# CPS : Small : Dynamically Managing the Real-time Fabric of a Wireless Sensor-Actuator Network

## 1- Introduction

Wireless sensor-actuator networks (WSAN) consist of numerous sensing and actuation devices that share information over an ad hoc wireless communication network. WSANs can be used to manage networked systems that distribute goods and services over large spatially distributed domains. Examples of such systems include the national power grid, ground/air traffic networks, and water/gas distribution networks. These particular examples are components of our national civil infrastructure and the optimal management of such networks is in the national interest.

This project studies the implementation of feedback control algorithms over WSANs, particularly with regard to the management of large-scale networked systems such as the electric power grid or water distribution networks. Controlling such physical processes traditionally requires some type of hard real-time support. This means that each packet of feedback data must be serviced within a specified deadline to assure the overall control application's performance level. It has, in practice, been difficult to provide such guarantees in real-life wireless networks. This project will address that issue by developing algorithms that allow control applications and network servers to work together in maximizing application performance subject to hard real-time service constraints.

This goal will be achieved using a model-based approach that is well suited for modeling large-scale networks of computational (cyber) processes. The project models the interaction between a network's cyber-processes as a *real-time fabric* [55]. This fabric model relies on network calculus [52] concepts used in the modeling of real-time systems [114]. The network's real-time fabric allows us to pose the problem of managing application data flows in a manner that scales well with network size. The underlying problem is seen as a multi-player game between applications (users) that is essentially a network utility maximization (NUM) problem. This project's algorithms use pricing structures to generate a Pareto-efficient Nash equilibrium that maximizes overall application performance subject to constraints on the service rates supplied by nodes within the wireless network. The project will build software for WSANs based on these distributed control algorithms. The software will be developed for TinyOS type sensor networks and will use innovative event-triggered mechanisms for reducing the algorithm's communication overhead. This software will be developed and evaluated on two WSAN hardware tesbeds at the University of Notre Dame.

**Intellectual Impact:** This project provides a "transformative" cross-disciplinary approach to the co-design of networked controls systems that will have a major impact on both the control systems and real-time systems community. The components developed under this project will enable the use of WSANs for control of systems found in a variety of disciplines including civil, electrical, and mechanical systems engineering.

**Broader Impact:** The impact of this project will be broadened through interactions with its industrial partner, EmNet LLC. EmNet LLC is interested in using the software developed by this project on its CSOnet system. CSOnet is a WSAN that EmNet is building in a handful of U.S. cities to address environmental problems arising from combined-sewer overflows (CSO). This environmental problem is faced by nearly 800 cities in the United States, so that integration of this project's software into CSOnet will have impact of national scope. The project's impact will also be broadened through educational outreach activities. The PIs will develop a senior-level or first year graduate course on formal methods in cyber-physical systems. The methods and outcomes of this project will be integrated into that course and the curriculum will be released over the world wide web. The principal investigator is lecturing on networked control at a European Ph.D. summer school in summer of 2009. These types of international outreach activities will be continued under this project. Finally, the principal investigator will build upon earlier interactions with a local middle school to develop a presentation on multi-robotic systems that can dovetail with after-school activities on autonomous robotics.

**CPS Relevance:** This project addresses the CPS theme of **Foundations** through its use of a network's real-time fabric to holistically trade off the performance of a networked control system's physical process against the capacity limits inherent in the communication network's cyber processes. This project addresses the CPS theme of **Components, Run-time Substrates, and Systems** through its development of middleware services supporting real-time WSAN control. The outcomes resulting from this project address the **CPS program's vision** of providing scalable and reliable coordination of large-scale components of our national infrastructure. The project will work to develop a future CPS workforce through the development of curriculum that directly integrates this project’s outcomes. Curriculum components are planned for undergraduate/graduate level courses in *Cyber-Physical System Design: Methods and Practice* as well as participation in an international Ph.D. school and an after school program at a local middle school.

The project team is well positioned to achieve these objectives. Dr. Lemmon and Dr. Hu have worked together for over 6 years seeking to integrate the design of real-time scheduling algorithms with controller design. Dr. Lemmon has a long record in hybrid systems and networked control systems. Dr. Hu has a well-known record in embedded real-time systems. Dr. Lemmon worked as part of DARPA’s NEST project and later worked to bring embedded sensor network technologies to the monitoring and control of environmental applications [83, 102]. He assisted the start-up of a private-sector partner (EmNet LLC) capable of deploying the CPS system, CSOnet, on a national scale. His participation in CSOnet has led to collaborative ties with a European Union project [9] that is using distributed model predictive control to build a CPS water distribution network for the City of Barcelona, Spain.

## 2 - Real-time Fabric: a model for network flows

This project uses a model-based approach to managing real-time flows in wireless networks. The proposed modeling framework has its origin in a calculus for network delay [27, 28] that was later used to model Internet flows [52]. The network calculus also served as the basis for the modular analysis of integrated service networks [89, 88] and real-time systems [114]. A network’s real-time fabric is a network of dataflow systems that uses the min-plus algebra to capture the way in which data packets flow through a multi-hop network. This approach allows a characterization of message delays that is crucial for modeling hard real-time service constraints. Since this model may not be familiar to everyone, this section provides a tutorial introduction to what we call the network’s real-time fabric.

To describe the real-time fabric, let’s consider the networked system shown in figure 1. The figure’s left side shows a WSAN consisting of six nodes that are connected in a linear manner. The picture on the right side is this WSAN’s real-time fabric. The first three nodes in the WSAN are ”sensors” and the last three nodes are ”actuators”. In this example, a sensor-actuator pair is associated with the feedback control of a physical process. The control application streams sensor data from the sensor to the actuator over the network. In this example we assume the ”controller” for the physical process is located at the actuator. Figure 1 shows three such application (what we will call ”user”) streams. The user streams must satisfy a hard real-time quality-of-service (QoS) constraint in order to guarantee the control application’s closed-loop stability. The main problem being addressed in this proposal concerns the development of algorithms and middleware supporting the hard real-time streams generated by the control applications.

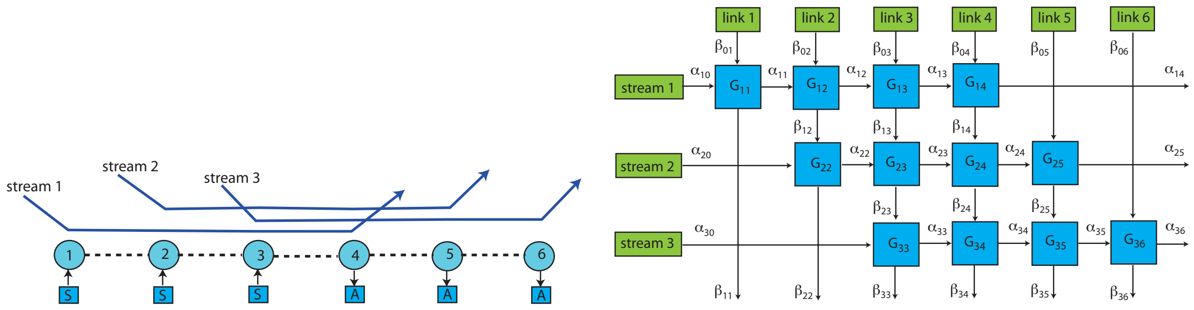


Figure 1: (left) Wireless Sensor Actuator Network; (right) Network’s Real-time Fabric

The user streams consist of sequences of ”messages”. Each message corresponds to a measurement that is used by the application’s controller to generate the actuation signal for the physical process. Each message may be broken up into a set of smaller packets that are transmitted across the network. In our proposed modeling scheme, we assume these messages are infinitely divisible so packets can be of arbitrarily small size. This is sometimes referred to as the *fluid* approximation. This approximation is an idealization of the actual nature of network flows. It provides, however, a convenient abstraction that allows us to more easily capture the underlying mechanisms behind network flow management.

The $i$th user stream is characterized by the function, $\alpha_{i0} : \Re \rightarrow \Re$, whose value $\alpha_{i0}(t)$ denotes the total number of *bits* that were transmitted by sensor $i$ over the time interval $[0, t]$. For this project description, we confine our attention to periodic user flows. The $i$th stream can be characterized by the following set of stream parameters, $\{Q_i, D_i, T_i, O_i\}$ where $Q_i$ is the number of bits transmitted per message (also called the message's quantization level), $D_i$ is the relative deadline (also called "delay") that these bits must be delivered by, $T_i$ is the time between consecutive messages (also called the stream's "period"), and $O_i$ is a time offset. The function $\alpha_{i0}$ is called the $i$th user (application) stream's *arrival curve*. While these parameters describe a periodic message stream, this project's methods can also be used for more general sporadic flows. It has the form

$$\alpha_{i0}(t) = \sum_{k=0}^{\infty} h_i(t - kT_i - O_i), \quad \text{where,} \quad h_i(t) = \begin{cases} 0 & t < 0 \\ \frac{Q_i}{D_i} t & 0 \leq t \leq D_i \\ Q_i & t \geq D_i \end{cases} \tag{1}$$

The function $h_i$ is the arrival rate of bits associated with a single message in the $i$th data stream. The solid black lines of figure 2 show an arrival curve with stream parameters $\{Q = 16, D = 2.5, T = 7.5, O = 2.5\}$ for $t \in [0, 30]$.

The messages in the user stream, $\alpha_{i0}$, are forwarded over the links in the multi-hop network shown on the left side of figure 1. Stream 2, for example, starts at sensor node 2 and forwards its messages through nodes 3 and 4 until it reaches actuator node 5. We therefore see each message as being forwarded across several links that form a route between the sensor and actuator node. A dynamical system, $G_{ij}$ models the fowarding of stream $i$'s bits across the $j$th link in the route. This system takes two types of input signals. The first input $\alpha_{ij}^{(\text{in})}(t)$ represents the total number of bits that arrive at $G_{ij}$ for service (i.e. forwarding over link $j$) in the time interval $[0, t]$. The second input $\beta_{ij}^{(\text{in})}(t)$ represents the maximum number of bits that $G_{ij}$ can forward over link $j$ in the time interval $[0, t]$. We refer to $\beta_{ij}^{(\text{in})}(t)$ as link $j$'s *supply curve* since it represents the total number of resources available to link $j$ that can be used in forwarding stream $i$'s bits. System $G_{ij}$ generates two types of outputs. The first output, $\alpha_{ij}(t)$, represents the total number of bits that were forwarded by link $j$ over $[0, t]$. The second output, $\beta_{ij}(t)$, represents the excess supply of the link; namely the number of available supply bits that were not used in forwarding stream $i$'s bits over link $j$.

System $G_{ij}$ may be viewed as a formal model for the cyber process controlling the forwarding of stream packets across a link in the network. In our work, we assume that each cyber process, $G_{ij}$, forwards a stream's bits at a constant rate. In particular, this means there are positive real constants $\overline{r}_{ij}$ and $\overline{d}_{ij}$ such that at most $\overline{\beta}_{ij}(t) = \max\{0, \overline{r}_{ij}(t - \overline{d}_{ij})\}$ bits of stream $i$ are forwarded over link $j$ in the interval $[0, t]$. The parameter $\overline{d}_{ij}$ represents a constant delay in processing the arriving bits and $\overline{r}_{ij}$ is that server's service bit rate. Note that for the purposes of this discussion, we are assuming that stream bits are transmitted at a constant bit rate across the network with a finite delay. With the definition of the $ij$th system's service rate, we can now formalize the relationship between $G_{ij}$'s inputs and outputs. In particular, we find that system $G_{ij}$ can be viewed as a formal model for the *cyber* process controlling the forwarding of the $i$th stream's messages over the $j$th link. The relationship between the inputs and outputs of $G_{ij}$ is given by the following relationship

$$\alpha_{ij}(t) = \overline{\beta}_{ij}(t) \otimes \alpha_{ij}^{(\text{in})}(t) \tag{2}$$

$$\beta_{ij}(t) = \max\left\{0, \beta_{ij}^{(\text{in})}(t) - \overline{\beta}_{ij}(t) \otimes \alpha_{ij}^{(\text{in})}(t)\right\} \tag{3}$$

where $\otimes$ denotes the min-plus algebra's convolution operator; namely

$$\alpha(t) \otimes \beta(t) = \inf_{0 \leq s \leq t} (\alpha(s) + \beta(t - s)) \tag{4}$$

The above equations show that the unused supply generated by $G_{ij}$ is the difference between its input supply $\beta_{ij}^{(\text{in})}(t)$ and what was actually forwarded over the link, $\alpha_{ij}(t)$.

The network's real-time fabric is formed by interconnecting the subsystems $G_{ij}$ that forward a stream's bits over the network's links. This interconnection is shown on the right side of figure 1. Consider stream $i$ and let $L_i = \{j_1, \cdots, j_M\}$ represent the links in stream $i$'s route of length $M$. The bits generated by the $i$th user stream are characterized by the arrival curve $\alpha_{i0}$. This function is the input to a cyber process, $G_{ij_1}$ that forwards bits over the first link of that stream's route. The arrival curve, $\alpha_{ij_1}$, generated by that process is the input to the next cyber-process on the route, $G_{ij_2}$. The arrival process $\alpha_{ij_2}$ generated by $G_{ij_2}$ then drives the next process, $G_{ij_3}$ on the route and so on until the destination is reached. We can therefore view each user stream as a *thread* that connects the sensor and actuator.

In addition to inputs from the sensor, each cyber-process receives a supply input. In particular, we assume that the $j$th link has an overall supply rate $\overline{r}_{0j}$ which represents that total bit rate available to all streams being forwarded by link $j$. Let $S_j = \{i_1, \cdots, i_N\}$ denote the set of streams serviced by link $j$. The overall supply curve $\overline{\beta}_{0j}(t) = \overline{r}_{0j}t$ is an input to the cyber-process $G_{i_1 j}$. This process forwards $\alpha_{i_1 j}$ bits and the remaining supply is given by the supply curve $\beta_{i_1 j}(t)$ which is used as input to the next cyber-process $G_{i_2 j}$ on the route. This cyber-process uses that supply to forward bits in stream $i_2$ and the remaining supply from $G_{i_2 j}$ is the supply curve $\beta_{i_2 j}$ that is used to drive $G_{i_3 j}$ and so on until all streams in $S_j$ have been served. We can therefore think of each link as generating a *supply thread* that traverses a set of cyber processes $\{G_{ij}\}$ in which $i \in S_j$.

In view of the above discussion, we can weave the threads of user stream bits and link supply bits into the fabric shown on the right side of figure 1. This figure arranges the source (sensor) nodes for a stream in a column on the left side of the picture. The supply nodes for the links are arrange in a row at the top of the figure. A user thread starts from each source node and extends to the right. A supply thread starts at each link and extends downward. A subsystem $G_{ij}$ is located at the intersection of these supply and user threads provided the $j$th link is actually on the stream's route.

From figure 2, one can easily characterize whether or not a stream's messages will be delivered within hard real-time deadlines. Stream 2, for example, is generated by the arrival curve $\alpha_{20}(t)$. The arrival curve arriving at the actuator is $\alpha_{25}(t)$. Due to the composable nature of the min-plus convolution operation and the fact that we're assuming constant bit rate supply functions, we can readily see that

$$\gamma_2(t) = \alpha_{25}(t) = \overline{\beta}_{25}(t) \otimes \overline{\beta}_{24}(t) \otimes \overline{\beta}_{23}(t) \otimes \overline{\beta}_{22}(t) \otimes \alpha_{20}(t) = \overline{\beta}_2^*(t) \otimes \alpha_{20}(t)$$

where $\overline{\beta}_2^*(t) = \overline{r}_2^*(t - \overline{d}_2^*)$. In the preceding equation $\overline{r}_2^* = \min_{1 \leq j \leq 6} \{\overline{r}_{2j}\}$ is the minimum bit rate seen by stream $i$ as it moves towards its destination. The other parameter $\overline{d}_2^* = \sum_j \overline{d}_{2j}$ represents the *end-to-end delay* seen by stream 2. We refer to $\overline{\beta}_i^*(t)$ as the overall supply curve seen by sensor stream $i$ and we refer to $\gamma_i(t)$ as the *throughput* function seen by stream $i$. The overall service seen by stream 2 in this example is shown on the left side of figure 2. The red line shows the overall supply curve $\overline{\beta}_2^*(t)$ seen by stream 2. The dashed line represents the min-plus convolution of $\alpha_{20}(t)$ with $\overline{\beta}_2^*(t)$ and the shaded blue area beneath this curve represents the total throughput, $\gamma_2(t)$, seen by stream 2.

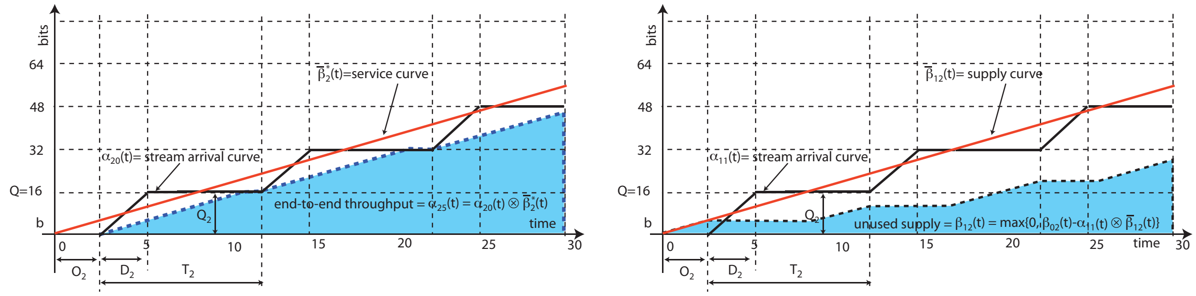


Figure 2: (left) throughput for stream 2 (right) unused supply for link 2

Whether or not the $i$th stream meets its real-time constraints can be determined by examining the stream's throughput function, $\gamma_i(t)$. The $i$th stream meets its real-time deadlines provided $Q_i$ bits in each message are delivered by the specified deadline. The relative deadline is $D_i$ and the absolute deadline for the $k$th message is therefore $kT_i + D_i + O_i$. We can therefore see that the $i$th stream meets its real-time deadlines if its throughput function satisfies

$$\gamma_i(kT_i + D_i + O_i) \geq kQ_i \tag{5}$$

for all $k$ within the window of interest. Equation (5) is a set of inequality constraints characterizing whether the stream satisfies its real-time constraints. The stream shown in figure 2, of course, does not meet its real-time constraints.

The real-time constraint represented by equation (5) is just one constraint that must be satisfied by the real-time flow. In addition to this, we need to make sure that the total supply rate, $\overline{\beta}_{0j}(t)$, is not exhausted in processing the streams in $L_j$. In this case, figure 2 shows what happens to the supply $\beta_{02}(t)$ on link 2 that is used to service the first stream. In this case, the supply unused by $G_{12}$ is

$$\beta_{12}(t) = \max\left\{0, \beta_{02}(t) - \overline{\beta}_{12} \otimes \alpha_{11}(t)\right\}$$

This unused supply is shown by the shaded blue region in figure 2 and this then becomes the supply that is input to $G_{22}$ in figure 1. Continuing on down, the final unused supply for link 2 may be expressed as

$$\beta_{22}(t) = \max\left\{0, \beta_{02}(t) - \sum_{i=1}^{2} \overline{\beta}_{i2} \otimes \alpha_{i1}(t)\right\}$$

This last supply curve $\beta_{22}(t)$ represents the total unused supply available at link 2. If this curve is zero for all time, then there is no unused capacity at the link. We can therefore view this curve as a measure of how close a link is to exhausting its capacity and we will use this later to help user streams coordinate how they set their performance levels.

**Prior Work:** Our so-called "real-time" fabric has its origins in a calculus for network delay [27, 28] that uses the min-plus algebra to model network flows [6]. This approach provides a powerful method of characterizing delays in a variety of distributed or networked systems ranging from integrated service networks [89, 88] to embedded real-time systems [114, 120]. With respect to real-time systems this modeling framework is sometimes called the "real-time" (RT) calculus so that our "real-time fabric" may be viewed as a convenient way of visualizing relationships within the RT-calculus.

The RT-calculus models a real-time process as a discrete-event system whose inputs and outputs are streams (periodic or sporadic) of "data" events. Under the RT-calculus, these event streams are characterized in a way that easily captures a variety of well-known real-time QoS constraints such as the $(m, k)$ model [94] [40], the skip-over model [51], and weakly-hard systems [11]. A stochastic extension of the RT-calculus could also be used to capture the Markov chain (MC) QoS constraint [70]. The RT-calculus, therefore, provides a flexible way of modeling real-time processes that can treat most of the best-known real-time QoS constraints.

The RT-calculus uses the *behavioral* framework for modeling dynamical systems [91]. Under this framework, a system is a set of "event streams" (behaviors) and the composition of systems is the intersection of these sets. The behavioral approach has proven valuable in modeling the composition of heterogeneous systems [95]. Behavioral models based on event streams preserve a form of "causality" under system composition [53, 54] that make them particularly well-suited for modeling the hierarchical scheduling of multiple resources [121, 105, 104, 81, 82]. Since models based on the RT-calculus have such attractive compositional properties, it is ideal for modeling the interaction of data streams and resources in a wireless network.

As noted above, the RT-calculus invokes the *fluid* approximation in which network flows can be divided up into infinitesimally small packets. The framework we're proposing here is strongly infuenced by the *generalized processor sharing* (GPS) approach to modeling network flows [89, 88] which also relies on the fluid approximation. GPS nodes, however, are theoretical concepts that ignore the innate discrete-nature of digital communication. So one natural question is whether or not we can implement real-life network nodes to behave like GPS nodes? This question is examined in more detail in section 5.

## 3 -Price-Based Control of Real-time Networks

This section discusses how this project will use the real-time fabric to pose an optimization problem to maximize the application's performance subject to network throughput constraints. At its heart, this is a problem about managing a network's real-time quality-of-service (QoS). Enforcing hard real-time constraints in wireless networks has proven to be very difficult due to the unpredictability of wireless communication. This project will overcome that difficulty by adjusting the user's performance requirements, rather than directly adjusting the link's service rate. In particular, this means that links provide feedback to the users about how "stressed" they are, and the users then adjust their performance levels to stay within the network's throughput limits. Since this approach to QoS management adjusts user "utilization", it is referred to as the network utility maximization or NUM problem. This project, therefore, is proposing to use the network calculus to pose a NUM problem for real-time QoS enforcement in wireless networks. NUM type algorithms have been well studied in the context of Internet congestion control [49, 72, 5] and mobile ad hoc networks [92, 26, 131]. NUM methods have recently been suggested as a method for enforcing real-time constraints in wireless networks [98, 45]. This project's methods are similar to what was done in [98] and [45]. Our approach, however, offers significant innovations with regard to the choice of utilization function and implementation. These innovations will make this project's methods particularly well-suited to real-time control over wireless networks.

Managing real-time flows in a WSAN requires that we balance the network's ability to support hard real-time guarantees against the user requirements. At first glance, one might formulate this as a single large optimization problem of the form

$$\begin{array}{ll} \text{minimize:} & \sum_{i=1}^{N} J_i(Q_i, T_i) + \sum_{i \in L_j} \overline{r}_{ij} \\ \text{subject to:} & \gamma_i(kT_i + D_i + O_i) \geq kQ_i \end{array} \tag{6}$$

In the above equation, $J_i$, represents the user's performance cost as a function of the application stream's parameters. The second term in the objective minimizes link energy consumption by reducing the link's supply rate. This overall cost is minimized subject to the real-time constraint of equation (5). This is a centralized optimization problem reminiscent of the control/scheduling co-design methods in [103, 106, 20, 4].

The problem statement in equation (6) is of limited value. The nonsmooth nature of the inequality constraints can make it very difficult to find solutions to the problem. Moreover, the problem uses a single cost function that would be difficult to evaluate in a real-time manner. In practice, the problem in equation (6) is often solved by invoking large-scale solvers that execute on a single computer. This "centralized" approach to managing network flows is highly undesirable. It requires that all users/links transmit their information to a central computer. This central computer represents a processing bottleneck that would generate a great deal of communication overhead while also representing a single point of failure in the system. We would therefore prefer to develop a *distributed* approach for managing network flows.

A distributed approach has each user and link reconfigure itself on the basis of local information. This distributed approach has several advantages. Since each link/user communicates to a small subset of the network, the distributed approach has much lower communication overhead than a centralized approach. Since this overhead determines the cost of the communication infrastructure, the distributed approach will cost less than a centralized solution. In addition to this, there is good reason to believe distributed implementations will be more robust to modeling errors. By reducing communication overhead, the distributed approach allows information to be fed back to users/links at a much greater rate than is possible in a centralized system of comparable capacity. The primary impact of this higher feedback rate is to reduce a management scheme's sensitivity to disturbances that arise due to unanticipated changes in network loads or inaccurate models of network flows. As a result, we anticipate the distributed implementation to have lower cost and greater robustness to modeling error than centralized approaches to network management.

Our challenge is how best to formulate this distributed management strategy. We propose formulating the problem as a non-cooperative game that uses pricing mechanism to achieve an efficient Nash equilibrium. This approach has been successfully used in distributed flow control over ad hoc wireless networks. This prior work has addressed congestion control [3, 132, 30, 25, 92], routing [3], and power control [131, 26, 101]. These methods are well understood and there is, therefore, good reason to believe that we can also apply them to enforcing hard real-time constraints on wireless flows.

We now characterize a user-driven method for managing real-time flows. In particular, let's assume that the user has decided that its performance level should be $\overline{J}_i$. The user can then select a quantization level, $Q_i$, and period $T_i$ such that $J_i(Q_i, T_i) = \overline{J}_i$. The selection of $Q_i$ and $T_i$ immediately specifies a desired level of real-time support. For a specific period, we know (see [125]) that there is a maximum allowable delay that can be tolerated by the application. If we denote this maximum delay as $D(T_i)$, then the obvious thing to do is let the arrival curve's deadline $D_i$ equal $D(T_i) - \overline{d}_i^*$ where $\overline{d}_i^*$ is the end-to-end delay experienced by stream $i$ due to service delays at the links. These selections immediately fix the smallest supply rate that will satisfy the user's real-time requirements. In particular, we can easily see that all links on the user's route must supply a minimum rate of $\frac{Q_i}{D(T_i)-\overline{d}_i^*}$ bits/second to ensure that all messages are received within the deadline $D_i$. Morever, if these links also want to minimize their energy consumption, then all links along the route should supply no more than the specified minimum bit rate. Therefore, as soon as a user selects $Q_i$ and $T_i$, the rational selection for each link is to simply set their supply rates, $\overline{r}_{ij}$, to $\frac{Q_i}{D(T_i)-\overline{d}_i^*}$. Note that this particular approach to selecting service rates may be conservative. Section 5 considers an alternate way of setting up service rates.

Since this selection for the supply rates automatically assures the satisfaction of the real-time constraint there is only one remaining constraint that the supply rates need to satisfy. In reviewing the real-time fabric model in figure 1, we see that each link has a maximum overall supply rate $\overline{r}_{0j}$. Clearly the selected rates, $\overline{r}_{ij}$, (for $i \in S_j$) cannot exceed this overall rate. With this observation in hand, we can now recast our underlying optimization problem as

$$\begin{array}{ll} \text{minimize:} & \sum_{i=1}^{N} J_i(Q_i, T_i) \\ \text{subject to:} & \sum_{i\in S_j} \overline{r}_{ij} \leq \overline{r}_{0j}, \quad j = 1, \ldots, M \end{array} \tag{7}$$

In general, we expect $J_i$ to be a convex function of the quantization level and broadcast period. The constraint given above is linear. So this problem can be seen as a special case of the network utility maximization (NUM) problem [49] used in Internet congestion control. As was done in [72], we use a dual formulation of the problem in equation (7) to obtain a distributed algorithm that generates a sequence of user parameters that asymptotically converge to the solution of the problem in equation (7).

The distributed solution to our real-time flow problem may be obtained by considering the dual problem

$$\max_{p\geq 0} \min_{Q_i,T_i} \left( \sum_{i=1}^{N} J_i(Q_i, T_i) - \sum_{j\in L_i} p_j \left( \sum_{i\in S_j} \overline{r}_{ij} - \overline{r}_{0j} \right)^{+} \right) \tag{8}$$

where $(x)^+ = \max\{0, x\}$. It is well known that a solution to the dual problem in equation (8) will also solve the original problem in equation (7). In the above equation $p = [p_1, \cdots, p_M]^T$ is a vector of Lagrange multipliers associated with the link's rate constraint. In particular, the vector's $j$th component, $p_j$, can be viewed as a *price* that link $j$ charges all users whose streams pass through that link. The term $\left(\sum_{i\in S_j} \overline{r}_{ij} - \overline{r}_{0j}\right)^+$ is a measure of how overloaded that link is. So each user pays a price that is proportional to how overloaded the link is. This cost exerts indirect pressure on all users to adjust their "performance" levels so that the requested service rates ($Q_i$ and $T_i$) remain with the network's capacity constraints.

The dual problem in equation (8) can be solved in a distributed manner using the so-called *dual-decomposition* algorithm. The dual decompostion algorithm was proposed in [72] as a means of solving the NUM problem. The convergence properties of this algorithm are well understood. This algorithm uses a recursion of the following form,

$$\begin{aligned} (Q_i[k+1], T_i[k+1]) &= \arg\min_{Q_i,T_i} \left\{ J_i(Q_i[k], T_i[k]) - \frac{Q_i[k]}{D(T_i[k]) - \overline{d}_i^*} \sum_{j\in L_i} p_j[k] \right\} && (9) \\ p_j[k+1] &= \max\left\{ 0, p_j[k] + \gamma \left( \sum_{i\in S_j} \frac{Q_i[k]}{D(T_i[k]) - \overline{d}_i^*} - r_{0j} \right)^{+} \right\} && (10) \end{aligned}$$

where $\gamma$ is a step size that must be chosen sufficiently small to assure the algorithm's convergence. The algorithm shown above is distributed. Equation (9 updates each user's quantization level and broadcast period. Note that this update only requires the "price" information from those links being used by the stream. Equation (10) adjusts the link prices and again these updates only require information from that link's users. Together these two recursions form a feedback loop between users and links that allow the users to adjust their required service levels in a manner that optimizes their performance while remaining within each link's rate constraints. This project proposes using the recursions in equations (9)-(10) as a starting point for managing real-time flows in wireless networks.

**Prior Work:** There has, of course, been other work attempting to develop real-time middleware for wireless networks. Examples include the RAP [73] and SPEED [41] protocols which use time-tagged events with best-effort delivery guarantees. These protocols have been used in middleware supporting real-time network databases [50, 61]. This prior work, however, only provides best-effort guarantees and is unsuitable for safety-critical control systems. Stronger guarantees may be obtained by adaptively managing network resources as has been done in [15]. Such flow management has been achieved through feedback-controlled scheduling protocols [74, 109]. This earlier work, however, relies on a heuristic application of ”control theory” to real-time scheduling and is therefore unable to provide predictable guarantees that are required for hard real-time.

Prior research has also used heuristic incentives to encourage node cooperation for packet forwarding, congestion control, and power control. One heuristic [76] uses of a ”watchdog” node to identify misbehaving nodes in the network. Distributed *reputation* [16] or *currency* [134, 17] based heuristics have also been used as incentives for cooperation. While these methods appear to work empirically, there is little analysis characterizing their behavior.

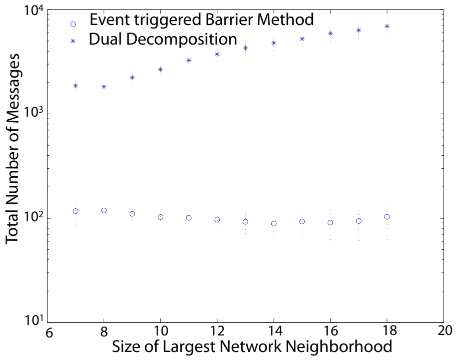
The price-based formalism described above is based on the network utility maximization problem (NUM) proposed by Kelly et al. [49]. The specific algorithm we use is a variation on the dual-decomposition algorithm suggested by Low [72, 5] for Internet congestion control. It should be noted, however, that this is not the only way of introducing pricing. We can use a primal, dual, or primal-dual characterizations of the problem to develop competing price-based algorithms [86, 130, 47, 87]. All of these approaches lead to distributed algorithms that are convergent and this can be used to identify alternative algorithms for our problem.

The desirability of a price-based algorithm’s equilibrium may be answered more rigorously using a game-theoretic view [108]. Game theory has a long history in network management. Game-theoretic ideas have been used in power control in cellular networks [46, 1] and data networks [34, 100, 101, 2]. Game theoretic ideas have also be used to address congestion/flow control [30, 3, 8, 25], resource (bandwith) allocation [132, 92, 8, 26, 131], routing [3], and packet-forwarding [92, 36] in ad hoc wireless networks. Much of this work has shown that suitably designed price-based formalism converge to Pareto-efficient Nash equilibria [101]. Such equilibria are desirable because they represent solutions that are fair and ”socially” optimal. In other words, the equilibria achieved using price-based mechanisms are highly desirable from the standpoint of the overall community of links and users. We believe that similar claims can be made of the price-based algorithm we’re proposing to use to solve our hard real-time network problem.

Our approach is most closely related to very recent work [98, 45] using network utility maximization as a basis for dynamically adjusting bandwidth for real-time flows. Saad et al. [98] and Jayachandran et al. [45] both use NUM formalisms to enforce end-to-end QoS requirements through the adjustment of user rates. In Saad et al., the underlying user objective is an unspecified utility function. Jayachandran introduce a specific utility function that is directly related to deadline satisfaction. The approach being proposed here differs significantly from that earlier work in its choice of user objective. This project focuses on control applications where there is a complex relation between deadlines, periods, quantization levels and application performance (utility). Recent results [113, 62, 33, 60, 67, 58, 85, 112, 18, 42, 125, 127] have begun to uncover what these relationships are and how we might exploit them in building reliable computer-controlled systems. The methods being proposed in this project, therefore, possess much greater flexibility than the earlier work by Saad [98] and Jayachandran [45] and we believe this flexibility represents an important way in which our proposed effort can transform the use of NUM methods into a practical method for building reliable real-time control over wireless networks.

**Research Challenges:** While we believe that our proposed method has significant potential, there are a number of research challenges that will need to be addressed. Each of these challenges is discussed in greater detail below.

One major challenge involves the **communication overhead** required by price-based methods. Each user needs to have ”price” information from its links and each link needs end-to-end delay information. Equations (9) and (10) represent a set of coupled dynamical equations in which the link broadcasts its price to the users *at each time step*. Transmitting price information at every time step leads to a tremendous amount of communication overhead. The challenge in this case is to develop algorithms whose overhead scales in an attractive manner with network size and complexity. This project proposes addressing this challenge using an **event-triggered** price-based NUM algorithm. Under event-triggered message passing, price information would only be broadcast when a link's internal ”error signal” exceeds a state dependent threshold. Our prior work applied such event-triggering to NUM problems using price-based barrier method [117], augmented Lagrangian [116] and primal-dual [119] algorithms. We've also applied event-triggering to NUM problems in wireless sensor networks [118] as well as power dispatch in microgrids [56]. The results from one of these earlier studies are shown in the figure below.



This plot graphs the total number of iterations used by traditional dual decomposition and an event-triggered NUM algorithm as a function of the network's diameter. The remarkable thing to be seen is that the total number of messages used by the event-triggered algorithm is nearly two orders of magnitude lower than what is found under dual-decomposition. The second observation is that the message passing complexity under event-triggering appears to be *scale-free*. In other words, event triggering appears to significantly reduce the communication complexity of these distributed optimization algorithms. We propose using this as a starting point for developing low-overhead price-based algorithms.

One important difference between this project and previous results [98, 45] lies in its choice of utilization or performance cost. This project will attempt to minimize the **control application's performance cost through adjustment of period, delay, and message quantization**. This approach is more desirable than the method used in [45] in which utilization is only a function of the end-to-end delay. We will need to directly relate application performance to delay, period, and quantization level. The question is whether or not such control-based performance functions are known? The answer, in part, is yes. While there is prior work relating control performance to sampling period [85, 112, 18, 125, 127] or to quantization level [113, 62, 33, 60, 67, 58] , there is very little work on how performance varies as a function of both period and quantization. Very recently one can find work [42, 125, 127] characterizing the maximum allowable delay $D(T_i)$ as a function of the sampling period. But we have yet to see how this might be related to our earlier work that characterizes control system performance as a function of quantization level. One of the important challenges that must be addressed in this project therefore involves a characterization of control performance as a joint function of period and quantization. Given our extensive prior contributions to sampling [125, 127, 128, 122, 126, 124] and quantization [60, 67, 58, 68, 66], this group is well qualified to address this challenge.

Wireless networks exhibit large variations in network throughput. In the face of such throughput variations, is it realistic to expect WSANs to support **safety-critical applications**? The answer to this question rests on how much flexibility we expect on the application side. For instance, the control performance functions cited in the preceding paragraph (see [60] for instance) assume a fixed controller in which reduced quantization levels reduce that controller's performance. We can also, however, think of changing the structure of the controller. In other words, if network throughput drops in a precipitous manner, it may make more sense to switch to a so-called *safe* controller that assures closed-loop stability under extremely low channel bandwidth. Switching controller structure in response to variations in network capacity lies at the heart of the so-called **anytime control** methods [12, 37]. Anytime control methodologies provide additional flexibility that can guarantee control system performance under extremely limited network bandwidth. This project will examine the extent to which anytime controls can be integrated with our proposed price-based algorithm. In particular, we'll study this question with regard to safety-critical control applications.

At its heart the proposed price-based algorithm is a feedback system that should be relatively insensitive to unmodeled disturbances. This observation means that the proposed method should be robust to **variations in the arrival and supply curves**. Recall that the previous section focused on a rather narrow class of periodic arrival curves and constant bit rate supply curves. Due to the feedback nature of our scheme, it should also be able to handle generalized arrival and supply curves. In particular, we'd like to consider sporadic arrival curves that are generated by self-triggered feedback control systems [125]. We'd also like to study the impact that random variations in link parameters such as link delay $\overline{d}_{ij}$ and link service rate $\overline{r}_{ij}$ have on QoS allocation. Due to its feedback nature, we expect the proposed algorithms to have robust performance with respect to such parameter variations. The challenge involves quantifying that uncertainty in a way which scales well with overall network size and complexity.

One thing that hasn't been addressed in this work concerns **malicious or irrational users**. An irrational user is one that fails to follow the price-based update in equation (9). Such irrational behavior may be unintentional. It is possible, for example, that a memory fault at the user results in what appears to be malicious or irrational decisions. In this case, heuristics similar to those used in [76, 134, 17] may be appropriate. This project will investigate how irrational user behavior may adversely impact our priced-based approach.

Another interesting challenge involves the role of **network links**. In the problem formulation given above, we focused on having users determine their desired quantization level, $Q_i$, and sampling periods, $T_i$. To meet real-time constraints, the links are then required to provide the associated minimum supply rates $Q_i/D(T_i)$. But an observant reader will note that these are not the only things that can be adjusted. The stream offset, $O_i$, can clearly be adjusted by the link without adversely impacting application performance. By adjusting stream offsets, the link may be able to increase its effective rate capacity. In addition to this the link can adjust the *priority* with which streams are processed. The ordering of the streams within the link set $L_j = \{i_1, \cdots, i_N\}$ actually specify a priority. Those streams appearing first in the set will take the link's overall supply $\overline{r}_{0j}$ first. By adjusting stream priority it may be possible for links to again provide users greater freedom in selecting their quantization levels and periods. One thing which we have yet to fully explore is how we might be able to use the unused degrees of freedom available to the link in optimizing overall network behavior. This issue will be discussed in greater detail in section 5.

## 5 - Link Level Resource Management

The price-based algorithms discussed above assume that link nodes follow the general processor sharing (GPS) model. This is an idealized model of node behavior that ignores the inherent discrete-nature of packet-switching. Because of the non-negligible packet sizes, we must consider two effects at the link level. First, each packet must be transmitted nonpreemptively. Secondly, no packet can be transmitted until its last bit has arrived. These restrictions pose one important issue that needs to be addressed in this project, that is, how to support the GPS model at the link level such that the price-based scheme indeed leads to a solution to the optimization problem that we intend to solve. On a more general term, we need to manage link-level resources locally within the price-based framework such that the overall objective defined in (6) can be achieved.

**Prior Work:** A well studied way of extending the GPS model to a more realistic package transmission environment is the Packet GPS (PGPS) service discipline [89, 88]. A PGPS server uses the packet departing time under GPS as priority for scheduling packet transmission. Implementation of PGPS requires each node to maintain a virtual time for "simulating" the execution of GPS. PGPS and its relatives (such as Weighted Fair Queuing (WFQ) [29] and Worst-case Fair WFQ [10]) have the desirable traffic protection property and can provide bounded delays. Such scheduling schemes would be a natural choice since they are closest to the fluid model assumed by our price-based resource allocation approach. However, one serious drawback of these schemes is that they have high computational complexity (due to the maintenance of virtual time) and can be too costly to be implemented at sensor nodes such as those assumed in this project.

Regardless of the GPS or PGPS models, the ultimate goal at a link node is to schedule packet transmission to meet the end-to-end deadline of each packet. Many packet scheduling schemes have been investigated. One widely used packet scheduler category is earliest-deadline-first (EDF). It has been shown that EDF scheduling policy is optimal in providing delay bounds at a single node [38]. For providing end-to-end delay bounds, EDF requires the use of per-node traffic shaper in order to regulate the distorted traffic coming out of an EDF scheduler. If per-node traffic shaping is employed, the augmented EDF is referred to as RC-EDF and has been shown to outperform PGPS in the end-to-end case [39, 133]. However, the performance of a RC-EDF scheduler (i.e., the size of its schedulable region) is critically dependent on the traffic shaper and it has been proved that finding "optimal" shaper is in general infeasible [107]. Furthermore, traffic shaping introduces additional packet delays which may not be desirable when the bandwidth of the network is already scarce.

Other alternatives in the general category of EDF have also been studied. For instance, deadline-curve based EDF (DC-EDF) [135] does not rely on traffic shaping but adopts a different way of setting local packet deadlines. By judiciously adjusting local deadlines, DC-EDF achieves the purpose of smoothing the traffic and hence can guarantee end-to-end delay bounds while providing schedulable regions as large as that of RC-EDF [135]. However, DC-EDF does not provide service differentiation and has no regards for load distribution in the network, which can lead to some nodes become bottlenecks. Some attempts have been made to address this concern. For example, The load-aware EDF (LA-EDF) [13] adjusts the local deadline of a flow at a node based on the overall load distribution along the path that the flow traverses. Implementing LA-EDF could be somewhat challenging since it needs the load information along the entire flow path for each flow. Though EDF based scheduling may lead to larger schedulable regions, it is not clear how to integrate EDF-based scheduling with the price-based rate allocation approach.

Fixed-priority scheduling schemes could be less costly in terms of implementation at the link-level. They also provide a means for service differentiation, which can be desirable for mixed classes of traffic. Authors in [45] have presented an approach for delay bound calculation and resource allocation for fixed-priority scheduling. The approach also relies on solving a version of the NUM problem in a distributed fashion. By using fixed-priority scheduling, the end-to-end delay bound of a flow can be expressed as a function of service rates along the path that the flow traverses. A fundamental challenge with any fixed-priority scheme is how to make the priority assignment. Choices of priorities can drastically influence the serve rates demanded by traffic flows and hence applications' QoS. This is particularly problematic when considering multiple applications competing for the same real-time network.

**Proposed Approach and Research Challenges:** Our goal is to develop a systematic approach for resource management at the link level such that it facilitates the decentralized priced-based scheme to maximize the overall cyberphysical system performance. There are several fundamental requirements that such an approach must satisfy. First, it should use locally available information that does not require expensive global information. Second, it should be a component for guaranteeing the overall packet end-to-end delay. Third, it should provide an efficient means for setting up "price" for message streams to use. Last but not least, it should avoid imposing additional constraints onto applications and should not be overly pessimistic. Based on these requirements, we believe that an EDF based scheduling strategy is a good choice. However, as we discussed above, existing EDF schedulers cannot be used in the proposed price-based framework. We will address this problem in this part of the project.

The price-based framework essentially allows each message stream to reserve a service rate based on the prices setting up by each link. Each link node sets up its price based on its capacity and requested service rate by each stream. With the GPS model this price is governed simply by the difference in the capacity and total request services as shown in (10). For an EDF-based approach, we again need to relate total service rates with link capacity. To achieve this, we notice that scheduling periodic message transmission at a link-level node is similar to scheduling periodic task execution on a CPU. A main difference is that a message can only be preempted at boundaries of packets. Assuming the maximum packet size among all message streams to be $S_{\max}$, we can simply treat packets as non-preemptable portions of a task [71].

Following the above observation, we can readily modify the well-known EDF schedulability result from [7] to relate service rates with schedulability. Let $D_{ij}$ be the local deadline of stream $i$ at link node $j$. Then we have,

$$\sum_{i=1}^{N}\left(\left\lfloor\frac{R-D_{ij}}{T_i}\right\rfloor+1\right)Q_i+S_{\max}\leq R \tag{11}$$

where $R\in\{lT_i+D_{ij}\leq\min(B_p,H)\}, l\in\mathcal{N}$, and $B_p$ and $H$ denote the busy period and hyperperiod of the periodic streams to be scheduled, respectively. Though (11) is a sufficient and necessary condition for schedulability, it does not provide an efficient way of setting up price since the inequality may need to hold for a potentially large set of different $L$ values. In our prior work, we have proposed a modified version of (11) which reduces the set of $L$ values to one of two values [21, 22]. That is,

$$\sum_{i=1}^{N}\left(\frac{R^*-D_{ij}}{T_i}+1\right)Q_i+S_{\max}\leq R^* \tag{12}$$

where

$$R^*=\begin{cases} D_{2j} & : \quad D_{1j}+T_1\leq D_{2j}\\ \min_{i=1}^{N}\left(T_i+D_{ij}\right) & : \quad \text{otherwise}\end{cases}$$

(It is assumed here that streams are ordered in the increasing order of their local deadlines.) In the price-based scheme, at step $k$, when the deadline, period, and message size for each stream are computed and local deadlines for stream $i$ at each node are determined, one of the $R^*$ values can be used in (12) to set up the price at each node as

$$p_j[k+1] = \max\left\{0, p_j[k] + \gamma\left(\sum_{i \in S_j}\left(\frac{R_j^* - D_{ij}}{T_i} + 1\right)Q_i[k] + S_{\max} - R_j^*\right)\right\} \tag{13}$$

Equation (13) is an alternative to equation (10)'s approach in generating price. In this case, the price measures how close the link is to violating its EDF scheduling constraint, whereas equation (10) represents how close the link is to violating its total supply capacity. This project will investigate various methods for setting prices in the real-time networks. Equations (10) and (13) represent two reasonable starting points for this investigation.

There are several challenges in using the approach outlined above. First is how to determine messages' local deadlines at each node based on the messages' end-to-end deadlines. In [99], three different deadline assignment methods were proposed. They include (i) fair deadline assignment where a local deadline is set to the average delay budget computed by equally distributing the end-to-end delay budget (deadline) over the number of nodes along the path that a stream traverses, (ii) utilization-based deadline assignment where the delay budget is proportional to the utilization at each node, and (iii) bandwidth-based deadline assignment where the delay budget is inversely proportional to available bandwidth at each node. Other laxity-based deadline assignments have also been proposed [84, 48]. Most of these methods (except the fair deadline assignment) require the knowledge of load distribution throughout the path traversed by a stream. This is intuitively understandable since local deadline assignments could impact the service rate requested by each stream which in turn impacts the overall QoS of the application. We believe that it is advantageous to make use of load information for local deadline assignments. However, gathering such information in a distributed fashion is not as easy. We would like to study and compare the effectiveness of existing deadline assignment strategies under our particular framework. Furthermore, we would like to investigate decentralized methods to gather load information within the real-time fabric.

Another challenge associated with our proposed approach is that the condition in (12) can be quite pessimistic especially when the traffic is heavy [22]. This pessimism is further aggravated since the conditions in both (11) and (12) assume that the message streams exhibit the worth-case phasing while in fact phase shifts naturally occur in the downstream traffic. If the arrival time of each message to a node is known a priori, one could consider modifying (12) by taking into account the phase shifts between messages when evaluating schedulability. However, in the price setting phase, such information is not available. We will consider improving the quality of the schedulability condition in (12) in two directions. One is to examine possible relationships between $D_{il}$ and $T_i$ that may lead to better approximations to (11) than (12). Note that we have some freedom in selecting $D_{il}$ as discussed above. If certain relationship between $D_{il}$ and $T_i$ could lead to less pessimistic approximation of (11), it could result in higher overall QoS. The other direction that we would like to explore is to leverage some decentralized load information gathering mechanism to estimate arrival patterns of different streams. Such patterns may then be exploited to improve the schedulability condition.

The above two efforts address the difficulties arising during the "negotiation" stage of our framework. Since a WSAN is a rather dynamic environment, network behavior would often deviate from the assumptions made during the negotiation stage. The advantages of using the price-based framework is that it can dynamically update resource allocation in response to network changes. However, this update is not free in terms of both computation and communication resource usage. It would be desirable to reduce such needed updates as much as possible. In the previous section, we have discussed one possible direction is to use event-triggered price-based algorithms to reduce the frequency of such updates. We think it is also possible to leverage local link-level schedule adjustments to help reduce this frequency. For example, in DC-EDF [135], messages' local deadlines are assigned based on the arrival times of the messages. We could use the same way to adjust message deadlines, that is, if a message arrives late (say due to unexpected network congestion), its priority at the current node could be raised (say by shortening its local deadline) so that the message is given a chance to "catch up". However, in doing so, other messages may be delayed. What is needed is a coordinated approach to adjusting message deadlines locally. This bears some similarity to our prior work on generalized elastic scheduling [21, 22]. Yet the focus is different as here we are interested in end-to-end delay satisfaction instead of minimizing utilization deviation. We plan to develop efficient methods for tackling this deadline adjustment problem.

## 6 - Evaluation Plan

This project will develop middleware solutions implementing the price-based network management algorithms. This middleware will be evaluated on two hardware testbeds. The first testbed interconnects a group of robots using a WSAN. The second testbed will experimentally validate the algorithms on a hardware model of CSOnet [83] which is a WSAN that was built by our industrial partner (EmNet LLC) to reduce the frequency of combined-sewer overflow events from municipal sewers.

**WSAN-Robot Testbed:** The proposed testbed consists of three Active Media Pioneer robots that communicate with each other over an ad hoc wireless network built from 40-50 Mica2 modules. The Pioneer robots use acoustic proximity to detect obstacles. Gyro-corrected wheel encoders determine local position and orientation. The robots are controlled through an on-board embedded Linux PC that can communicate to the outside world through an 802.11 wireless LAN card and a Mica2 node. Low level robot motion controls were developed under an earlier NSF project (ECS04-00479) using a set of C++ classes developed for controlling high autonomy mobile robots. These robots provide a reasonable testbed for studying multi-robot coordination over ad hoc wireless networks. The wireless LAN connection can be used to monitor overall system behavior and thereby evaluate how well robot motion planning is performing. In our experiments, we propose coordinating the three robot actions over the MICA2 ad hoc wireless network.

This testbed will be used to develop a distributed receding horizon controller (RHC) for robot formation stabiliztaion [32]. The RHC controller requires state information from neighboring robots. The communication infrastructure enabling robot-to-robot communication will be an ad hoc network of Mica2 nodes. In order to assure stable robot formation control, it is essential that these data streams satisfy hard real-time deadlines. The price-based network algorithms being developed by this project will be used on this network of Mica2 nodes to evaluate how well our proposed methods work.

We'll develop our middleware algorithms using solutions that the PI previously wrote for an earlier DARPA project [35] and the LakeNet system [102]. This middleware was built to stream logged sensor data to a data fusion center. It used stateless gradient-based routing [14], where the gradient is determined by "nearness" to the data fusion center. This approach builds a local routing table that allows each node to select the most reliable nearest neighbor in forming the routes. Sensor data begins streaming to the data fusion center, once the route has been established. These streams consist of periodically transmitted fixed length packets. Hop-by-hop acknowledgements are used to guarantee sensor data delivery. These acknowledgements are also used to compute estimates of link reliability to adaptively reroute packets when a link fails. Earlier testing of these services on the LakeNet project [102] showed that they provided a simple and robust approach to reliably stream data between sensors and data fusion centers.

We will use these earlier middleware services as a starting point in developing a real-time middleware service for the robotic formation control application. While our previous solutions worked well, they provided no guarantees on real-time message delivery. The price-based mechanisms described above will be integrated into our earlier middleware and the real-time performance of the resulting system will be experimentally measured and compared against theoretical predictions of the services' expected performance.

**CSOnet Testbed:** For the past four years, the project PI has been working with a company (EmNET LLC) to build a metropolitan scale wireless sensor-actuator network called CSOnet. CSOnet [96, 115, 83] is being built in the City of South Bend Indiana to control the frequency of combined sewer overflow (CSO) events. The construction of the initial prototype was funded by Indiana's 21st Century Technology Fund with subsequent funding in 2007 to expand the system to cover the entire city of South Bend. At present the system consists of a 110+ sensor network covering a 13,000 acre area with an additional 18-20 actuation points being added by summer 2009. Other CSOnet systems are currently being developed for Fort Wayne Indiana, Indiannapolis, and Omaha Nebraska.

CSO events occur when stormwater flows overload a sewer's capacity, thereby forcing city enginners to divert excess storm water into a river [77] [80]. CSO events represent a major environmental hazard since the diverted waters contain chemical and biological contaminants. One way for addressing the CSO problem is to store excess water in unused parts of the city's sewer system (also called in-line storage). Currently in-line storage uses a centralized model-predictive control method [75, 31, 19] in which all sensors transmit their data over a SCADA network to a central computer. This computer makes a global control decision and broadcasts its decision to the actuators. Prior implementations of this centralized approach in Quebec Canada [90] and Milwaukee Wisconsin [97] demonstrated that the cost scales poorly in large metropolitan areas. CSOnet addresses the issue of cost scalability by distributing the control decisions throughout the entire sewer system. This means that actuation (control) decisions are made locally at the actuation point using information from neighboring actuation points. Because decision making is distributed, we no longer need a SCADA network and can, instead, use a lower cost mesh radio network. CSOnet is therefore a wireless sensor-actuator network that uses controllers distributed across the physical sewer network.

The current controller for CSOnet is a decentralized controller that only uses discrete events to coordinate actions between nodes. It was soon realized that better performance could be achieved if real-time information from neighboring nodes was available for control also. CSOnet's current middleware, however, is unable to provide hard real-time guarantees on data streams between adjacent nodes due to the unreliability of the wireless communication network. The objective of this testbed is to see whether the real-time middleware developed under this project will support real-time data streams between adjacent CSOnet controllers. The resulting algorithms will be validated on the hardware testbed currently at our industrial collaborator's (EmNet LLC) offices. EmNet will commit time and financial resources to assist in the project's completion (see EmNet's letter of commitment).

## 7 - Student Training and Curriculum Development Activities

As an integral part of the proposed work, the PIs will focus on several activities related to student training and curriculum development. The ultimate goal of these activities is to help build a diversified and well-qualified participant basis in the much needed cyperphysical system research area.

One of our efforts is on curriculum development aimed at preparing students with better mastery of formal methods for designing cyberphysical systems. As cyberphysical systems become more prevalent, many of our engineering and science graduates will eventually expect to work with them in one form or another. However, we feel that our current curricula are inadequate in providing students with the proper training for this need. As an example, the curricula from CSE (Computer Science and Engineering and other engineering departments at Notre Dame do not have any course that focuses on systematic design and analysis approaches for complex systems such as CPS at the undergraduate level. Though each department has courses on formal analysis and design of either physical systems or computer systems, there is no course on formal methods for studying the interactions between cyber and physical systems. We believe that this is true for many universities in the U.S., and intend to improve this situation under this NSF project.

As a starting point, the PI and co-PI will jointly develop a course, titled *Cyberphysical System Design: Methods and Practice*. The focus of the course is exposing students to formal design methods while allowing them to practice with the methods in actual design projects. Specifically, the course will introduce students to formal modeling, analysis and verification of concurrent processes with an emphasis on networked control systems. As we have pointed out in the earlier part of this proposal, networked control systems are a good representative of cyberphysical systems. Students will also learn basic concepts in control system modeling and performance estimation. Students will make use of formal verification tools for cyber-physical (hybrid) systems. We may use Uppaal as the verification tool for Stateflow models of cyber-physical systems. An alternative would be to use synchronous programming languages as a basis for CPS modeling. We will also introduce the use this project's real-time fabric for CPS modeling and analysis. As an integral part of the course, students will apply the formal design and evaluation methods to design a small scale networked control system as a course project. The course will be open to senior level undergraduates and entry level graduate students. It may also be used as a senior capstone course in CSE. We believe that this course will benefit greatly from the collaboration funded by this NSF project. The findings generated in this project are of direct relevance to the model and analysis of cyberphysical systems. We will in fact integrate some of the results into this course, thereby broadening the impact of the proposed research. Eventually, we would like to expand the above proposed course into a two-course sequence in which the first one covers more in-depth formal analysis and design methods while the second one focus on larger scale design projects.

On another training related effort, in the summer of 2009, the PI will be lecturing on event-triggered control to a Ph.D. summer school at the University of Siena, Italy. The substance of this lecture will be expanded into a set of on-line lectures that can be accessed from the world-wide web. A print version of these lectures will also be included in a book on networked control systems. The lectures will provide fundamental instruction on event-triggered optimization, estimation, and control. The lectures will be used to instruct the students of the PI and his collaborators. In making these lectures freely available, we hope to more broadly disseminate some of this project's results.

The PIs have a strong record of outreach activities. We have had 3-5 undergraduate students working at our respective groups in any given year. Furthermore, we have each advised a number of female graduate students. In fall of 2008, the PI began working with a local middle school in South Bend Indiana to discuss some of our recent work with embedded control of sewer networks [83]. It was observed that many of these students are taking part in an after school program using Lego's mind-storm system to develop autonomous robots. We plan to continue our outreach effort by hosting REU students and working with local middle schools as part of this project. Since part of this project involves using our multi-robotic testbed, we could with a little additional effort develop a demonstration using the project's multi-robotic testbed to show how their mind storm robots would scale up. So as part of this project, we intend to develop graphical and programming interfaces that these students can use to control our more powerful robots. We feel that this would help students see beyond the "toy-like" nature of the mind-storm robots and would stimulate their interest in entering systems engineering. Effort in reaching out to under-represented groups, particularly female students, will be made. Since one of the PIs is a female faculty member, this gives us a unique position to recruit female undergraduate and graduate students to participate in the project.

## 8 - Results from Prior NSF Sponsored Research

Dr. Lemmon and Dr. Hu have jointly received prior support under NSF grants **Flexible Scheduling in Real-Time Control Systems with Uncertainty**, CNS-0410771, $240,000, 2004–2007 and **Integrating Decentralized Control and Real-Time Scheduling for Networked Dynamical Systems**, CCF-0720457, $160,000, 2007–2010. Dr. Lemmon also received support under NSF grant ECS04-00479 (2004-2007), titled **Scalable Decentralized Control over Ad Hoc Sensor Actuator Networks** ($210,000).

These grants studied modeling, performance evaluation and design of networked control systems and real-time systems. Our earliest work developed a power spectral analysis for linear control systems with dropped feedback messages [63, 64, 66]. A Markov-chain model for data dropouts in control was considered in [65, 59]. These Markov-chain models were later used in developing a novel real-time QoS constraint for control systems [70, 69, 44]. Later papers [58, 68] studied the stability of dynamically quantized feedback systems. We also studied the use of periodic communication logics for multi-agent systems [110] and the interconnection of swarms with consensus filters in [60, 111]. We developed an event-triggered and self-triggered approach to feedback control [129, 123, 122, 126, 124, 125, 57]. Event-triggering was applied to utility maximization problems in [116, 117]. We also examined extensions of elastic scheduling as they might apply to networked control [22, 23, 24, 21, 43]. Besides control and real-time performance, we also studied the problem of power/energy consumption in real-time control systems. Specifically, we examined how dynamic voltage scaling can improve jitter in real-time systems [78]. We proposed methods for optimizing power consumption in distributed real-time systems [93]. We studied the problem of overall networked system energy consumption [79].

This proposed project is most closely related to NSF grant CNS07-20457. The work proposed in this project is significantly different from the work in CNS07-20457. The work proposes developing middleware and real-time congestion algorithms for wireless sensor networks. CNS07-20457, on the other hand, focuses on self-triggering in real-time computing systems.

# References

[1] T. Alpcan, T. Basar, R. Srikant, and E. Altman. Cdma uplink power control as a non-cooperative game. *Wireless Networks*, 8:659–670, 2002.

[2] E. Altman and Z. Altman. S-modular games and power control in wireless networks. *IEEE Transactions on Automatic Control*, 45(5):839–842, 2003.

[3] E. Altman, T. Basar, and R. Srikant. Nash equilibria for combined flow control and routing in networks: asymptotic behavior for a large number of users. *IEEE Transactions on Automatic Control*, 47(6):917–930, 2002.

[4] K-E Arzen, A. Cervin, J. Eker, and L. Sha. An introduction to control and scheduling co-design. In *IEEE Conference on Decision and Control*, 2000.

[5] S. Athuraliya and S.H. Low. Optimization Flow Control with Newton-like algorithm. *Telecommunication Systems*, 15(3-4):345–358, 2000.

[6] F.L. Baccelli, G. Cohen, G.J. Olsder, and J.P. Quadrat. *Synchronization and Linearity*. John-Wiley, 1992.

[7] S. Baruah, L. Rosier, and R. Howell. Algorithms and complexity concerning the preemptive scheduling of periodic, real-time tasks on one processor. *Real-Time Systems*, 2(4):301–324, 1990.

[8] R. Battiti, M. Conti, and E. Gregori. Price-based congestion control in WiFi hot spots. In *Proceedings of WiOpt'03*, 2003.

[9] A. Bemporad. Decentralized wireless control of large-scale systems. WIDE project - University of Siena - http://www.wide.dii.uinisi.it, 2008.

[10] J. C. R. Bennett and H. Zhang. Wf2q: worst-case fair weighted fair queueing. *IEEE INFOCOM*, pages 1080–1085, 1996.

[11] G. Bernat, A. Burns, and A. LLamosi. Weakly hard real-time systems. *IEEE Transactions on Computers*, 50(4), 2001.

[12] R. Bhattacharya and G.J. Balas. Anytime control algoirthm: model reduction approach. *Journal of Guidance, Control and Dynamics*, 27(5):767–776, 2004.

[13] C. Bouras and A. Sevasti. An analytical qos service model for delay-based differentiation. *Computer Networks*, 51:3549–3563, 2007.

[14] D. Braginsky and D. Estrin. Rumor routing algorthim for sensor networks. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 22–31. ACM Press New York, NY, USA, 2002.

[15] S. Brandt, G. Nutt, T. Berk, and J. Mankovich. A dynamic quality of service middleware agent for mediating application resource usage. In *Proceedings of 19th IEEE Real-time Systems Symposium (RTSS)*, 1998.

[16] S. Buchegger and J-Y Le Boudec. Performance analysis of CONFIDANT protocol (cooperation of nodes - fairness in dynamic ad-hoc netowrks). In *Proceedings of third ACM International Symposium on Mobile Ad-Hoc Networking and Computing (MobiHoc)*, 2002.

[17] L. Buttyan and J-P Hubaux. Enforcing service availability in mobile ad hoc WANs. In *Proceedings of First ACM/IEEE International Workshop on Moble Ad Hoc Networking and Computing (MobiHoc)*, 2000.

[18] D. Carnevale, A.R. Teel, and D. Nesic. A lyapunov proof of improved maximum allowable transfer interval for networked control systems. *IEEE Transactions on Automatic Control*, 52:892–897, 2007.

[19] G. Cembrano, J. Quevedo, M. Salamero, V. Puig, J. Figueras, and J. Marti. Optimal control of urban drainage systems. A case study. *Control Engineering Practice*, 12(1):1–9, 2004.

[20] A. Cervin, J. Eker, B. Bernhardsson, and K-E Arzen. Feedback-feedforward scheduling of control tasks. *Real-time Systems*, 23(1-2):25–53, 2002.

[21] T. Chantem, X. Hu, and M.D. Lemmon. Generalized elastic scheduling. In *IEEE Real Time Systems Symposium*, 2006.

[22] T. Chantem, X.S. Hu, and M.D. Lemmon. Generalized elastic scheduling for real-time tasks. to appear in IEEE Transactions on Computers, 2008.

[23] T. Chantem, X. Wang, M.D. Lemmon, and X. Hu. Period and deadline selection for schedulability in real-time systems. In *Proceedings of the European Conference on Real-time Systems*, 2008.

[24] T. Chantem, X.Hu, and M.D. Lemmon. Period and deadline section problem for real-time systems. In *Proceedings of the IEEE Real-time Systems Symposium - work in progress track*, 2007.

[25] L. Chen, SH Low, and JC Doyle. Joint congestion control and media access control design for ad hoc wireless networks. *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, 3, 2005.

[26] M. Chiang and J. Bell. Balancing supply and demand of bandwidth in wireless cellular networks: utility maximization over powers and rates. *Proc. IEEE INFOCOM*, 4:2800–2811, 2004.

[27] R.L. Cruz. A calculus for network delay i: network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–131, 1991.

[28] R.L. Cruz. A calculus for network delay ii: network analysis. *IEEE Transactions on Information Theory*, 37(1):132 –141, 1991.

[29] A. Demers, S. Keshav, and S. Shenkar. Analysis and simulation of a fair queueing algorithm. *Internetworking Res. and Experience*, 1, 1990.

[30] C. Douligeris and R. Mazumdar. a game theoretic perspective to flow control in telecommunication networks. *Journal of the Franklin Institute*, 329(2), 1992.

[31] S. Duchesne, A. Mailhot, and J.P. Villeneuve. Global Predictive Real-Time Control of Sewers Allowing Surcharged Flows. *Journal of Environmental Engineering*, 130(5):526–534, 2004.

[32] W.B. Dunbar and R.M. Murray. Distributed receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549–558, 2006.

[33] N. Elia and S. Mitter. Stabilization of linear systems with limited information. *IEEE Transactions on Automatic Control*, 46(9):1384–1400, 2001.

[34] D. Falomari, N. Mandayam, and D. Goodman. A new framework for power control in wireless data networks; games utility and pricing. In *Proceedings of the Allerton Conference on Communication, Control, and Computing*, pages 546–555, 1998.

[35] L. Fang, P.J. Antsaklis, L. Montestruque, B. McMickell, M.D. Lemmon, Y. Sun, H. Fang, I. Koutroulis, M. Haenggi, M. Xie, and X. Xie. Design of a wireless dead reckoning pedestrian navigation systems: the navmote experience. *IEEE Transactions on Instrumetation and Measurement*, 54(6):2342–2358, 2005.

[36] M. Felegyhazi, J-P Hubaux, and L. Buttyan. Nash equlibria of packet forwarding strategies in wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 5(5):463–476, 2006.

[37] D. Fontanelli, L. Greco, and A. Bicchi. Anytime control algorithms for embedded real-time systems. In *Hybrid Systems: computation and control*, 2008.

[38] L. Georgiadis, R. Guerin, and A. Parekh. Optimal multiplexing on a single link: delay and buffer requirements. *IEEE/ACM Transactions on Information Theory*, 43:1518–1535, 1997.

[39] L. Georgiadis, R. Guerin, V. Peris, and K. Sivarajan. Efficient network qos provisioning based on per node traffic shaping. *IEEE/ACM Transactions on Networking*, 4:482–501, 1996.

[40] M. Hamdaoui and P. Ramanathan. A dynamic priority assignment technique for streams with $(m, k)$-firm deadlines. *IEEE Transactions on Computers*, 44(12):1443–1451, 1995.

[41] T. He, J.A. Stankovic, C. Lu, and T. Abdelzaher. Speed: a stateless protocol for real-time communication in ad hoc sensor networks. In *Proceedings of the 23rd International Conference on Distributed Computing Systems*, 2003.

[42] W.P.M.H. Heemels, A.R. Teel, N. van de Wouw, and D. Nesic. Networked control systems with communication constraints: tradeoffs between sampling intervals, delays and performance. submitted to the 2009 European Control Conference (ECC), 2008.

[43] X. Hu, T. Chantem, and M.D. Lemmon. Optimal elastic scheduling. In *IEEE Real-time and Embedded Technology and Applications Symposium (works in progress track)*, 2006.

[44] X. Hu, D. Liu, M.D. Lemmon, and Q. Ling. Firm real-time system scheduling based on a novel qos constraint. In *Real Time Systems Symposium*, 2003.

[45] P. Jayachandran and T. Abdelzaher. Bandwidth allocation for elastic real-time flows in multihop wireless networks based on network utility maximization. In *Proceedings of the 28th International Conference on Distributed Computing Systems*, 2008.

[46] H. Ji and C. Huang. Non-cooperative uplink power control in cellular radio systems. *Wireless Networks*, 4(3):233–240, 1998.

[47] B. Johansson, P. Soldati, and M. Johansson. Mathematical Decomposition Techniques for Distributed Cross-Layer Optimization of Data Networks. *Selected Areas in Communications, IEEE Journal on*, 24(8):1535–1547, 2006.

[48] J. Jonsson and K. Shin. A robust adaptive metric for deadline assigment in heterogeneous distributed real-time systems. *Real-Time Systems*, 23:239–271, 2002.

[49] F.P. Kelly, A.K. Maulloo, and D.K.H. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, 1998.

[50] S. Kim, S.H. Son, J.A. Stankovic, S. Li, and Y. Choi. Safe: a data dissemination protocol for periodic updates in sensor networks. In *IEEE Workshop on Data Distribution for Real-Time Systems (DDRTS)*, 2003.

[51] G. Koren and D. Shasha. Skip-over: algorithms and complexity for overloaded systems that allow skips. In *Proceedings of the IEEE Real-time Systems Symposium (RTSS)*, 1995.

[52] J.-Y. Le Boudec and P. Thiran. *Network Calculus*. Springer-Verlag, 2001.

[53] E.A. Lee. Modeling concurrent real-time processes using discrete-events. *Annals of Software Engineering*, 7:25–45, 1999.

[54] E.A. Lee and A. Sangiovanni-Vincentelli. A framework for comparing models of computation. *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, 17(12):1217–1229, 1998.

[55] M.D. Lemmon. Dynamic management of network's real-time fabric. submitted to the IEEE Conference on Decision and Control, 2009.

[56] M.D. Lemmon. Final report for odyssian sttr project entitled "advanced distribution and control for hybrid intelligent power systems. Final Report submitted to Odyssian LLC, February, 2009.

[57] M.D. Lemmon, T. Chantem, X. Hu, and M. Zyskowski. On self-triggered full information h-infinity controllers. In *Hybrid Systems: computation and control*, 2007.

[58] M.D. Lemmon and Q. Ling. Control system perforamnce under dynamic quantization: the scalar case. In *IEEE Conference on Decision and Control*, 2004.

[59] M.D. Lemmon, Q. Ling, and Y. Sun. Overload management in sensor-actuator networks used for spatially-distributed control systems. In *1st ACM Conferenced on Embedded Network Sensor Systems (SenSys03)*, 2003.

[60] M.D. Lemmon and R. Sun. Performance rate curves for dynamically quantized feedback systems. In *IEEE Conference on Decision and Control*, 2006.

[61] S. Li, S. Son, and J. Stankovic. Event detection services using data service middleware in distributed sensor networks. In *Proceedings of 2nd International Workshop on Information Processing in Sensor Networks*, 2003.

[62] D. Liberzon. On stabilization of linear systems with limited information. *IEEE Transactions on Automatic Control*, 48:304–307, 2003.

[63] Q. Ling and M.D. Lemmon. Robust performance of soft real-time networked control systems with data dropouts. In *IEEE Conference on Decision and Control*, 2002.

[64] Q. Ling and M.D. Lemmon. Optimal dropout compensation in networked control systems. In *IEEE Conference on Decision and Control*, 2003.

[65] Q. Ling and M.D. Lemmon. Soft real-time scheduling of networked control systems with dropouts governed by a markov chain. In *American Control Conference*, 2003.

[66] Q. Ling and M.D. Lemmon. Power spectral analysis of networked control systems with data dropouts. *IEEE Transactions on Automatic Control*, 49(6):955–959, 2004.

[67] Q. Ling and M.D. Lemmon. Optimal dynamic bit assignment in noise-free quantized linear control systems. In *IEEE Conference on Decision and Control*, 2005.

[68] Q. Ling and M.D. Lemmon. Stability of quantized control systems under dynamic bit assignment. *IEEE Transactions on Automatic Control*, 50(5):734–740, 2005.

[69] D. Liu, X. Hu, M.D. Lemmon, and Q. Ling. Scheduling tasks with markov-chain constraints. In *17th Euromicro Conference on Real-time Systems*, 2005.

[70] D. Liu, X. Hu, M.D. Lemmon, and Q. Ling. Firm real-time system scheduling based on a novel qos constraint. *IEEE Transactions on Computers*, 55(3):320–333, 2006.

[71] J.W.S. Liu. Real-time systems. 2000.

[72] S.H. Low and D.E. Lapsley. Optimization flow control, I: basic algorithm and convergence. *IEEE/ACM Transactions on Networking (TON)*, 7(6):861–874, 1999.

[73] C. Lu, B. Blum, T. Abdelzaher, J.A. Stankovic, and T. He. Rap: A real-time communication architecture for large-scale wireless sensor neworks. In *Proceedings of the 8th IEEE Real-time Technology and Applications Symposium*, 2002.

[74] C. Lu, J.A. Stankovic, S.H. Son, and G. Tao. Feedback control real-time scheduling: Framework, modeling and algorithms. *Real-time Systems*, 23(1-2):85–126, 2002.

[75] M. Marinaki and M. Papageorgiou. A non-linear optimal control approach to central sewer network flow control. *International Journal of Control*, 72(5):418–429, 1999.

[76] S. marti, T.J. Guili, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of ACM IEEE International Conference on Mobile Computing and Networking (Mobicom'00)*, 2000.

[77] L.W. Mays. *Stormwater Collection Systems Design Handbook*. McGraw Hill, 2001.

[78] B. Mochocki, X. Hu, R. Racu, and R. Ernst. Dynamic voltage scaling for the schedulability of jitter-constrained real-time embedded systems. In *Proceedings of International Conference on Computer Aided Design*, 2005.

[79] B. Mochocki, D. Rajan, X. Hu, C. Poellabauer, K. Otten, and T. Chantem. Network-aware dynamic voltage and frequency scaling. In *Proceedings of Real-Time and Embedded Technology and Applications Symposium*, 2007.

[80] P.E. Moffa. *Control and Treatment of Combined Sewer Overflows*. Van Nostrand Reinhold, 1990.

[81] A.K. Mok and X. Feng. Towards compositionality in real-time resource partitioning based on regularity bounds. In *Proceedings of the 22nd IEEE International Real-Time Systems Symposium*, 2001.

[82] A.K. Mok, X. Feng, and D. Chen. Resource parition for real-time systems. In *Proceedings of the 7th IEEE Real-time Technology and Applications Symposium (RTAS)*, 2001.

[83] L. Montestruque and M.D. Lemmon. Csonet: a metropolitan scale wireless sensor actuator network. In *Proceedings of the International Workshop on Mobile Device and Urban Sensing (MODUS)*, 2008.

[84] M. Di Natale and J.A. Stankovic. Dynamic end-to-end guarantees in distributed realtime systems. In *Proceedings of Real-Time Symposium*, 1994.

[85] D. Nesic and A.R. Teel. Input-output stability properties of networked control systems. *IEEE Transactions on Automatic Control*, 49:1650–1667, 2004.

[86] F. Paganini. A global stability result in network flow control. *Systems and Control Letters*, 46(3):165–172, 2002.

[87] DP Palomar and M. Chiang. Alternative Distributed Algorithms for Network Utility Maximization: Framework and Applications. *Automatic Control, IEEE Transactions on*, 52(12):2254–2269, 2007.

[88] A.K. Parek and R.G. Gallager. A generalized processor sharing approach to flow contorl in integrated services networks: The multiple node case. *IEEE/ACM Transactions on Networking*, 2:137–150, 1994.

[89] A.K. Parekh and R.G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: the single node case. *IEEE/ACM Transactions on Networking*, 1:344–357, 1993.

[90] M. Pleau, H. Colas, P. Lavalee, G. Pelletier, and R. Bonin. Global optimal control of the quebec urban drainage system. *Environmental Modelling and Software*, 20:401–413, 2005.

[91] J.W. Polderman and J.C. Willems. *Introduction to Mathematical Systems Theory: a behavioral approach*. Springer, New York, 1998.

[92] Y. Qiu and P. Marbach. Bandwidth allocation in ad hoc networks: a price-based approach. *INFOCOM 2003*, 2003.

[93] R. Racu, R. Ernst, A. Hamann, B. Mochocki, and X. Hu. Methods for power optimization in distributed embedded systems with real-time requirements. In *Proceedings of International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, 2006.

[94] P. Ramanathan. Overload management in real-time control applications using $(m, k)$-firm guarantee. *IEEE. Transactions on Parallel and Distributed Systems*, 10(6):549–599, 1999.

[95] K. Richter and R. Ernst. Event model interfaces for heterogeneous system analysis. In *Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE)*, 2002.

[96] T. Ruggaber, J. Talley, and L. Montestruque. Using embedded sensor networks to monitor, control, and reduce cso events: A pilot study. *Environmental Engineering Science*, 24(2):172–182, 2007.

[97] N. Schultz S. Heinz and D. Wre. Milwaukee case study in example evolution of sewer control. In *Proceedings of World Water and Environmental Resources Congress*, 2006.

[98] M. Saad, A. Leon-Garcia, and Y. Wei. Optimal network rate allocation under end-to-end quality-of-service requirements. *IEEE Transactions on Network and Service Management*, 4(3):40–49, 2007.

[99] A. Sahoo, W. Zhao, and W Jia. Partition-based admission control in heterogeneous networks for hard real-time connections. In *Proceedings of International Conference on Parallel and Distributed Computing*, 1997.

[100] C.U. Saraydar, N. Madayam, and D. Goodman. Pricing and power control in a multicell wireless network. *IEEE Jornal on Selected Areas in Communications*, TBD:1883–1898, 2001.

[101] C.U. Saraydar, N. Mandayam, and D. Goodman. Efficient power control via pricing in wireless data networks. *IEEE Transactions on Communications*, 50(2):291–303, 2002.

[102] L.A. Seders, C.A. Shea, M.D. Lemmon, P.A. Maurice, and J.W. Talley. Lakenet: an integrated sensor network for environmental sensing in lakes. *to appear in Environmental Engineering Science*, 24(2):183–191, 2007.

[103] D. Seto, J.P. Lehoczky, L. Sha, and K.G. Shin. On task schedulability in real-time control systems. In *IEEE Real-time Technology and Applications Symposium (RTAS)*, pages 13–21, 1996.

[104] I. Shin and I. Lee. Periodic resource model for compositional real-time guarantees. In *Proceedings of the 24th IEEE International Real-Time Systems Symposium (RTSS)*, pages 2–13, 2003.

[105] I. Shin and I. Lee. Compositional real-time scheduling framework. In *Proceedings of the 25th IEEE International Real-Time Systems Symposium (RTSS)*, pages 56–67, 2004.

[106] K. Shin and C. Meissner. Adaptation and graceful degradation of control system performance by task reallocation and period modification. In *EuroMicro Conference on Real-Tiem Systems (ECRTS)*, pages 29–36, 1999.

[107] V. Sivaraman, F.M. Chiussi, and M. Gerla. Deterministic end-to-end delay guarantees with rate controlled edf scheduling. *Performance Evaluation*, 63(4):509–519, 2005.

[108] V. Srivinasan, P. Nuggehalli, C.F. Chiasserini, and R.R. Rao. Cooperation in wireless ad hoc netowrks. In *Proceedings of IEEE INFOCOM*, 2003.

[109] J.A. Stankovic, T. He, T. Abdelzaher, M. Marley, G. Tao, S. Son, and C. Lu. Feedback control scheduling in distributed real-time systems. In *Proceedings of the 22nd IEEE Real Time Systems Symposium (RTSS)*, 2001.

[110] Y. Sun and M.D. Lemmon. Periodic communication logics for the decentralized control of multi-agent systems. In *IEEE Conference on Control Applications*, 2005.

[111] Y. Sun and M.D. Lemmon. Swarming under perfect consensus using integral action. In *American Control Conference*, 2007.

[112] M. Tabbara, D. Nesic, and A.R. Teel. Stability of wireless and wireline networked control systems. *IEEE Transactions on Automatic Control*, 52(9):1615–1630, 2007.

[113] S. Tatikonda and S.K. Mitter. Control under communication constraints. *IEEE Transactions on Automatic Control*, 49(7):1056–1068, 2004.

[114] L. Thiele, S. Chakrabory, and M. Naedele. Real-time calculus for scheduling hard real-time systems. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume 4, pages 101–104, 2000.

[115] P. Wan and M.D. Lemmon. Distributed flow control using embedded sensor-actator networks for the reduction of combined sewer overflow (cso) events. In *Proceedings of the IEEE Conference on Decision and Control*, 2007.

[116] P. Wan and M.D. Lemmon. Distributed network utility maximization using event-triggered augmented lagrangian methods. submitted to American Control Conference, 2009.

[117] P. Wan and M.D. Lemmon. Distributed network utility maximization using event-triggered barrier methods. submitted to the European Control Conference, 2009.

[118] P. Wan and M.D. Lemmon. Event-triggered distributed optimization in sensor networks, 2009.

[119] P. Wan and M.D. Lemmon. An event-triggered distributed primal-dual algorithm for network utility maximizatin. submitted to the IEEE Conference on Decision and Control, 2009.

[120] E. Wandeler, A. Maxiaguine, and L. Thiele. Quantitative characterization of event streams in analysis of hard real-time applications. *Real-time Systems*, 29:204–225, 2005.

[121] E. Wandeler and L. Thiele. Interface-based design of real-time systems with hierarchical scheduling. In *Proceedings of the 12th IEEE Real-time and Embedded Technology and Applications Symposium (RTAS)*, pages 243–252, 2006.

[122] X. Wang and M.D. Lemmon. Decentralized event-triggering broadcast over networked systems. In *Proceedings of Hybrid Systems: computation and control*, 2008.

[123] X. Wang and M.D. Lemmon. Event design in event-triggered feedback control systems. In *Proceedings of the IEEE Conference on Decision and Control*, 2008.

[124] X. Wang and M.D. Lemmon. Event-triggered broadcasting across distributed networked control systems. In *Proceedings of the American Control Conference*, 2008.

[125] X. Wang and M.D. Lemmon. Self-triggered feedback control systems with finite-gain l2 stability. to appear in the IEEE Transactions on Automatic Control, 2008.

[126] X. Wang and M.D. Lemmon. State based self-triggered feedback control systems with l2 stability. In *Proceedings of the 17th IFAC World Congress*, 2008.

[127] X. Wang and M.D. Lemmon. Event-triggering in distributed networked systems with data dropouts and delays. In *Proceedings of Hybrid Systems: computation and control*, 2009.

[128] X. Wang and M.D. Lemmon. Finite-gain l2 stability in distributed event-triggered networked control systems with data dropouts. submitted to the European Control Conference, 2009.

[129] X. Wang and M.D. Lemmon. Self-triggered feedback systems with state-independent disturbances. submitted to the American Control Conference, 2009.

[130] JT Wen and M. Arcak. A unifying passivity framework for network flow control. *Automatic Control, IEEE Transactions on*, 49(2):162–174, 2004.

[131] Y. Xue, B. Li, and K. Nahrstedt. Optimal resource allocation in wireless ad hoc networks: a price-based approach. *IEEE Transactions on Mobile Computing*, 5(4):347–364, 2006.

[132] H. Yaiche, R. Mazumdar, and C. Rosenberg. A game theoretic framework for bandwidth allocation and pricing of elastic connections in broadband networks: theory and algorithms. *IEEE/ACM Transactions on Networking*, 8(5):667–678, 2000.

[133] H. Zhang and D. Ferrari. Rate-controlled service disciplines. *Journal of High Speed Networks*, 3(4):389–412, 1994.

[134] S. Zhong, Y.R. Yang, and J. Chen. SPRITE: a simple, cheat-proof credit-based system for mobile ad hoc networks. In *Proceedings of IEEE INFOCOM*, 2003.

[135] K. Zhu, Y. Zhuang, and Y. Viniotis. Achieving end-to-end delay bounds by edf scheduling without traffic shaping. *IEEE INFOCOM*, pages 1493–1501, 2001.