Evolutionary Algorithm Memory Enhancement for Real-Time CSO Control

Andrea Zimmer[1], Barbara Minsker[1], Arthur Schmidt[1], Avi Ostfeld[4],

[1]Civil and Environmental Engineering, University of Illinois at Urbana-Champaign,
[4]Civil and Environmental Engineering, Technion - Israel Institute of Technology

ABSTRACT

Control of combined sewer overflows (CSOs) may be enhanced through real-time decision support. An optimization algorithm adapted for changing rainfall is used to dynamically control complex sewer hydraulics to minimize CSO volume. Different methods of enhancing the optimization for real-time processing consist of: separating the hydraulic model for multi-objective optimization or to optimize only critical portions of the sewer system, using an efficient optimization technique, and incorporating memory into the optimization to speed convergence to a solution for each forecasted rainfall change. Potential optimal management solutions and the associated environmental characteristics can be stored and used to re-initialize the optimization at each environmental change. The memory may also be altered to indicate what precision of hydraulic model should be used for different rainfall conditions.

INTRODUCTION

Combined sewer overflows are most frequent during high intensity or long rainfall events. Combined sewage may back up into basements or overflow to the adjacent waterway when the wastewater storage and treatment system reaches capacity. If a deep tunnel (below sewer and interceptor grade) is incorporated into the system, potential overflows may be directed via sluice gates and dropshafts to the tunnel. An accurate hydraulic model of the combined sewer system is necessary to depict when and where CSOs might occur, but real-time sewer optimization necessitates fast computation (Schutze et al. 2004). Many models represent combined sewer systems for real-time operation (Duchense et al., 2001; Darsono and Labadie, 2007; Vanrolleghem et al., 2005), but have not yet incorporated deep tunnel hydraulics. The inclusion of a deep tunnel to mitigate CSO contamination necessitates the analysis of real-time optimization strategies for a combined sewer and tunnel wastewater system. The strategies will be analyzed under dynamically changing precipitation forecasts. Evolutionary optimization will be examined for its efficacy in reducing system CSOs.

BACKGROUND

Real-time decision support is based on several components: a simulation model of the hydrologic-hydraulic system, an optimization algorithm, and improvements to the optimization algorithms for real-time operation. Simulation of the combined sewer system includes watershed rainfall transformation, which is accomplished through the cell model introduced by Diskin et al. (1984). Interceptor and connecting tunnel hydraulics are represented with the Hydraulic Performance Graph, or HPG, (Yen and

Gonzales-Castro, 2000) which is a collection of backwater profiles created through the gradually varied flow equation (Chow, 1959). The HPG ensures conservation of momentum and is established offline for each tunnel or interceptor segment. During runtime, the HPGs allow an upstream water surface elevation to be interpolated based on a known flow rate and downstream water surface elevation for each reach (Oberg et al., 2008). Flow through the sluice gates to the deep tunnel dropshaft is based on mass and momentum conservation as established within the USGS FEQ model (Franz and Melching, 1997).

Techniques applied to the optimization of real-time systems include genetic algorithms (Holland, 1975; Goldberg, 1989), ant colony optimization (Dorigo et al., 1996), and approximate dynamic programming (Powell, 2007). Some of these approaches have been compared by Elbeltagi et al. (2005) for several optimization problems. Genetic Algorithms and Shuffled Frog Leaping are used to determine the optimal bridge maintenance scheme that will bring the highest return on the budget (Elbehairy et al., 2006); both methods appeared equally well suited to solve the problem when proper parameters were used. Maier et al. (2003) note that Ant Colony Optimization Algorithms (ACOAs) may be more efficient than GAs for dynamically changing systems.

Branke (1999) advocates the use of memory in optimization algorithms when the temporally changing objective function revisits previous states. For the combined sewer system studied, overflows will be simulated based on hydraulics resulting from precipitation. The rainfall patterns and the resulting hydraulics will likely resemble previous patterns, thus demonstrating a cyclic environment. Memory enhanced algorithms can be used to provide quicker convergence to management solutions. Approaches to memory include data-mining search space information at the previous state to generate new solutions (Koonce and Tsai, 2000) and using classifier-based memory (Barlow and Smith, 2008). Hu and Chen (2005) adapt a genetic algorithm to model predictive control by defining a control interval, shorter than the total optimization horizon, for which a genetic algorithm (GA) can be used to optimize decisions. At each interval, memory can be used to re-seed the population. Celeste et al. (2004) examine the use of a genetic algorithm using model predictive control for a water distribution system. A continuous optimization can also be used in which the population is injected with individuals from the memory at varying time intervals. Yang (2005) creates a hybrid approach combining both the random immigrants and memory methodologies. At each environmental change, the best solution in the memory is retrieved and used to guide the formation of randomly generated individuals. Yang (2006) improves the memory accessing process by developing an associative memory scheme which stores environmental information as well as the good solutions.

METHODS

A portion of the simulated combined sewer system is shown in Figure 1. Within each watershed, rainfall is collected and conveyed to the interceptor inlets through the cell model introduced by Diskin et al. (1984), and further utilized by Karnieli et al. (1994) and Ostfeld and Pries (2003). Rainfall is transformed through two linear reservoirs in series (Dooge, 1973) that represent each watershed.

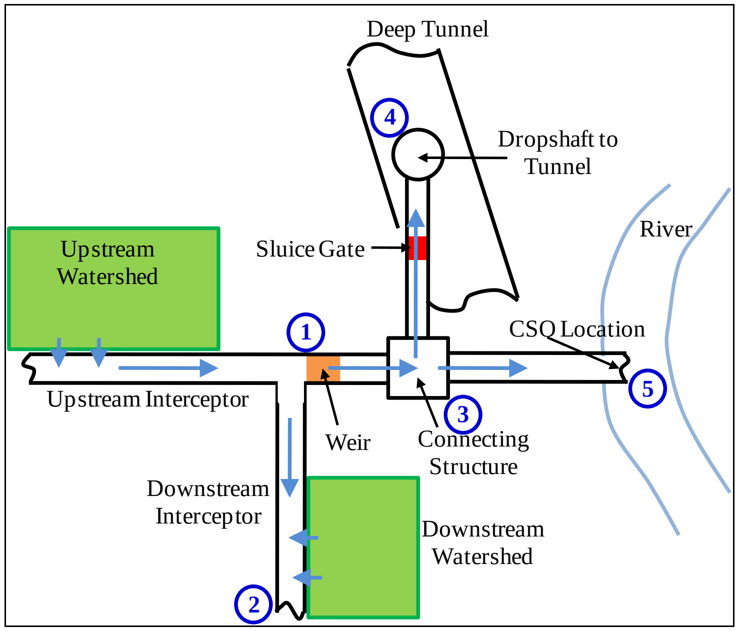

Figure 1

All water that enters the interceptor system is immediately transferred downstream (point 2 in Figure 1) and from there is routed back upstream to establish conduit water levels in accordance with work done by Oberg et al. (2008). At each time step, the HPG is used to find the upstream water surface elevation for each conduit given the flowrate and the downstream water surface elevation. The HPG files are created offline for a range of flow rates and downstream boundary conditions for the sub-conduits through tools scripted in *C++* (Oberg, 2008). If water elevations in the interceptors (at point 1) are high enough, flows from each interceptor towards the outflow structures are calculated by weir equations derived by Franz and Melching (1997) in the USGS FEQ model.

Water that flows from the interceptors towards the outflow location (point 5) is routed via HPGs through connections (point 3) and sluice gates (point 4). These structures partition the water to the deep tunnel or the outfall. The sluice gate equations are applied (Franz and Melching, 1997) and consist of mass and momentum balance for free-orifice, free-weir, submerged-orifice, or submerged-weir flow. The type of flow depends on the relative connecting structure and deep tunnel water surface elevations. If the sluice gate is closed, water will back up and overflow. Water that enters the deep tunnel is routed through HPGs to establish tunnel water surface elevations for the next time step.

For both the interceptor conduits and the deep tunnel, the stepwise steady HPG routing approach does not portray hydraulic transients that could cause overflows due to rapid gate closures, air entry through tunnel dropshafts, and other possible situations that could be identified within a more accurate physics-based (but computationally more demanding) model such as the ITM, or Illinois Transient Model (Leon et al., 2009). The analysis of model performance for historic rainfall events will reveal which type of model is needed for similar rainfall events during real-time calculation.

A simulation of the system is used as a first step towards testing several optimization approaches (genetic algorithms, ant colony optimization, and approximate dynamic programming). The genetic algorithm initializes a population of chromosomes composed of genes, and iteratively applies the operators of selection, crossover, and mutation to produce chromosomes of higher fitness. Chromosome genes represent percentages of sluice gate closures and pumping rates and are coded between 0 and 1 and rounded to one decimal place. Each gene can take one of 11 possible values. The real-coded genes necessitate high mutation rates (Tate and Smith, 1993; Wright, 1991) and a mutation rate of 0.05 is used. Non-uniform mutation (Herrera et al. 1998) and blended crossover, BLX-0.5, (Chen, 2003; Chang and Chen, 1998), and binary tournament selection are used in combination with the real coded GA. The maximum number of generations is 100; the GA seeks the chromosome with the best fitness, or lowest CSO volume.

Two approaches are compared for applying the control algorithm: continuous optimization and model predictive control (MPC). Continuous optimization consists of minimizing the CSO volume, *V*, over an entire storm event of duration T, by summing the volume for each time interval, as in Equation 1.

$$\text{minimize} \sum_{t=1}^{T} V(t)$$ **Equation 1**

The MPC formulation involves minimizing the predicted volume of overflow calculated at sequential 15-minute time increments Δt, to be evaluated out to time horizon $t + (\Delta t \times N)$ based on system information available at time *t* (Equation 2.)

$$\text{minimize} \sum_{k=1}^{N} V\left(t + (k \times \Delta t) \mid t\right)$$ **Equation 2**

The variable *t* indicates the current time, and *N* is the number of time intervals in the prediction horizon. The value of *N* is set to 8 to designate a 2-hour prediction horizon divided into 15-minute intervals. At a given time, the model would predict overflows in 15-minute intervals over the next two hours and identify control options to take at the beginning of each 15 minute time step to minimize those overflows. The volume depends on the values of the current and future control signals, which consist of the pumping rate and sluice gate positions and are designated by the vector *Y* in Equation 3.

$$\sum_{k=0}^{N-1} Y\left(t + (k \times \Delta t) \mid t\right)$$ **Equation 3**

The summation indicates the inclusion of all control signals within the prediction window. At time *t+1*, the control signal $Y(t)$ is implemented in the hydrologic-hydraulic model, and the computation is repeated with the prediction horizon (*k*) moving forward one interval. The interval *t* is advanced with *k* at Δt until no additional overflows are predicted within the window $t + (\Delta t \times N)$. The MPC algorithm implemented for this study is similar to the algorithms used by Onnen et al. (1997) and Naeem et al. (2005).

Branke (1999) details several ways of replacing solutions stored in algorithm memory; the replacement method used in this project will be to substitute the best individual in the evolving population for the most similar one in memory at any change in the environment. The number of solutions stored in evolutionary algorithm memory will initially be based on the methods used by Yan and Minsker (2006), in which a fixed memory size will be set, and old individuals will be discarded when the limit is reached. Different memory populations may come online for different seasons; rainfall patterns will be analyzed for the recurrence probability of similar events.

CASE STUDY

The combined hydraulic and optimization system is demonstrated using a small test case that is roughly based on a portion of the Chicago Tunnel and Reservoir Plan (TARP) system and consists of 11 sewersheds. A portion of the combined sewer system is shown in Figure 2, in which arrows indicate the direction of water flow. The system is comprised of six sewersheds on the south side of the river and four on the north side of the river.

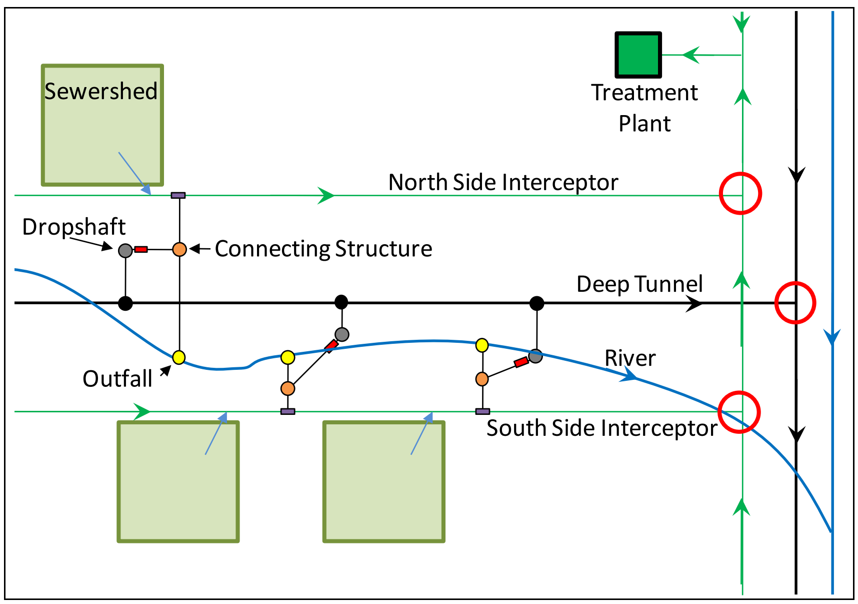

Figure 2

Two interceptors (on the south and north sides of the river) collect the runoff and flow to a main interceptor which conveys water north to a treatment plant. The deep tunnel flows under the river and directs water to a main tunnel which flows south. The interceptors are connected to the deep tunnel through weirs, connecting structures, sluice gates, and dropshafts as shown in Figure 2. Circles indicate the intersection between interceptors or deep tunnel portions; water elevations are defined at these boundary conditions. The effects of the treatment plant on the interceptors are portrayed as a reduction in the interceptor boundary condition elevation. Water in both interceptors usually flows downstream in the direction indicated by the arrows, but can also back up. If the water level at any point in the interceptor is greater than the weir elevation at that point, water will flow over the weir towards the connecting structure. An increase in elevation at the weir structure will occur if the flow rate is high enough and if the water elevation at the downstream end of the interceptor is high enough to allow the upstream water elevation to exceed the weir height. Once water flows over the weir, it flows into the connecting structure, through the sluice gate, and into the dropshaft to the deep tunnel. If the deep tunnel water elevation is high or the gate is closed sufficiently, water will accumulate in the connecting structure and overflow to the CSO location at the river. CSOs are most likely to occur when there is a large volume of water in the system and when the sluice gate is closed.

For this case study, the system of 11 dropshafts is simulated for a hypothetical 6 hour storm that allows sufficient flows to accumulate, and boundary conditions (upstream inflows and downstream water elevations) are set in order to allow the tunnel to potentially reach capacity during the rainfall event. Operations similar to current strategies are simulated. Table 1 shows the simulated rainfall for the system.

Time (min)	15	30	90	150	270	330	390
Rainfall Rate (cm/hr)	0	1.27	2.54	4.572	2.54	1.016	0

The simulation stops after 13 hours when all gate-instigated overflows have subsided. No changes in the rainfall are assumed to occur for the operations shown.

RESULTS AND DISCUSSION

The hydrologic-hydraulic system was run without optimization to simulate current operating policies for a storm duration of 12.5 hours. Figure 3 shows the outcome of a strategy in which all gates are left fully open until the tunnel reaches 70 percent of its diameter, at which point all gates are closed. A constant downstream boundary condition is maintained.

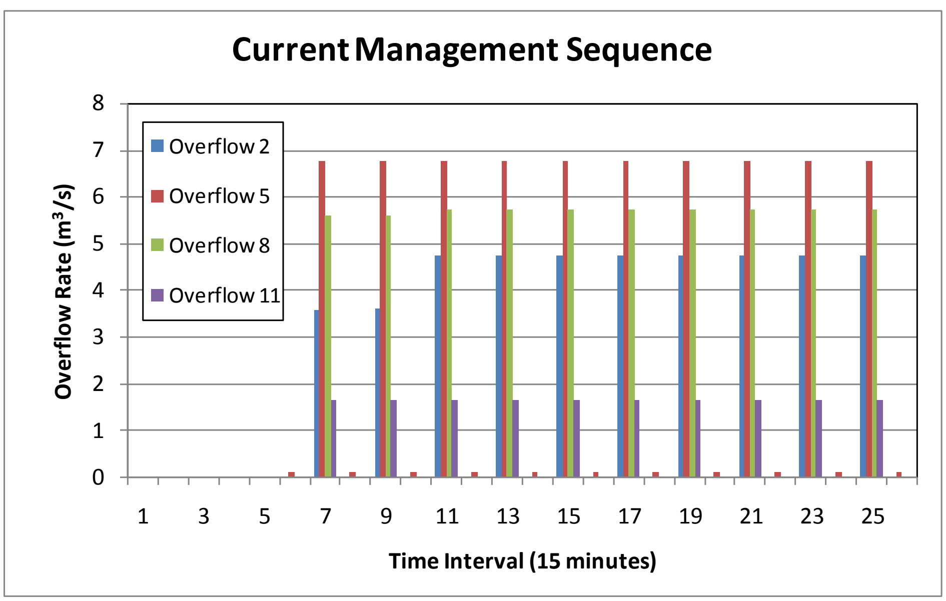

Figure 3

The CSOs start each time the gates are closed, at which point the tunnel level decreases to below 70% diameter. CSOs stop as the gates are opened again, allowing the tunnel to fill. CSO points 2 and 8 exhibit higher CSO volumes for this storm event than the locations 2 and 11, which may result from the larger drainage areas serviced by these outfalls. CSOs start one and a half hours into the storm, and the gate-instigated CSOs stop after seven hours.

CONCLUSIONS AND FUTURE WORK

A hydraulic model is constructed to simulate a combined interceptor and deep tunnel system and identify CSO occurrences for a given set of rainfall and gate conditions. Model simulations under conservative deep tunnel operation strategies show that benefits in CSO reduction can be achieved through optimization. Most CSOs occur due to a closed sluice gate, and changes in operations may be able to lessen the frequency with which gates are closed.

Future research will examine further reductions in CSO volume as well as in other objectives such as energy and contamination that can be achieved through multi-objective optimization. Several optimization techniques will be analyzed for their efficacy; a rainfall forecast that yields a changing optimum implies that these methods should be examined for continuous optimization. This includes hypermutation, or restarting the optimization, and the injection of solutions from memory. We will also examine which hydraulic models to use during real-time operation based on the intensity and placement of rainfall. Potential models include a decision tree, the stepwise steady HPG approach, and the ITM. Analysis will also be extended to determine if, as with the hydraulic models, the optimization techniques should be changed based on real-time rainfall patterns.

REFERENCES

Bhattacharya, B., and Solomatine, D. (2006). Machine learning in sedimentation modeling. *Neural Networks*, 19, 208-214.

Branke, J. (1999). Memory Enhanced Evolutionary Algorithms for Changing Optimization Problems. *Congress on Evolutionary Computation*, CEC99.

Chang, F., and Chen, L. (1998). Real-Coded Genetic Algorithm for Rule-Based Flood Control Reservoir Management. *Water Resources Management*, 12, 185-198.

Chen, L. (2003). Real Coded Genetic Algorithm Optimization of Long Term Reservoir Operation. *J. of the American Water Resources Association*, 39(5), 1157-1165.

Chow, V. T. Open Channel Hydraulics. McGraw-Hill, New York, 1959.

Darsono, S., and Labadie, J. (2007). Neural-optimal control algorithm for real-time regulation of in-line storage in combined sewer systems. *Environmental Modeling and Software*, 22, 1349-1361.

Diskin, M.H., G. Wyseure, and J. Feyen. (1984). Application of a Cell Model to the Bellebeek Watershed. *Nordic Hydrology*, 15, 25-38.

Dooge, J. (1973). Linear Theory of Hydrologic Systems. US Dep. Agric. Tech. Bull., 1468, 327.

Dorigo, M., Maniezzo, V., and Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Trans. Syst. Man. Cybern*., 26(1), 29-41.

Duchense, S., Mailhot, A., Dequidt, E., and Villeneuve, J. (2001). Mathematical modeling of sewers under surcharge for real time control of combined sewer overflows. *Urban Water*, 3, 241-252.

Elbeltagi, E., Hegazy, T., and Grierson, D. (2005). Comparison among five evolutionary-based optimization algorithms. *Adv. Eng. Infomatics*., 19, 43-53.

Elbehairy, H., Elbeltagi, E., Hegazy, T., and Soudki, K. (2006). Comparison of Two Evolutionary Algorithms for Optimization of Bridge Deck Repairs. *Computer-Aided Civil Infrastructure Eng.*, 21, 561-572.

Eusuff, M.M., and Lansey, K.E. (2003). Optimization of water distribution network design using the shuffled frog leaping algorithm. *J. Water Resourc. Plan. Manage.,* 129(3), 210-225.

Franz, D., and Melching, C. (1997). Full Equations (FEQ) model for the solution of the full, dynamic equations of motion for 1-D unsteady flow in open channels and through control structures. USGS Water-Resources Investigation Report 96-4240. Washington, D.C.: U.S. Geological Survey.

Goldberg, D.E. Genetic algorithms in search, optimization and machine learning. Reading, MA. Addison-Wesley Publishing Co. 1989.

Herrera, F., Lozano, M., and Verdegay, J. (1998). Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioral Analysis. *Artificial Intelligence Review*, 12(4), 265-319.

Karnieli, A.M., M.H. Diskin, and L.J. Lane. (1994). CELMOD5 – a semi-distributed cell model for conversion of rainfall into runoff in semi-arid watersheds. *Journal of Hydrology*, 157, 61-85.

Leon, A., Choi, N., Schmidt, A., and Garcia, M. (2009). The Illinois transient Model. V.Y. Chow Hydrosystems Lab, Dept. of Civil and Env. Eng., University of Illinois at Urbana Champaign.

Maier, H. R., Simpson, A., Zecchin, A., Foong, W., Phang, K., Seah, H., and Tan, C. (2003). Ant colony optimization for design of water distribution systems. *J. Water Resour. Plann. Manage.*, 129(3), 200-209.

Naeem, W., Sutton, R., Chudley, J., Dalgleish, F.R., and Tetlow, S. (2005). An online genetic algorithm based model predictive control autopilot design with experimental verification. *International Journal of Control*, 78(14), 1076-1090.

Oberg, N., A.S. Leon, A.R. Schmidt, and M.H. Garcia. Illinois Hydraulic Conveyance Analysis Program (ICAP) Version 1. University of Illinois at Urbana Champaign, June 2008.

Onnen, C., Babuska, R., Kaymak, U., Verbruggen, H.B., and Isermann, R. (1997). Genetic Algorithms for Optimization in Predictive Control. *Control Eng. Practice*, 5(10), 1363-1372.

Ostfeld, A., A. Pries. (2003). Lake Kinneret watershed contamination transports – a GIS based hydrological model. *Water Science and Technology*, 48(10), 63-70.

Powell, W. Approximate Dynamic Programming. John Wiley & Sons, Hoboken, New Jersey, 2007.

Schutze, M., Campisano, A., Colas, H., Schilling, W., and Vanrolleghem, P. (2004). Real time control of urban wastewater systems – where do we stand today? *J. of Hydrology*, 299, 335-348.

Tate, D.M., and E.A. Smith. (1993). Expected allele coverage and the role of mutation in genetic algorithms. In: Proc. 5th Int. Conf. Genetic Algorithms, S. Forrest, Ed. San Mateo, CA: Morgan Kaufmann, 1993. 31–37.

Vanrolleghem, P., Benedetti, L., and Meirlaen, J. (2005). Modelling and real-time control of the integrated urban wastewater system. *Environ. Modeling and Software*, 20(4), 427-442.

Wright, A. (1991). Genetic Algorithms for Real Parameter Optimization, in: G.J.E. Rawlins, *Foundations of Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA, 205-218.

Yan and Minsker (2006). Optimal groundwater remediation design using an Adaptive Neural Network Genetic Algorithm. *Water Resources Research*, 42.

Yang, S. (2005). Memory-Based Immigrants for Genetic Algorithms in Dynamic Environments. *GECCO 2005*, Washington DC.

Yang, S. (2006). Associative Memory Scheme for Genetic Algorithms in Dynamic Environments. In F. Rothlauf et al. (Eds.): EvoWorkshops 2006, LNCS 3907, 788-799, Springer-Verlag Berlin Heidelberg.

Yen, B. and Gonzalez-Castro, J. (2000). Open-channel capacity determination using hydraulic performance graph. *Journal of Hydraulic Engineering*, 126(2), 112–122.