Towards More Effective Spectrum Use Based on Memory Allocation Models

John Meier
Christopher Gill
Roger D. Chamberlain

John Meier, Christopher Gill, and Roger D. Chamberlain, "Towards More Effective Spectrum Use Based on Memory Allocation Models," in *Proc. of 35th IEEE Annual Computer Software and Applications Conference* (COMPSAC), July 2011.

Dept. of Computer Science and Engineering
Washington University in St. Louis

2011 35th IEEE Annual Computer Software and Applications Conference

Towards More Effective Spectrum Use Based on Memory Allocation Models

John Meier, Christopher Gill, and Roger D. Chamberlain
Department of Computer Science and Engineering, Washington University, St. Louis, Missouri

***Abstract*—Modern embedded systems are increasingly likely to be distributed across multiple devices and platforms that must interact with high precision across wireless networks. Traditional ways of managing the wireless radio spectrum suffer from two fundamental limitations, which the research presented in this paper addresses: (1) spectrum is divided *a priori* into static coarse-grained partitions without reference to details of particular applications; and (2) partitions are non-overlapping, which although beneficial to reduce interference prevents a much greater utilization of the spectrum through carefully allowing overlap of spectrum allocations. To overcome these limitations, we propose an approach to spectrum allocation based on dynamic allocation of diverse portions of the overall spectrum and overlapping allocations to increase utilization. This paper makes three main contributions to the state of the art in spectrum management for embedded systems: (1) it examines how memory management techniques such as Knuth's buddy algorithm can be applied to spectrum management, in the face of transmission failures that may arise from the physical environment; (2) it extends that approach to consider transmission failures resulting from interference, when overlapping regions of spectrum are allocated to increase utilization; and (3) it presents results of simulation experiments we conducted to evaluate those approaches, which demonstrate their efficacy and suggest future extensions based on them.**

Keywords—wireless radio spectrum allocation; modeling and simulation; buddy algorithm; distributed embedded systems.

I. INTRODUCTION

In a variety of emerging application domains, ranging from next-generation medical devices systems for urgent care [1] to cognitive radios for coordinated multi-agency disaster response [2], dynamic and fine-grained management of wireless radio spectrum use among embedded devices and platforms is increasingly critical. However, allocating the wireless radio spectrum is not efficient today because the authorities responsible for such allocation tend to divide the available radio frequency (RF) range coarsely into static non-overlapping blocks, which can accommodate only a limited number of simultaneous users. Traditional spectrum management involves (1) fixing RF partitions and the allocation of blocks within the partitions for particular categories of use (e.g., broadcasters, military personnel, or mobile device users) and then (2) performing (primarily static) assignment of specific frequency ranges within allocated blocks to different licensees.

For example, in the United States of America, radio spectrum use is regulated by the Federal Communications Commission (FCC). FCC rules regulate the use of available spectrum by competing applications, through multiple physical layer standards such as 802.11 [3] and 802.15.4 [4]. Even within a standard, technologies such as ultra-wideband radios operating in the 7.5 GHz range, or cognitive radios operating in the industrial, scientific and medical (ISM) band, may support multiple applications concurrently over multiple frequency ranges, thus competing dynamically for the spectrum range specified by the relevant standard.

Ample spectrum exists if secondary users are allowed to employ spectrum in a priority scheme and pool the existing spectrum [5]. Currently, however, the majority of techniques use fixed allocation that results in suboptimal spectrum use. For example, coordinated frequency planning is used today within conventional cellular networks but is not likely to be used in next generation multi-tiered cellular networks due to the lack of a rigorous effective approach. Effective spectrum reuse thus requires the development of new models to allocate spectrum dynamically and with fine granularity for multiple applications concurrently.

Efforts have been made to use the RF spectrum more efficiently, including the idea of *full radio spectrum dominance* in the 802 (radio frequency protocol) spectrum coexistence study, where scalability is envisioned as the ability of multiple systems using different sets of rules, policies or protocols to perform separate tasks [6]. However, coordinated frequency planning across multiple distributed *communication endpoints* (e.g., wireless radios used by different devices or platforms) is difficult and inefficient atop traditional spectrum allocation approaches.

In [7] a two-tier network attempted to use compatible modulation methods to improve area spectral efficiency (ASE). A decentralized spectrum allocation policy for ASE is essential to distributed sensor coordination [6] but requires dynamic, multi-tiered spectrum allocation to address cross-tier interference. For example, the capacity-limiting factor in shared tiered cellular networks is cross-tier interference that can severely reduce capacity when either centralized or coordinated frequency planning is employed atop traditional static allocation approaches. Furthermore, the inefficiency of centralized frequency allocation makes it infeasible for two-tier networks. Centralized spectrum use requires coordination of distributed resources, which may cause increased bandwidth overhead and system latency. This increased overhead may result in poor real time performance and scalability limitations. The ever growing number of distributed wireless communication systems (e.g., cell phones and GPS navigation) requires spectrum solutions that can scale while achieving real time performance. For

0730-3157/11 $26.00 © 2011 IEEE
DOI 10.1109/COMPSAC.2011.63

IEEE computer society

example, cell phone systems use a dedicated set of channels when requesting a channel for spectrum allocation.

This paper proposes and analyzes a new allocation approach for (potentially multi-tiered) wireless networks, by evolving techniques originally developed for memory management to reflect spectrum-specific details. While the uses of memory and spectrum often differ (e.g., memory is used to *store* persistent data over time, while spectrum is used to *transmit* data transiently) memory and spectrum have sufficient commonalities (e.g., both memory addresses and spectrum frequencies *index* their available ranges) that common models, algorithms, and analyses can be applied or adapted to account for how issues such as fragmentation, segmentation, overlaps, and transmission play out in wireless spectrum management. We explore differences between memory and spectrum using a modified allocation model in which multiple levels of overlapping allocation may be allowed and transmission failures due to environmental factors or collisions resulting from those overlaps can be considered. We consider how these issues can be modeled and evaluated, towards our long-term goal of developing new spectrum allocation strategies that can exploit these nuances effectively as future work. The major contribution of this work is the creation of a model enabling evaluation of diverse methods for efficient decision making using spectrum reuse policies for the scalable set of distributed resources. Current results show deviation from simulation predictions as compared with the theoretical queuing model at higher arrival rates, in the saturation region. As future work we plan to refine the model using queuing theory and Markov decision processes to improve spectrum use.

Among the many differences between memory and spectrum, the most significant include probable transmission failures due to the environment or to collisions. This paper explores how these essential differences, along with key features that are common to both memory and spectrum, can be modeled and evaluated, establishing a basic foundation towards the development of new allocation strategies that can exploit spectrum features, particularly those that distinguish spectrum from memory.

In this paper we explore two hypotheses related to the differences between memory and spectrum: (1) that *environmental probability of failure*, P_{env}, will induce a message throughput that is less than but directly related to the available spectrum and the offered load; (2) that multiple *levels* of overlapping allocation to the same frequency range, combined with a corresponding *probability of conflict*, P_{conf}, between messages that *collide* (i.e., their allocated spectrum ranges *overlap* at any time), will induce a similar but distinct relationship between message throughput, spectrum, offered load, and the number of degrees of overlap that are allowed.

II. BACKGROUND

In Section II.A we describe approaches that historically have been used for memory management. This includes a discussion of the performance impact of different offered loads relative to a constrained resource. While these considerations are common to spectrum and memory allocation, two additional issues differentiate these media. In Section II.B we examine how environmental factors may affect message transmission, and identify the ability to overlap allocations for different messages and the potential resulting consequence of failures due to conflicts.

A. Memory management approaches

Existing memory management approaches provide rigorous examples for how to design and evaluate spectrum management techniques. Considering properties of memory allocation methods provides a baseline for evaluating and comparing memory and spectrum management. A variety of memory allocation strategies have been developed in prior work. Each strategy offers different benefits and potential drawbacks. For example, the *first fit* algorithm [8] is straightforward to implement but its performance may suffer due to external fragmentation. In contrast, the *buddy* algorithm [8] avoids external fragmentation but is more complex to implement and may suffer its own potential inefficiencies due to internal fragmentation. A key strength of the buddy algorithm is the efficient use of memory through its binary method of allocation and reclamation. Each allocation step splits a block into two new (binary) sized blocks which allows efficient reclamation by combining adjacent blocks. In this paper, we focus on the buddy allocation algorithm as a useful exemplar of the broader set of possible allocation strategies. Moreover, the modeling and analysis techniques presented here may be adapted for use with other allocation strategies.

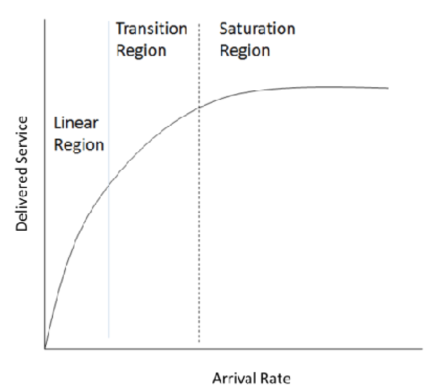

Figure 1. Effects of contention on delivered service for any bounded resource (memory or spectrum).

We are interested in how an allocation strategy responds to offered loads. Our trials use loads high enough to cause resource contention, depicted by the transition and saturation regions in Figure 1. Figure 1 qualitatively illustrates a typical response curve for how delivered service varies under an increasing request arrival rate (i.e., an increasing load). One possible implementation is multiple transmitters coordinating the allocation of spectrum using the distributed algorithm described. We assume a centralized coordination scheme, such as that used for cellular systems. The three regions of the response curve indicate the effects of contention for the limited resource. In the *linear region* of operation there is ample space to allocate the resource, we expect little or no contention, and the increase in the delivered service is roughly proportional to the increase in the request arrival rate. In the *transition region*, contention in allocating the resource increases so that some newly arriving requests might not find sufficient resource available. In the transition region, the delivered service grows more slowly as a function of the request arrival rate. The *saturation region* is where consistent contention severely limits further increases in delivered service which is thus near the capacity of the resource, and newly arriving requests only are serviced if a previous request has completed recently to make room – otherwise they must be dropped (which we assume in Section III) or queued for later service, depending on the allocation strategy being used.

B. Failure due to environmental factors or conflicts

While successful allocation of an unoccupied memory address range ensures storage of a message, with spectrum transmission may fail even if allocation is successful. The most basic kind of transmission failure is due to features of the physical environment such as obstacles, reflections, or background RF noise. We consider these features in the aggregate, and represent the probability of an environmental transmission failure with a single parameter, P_{env} (as in [9]).

Today unlicensed spectrum is used without de-confliction, resulting in potential accidental *conflicts* (transmission failures resulting from a message *colliding* with other messages whose allocated frequency ranges overlap with its own range). Towards the design of allocation strategies that can mitigate the effects of contention and saturation described in Section II.A, particularly as the request arrival rate increases relative to the spectrum capacity, we also consider the possibility of *intentionally allowing and recording* the allocation of different messages to overlapping ranges of spectrum.

As with environmental failures, we also represent the probability of a conflict with a single parameter, P_{conf}. Since the message throughput is then a function of both the conflict probability and the number of overlaps, we also restrict the allowed overlap in our experiments (described in Section IV) so that we can quantify the spectrum capacity, which when approached again results in contention and saturation.

III. MODELING APPROACH

To evaluate the semantics and performance of different memory and spectrum allocation models, we have built a simulator written in C that can be configured with different parameter settings (including those common to memory and spectrum as well as those unique to spectrum) to model different kinds of media and to explore and record their behaviors by executing simulation runs under a variety of operating conditions and allocation strategies. Figure 2 illustrates a possible configuration of the simulator, in a possible state of the simulation at a particular point in a run.

Each rectangle in Figure 2 corresponds to a fixed sized unit of allocation of the medium (which we call a *channel*). Each message is allocated a fixed number of contiguous channels, and occupies those channels until a given time relative to the time of allocation (called *EOL* for message end-of-life). The messages shown in Figure 2 have been allocated to adjacent channel ranges, leaving a single unallocated gap in the region shown. Different models or allocation strategies would naturally give other states: for example in a buddy allocation scheme for memory messages would be aligned on boundaries that are binary exponentials, while in a spectrum allocation scheme that allows overlaps, messages could be allocated atop regions already occupied by other messages.

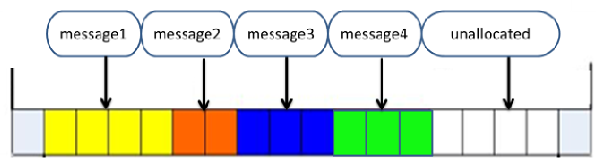

Figure 2. Example simulator state during a run.

During each run, the simulator generates different arrival and departure patterns (according to how specific message arrival rate and message lifetime parameters are configured), and records data about the evolving state of the messages and the medium as the simulation progresses. For the experiments discussed in this paper, we assume that if a message cannot be allocated (i.e., if there is no open region of a sufficient size in the medium when the allocation request arrives) it is simply dropped. For spectrum allocation, the simulator also is configured to determine whether or not a successful allocation resulted in a successful transmission.

Our simulator employs a pseudo-random number generator to produce allocation requests for messages of varying sizes (i.e., the number of channels needed), which is distributed uniformly between 1 and the maximum message size m_{MAX}. Each allocation request specifies the message size, and a particular allocation strategy determines the number of channels to allocate to the message (and if allocation is possible which channels will be used). Message arrivals are sampled from a Poisson process with mean rate λ messages per millisecond.

The simulator generates a *message arrival event* for each allocation request, runs the allocation strategy, and if the allocation is successful records which channels are occupied by the message. Each arrival event contains a *message number* that uniquely identifies each allocated message. For successfully allocated messages, a pseudo-random number generator is used to determine the *EOL* for the message (uniformly distributed between 1 and an upper bound of EOL_{MAX}). The simulator tracks the passage of time since each message's allocation event, and whenever the *EOL* for an allocated message is reached it generates a *message end-of-life event* and records the release of the channels that the message previously had occupied. As was noted in Section II.A., the policy for all of the allocation strategies presented in this paper was to drop any message request for which allocation failed. However, as future work we plan to evaluate strategies that allow such requests to remain in an allocation request queue for possible eventual allocation.

A. Modeling Fragmentation

Internal fragmentation is caused when more of the resource than is necessary to satisfy a request is allocated to it, thereby wasting the remainder. External fragmentation occurs as allocations used to satisfy requests are freed, but other allocations persist, leaving regions of unallocated resource that may not be sufficiently large enough to satisfy a subsequent request.

One kind of data collected by the simulator that is especially useful, both for validating our simulator's representation of fragmentation behavior and for evaluating the performance of different allocation strategies in the face of fragmentation, is the distribution of the different sizes of regions of the medium that are contiguously occupied (*fills*) vs. contiguously empty (*holes*). From those data we can extract fragmentation maps, such as the one shown in Figure 3 under a buddy allocation scheme, that capture the state of a given simulation at a particular point in a run, and help to explain its overall performance. The fragmentation map indicates the occupancy of the buddy levels associated with the spectrum allocation for seven different arrival rates. The available spectrum is represented in a series of binary hole sizes for each of the rates. Figure 3 presents a snap shot of the available memory after 10000 iterations. These maps allow side-by-side comparison of hole sizes and the number of holes of each size for different request arrival rates.

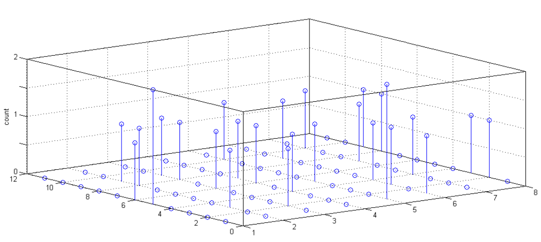

Figure 3. Fragmentation map.

We validated our simulator by confirming that each set of arrival patterns produced the same maps, even with differing environmental and conflict transmission failure probabilities.

B. Modeling memory allocation

As was reported in [8], Knuth conducted experiments involving the expected occupancy, message size and spectrum size as independent variables. To validate our simulator, and a basic allocation model for features common to memory and spectrum, we configured simulation experiments according to Knuth's experiment set [8]. In these validation experiments the spectrum size was set to 8192 channels, selecting message size between 1 to 1,000 and the occupancy deadline (*EOL*) was randomly selected from 1 to 1,000. These experimental trials varied the message allocation request arrival rate (λ) by powers of two ranging from 1 to 256 and confirmed that the point where the first failure to allocate (representing e.g., when a program ran out of memory) occurred at a point consistent with Knuth's analysis.

C. Modeling spectrum allocation

After reproducing Knuth's results, we then modified the simulation to continue allocating even after the first allocation failure, in order to evaluate the simulation's ability to represent a more spectrum-like use of the medium, even without transmission failures. Our experiments targeted the relationships between offered load, message size, available spectrum, and message throughput. For those experiments, as well as the experiments described in the rest of the paper, we continued to use Knuth's buddy allocation strategy. The message throughput is reported as the number of successfully allocated messages per unit of time.

We evaluated saturation effects across a range of request arrival rates, and confirmed consistent behavior spanning the operating regions shown in Figure 1. We also verified the model using queuing theory to create analytical expressions that predict the simulation results.

We then evaluated our hypothesis that a non-zero environmental probability of failure (P_{env}) would induce a message throughput that is less than but directly related to the available spectrum and the offered load. These experiments modified the definition of message throughput to include only the allocated messages that were *transmitted* successfully. To verify these experiments, we extended the queuing theoretic analysis to reflect the probability of environmental transmission failure.

Building upon basic allocation and environmental transmission failure, we then add message overlap to our model and consider the resulting probability of conflict (P_{conf}) to evaluate the possibility of transmission loss due to the overlaps. We again extend our queuing theoretic analysis to predict simulation performance with conflict failures.

The resulting spectrum channel allocation model (with overlaps) considers two types of effects: conflict failure and environmental transmission failure. Our simulation model uses two separate continuous random variables sampled uniformly from 0 up to but not including 1, to represent the probability of environmental or conflict failures. Each random variable is compared to its corresponding threshold which represents the probability of environmental failure (P_{env}) the probability of a conflict (P_{conf}). Our model evaluates the success or failure of a message *m* using the steps of the following pseudo-code algorithm.

```
sample random variable X ∈ U[0,1)
if X < P_env {
        message m fails
} else {
        sample random variable Y ∈ U[0,1)
        if m overlaps another message in any channel
        and Y  < P_conf {
                message m fails
        } else {
                message m succeeds
        }
}
```

The random variables in this approach are sampled separately to insure independence of events associated with environmental failures or conflict failures.

IV. Evaluation

This section describes how the simulation model presented in Section III was evaluated. We first articulate the experimental design, including a listing of the parameter settings used for empirical evaluation. Next, we present a queuing theoretic model that approximates the simulation model. Following that, the empirical results are presented and assessed.

A. Experimental design

For all of the experiments described in this paper, we fixed the number of channels, C, maximum message size, m_{MAX}, and maximum end-of-life for a message, EOL_{MAX}. Parameters that are varied include the number of *levels* (encoding the degree of spectrum overlap allowed), failure probabilities, P_{env} and P_{conf}, and the message arrival rate, λ. The ranges of values used in the simulations are listed in Table I.

TABLE I. MODEL PARAMETERS

Symbol	*Parameter*	*Value(s)*
C	number of channels	1024 channels
m_{MAX}	maximum message size	256 channels
EOL_{MAX}	maximum end of life	64 ms
levels	number of levels	{1, 3}
P_{env}	environmental failure probability	{0.0., 0.1, 0.2}
P_{conf}	conflict probability	{0.0., 0.2}
λ	mean msg. arrival rate	(0 to 1] msgs/ms

For all of the simulation results, the plotted mean delivered throughputs represent the mean value across five distinct simulation executions (i.e., 5 distinct pseudo-random number generator seeds). Error bars represent 99% confidence intervals based on the student's t-distribution.

B. Queuing theoretic approximate model

We use queuing theory to provide an approximate model of the resource (spectrum) usage. This helps us both validate the simulation models and better understand the underlying causes for the effects that are observed. Additional symbols used by the analytic model are shown in Table II.

TABLE II. SYMBOL DEFINITIONS

Symbol	*Description*	*Units*
r	delivered message throughput	msgs/ms
μ	mean message service rate	msgs/ms
c	number of servers	
$B(c, \lambda/\mu)$	Erlang's loss formula	

Using Kendall's notation, we model the spectrum allocation as an *M/U/c/c* queuing system (i.e., *M*arkovian (Poisson) arrival process with mean arrival rate λ msgs/ms; *U*niformly distributed service process with mean service rate μ msgs/ms; c servers; and c total jobs (msgs) allowed in the system). Given a uniformly distributed service time with a maximum of EOL_{MAX} ms,

$$\mu = \frac{2}{EOL_{MAX}+1} \text{ msgs/ms.}$$

Also given a uniformly distributed message size with a maximum of m_{MAX} channels,

$$c = levels \times \frac{1}{m_{MAX}} \sum_{i=1}^{m_{MAX}} 2^{\lceil \log_2 i \rceil}$$

where the ceiling function accounts for the internal fragmentation due to the buddy algorithm.

The actual spectrum use differs from the queuing model primarily in the fact that the queuing theoretic results make the assumption that the number of servers is fixed, whereas the simulation model accounts for the varying instantaneous number of concurrent messages that can be supported by the spectrum. This discrepancy will result in the queuing theory model not necessary precisely matching the simulation results, but we expect both throughput predictions to be reasonably close to one another. Where they diverge, the simulator is likely to be more accurate, since its model more closely matches reality.

Using the above queuing model, the delivered throughput in the absence of failures is predicted as

$$r = \lambda(1 - B(c, \lambda/\mu)),$$

where B is Erlang's loss formula [10]. The modeled capacity of the spectrum is μc. The degradation in throughput due to environmental causes is modeled as being linear, e.g.,

$$r = (1 - P_{env})\lambda(1 - B(c, \lambda/\mu)),$$

and the degradation in throughput due to conflicts is linear when the expected number of messages in the system,

$$L = (\lambda/\mu)(1 - B(c, \lambda/\mu)),$$

is greater than $c/levels$.

$$r = \begin{cases} (1 - P_{env})\lambda\left(1 - B\left(c, \dfrac{\lambda}{\mu}\right)\right), & \text{if } L < c/levels \\ (1 - P_{conf})(1 - P_{env})\lambda\left(1 - B\left(c, \dfrac{\lambda}{\mu}\right)\right), & \text{if } L > c/levels \end{cases}$$

C. Experimental results

All of the experimental result graphs contain the following elements: (1) each plot has fixed values for number of levels and failure probabilities, these quantities vary across plots; (2) the horizontal axis shows the mean message arrival rate, λ, in messages per ms; (3) the vertical axis shows the delivered throughput, *r*, in messages per ms; (4) the dotted line (labeled q.t. capacity) represents the constant-valued queuing theoretic capacity approximation, μc; (5) the dashed line (labeled q.t. tput) shows the queuing theoretic model of delivered throughput, *r*, in messages per ms; and (6) the points (labeled mean sim tput) show the simulation model predictions for delivered throughput.

Figure 4 shows the baseline performance predictions: 1 level and failure probabilities all 0. In this plot there is no distinction between memory allocation and spectrum allocation. As can be seen, there is reasonable alignment between the simulation model predictions and the approximate analytic model derived from queuing theory. At low arrival rates (i.e., $\lambda \lesssim 0.1$ msgs/ms) the delivered throughput is essentially equal to the arrival rate for both the simulation model and the analytic model. The transition region is entered at only slightly higher arrival rates, and while the alignment is not perfect, the simulation results and analytic approximations are reasonably close across the rest of the graph.

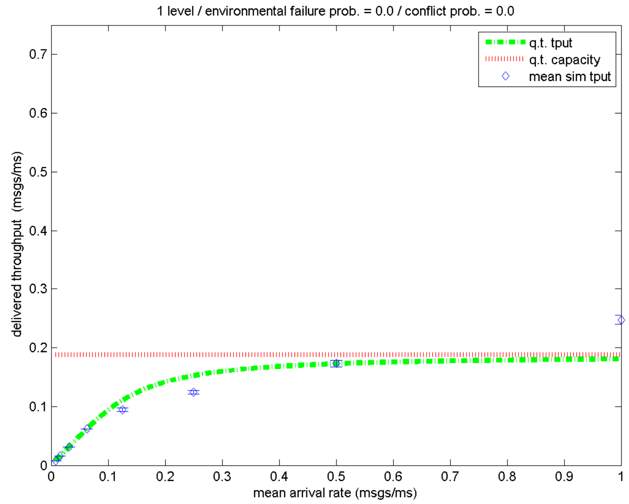

Figure 4. Baseline performance, no overlap, no failures.

This graph gives us assurance that the simulation model is reasonable (i.e., it tracks the general trends of the analytic approximation). We anticipate that the discrepancies that exist are due primarily to the fact that the analytic model makes the simplifying assumption that the number of queuing theoretic servers (*c* in the *M/U/c/c* queuing system) is constant, while the underlying truth is that the number of messages that can be simultaneously delivered by the spectrum depends upon the instantaneous message size. As is true for all of the simulation results presented in this paper, the error bars are very tight, indicating relatively small uncertainty in predictions due to statistical variability.

Figure 5 explores how this baseline performance is altered for spectrum when multiple simultaneous messages are allowed to overlap in each channel. Here, the number of levels of allocation has been increased to 3, yet the failure probabilities are still all 0. There are several features of this graph that reflect the trebling of effective capacity. First, the analytic model shows a three-fold increase in delivery rate at or near saturation. Second, both analytic and simulation models show the transition region extending approximately three times as wide as the previous case (i.e., to $\lambda \lesssim 0.3$ msgs/ms).

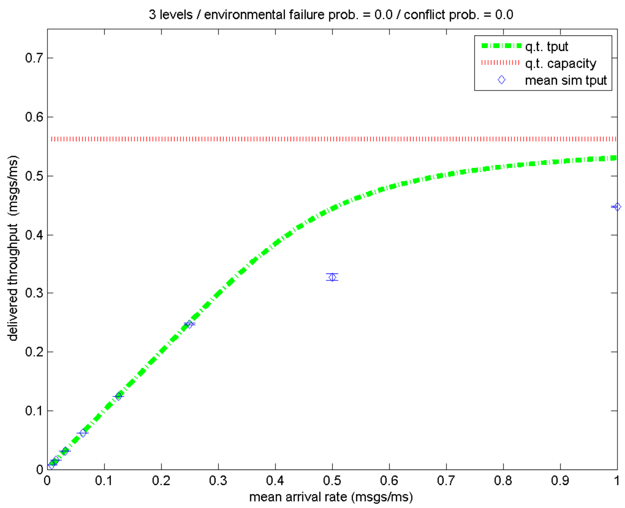

Figure 5. Overlap allowed (3 levels), no failures.

Figure 6 and Figure 7 return to the no overlap case (1 level) and explore the impact on delivered throughput of environmental failures. The plots for P_{env} = 0.1 and P_{env} = 0.2 can be compared to Figure 4 (with P_{env} = 0.0). Across the board, the analytic and simulation models show a linear decrease in throughput as the environmental failure probability increases. Note that the queuing theoretic capacity approximation does not change: only the predicted throughput is impacted.

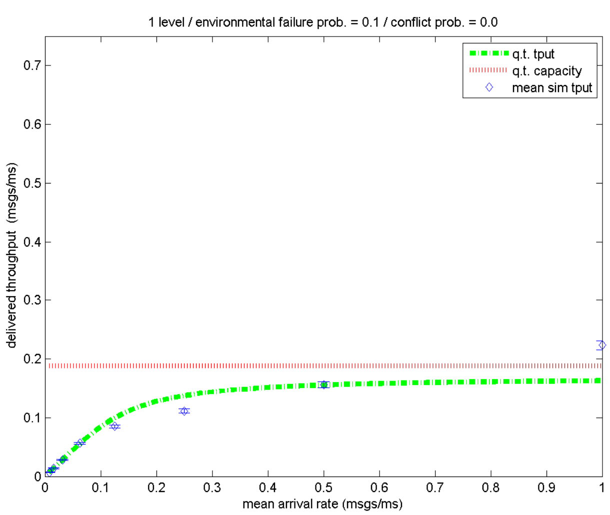

Figure 6. No overlap (1 level), P_{env} = 0.1.

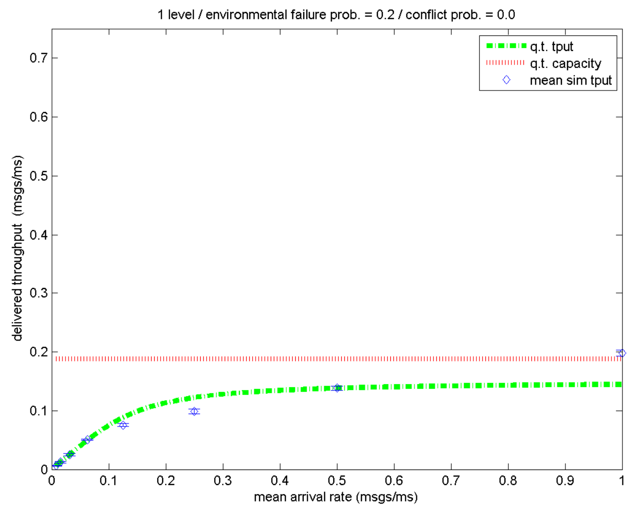

Figure 7. No overlap (1 level), P_{env} = 0.2.

This same effect is illustrated in the presence of overlap (3 levels) in Figure 8 and Figure 9 with the same results. When compared to the corresponding no overlap curves, delivered throughput generally increases by a factor equal to the number of levels of overlap allowed and the boundary between the linear and transition regions generally moves to the right, again by a factor roughly equal to the allowed overlap.

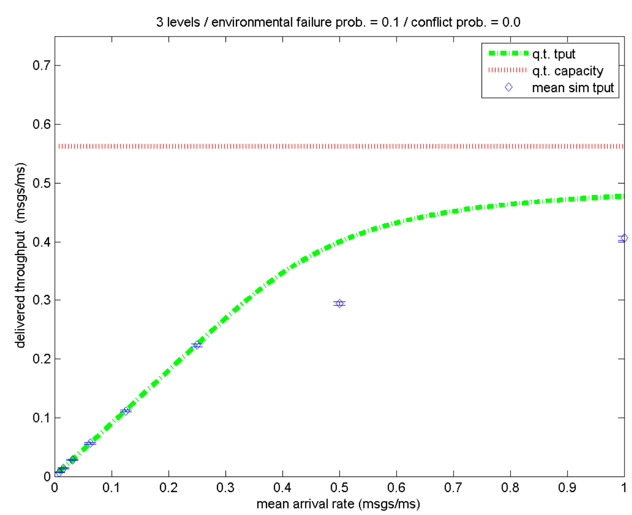

Figure 8. Overlap allowed (3 levels), P_{env} = 0.1.

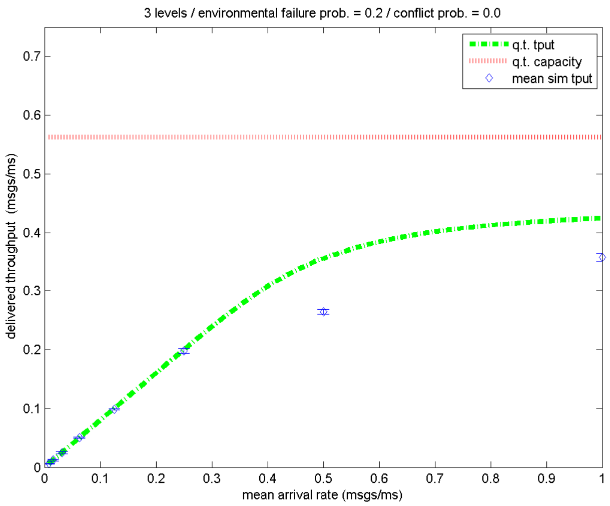

Figure 9. Overlap allowed (3 levels), P_{env} = 0.2.

Figure 10 through Figure 12 combine all of the effects that differentiate spectrum from memory: number of levels (allowing spectrum overlap), the existence of environmental failures, and conflict failures. As in the previous 4 graphs, the queuing theory approximated capacity increases with the allowed levels but is not impacted by the failure probabilities. In each of these graphs, the analytic expressions are distinct for the linear region and the combination of the transition and saturation regions, which explains the discontinuity in the analytic predictions for delivered throughput. As these are only approximate models, we leave it to future work to further refine these analytic expressions to remove this artifact; we will restrict our assessment to considering the two regions separately.

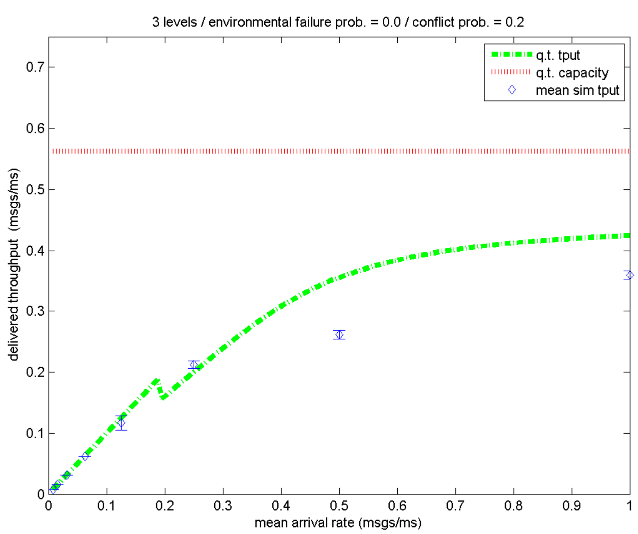

Figure 10. Overlap allowed (3 levels), P_{env} = 0.0, P_{conf} = 0.2.

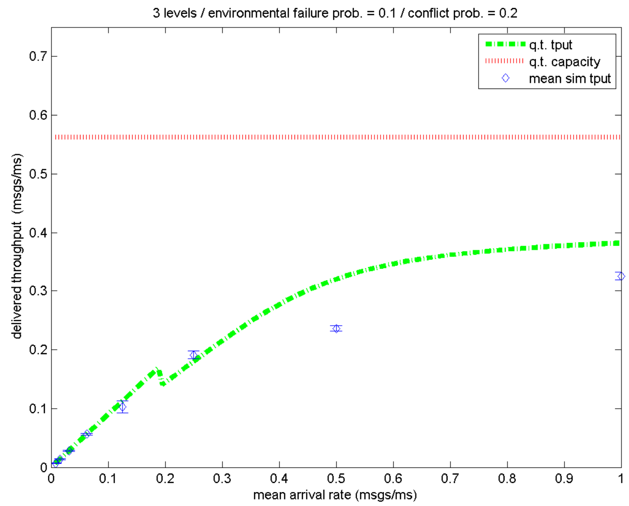

Figure 11. Overlap allowed (3 levels), P_{env} = 0.1, P_{conf} = 0.2.

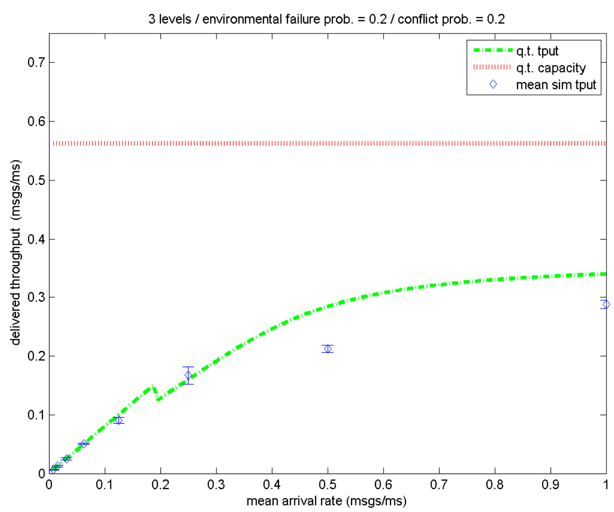

Figure 12. Overlap allowed (3 levels), P_{env} = 0.2, P_{conf} = 0.2.

First, we notice that once again the analytic model predictions and simulation model predictions agree quite well both within the linear region and in defining the boundary of the linear region (the latter being the more interesting prediction). This correspondence between the queuing theoretic predictions and simulation persists across the entire range of experimental results. Second, the degradation in delivered throughput due to both environmental impacts and spectrum reuse (resulting in conflicts) is consistent between the analytic expressions and the simulation model. Third, this set of curves allows for the richest comparisons between memory allocation (with performance predictions shown in Figure 4) and spectrum allocation in the presence of transmission failures as we discuss further in Section V.

V. CONCLUSIONS

Traditional ways of managing the wireless radio spectrum in distributed embedded systems suffer from inefficiency and inflexibility which potentially can be overcome through new approaches to spectrum management. In this paper we have conducted preliminary investigations into the features, semantics, and performance of wireless spectrum allocation based on extending techniques and models developed initially for memory management. We first summarize the salient observations and lessons from these investigations and then outline plans for future research.

A. Observations and lessons learned

Techniques that were developed originally for memory allocation and management serve as excellent starting points for a broader investigation of other media. By demonstrating that Knuth's buddy algorithm can be applied effectively to RF spectrum management, we establish a baseline for extending the buddy algorithm to include probabilistic transmission due to environmental failures and allowed overlap, leading to both increased capacity and the need to consider the associated probability of conflict failures.

Our performance evaluations span both a discrete-event simulation model and a queuing theoretic model, which are generally in agreement. Features of the performance curves generated from these models diverge in ways that suggest specific refinements to both models. For example, the conflict probability in our simulation model does not yet vary with an increasing number of conflicts so that the performance consequences of allowing overlaps in our studies may be somewhat optimistic. Similarly, the queuing theoretic representation of message throughput, while able to identify the boundary between the linear and transition regions of the response curves, needs to better reflect the semantics of the system as saturation increases.

The experimental evaluation yielded several concrete results. First, the message throughput under low load increases linearly with the arrival rate, as expected. Second, as the load increases the message throughput follows a canonical response curve, bending over as saturation increases. Third, with the introduction of environmental failures, the message throughput decreases linearly with P_{env}. Fourth, with overlapping allocation and corresponding collisions, the message throughput decreases as a more complex function of the multi-level fragmentation and P_{conf}.

From these observations, we draw the lesson that allowing overlaps provides access to more of the available spectrum capacity. However, it is then necessary to consider conflict failures, which incur a real cost to message throughput. While our experiments demonstrate that a suitable balance can be achieved between the additional opportunities afforded by overlap and the additional risks associated with collisions, a more sophisticated treatment of the benefits and risks is needed under various realistic operating conditions.

B. Directions for future work

The results presented in this paper establish a solid foundation for the development of new spectrum allocation strategies. These strategies will focus on how the features that are unique to spectrum can be exploited beyond those that are common to both spectrum and memory. Building on this foundation, new spectrum allocation strategies using both analytic and simulation analysis techniques methods will be developed. For example, finding optimal levels of overlap requires balancing increases in throughput capacity with possible message conflicts. Refinement of these models will incorporate a probability of conflict that increases proportionally to the amount overlap.

Analytic models also will be evolved to achieve a more complete and predictive representation based on the simulation results, including varying the probability of conflict with the degree of overlap. We will use fragmentation maps (like the one illustrated by Figure 3) to predict response behavior, including visual representations that will guide our experiments. Next, the physical implementation and performance of these strategies will be compared to existing technology standards (e.g., 802.11 and 802.15.4). Communication model refinement will provide realistic test conditions for commonly used ranges of spectrum in the ISM band regions around 2.4 GHz or 5.2 GHz.

We plan to extend the spectrum allocation models presented in this paper with a state space that also encodes the fragmentation map information. We will first describe a queue-less case with channels acting as servers, and then obtain a Markov decision process with which to generate policies that can optimize expected throughput. Towards that end, we will create probabilistic models for transmission speed and capacity.

We plan to use these models to quantify the relationships between the environmental probability of failure and message throughput, and between the number of levels of overlapping allocation and the corresponding probability of conflict. In light of these relationships we also will explore the trade-offs between each given message using fewer channels for a longer time versus more channels for a shorter time, and identify points of optimality.

Our longer term objectives include physical evaluation of the strategies we develop and comparison to existing technology standards such as 802.11 and 802.15.4, under realistic conditions in commonly used ranges of spectrum such as the 2.4 GHz or 5.2 GHz ISM bands.

ACKNOWLEDGEMENTS

This research was supported in part by NSF grants CNS-0931693 (CPS) and CCF-0448562 (CAREER). We also wish to thank the anonymous reviewers for their suggestions for this paper and our potential future work.

REFERENCES

[1] D. Arney, M. Pajic, J. Goldman, I. Lee, R. Mangharam, and O. Sokolsky, "Toward Patient Safety in Closed-Loop Medical Device Systems," 1st International Conference on Cyber-Physical Systems, April 2010.

[2] M. Silvius, F. Ge, A. Young, A. MacKenzie, C. Bostian, "Smart radio: spectrum access for first responders", Wireless Sensing and Processing III (S. Dianat, M. Zoltowski, eds.). *Proceedings of the SPIE*, Volume 6980, 2008.

[3] IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std 802.11-2007, December 2007.

[4] IEEE Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks-Specific Requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), IEEE Std 802.15.4-2006, September 2006.

[5] D. Cabric, S. Mishra, D. Willkomm, R. Brodersen, and A. Wolisz, "A Cognitive Radio Approach for Usage of Virtual Unlicensed Spectrum," 14th IST Mobile and Wireless Communications Summit, June 2005.

[6] K. Bian and J.-M. Park, "A Coexistence-Aware Spectrum Sharing Protocol for 802.22 WRANs," 18th International Conference on Computer Communications and Networks, August 2009.

[7] V. Chandrasekhar and J. Andrews, "Spectrum allocation in tiered cellular networks," *IEEE Transactions on Communications*, vol. 57, no. 10, pp. 5314-5327, October 2009.

[8] D. Knuth, *Art of Computer Programming, Volume 1 - Fundamental Algorithms*, Addison-Wesley, 1973.

[9] I.-H. Hou, V. Borkar, and P.R. Kumar, "A Theory of QoS for Wireless," INFOCOM, April 2009, pp. 486-494.

[10] A. Allen, *Probability, Statistics, and Queuing Theory with Computer Science Applications,* Academic Press, 1978.