, Dconst$) ▷ $Dconst$ is the waiting time bound
  Pool_L_v,Pool_vnum Initialisation

  $minP = 1$,$optL_v = 0$ and $optVnum = 0$
  while $Pool_{vnum}$ not exhausted **do**
    $vnumIndex = vnumIndex + 1$
    while $Pool_{L_v}$ not exhausted **do**
      $L_vIndex = L_vIndex + 1$
      $E = Power\_Analysis\_Function$
      $W = Waiting_Time\_Analysis\_Function$
      if $W < Dconst$ **then**
        if $E < minP$ **then**
          $minP = E$
          optVnum= $Pool_{vnum}[vnumIndex]$
          optL_v= $Pool_{L_v}[L_vIndex]$

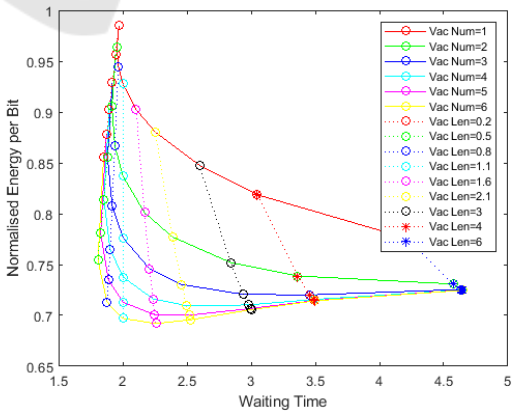


Figure 6: Energy-Delay vs (L_v, N_v).

The plot is similar to Figure 3 in (X.Guo et al., 2013) when the vacation number is 1 and the vacation length is a constant. As in paper (X.Guo et al.,

2013), the traffic rate and service rate have not been mentioned for generating Figure 3, the comparison stops where the plots have similar curves but not exact values. With the increase of the vacation length, the normalised energy per bit decreases while the waiting time increases. And the same rule applies to the change of the vacation number. Suppose the expected maximum delay is set to 2, the derived optimal solution is $(0.8\ , 6)$.

To validate the results, the simulation results are collected and a brute force method used to locate the ground truth and it can be seen in the figure 7 below that the derived minimum is 2 steps away from the ground truth $(0.8\ , 3)$.The derived solution has a relative error rate of $[\ 0.0299, 0.022\ ]$ from the ground truth value in this particular case study.

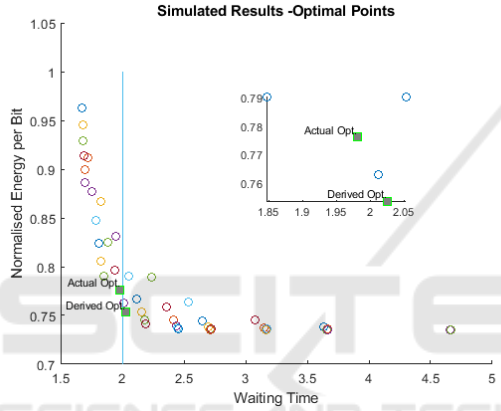


Figure 7: Effectiveness of the Derived Solution.

4 CONCLUSIONS

This work first discusses the advantage of the newly devised multi-stage sleep mode control for telecommunication networks and then presents a validated analytical model for it regarding energy efficiency and system delay. Lastly, the method as being integrated into a general algorithm design, is guaranteed to produce a solution that is deviated from the ground truth by minute discernible error.

5 FUTURE WORKS

Future works will investigate further into the delay modelling determining whether the discrepancy matters in practical engineering settings. The author will look into end-to-end delay bounds as specified in next generation telecommunication networks.

REFERENCES

B.T.Doshi (1985). A note in stochastic decomposition in a GI/G/1 queue with vacations or set-up times. *Journal of Applied Probability*, pages 419–428.

B.T.Doshi (1986). Queueing systems with vacations — a survey. *Queueing Systems*, pages 29–66.

J.Chen. A bounded multi-vacation queue model for multi-stage sleep control. *https://shorturl.at/EGOTU*.

J.Chen, B.Sikdar, and M.Hamdi (2018). An adaptive n-policy queueing system design for energy efficient and delay sensitive sensor networks. *Proceedings of the 2018 IEEE Global Communications Conference (GB2018)*.

J.Liu, S.Jin, and W.Yue (2019). Performance evaluation and system optimization of green cognitive radio networks with a multiple-sleep mode. *Annals of Operations Research*, pages 371–391.

J.Pender and T.Phung-Duc (2016). A law of large numbers for M/M/C/delayoff-setup queues with nonstationary arrivals. *International Conference on Analytical and Stochastic Modelling Techniques and Applications*.

J.Wu, E.W.M.Wong, Y.Chan, and M.Zukerman (2020). Power consumption and gos tradeoff in cellular mobile networks with base station sleeping and related performance studies. *Transactions on Green Communications and Networking*.

M.Feng, S.Mao, and T.Jiang (2017). Base station on-off switching in 5G wireless networks: Approaches and challenges. *IEEE Wireless Communications*.

P.Badian-Pessot, D.G.Down, and M.E.Lewis (2016). Optimal control policies for an M/M/1 queue with a removable server and dynamic service rates. *International Conference on Analytical and Stochastic Modelling Techniques and Applications*.

Q.Wu, X.Chen, Z.Zhou, L.Chen, and J.Zhang. Deep reinforcement learning with spatio-temporal traffic forecasting for data-driven base station sleep control. *IEEE/ACM Transactions on Networking*.

T.Phung-Duc (2020). Batch arrival multiserver queue with state-dependent setup for energy-saving data center. *Applied Probability and Stochastic Processes*, pages 421–440.

X.Guo, S.Zhou, Z.Niu, and P.R.Kumar (2013). Optimal wake-up mechanism for single base station with sleep mode. *Proceedings of the 2013 25th International Teletraffic Congress (ITC)*.

Yazici.M.A and T.Phung-Duc (2020). M/M/1 vacation queue with multiple thresholds : A fluid analysis. *International Conference on Quantitative Evaluation of Systems*.