

CSONet: A Metropolitan Scale Wireless Sensor-Actuator Network

Luis Montestruque
EmNet LLC
12441 Beckley St. # 6
Granger IN 46530
lmontest@heliosware.com

M.D. Lemmon
University of Notre Dame
Dept. of Electrical Eng.
Notre Dame, IN 46556
lemmon@nd.edu

Abstract—CSONet is a sensor actuator network that monitors and controls the frequency of combined sewer overflow (CSO) events in city sewer systems. CSO events constitute a major health and environmental hazard which numerous U.S. cities have been attempting to address. This paper describes an ongoing project building a metropolitan scale CSONet in the City of South Bend Indiana.

I. INTRODUCTION

Wireless sensor actuator networks or WSANs consist of computer controlled sensors and actuators that communicate over a wireless (usually RF) communication network. WSAN's use sensed data to power actuators which can then effect the sensed environment. The resulting changes in that environment can then be sensed by the network. This forms a distributed feedback loop that has the potential for efficiently controlling geographically distributed processes at a scale that was previously unthinkable.

WSANs are examples of so-called Cyber-Physical Systems (CPS). CPS are networks of computers that are tightly integrated into the real world for the purpose of both monitoring and controlling that world. CPS applications are found in automated highways, unmanned aerial vehicle swarms, traffic control, smart wastewater treatment plants, automatic chemical processes, energy efficient HVAC systems and power grid control. The hope is that CPS type systems will enable the cost effective control of such large-scale processes.

This paper describes a metropolitan scale (city wide) WSAN that is currently being built by a unique partnership of private (EmNet LLC), public (City of South Bend), and academic (University of Notre Dame and Purdue University) agencies. The WSAN is being built to control the frequency of combined-sewer overflow (CSO) events in a mid sized U.S. city (South Bend Indiana). The system is called CSONet. The problem addressed by CSONet represents a major public health and environmental issue faced by many U.S. cities. The South Bend CSONet will be completed by summer 2009 and if successful should serve as a model for a large-scale CPS system that can be adopted by other cities in the U.S.

The remainder of this paper is organized as follows. Section II discusses the CSO problem and identifies the particular issues relevant to the South Bend CSO system. Section III discusses the architecture of the WSAN to be used. Section

IV discusses CSONet hardware and middleware. The particular control application being implemented over CSONet is described in section V.

II. COMBINED-SEWER OVERFLOW PROBLEM

More than 700 cities in the U.S. have sewer systems that combine sanitary and storm water flows in the same system. During rain storms, wastewater flows can easily overload these combined sewer systems, thereby causing operators to dump the excess water into the nearest river or stream. The discharge is called a combined sewer overflow (CSO) event [1]. The discharged water is highly impacted with biological and chemical contaminants, thereby creating a major environmental and public health hazard. Under the provisions of the 1972 clean water act, the environmental protection agency (EPA) has begun fining municipalities for CSO events. These fines are substantial, sometimes running into the tens of millions of dollars. Municipalities have therefore begun looking for cost effective ways of reducing the frequency of CSO events.

The straightforward solution to the CSO problem is to enhance existing sewer infrastructure by separating storm and sanitary flows. Other solutions involve increasing the capacity of the wastewater treatment plant (WWTP) or building large off-line storage reservoirs. All of these options are extremely expensive and highly disruptive to the community.

Another solution uses the excess storage capacity in a city's sewer to temporarily store water during a storm. This option is referred to as *in-line storage*. The economical and reliable control of CSO events through in-line storage requires real-time monitoring and control.

Current approaches to real-time monitoring and control of sewer systems do not scale well. Sensor data is usually collected by a single computer over a Supervisory Control and Data Acquisition (SCADA) network. This computer determines the control action and distributes it back to the system through the SCADA network. It takes time to gather all of the sensor data and the delay introduced by gathering this data will also limit the rate at which control commands can be fed back to the system. Due to these delays, the control must be computed using complex simulation models of the sewer system. The entire control problem is therefore viewed as a large-scale nonlinear optimal control problem [2] [3] which can be addressed using linear quadratic approaches [4] or

model predictive control methods [5] [6]. These controllers are always implemented in a centralized fashion for very high-order plants. The system model is highly nonlinear with a great amount of uncertainty. As a result, centralized control of sewer systems tends to be complex, computationally intensive, and certainly is not robust to modeling error. All of these factors conspire to limit the scalability of centralized approaches to sewer flow control.

An alternative "distributed" approach to CSO control was presented by Ruggaber et al [7]. This case study used an embedded network of microprocessor controlled sensors and actuators to control CSO events. The network used a simple local feedback scheme to control a stretch of sewer system fed by a 1500 foot wide by 3.2 mile long corridor. In its first month of service the network prevented a 2 million gallon CSO event. The cost of the deployed network was around \$25,000, which was half of what it would have cost using existing SCADA network technologies.

The sensor-actuator network used by Ruggaber et al, therefore appeared to provide a cost-effective solution for controlling CSO events. The control used in that study was a simple switching law. A more sophisticated distributed control strategy was developed by Wan et al. [8]. This controller was a distributed control scheme that would be implemented on the WSAW used by Ruggaber et al. for a city the size of South Bend Indiana. Wan et al. were able to establish the optimality of their distributed scheme and through simulation studies were able to show that such an approach could reduce CSO overflows by 20 % over the existing passive strategies in use. On the basis of this early work, the control strategy developed by Wan et al. is being proposed for implementation on the metropolitan scale CSOnet for South Bend Indiana.

III. CSO NET ARCHITECTURE

CSOnet's architecture was designed to be a set of local WSAW's that connect to an existing wide area network (WAN) through gateway devices. CSOnet can therefore be viewed as a heterogeneous sensor-actuator network. It consists of four types of devices:

- Instrumentation Node or INode: these nodes are responsible for retrieving the measurement of a given environmental variable, processing that data and forwarding the data to the destination gateway through a radio transceiver.
- Relay Node or RNode: these nodes aid in forwarding data collected by INodes that are more than one-hop away from the gateway node. The RNodes only serve to enhance the connectivity in the wireless network.
- Gateway Node or GNode: these nodes serve as gateways between the WSAW used to gather data from the INodes and a Wide Area Network (WAN) which allows remote users easy access to CSOnet's data.
- Actuator Node or ANode: these nodes are connected to valves (actuators) that are used to hold back water in the sewer system.

Figure 1 shows the prototype CSOnet built by Ruggaber et al [7]. This network shows a single ANode (marked by

the "V") that receives its feedback sensor signals from three INodes (marked by the "I"). One of the INode's is located at the river to monitor actual CSO discharge into the river. The other INode's are used to measure the water level in a retention basin. The distance between the CSO outfall at the river and the retention basin is about 3 miles. Feedback information from the CSO outfall is forwarded over a line of RNodes (marked by the "R"). This figure can be taken as a very simple example of a single WSAW. This particular WSAW has been in continuous operation since 2005 and has been extremely useful in refining CSOnet's hardware and middleware components to ensure long-life and economical maintenance.

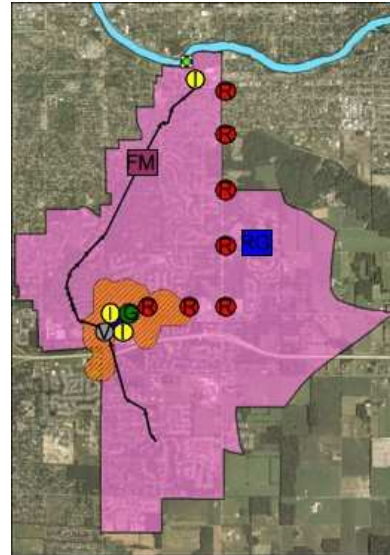


Fig. 1. Layout of Ruggaber's [7] prototype CSOnet at Ireland-Miami Road in South Bend Indiana (2005)

In order to scale CSOnet up to an entire metropolitan area, it was necessary to adopt a hierarchical architecture consisting of several WSAW's that are interconnected over an existing wide area network. One reason for this is the well-known inability of WSAW's to provide an acceptable quality of service when the network becomes too large. This is a consequence of the well-known theoretical limitations on wireless network throughput [9]. Empirical studies from DARPA's NEST program [10] have suggested that flat WSAW's should be limited to a diameter of 5-6 hops to prevent excessive congestion. If South Bend's CSOnet were to be built as a single flat WSAW covering the entire city, it would consist of several hundred nodes covering a 13,000 acre area. Such a deployment would require a prohibitive number of INodes and RNodes. For this reason, CSOnet consists of a set rather small WSAW's that forward their data to GNodes. The GNodes then forward the received packets to other WSAW's in the system.

To understand CSOnet's hierarchical structure, we first need to examine the actual sewer system to be controlled. Figure 2 shows a sewer system in which combined sewer trunk lines feed into a large *interceptor sewer*. Prior to 1972, municipal combined sewer lines were allowed to dump directly into

ivers and streams. Under the clean water act, cities were forced to treat the water from these combined sewer lines before they were released into a river or stream. One common way to meet this regulatory burden was to build an interceptor sewer along the river. This sewer would intercept the flow from the combined sewer trunk lines and convey that flow to a wastewater treatment plant (WWTP). Under dry weather conditions the flows were small enough to be handled by the WWTP. Under wet weather conditions (storms), the flows often overwhelmed the WWTP's capacity, thereby forcing operators to dump the excess directly into the water. As noted above such discharges constitute the CSO events described earlier.

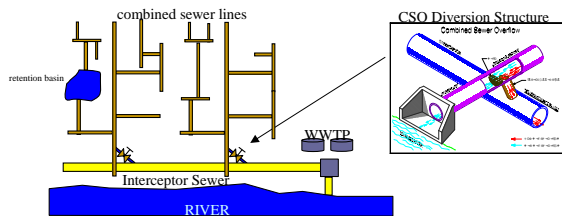


Fig. 2. South Bend Interceptor Sewer and CSO Diversion Structure

From figure 2 we can see that the combined sewer trunk lines and interceptor sewer connect at a *CSO diversion structure*. This is the point where we can apply control. The current system in South Bend Indiana uses a passive thresholding control. When the depth of the flow is below a fixed preset threshold, the flow is diverted into the interceptor sewer. Above this threshold, the flow is dumped out into the river. The problem is that these thresholds are set for the worst-case storm scenario. By placing a WSAN in the combined sewer trunk line above the CSO diversion structure, we can estimate the actual flows into the interceptor line and thereby make closed-loop control decisions that optimize the flow into the interceptor line such that WWTP capacity limits are never exceeded. This means that the natural place to place ANodes is at the CSO diversion points. These ANodes would then adjust the amount of water diverted into the interceptor sewer line based on an adaptive threshold that is a function of the current flows into the system. Because this scheme is adaptive, it need not be as conservative as the original passive thresholding scheme.

The scenario outlined above therefore indicates that CSOnet consists of a collection of WSAN's that forward flow measurements in a combined sewer trunk line to its associated CSO diversion structure. At this diversion structure would be a GNode and ANode. The ANode would adjust the flow into the interceptor line and the GNode would serve as a gateway between this particular WSAN and neighboring WSAN's up and down the interceptor line. Figure 3 illustrates this system architecture with 2 different WSAN's controlling the two diversion structures into the interceptor line. GNodes at these diversion structures and the WWTP are used to exchange control information in a way that allows coordinated flow

control across the city's entire sewer system.

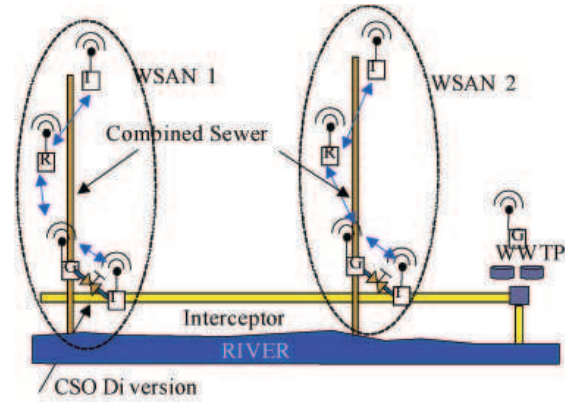


Fig. 3. CSOnet's Hierarchical Architecture

IV. CSOnet HARDWARE AND MIDDLEWARE

The actual hardware and middleware used to realize CSOnet are based, in part, on earlier experience with MICA2 sensor networks [11] [12] under DARPA's NEST program as well as more recent experience obtained in building the prototype CSOnet described by Ruggaber et al. [7]. This has led to a number of important design decisions whose robustness and reliability have been established on an empirical basis using Ruggaber's CSOnet prototype.

A. CSOnet Hardware

Many sensor networks developed by academia are based on some variation of U.C. Berkeley's MICA2 module [13]. The MICA2 processor module provided a low cost module integrating an easily programmed microprocessor with sensor interface that could communicate with other MICA2 modules through an embedded radio transceiver. While the MICA2 provided a convenient platform for experimental sensor networks, it was not sufficiently rugged for long-term industrial applications such as the CSOnet system.

The basic building block of CSOnet's WSAN is a more rugged version of the MICA2 processor module called the Chasqui wireless sensor node. The Chasqui node was developed by EmNet LLC to address a number of real-life issues that were found in platforms such as the MICA2. These issues concerned the limited radio range of the MICA2 and the need for specialized sensor-actuator interface that had sufficient power to drive commercial off the shelf environmental sensors.

EmNet's Chasqui node started with the original embedded node designs developed by U.C. Berkeley. EmNet enhanced the radio subsystem and sensor/actuator interface subsystems of this earlier design. The Chasqui node uses a 115 kbps MaxStream radio operating at 900 MHz. It uses frequency hopping spread spectrum (FHSS) signalling to reduce the radio's sensitivity to interference. The radio has a larger maximum transmission power (1 watt) than the earlier MICA2 processor module. This means the Chasqui node has a range

of over 700 meters in urban environments and up to a 5 km range for line-of-sight connections. The radio complies with FCC regulations for the use of license free ISM spectrum. The longer range of the Chasqui processor fits well with the distances required by the CSOnet application.

In spite of the higher transmission power required by the MaxStream module, careful design of the CSOnet middleware and hardware allows the WSAN's based on the Chasqui node to operate for several years before changing batteries. The Chasqui node consumes up to 5W when fully active and drops down to 0.14 mW in sleep mode. By using a precision real time clock, Chasqui nodes can coordinate their active and sleep cycles with sufficient precision to reliably function at a 2 % duty cycle. With such a duty cycle, the CSOnet applications based on the Chasqui processor node have a service life in excess of three years with a 4 cell lithium battery pack.

Another major feature of the Chasqui processor is its more rugged sensor-actuator interface subsystem. The Chasqui node uses a highly efficient switching power supply that generates 3.5 V for the microprocessor , 5V for the radio, and 12V for the sensors. The sensor node's interface also uses a MOSFET switch that allows the processor to completely switch off the sensor when not in use. This also helps minimize the module's energy usage which helps prolong the sensor node's service life.

CSOnet's INode and RNode are both based on the Chasqui processor module. The difference between these two devices is that the INode has a sensor management subsystem, whereas the RNode requires no such subsystem (since it is only used to forward data). INodes are typically located within the sewer system's manholes as shown in figure 4. The INode is attached to the manhole cover and is connected by cable to a pressure or flow sensor located within the sewer conduit. The INode transmits its sensor data out of the sewer manhole to an RNode that is usually located on a traffic or utility pole. Note that since conventional manhole covers are made of solid metal, radio waves have difficulty broadcasting out of the sewer system. Therefore as part of the CSOnet project our academic partners [14] at Purdue University have developed a composite fiber glass manhole cover with an embedded antenna that specially designed to broadcast efficiently out of the manhole.

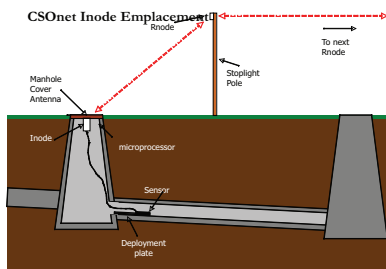


Fig. 4. Diagram showing the emplacement of INodes and RNodes in the city sewers.

GNodes and ANodes consist of a Chasqui processor module that is interfaced to a single board computer (SBC) running

linux. The SBC serves as a host computer that is connected directly to a wide area network (WAN) through either a hardwired ethernet connection, an 802.11 wireless card, or a cellular card. Data can then be transmitted between neighboring WSAN's over these gateways. For the CSOnet system shown in figure 3, we see that the GNode is located at the CSO diversion structure. So in this case the GNode can also use the actuator interfaces on the Chasqui processor module to actuate valves controlling water flows into the interceptor line.

B. CSOnet Middleware

Middleware is software that maintains a high level abstraction of the communication network that application software can use in a reliable manner. CSOnet's distributed control algorithm (introduced below in section V) requires a network abstraction that includes fast and reliable nearest neighbor communication as well as services supporting less frequent multicasts of control messages. Four middleware services had to be developed to help maintain this network abstraction. These were 1) a networking service used to construct reliable local routing tables, 2) a routing service for directing messages towards the gateway, 3) a synchronization service that maintains a time-slotted network abstraction, and 4) a power management service to help coordinate the network's waking and sleep cycles.

The underlying networking abstraction is a time-slotted network in which all nodes within the WSAN synchronize the times when they switch on (waking) and when they switch off (sleep). The synchronization of waking and sleeping periods must be very precise over extremely long periods of time. The heart of this is a clock synchronization middleware service similar to algorithms used in [15] that is triggered by synchronization beacons broadcast by the GNodes once per day. The GNodes themselves have their own clocks synchronized to the NIST timer server every six hours.

Since INodes and RNodes are usually emplaced in remote positions without access to external power, it is crucial that these devices be extremely miserly in their use of battery power. For CSOnet to be economically viable, these remote nodes need to have a service life of 2-3 years between battery changes. CSOnet achieves this long service life through a power management middleware service. CSOnet's power management service cycles all the nodes in a WSAN between waking and sleeping modes at a two percent duty cycle. Since synchronization beacons only occur once a day, the clock drift must be very small to ensure that all nodes are awake at the same time. Typical Real Time Clock (RTC) crystal tolerances are in the order of 15ppm at nominal temperature (25C) yielding drifts of up to 1.3 seconds per day at nominal temperature and up to 3 seconds on the extreme temperatures typically found outdoors. The Chasqui node implements a precision RTC (Dallas DS3231) with a typical drift of only 2ppm giving CSOnet tight synchronism between clock updates.

CSOnet must maintain a network abstraction that forwards sensor data to a gateway for subsequent rebroadcast to other

WSANs. CSOnet must therefore develop network and routing services enforcing this abstraction on a mesh radio network. Several approaches to the realization of mesh network communication protocols exist and have been researched by the sensor network community [16] [17] [18] [19] [20] [21]. These protocols typically assume a dense node population and good internodal connectivity. CSOnet radically differs from this paradigm. Nodes in CSOnet are sparse and have poor internodal reception. This is due to the extensive geographical area covered and the urban environment. A particular approach called Stateless Gradient-Based Persistent Routing was chosen due to its low computational requirements and robustness properties.

Stateless Gradient-Based Persistent Routing establishes routes from the source to the destination by imposing a gradient structure on the network (in a similar fashion to [22] [23]). Each node in the network has a gradient number that is an indication of how close the node is to the destination. Since there might be several destinations, each node stores one gradient number per destination in the network. A destination initiates the generation of the gradient number by sending out a beacon message. As the beacon message travels outward from the data destination point, nodes receiving the beacon generate their gradient based on the number of hops traveled by the beacon and their previous gradient number. The beacon message is transmitted using traditional flooding techniques.

When a node in the network desires to transmit a data message it appends its own gradient information corresponding to the destination and the destination ID. The message is then sent to all neighbors. The message will be forwarded only by those neighboring nodes with lower gradient number than the transmitter. In this way, messages travel down-gradient towards the destination. This method resembles the so called Directed Diffusion algorithm [23]. In order to increase reliability, anonymous acknowledgement messages are used for each forwarded message. If a forwarding node does not receive an acknowledgement that its message was heard by a lower gradient node, it will try to retransmit.

Since there is no explicit routing information generated, the computational complexity of the protocol is minimum as opposed to traditional Bellman-Ford or Dijkstras based approaches [24] [25]. Moreover, the network is inherently resilient to node failure as long as network remains connected with some link success probability. The number of retransmissions for lack of acknowledgement can be adjusted depending on those probabilities.

V. CSOnet's DISTRIBUTED CONTROL APPLICATION

A. WSAN's as Networked Control Systems

WSAN's add a new dimension to the traditional *sensor network* paradigm that has been of such interest in recent years. WSANs such as CSOnet are not content to simply monitor events. WSANs use sensor measurements to effect their external environment. Since such effects can again be measured by the sensors, this leads to a feedback interaction between the WSAN and the external world. In other words,

WSANs can act as distributed controllers for geographically distributed processes. We often refer to such systems as *Networked Control Systems* or NCS [26].

The feedback nature of a WSAN means that its performance is not well represented by its ability to communicate accurate information about the environment. Instead, the WSAN's performance is measured best by its ability to regulate that environment's behavior. Are both measures the same? No. Feedback systems have an inherent robustness to modeling error. This is why feedback is used. We measure the difference between what we expected and what actually happened in the environment and we use that difference to adjust the overall system's behavior. Feedback therefore reduces a system's sensitivity to unpredicted variations in sensor measurements and process models. What this means is that the communication requirements for a WSAN may be significantly less demanding than those required for a traditional sensor network.

What has recently become apparent is that high quality sensor data may not be needed to assure an acceptable level of NCS performance. This observation was first discussed by Brockett et al. [27] [28] with regard to his network of servo systems. Since that time a number of researchers have established bounds on dropped messages [29] [30], message delays [31] [32] [33] [34] [35], and quantization levels [36] [37] [38] [39] under which NCS can deliver acceptable levels of performance. These results indicate that the inherent requirements for NCS communication need not strive for the delivery of high-fidelity data. Instead, these results indicate that low-fidelity streams can be sufficient provided they respect the aforesaid bounds on delay and dropout rate.

B. Distributed Control of CSO Events

As noted above, CSOnet implements a control strategy that controls the amount of water diverted into the interceptor sewer from the combined sewer trunk line. Control actuation therefore occurs at the CSO diversion structure (see figure 2). This control action must maximize the total water diverted into the interceptor as this minimizes the total CSO discharge. This maximization, however, must be done subject to various safety constraints. First the flows in the interceptor line cannot exceed the capacity of the wastewater treatment plant. Second, the flows must not result in localized flooding or surcharge of the interceptor line.

Wan et al. cast this problem as an optimal control problem. To see how we might do this, we first note that the interceptor sewer in figure 2 can be abstracted to a straight line of $N + 1$ interconnected nodes forming a graph (figure 5).

We associate states with the links and nodes in figure 5's graph. The state of the link leaving the i th node is the flow Q_i (gallons per minute). The state of the i th node along the line is the head level (water level - feet) H_i for $i = 1, \dots, N$. These N nodes represent the manholes along the interceptor sewer. The $N + 1$ st node in the system is the WWTP, where its head level H_{N+1} is the ground level. Above each manhole node is a CSO diversion node. The flow entering this node is the external inflow w_i , the input from the old sewer lines (sanitary

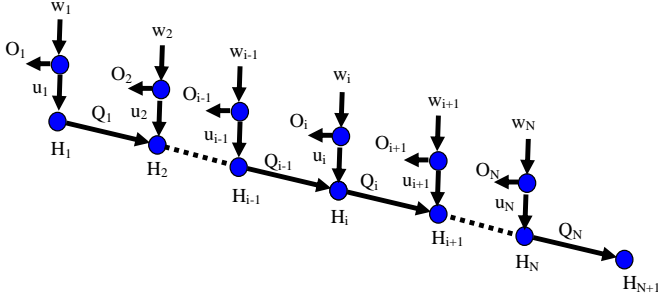


Fig. 5. Graph of South Bend Interceptor Sewer

water, rainfall, etc). The two flows leaving each CSO diversion node are O_i the overflow dumped into the river (overflow) and u_i the flow diverted into the i th manhole node from the i th CSO diversion node. This diverted flow, u_i , represents our control variable.

Our control problem seeks to minimize the total overflow from all CSO diversion nodes subject to state constraints on the nodes/links and a maximum flow limit for the entire network. Minimizing the total overflow is equal to maximizing the total diverted flow. Our problem therefore seeks to maximize

$$J(u_1, \dots, u_N) = \sum_{i=1}^N \int_0^{T_s} C_i u_i(\tau) d\tau \quad (1)$$

subject to

$$a_i(\tau) \frac{dH_i(\tau)}{dt} = u_i(\tau) + Q_{i-1}(\tau) - Q_i(\tau) \quad (2)$$

$$0 = H_i(\tau) - H_{i+1}(\tau) - k_i(\tau) Q_i^2(\tau) \quad (3)$$

$$0 \leq u_i(\tau) \leq w_i(\tau) \quad (4)$$

$$\bar{H}_i \geq H_i(\tau) \quad (5)$$

$$\bar{Q} \geq \sum_{j=1}^N u_j(\tau) \quad (6)$$

for $i = 1, \dots, N$ and $\tau \in [0, T_s)$ where $Q_0(\tau) = 0$ and $H_{N+1}(\tau) = 0$. In equation 1, C_i is a set of weighting coefficients (costs), and T_s is the horizon (storm duration) over which to maximize the diverted flow.

Equations 2-3 are abstracted from the equations characterizing open channel flows using the complete dynamic wave model [40]. In particular, equation 2 represents the conservation of mass. Equation 3 represents the conservation of momentum relation assuming flow rates are relatively slow varying.

The optimization is done subject to the control being admissible (equation 4 and 6) and state constraints in equation 5. The constant \bar{H}_i in equation 5 represents the head level above which the manhole begins flooding. The constant \bar{Q} represents a maximum flow limit for the entire network. This maximum flow limit may originate in limitations on the WWTP's capacity.

Wan et al. [8] showed that this problem could be solved using a switching control scheme. Moreover, it was shown that

switching decisions could be made in a distributed manner over a multi-hop network similar to that used by CSOnet. Assume that the node costs C_i are all distinct and can be ordered as $C_{i_{j+1}} < C_{i_j}$ for $j = 1, \dots, N_1$. At a given time instant t , let Ω_t represent those nodes who are flooding (i.e., $H_{i_j} \geq \bar{H}_{i_j}$). Each node i_j selects to divert the amount u_{i_j} . The selection falls into one of two cases.

- In the first case, i_j does not hit the head level constraint ($i_j \notin \Omega_t$). This node selects u_{i_j} according to the following equation

$$u_{i_j} = \begin{cases} w_{i_j} & \tilde{Q} - W \geq w_{i_j} \\ \tilde{Q} - W & 0 < \tilde{Q} - W < w_{i_j} \\ 0 & \tilde{Q} \geq W \end{cases} \quad (7)$$

where

$$\tilde{Q} = \bar{Q} - \sum_{i_j \in \Omega_t} (Q_{i_j} - Q_{i_{j-1}}), \quad W = \sum_{i_k \notin \Omega_t, k=1}^{j-1} u_{i_k} \quad (8)$$

\tilde{Q} represents the total free capacity for node i_j and W is the total used capacity for node i_j .

- If node i_j is flooding (i.e. $i_j \in \Omega_t$) then the control u_{i_j} is selected according to the feedback control law

$$u_{i_j} = K(H_{i_j}, H_{i_{j+1}}) \quad (9)$$

where K is a feedback controller mapping the head levels of the current node and the downstream node into the desired control action.

The workings of this control law are rather easy to explain. In the first case when flooding is not an issue (equation 7), the CSO diversion structure opens up the valve all the way. This is done first for those nodes with the highest cost C_i and continues until the total diverted flow begins to approach the capacity limit \bar{Q} . Once the total diverted flow reaches its limit, the remaining nodes with smaller costs, C_i , restrict their diverted flows to avoid exceeding the total flow limit, \bar{Q} . This type of "all-or-nothing" strategy is a common feature of many optimal controllers.

If node i_j is flooding, this node's objective changes from maximizing diverted flow to preventing flooding. In this case the feedback controller K_{i_j} (equation 9) is chosen to ensure that the current node's head level stabilizes at the maximum head level \bar{H}_{i_j} . The controller only needs feedback information from its immediate downstream node. This is an important feature of the controller for it means that it can be implemented in a distributed manner (i.e. feedback connections only take place between nearest neighbor nodes).

The algorithm was simulated on the real South Bend interceptor sewer line which consists of 36 CSO diversion structures. The following three different storm scenarios were considered. Each storm drops rain nonuniformly over the city and moving from west to east over city at 20mph.

- S1. 0.485 inch of rain in 11 hours
- S2. 0.799 inch of rain in 13 hours
- S3. 2.046 inch of rain in 19 hours

The simulation results (table I show that the proposed controller reduces total storm overflows by 24%–40% over existing fixed thresholding strategies. This reduction is significant.

Storm	existing overflow ($ft^3 \times 10^6$)	controlled overflow ($ft^3 \times 10^6$)	overflow decrease
S1	0.46	0.28	40%
S2	2.51	1.90	24%
S3	6.04	3.79	37%

TABLE I
OVERFLOW COMPARISON

VI. CONCLUSIONS

This paper gave a broad overview of the metropolitan scale sensor-actuator network that is currently being built in South Bend IN USA to reduce the frequency of CSO events. At present, the system consists of 150 wireless sensor nodes monitoring 111 locations in the South Bend sewer system are deployed and functional. Actuation nodes are scheduled to be completed in 2009. When fully implemented, CSOnet will control the diverted flows of at least 20 CSO diversion structures. If successful, South Bend's CSOnet will serve as a model that other U.S. municipalities can follow to address their own CSO problems in a cost effective manner.

REFERENCES

- [1] M. Schutze, A. Campisano, H. Colas, W. Schilling, and P. A. Vanrolleghem, "Real time control of urban wastewater systems; where do we stand today?" *Journal of Hydrology*, vol. 299, pp. 335–348, 2004.
- [2] M. Marinaki and M. Papageorgiou, "Central Flow Control in Sewer Networks," *Journal of Water Resources Planning and Management*, vol. 123, no. 5, pp. 274–283, 1997.
- [3] —, "A non-linear optimal control approach to central sewer network flow control," *International Journal of Control*, vol. 72, no. 5, pp. 418–429, 1999.
- [4] M. Marinaki, M. Papageorgiou, and A. Messmer, "Multivariable Regulator Approach to Sewer Network Flow Control," *Journal of Environmental Engineering*, vol. 125, no. 3, pp. 267–276, 1999.
- [5] S. Duchesne, A. Mailhot, and J. Villeneuve, "Global Predictive Real-Time Control of Sewers Allowing Surcharged Flows," *Journal of Environmental Engineering*, vol. 130, no. 5, pp. 526–534, 2004.
- [6] 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*, vol. 12, no. 1, pp. 1–9, 2004.
- [7] T. Ruggaber, J. Talley, and L. Montestruque, "Using embedded sensor networks to monitor, control, and reduce cso events: A pilot study," *Environmental Engineering Science*, vol. 24, no. 2, pp. 172–182, 2007.
- [8] P. Wan and M. Lemmon, "Distributed flow control using embedded sensor-actuator networks for the reduction of combined sewer overflow (cso) events," in *Proceedings of the IEEE Conference on Decision and Control*, 2007.
- [9] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, 2000.
- [10] A. A. et al., "Exscal: elements of an extreme scale wireless sensor network," in *Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*, 2005.
- [11] L. Seders, C. Shea, M. Lemmon, P. Maurice, and J. Talley, "Lakenet: an integrated sensor network for environmental sensing in lakes," *Environmental Engineering Science*, vol. 24, no. 2, pp. 183–191, 2007.
- [12] L. Fang, P. Antsaklis, L. Montestruque, B. McMickell, M. Lemmon, Y. Sun, H. Fang, I. Koutroulis, M. Haenggi, M. Xie, and X. Xie, "Design of a wireless dead reckoning pedestrian navigation system: the navmote experience," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 6, pp. 2342–2358, 2006.
- [13] J. Hill, "A software architecture supporting networked sensors," Master's thesis, U.C. Berkeley, Department of Electrical Eng. and Computer Science, 2000.
- [14] J. Mastarone and W. Chappell, "Urban sensor networking using thick slots in manhole covers," in *Proceedings of IEEE Antennas and Propagation Society International Symposium*, 2006, pp. 779–782.
- [15] S. Ganerwal, R. Kumar, and M. Srivastava, "Timing-sync protocol for sensor networks," in *ACM Conference on Embedded Networked Sensor Systems (SENSYS 2003)*, 2003.
- [16] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.
- [17] W. Henzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless sensor networks," in *Proceedings of 33rd Hawaii International Conference on System Science (HICSS00)*, 2000.
- [18] C. Intangaonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of ACM MobiCom*, 2000.
- [19] W. Henzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of 5th ACM/IEEE MobiCOM Conf.*, 1999.
- [20] A. Mangeshwar and D. Agarwal, "Apteen: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," in *Parallel Distributed Processing Symposium (IPDPS)*, 2002.
- [21] T. He, J. Stankovic, C. Lu, and T. Abedelzaher, "Speed: a stateless protocol for real-time communication in sensor networks," in *International Conference on Distributed Computing Systems*, 2003.
- [22] D. Braginsky and D. Estrin, "Rumor routing algorithm for sensor networks," in *Intl. Conf. Distributed Computing Syst. (ICDCS-22)*, 2001.
- [23] M. Maroti, "Directed flood routing framework," Technical Report ISIS-04-502, Institute for Software Integrated Systems, Vanderbilt University, 2004.
- [24] C. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999.
- [25] D. Johnson and D. Maltz, "Source routing in ad hoc wireless networks," in *Mobile Computing*, Ch. 5, pp. 153–181, 1996.
- [26] D. Hristu-Varakelis and W. Levine, *Handbook of Networked and Embedded Control Systems*. Birkhauser, 2005.
- [27] R. Brockett, "Stabilization of motor networks," in *Proceedings of IEEE Conference on Decision and Control*, 1995.
- [28] —, "Minimum attention control," in *Proceedings of IEEE Conference on Decision and Control*, 1997.
- [29] Q. Ling and M. Lemmon, "Power spectral analysis of networked control systems with data dropouts," *IEEE Transactions on Automatic Control*, vol. 49, no. 6, pp. 955–959, 2004.
- [30] B. H. V. Gupta and R. Murray, "Optimal lqg control across packet-dropping links," *Systems and Control Letters*, vol. 56, no. 6, pp. 439–445, 2007.
- [31] W. Zhang, M. Branicky, and S. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, pp. 84–99, 2001.
- [32] G. Walsh, H. Ye, and L. Bushnell, "Stability analysis of networked control systems," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 3, pp. 438–446, 2002.
- [33] D. Netic and A. Teel, "Input-output stability properties of networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1650–1667, 2004.
- [34] D. Carnevale, A. R. Teel, and D. Netic, "Further results on stability of networked control systems: a lyapunov approach," *IEEE Transactions on Automatic Control*, vol. 52, pp. 892–897, 2007.
- [35] X. Wang and M. Lemmon, "Self-triggered feedback control systems with finite gain l2 stability," to appear in *IEEE Transactions on Automatic Control*, 2008.
- [36] N. Elia and S. Mitter, "Stabilization of linear systems with limited information," *IEEE Transactions on Automatic Control*, vol. 46, no. 9, pp. 1384–1400, 2001.
- [37] D. Liberzon, "On stabilization of linear systems with limited information," *IEEE Transactions on Automatic Control*, vol. 48, pp. 304–307, 2003.
- [38] S. Tatikonda and S. Mitter, "Control under communication constraints," *IEEE Transactions on Automatic Control*, vol. 49, no. 7, pp. 1056–1068, 2004.
- [39] Q. Ling and M. Lemmon, "Stability of quantized control systems under dynamic bit assignment," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 734–740, 2005.
- [40] M. Chaudhry, *Open Channel Flow*. Prentice-Hall, 1993.