

Nonlinear Programming Strategies for Source Detection of Municipal Water Networks

Bart G. van Bloemen Waanders, Roscoe A. Bartlett
Optimization and Uncertainty Quantification Department, Sandia National Laboratories *

Lorenz T. Biegler, Carl D. Laird
Chemical Engineering, Carnegie Mellon University

February 11, 2003

Abstract

Increasing concerns for the security of the national infrastructure have led to a growing need for improved management and control of municipal water networks. To deal with this issue, optimization offers a general and extremely effective method to identify (possibly harmful) disturbances, assess the current state of the network, and determine operating decisions that meet network requirements and lead to optimal performance. This paper details an optimization strategy for the identification of source disturbances in the network. Here we consider the source inversion problem modeled as a nonlinear programming problem. Dynamic behavior of municipal water networks is simulated using EPANET. This approach allows for a widely accepted, general purpose user interface. For the source inversion problem, flows and concentrations of the network will be reconciled and unknown sources will be determined at network nodes. Moreover, intrusive optimization and sensitivity analysis techniques are identified to assess the influence of various parameters and models in the network in a computational efficient manner. A number of numerical comparisons are made to demonstrate the effectiveness of various optimization approaches.

1 Introduction

Water distribution networks are highly vulnerable to chemical and biological terrorist attacks. Physical security can only be enforced to a limited extent, primarily confined to the external boundaries of a distribution system and perhaps exclusively applied to storage tanks and treating facilities. Other access points remain relatively unprotected such as fire hydrants or even households taps. Once this physical security has been breached, the only line of defense may be our ability to monitor disturbances associated with chemical concentrations and hydraulics. This assumes that in-situ sensors are available and capable of measuring important quantities such as hydraulic velocities and in particular chemical concentrations. Given these observations, our research effort considers the use of optimization algorithms to locate the original points of attack and help mitigate the effects of such an attack.

*Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed-Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

In addition, we investigate efficient algorithms to address inversion of large datasets. In an “all pipe model” of large cities not only will the forward simulation be computationally expensive but certainly an inversion process where all the nodes are possible inversion parameters may be intractable using traditional “black box” optimization methods. The primary source of the computational expense is the finite difference method to calculate the objective function gradient. To avoid these costly gradient calculations which require converging many forward simulations, intrusive methods can be considered. For example, sensitivity information could be formulated within the simulation code and be used to calculate the objective function gradient. The underlying assumption of intrusive methods is that certain linear objects can be accessed (such as the stiffness matrix). Chemical transport in state-of-the-art water distribution simulation codes, such as EPANET [9], use explicit solution schemes never allowing the access of such linear objects. This suggests the need to consider a reformulation so that an implicit solution scheme solves at least the chemical transport part of EPANET and makes the linear algebra suitable to intrusive methods.

In this report, we first review categorization of intrusive optimization algorithms and we demonstrate the computational efficiencies of each of these levels using a simple convection-diffusion system. Source inversion is applied to a water distribution model and several scenarios are investigated. Finally, we discuss a new formulation for handling the chemical transport calculation in a water distribution system that can take advantage of the computational efficiencies associated with intrusive methods.

2 Background

Relatively little work has been done on source inversion for water distribution systems. Most optimization studies have dealt with discrete variables requiring non-gradient based methods, including simplex methods [11], particle backtracking [10], feedback and auto-calibration methods [14]. All of these studies appear to be conducted on small datasets in serial mode. Inversion of large datasets with a large number of inversion parameters is an intractable problem for non-gradient based methods and we therefore concentrate on gradient based methods.

Gradient based methods can be classified into Nested Analysis and Design (NAND) and Simultaneous Analysis and Design (SAND) [1] [8]. Each classification can be further subdivided depending primarily on how the sensitivity information is calculated. In total, seven levels can be identified [12]. The NAND classification, which is also referred to as the black box approach, is limited to smaller design spaces but can be interfaced easily assuming very little about the internal workings of the simulation code. Difficult and large problems can be solved by utilizing powerful frameworks such as DAKOTA [5]. However, large design spaces as in the case of source inversion, require either very large computational resources or intrusive approaches. These methods have great potential for solving large problems but there are many assumptions associated with the application of SAND algorithms. Probably the most obvious disadvantage is the implementation cost necessary to equip PDE solvers with the necessary facilities to compute gradient information.

The seven categories are briefly described. *Level 0* is a NAND non-gradient approach where the optimizer does not require any information from the PDE code other than the objective function value per optimization iteration. *Level 1* is a NAND gradient-based approach where the optimizer requires the objective function gradient per optimization iteration. The gradient is typically calculated using a finite difference method. The interfacing cost for both levels 0 and 1 is minimal, often utilizing file system type communication. *Level 2, 3* are NAND gradient-based methods that use direct sensitivities and adjoints from the simulation code [6] [7]. *Level 4, 5* are SAND gradient-based methods dependent on direct and adjoint sensitivities, respectively

[3]. Instead of passing this information to a black-box optimizer, it is passed directly to algorithms closely coupled to the simulation. **Level 6** is known as the full-space method [4] and has the most computational potential for very large design spaces. This level of optimization is the most intrusive as it requires assembly and solution of a full Karush-Kuhn-Tucker (KKT) system. In the next section we present numerical results for a simple convection-diffusion source inversion problem in a single pipe.

3 Optimization for Source Inversion of a Convection Diffusion System

A comparison of execution time for source inversion of a convection-dispersion system is presented. The selection of this example was based on our ability to apply all seven levels of optimization to the same problem, implementation software considerations, and the relation to water distribution simulation. The primary goal of this numerical experiment is to determine the computational efficiencies of more intrusive methods.

By specifying a limited number of state values (concentrations) at various points in the domain as targets, a nonlinear least-squares formulation constrained by a convection-diffusion PDE is used to determine the location of the original sources on the boundary. From a forward simulation we obtained 16 “sensor” locations out of 1600 grid points as targets, which were used in the inversion problem. Since this is an ill-posed problem, a regularization term needs to be added to the objective function. Three obvious options can be considered: the square of f , the square of ∇f and finally the square root of ∇f . Unfortunately as a result of an implementation limitation, the boundary inversion can not make use of gradient based terms for the regularization and therefore the numerical experiments were conducted with the square of f . Our formulation allows locating the source term anywhere on one of the boundaries (i.e. Γ_F):

$$\min_{c,f} \quad \frac{1}{2} \sum_i^s \int_{\Omega} \delta(\mathbf{x} - \mathbf{x}_i) (c - c^*)^2 + \frac{\rho}{2} \int_{\Gamma_F} f^2 \quad (1)$$

$$\text{s.t.} \quad -k\Delta c + \nabla c \cdot v = 0, \text{ in } \Omega \quad (2)$$

$$\frac{\partial c}{\partial n} = 0, \text{ on } \Gamma_N \quad (3)$$

$$c = 0, \text{ on } \Gamma_D \quad (4)$$

$$c = f, \text{ on } \Gamma_F \quad (5)$$

where:

$$\begin{aligned} \begin{bmatrix} v_x \\ v_y \end{bmatrix} &= \begin{bmatrix} (L+y)(L-y) \\ 0 \end{bmatrix} \\ \Omega &= \{(x, y) : (-L \leq x \leq +L) \wedge (-L \leq y \leq +L)\} \\ \Gamma_N &= \{(x, y) : (-L \leq x \leq +L) \wedge (y = -L \vee y = +L)\} \\ \Gamma_D &= \{(x, y) : (x = +L) \wedge (-L \leq y \leq +L)\} \\ \Gamma_F &= \{(x, y) : (x = -L) \wedge (-L \leq y \leq +L)\} \end{aligned}$$

where $\delta(\mathbf{x} - \mathbf{x}_i)$ is a delta function that specifies the location of the sensors, c is the vector of calculated state value (concentrations), c^* is the vector of concentration measurements (or targets), ρ is the regularization

parameter which is set to $1E-5$ for our numerical experiments, f is the source/inversion term, k is the diffusivity constant, and v is the velocity field. The velocity field is given for this problem and makes (2) linear in c and therefore no further linearization is required. Figure 1(a) shows the forward simulation on a 40×40 grid (i.e. $n_x = 40$ and $n_y = 40$ finite elements per dimension) for a Gaussian-like source on the left boundary. The convection-diffusion PDE is discretized in space using a finite element Galerkin method.



Figure 1: (a) Forward Simulation (b) Inverse Solution, for 40×40 grid. The axes from left to right are c , y and x .

Figure 1(b) shows the solution for the inversion problem defined in (1)–(5) on a 40×40 grid. A Sequential Quadratic Programming (SQP) algorithm was able to successfully solve the problem and recover the entire profile. Small oscillations on the boundary are observed which may be reduced by choosing a different regularization term. Additional experiments were conducted to evaluate different terms but are beyond the scope of this article.

The source inversion problem was used to demonstrate the numerical efficiencies of the 7 optimization levels by inverting for the boundary source using different grid resolutions. For Levels 1-3 we used an SQP algorithm through the DAKOTA framework and for Levels 4 and 5 we used an intrusive reduced space SQP algorithm. Level 6 was not solved for the boundary inversion problem because of implementation limitations. Instead an inversion problem was solved using the full space method where inversion parameters are located within the domain. As the formulation in (5) suggests, the number of inversion parameters scales with the size of the boundary. Each optimization level was used to complete the inversion for a grid size of 10×10 , 20×20 , 40×40 , 80×80 , and a 160×160 grid. The number of inversion parameters matched the size of the grid dimension of a single sided boundary (10, 20, 40, 80 and 160 inversion parameters). The convergence criteria are controlled by various tolerances, but in our experiment we choose to match objective functions as closely as possible. Table 1 shows CPU times for various levels of optimization methods.

Figure 2 shows the numerical results, graphically. Level 0 used a local coordinate pattern search and it is the least efficient algorithm for this problem. These methods are obviously not preferred for smooth and differentiable processes but we have included the results for completeness. Level 1 shows a considerable improvement over Level 0 as a result of using gradient information. Direct sensitivities for both NAND and SAND show significant improvements over Level 1 because the reduced gradients for Level 1 are calculated through finite differences which requires the convergence of a simulation for each inversion parameter. Calculating reduced gradients with direct sensitivities avoids this numerical overhead. Additional separation between NAND and SAND methods using direct sensitivities can be expected if this had been a non-linear problem.

The adjoint sensitivities are by far the most efficient method to calculate the reduced gradient. There is

Method	$n_x = n_y = 10$	$n_x = n_y = 20$	$n_x = n_y = 40$	$n_x = n_y = 80$	$n_x = n_y = 160$
Sim	0.591	2.119	8.214	32.831	134.396
L-0 Inv	13974.8	31239.3	-	-	-
L-1 Inv	1278.63	1642.32	5385.14	27128.3	-
L-2 Inv	58.5	182.5	293.4	1840.8	22003.2
L-3 Inv	55.1	165.8	465.8	882.8	3620.4
L-4 Inv	9.47	17.32	55.87	835.65	13911
L-5 Inv	8.6	13.0	26.6	151.1	986.5

Table 1: Summary of CPU times (secs) on a boundary.

a significant difference between NAND and SAND because of the simulation overhead that NAND incurs at each optimization iteration. This difference is better observed in the right graph (Figure 2). One would expect that a non-linear simulation problem would incur additional Newton iterations which would add to the NAND expense for each optimization iteration and the gap between Levels 3 and 5 would be even greater. Estimated times for Level 6 are presented that equals three times the forward simulation cost. This is a conservative estimate considering the full space inversion of a 40x40 grid with 1600 inversion parameters required less than 10 seconds to converge.

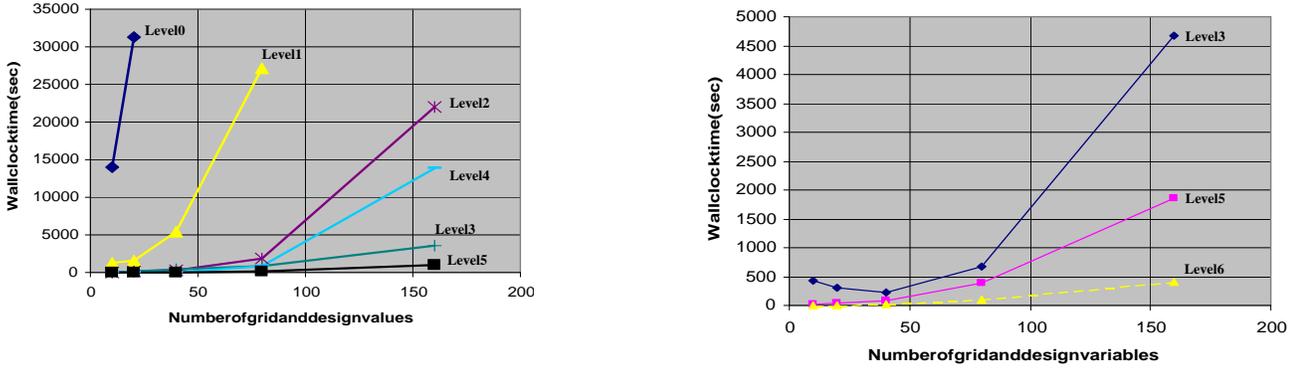


Figure 2: Numerical Results for Levels 0-5 (left) and Levels 3, 5 and 6 (right)

4 Optimization Interface For Water Distribution Systems

For the water distribution system, our initial objective is to test our source inversion algorithms and conduct numerical experiments to evaluate the sensitivity of various model parameters on the optimization algorithms. Our long term interest is to eventually interface water distribution simulation with intrusive optimization in an attempt to achieve the highest possible computational efficiencies for a source inversion problem. However, intrusive optimization algorithms need access to various linear objects in the simulation code that do not exist in the state-of-the-art water distribution models, such as EPANET. The solution methods of these models have evolved to efficient and stable techniques for the forward simulation mode. One of the more popular techniques, which is currently the preferred solution method in EPANET, is the

Lagrangian discretization and associated solution method. Although efficient, it is not obvious how it would allow intrusive solution methods to access implicit linear objects and to consider design, control, or inversion problems with a number of state and design parameters. The current implementation of the Lagrangian solution technique only allows a non-intrusive optimization interface. Before considering intrusive sensitivity calculations for higher optimization interface levels, we therefore investigate the performance of a NAND gradient based interface (Level 1). This level of optimization treats the simulation as a black box and requires no additional information from the simulator. In addition, we can make use of powerful optimization frameworks, such as DAKOTA, utilizing a variety of capabilities and course grained parallelism.

Our numerical experiments are based on the source inversion concepts in the previous section, with the EPANET model replacing the convection diffusion system. Our assumption is that an attack occurs at zero time at a nodal point in the network, and a finite number of sensors record concentration values at specified time intervals. Performance of our source inversion algorithm is evaluated both serially and in parallel. Other issues are investigated, such as source location, number of sensors, and finite difference step.

We start with a mathematical description of the non-linear least squares problem which is used to minimize the difference between the calculated and target concentrations:

$$\min_{f_j, j \in I \cup S} \frac{1}{2} \sum_{k \in K} \{ \sum_{i \in I} (c_i(0, t_k) - c_i^*(0, t_k))^2 + \sum_{s \in S} (c_s(t_k) - c_s^*(t_k))^2 \} + \frac{\beta}{2} \sum_{j \in I \cup S} f_j^2 \quad (6)$$

$$\text{s.t.} \quad \frac{\partial c_i(x, t)}{\partial t} + u_i \frac{\partial c_i(x, t)}{\partial x} - r(c_i) = 0 \quad i \in I \quad (7)$$

$$c_i(0, t) = \frac{\sum_{j \in I_k} Q_j c_j(L_j, t) + Q_{k, ext} c_{k, ext}(t)}{\sum_{j \in I_k} Q_j + Q_{k, ext}} + f_i \quad i \in I \quad (8)$$

$$\frac{\partial (V_s c_s(t))}{\partial t} = \sum_{i \in I_s} Q_i c_i(L_i, t) - \sum_{j \in O_s} Q_j c_s(t) + r(c_s) + f_s \quad s \in S \quad (9)$$

$$(10)$$

Given the sensor measurements c^* at times t_k , we need to determine the unknown source terms f_j that are specified as initial conditions for the concentration variables. Here the constraints are the primary chemical transport equations, I is the set of pipe segments, S is the set of source tanks, I_s is the set of pipes flowing into tank s , O_s is the set of pipes flowing out of tank s , $c_i(x, t)$ are the pipe concentrations at time t and position $x \in [0, L_i]$, $V_s, c_s(t)$ are the tank volumes and concentrations and $r(c)$ are reaction rates that occur in tanks or pipes. The external flowrates Q_{ext} and concentrations $c_{ext}(t)$ are specified and the velocity u_i and flowrates, Q_i are obtained by first solving the hydraulic equations over the time period. For an inversion problem in one spatial dimension, recovering small scale features at the source is not needed and the requirement for regularization is not anticipated. The term β is therefore set to zero.

4.1 Numerical results

We describe several results to establish the performance of source inversion for a water distribution system. A dataset from a local industrial area (LIA) was used to perform numerical tests. Figure 3(a) shows the general outline of the network and Table 2 describes some of the details. Figure 3(b) shows the scenario for the first experiment, in which the point of attack is at the main feeder tank. Approximately one fourth of the entire top half of the grid is contaminated (colored red) after 15 hours. Only the top portion of the total network is shown because the spread of the chemical is limited to that area.

With an attack initiated from the main feeder tank (i.e., all $f_i, i \in I$ are zero and all but one $f_s, s \in S$ are zero) all of the nodal concentrations are calculated from a forward simulation in EPANET and a subset

Item	Description	Item	Description
Number of Junctions	466	Maximum Trials	40
Number of Tanks	4	Quality Analysis	Chemical
Number of Pipes	621	Total Duration	15.00 hrs
Number of Pumps	3	Water Quality Time Step	5.00 min
Number of Valves	12	Water Quality Tolerance	0.01 mg/L
Headloss Formula	Hazen-Williams	Specific Gravity	1.00
Hydraulic Timestep	1.00 hrs	Relative Kinematic Viscosity	0.98
Hydraulic Accuracy	0.001	relative Chemical Diffusivity	1.00

Table 2: Operating conditions for the LIA dataset

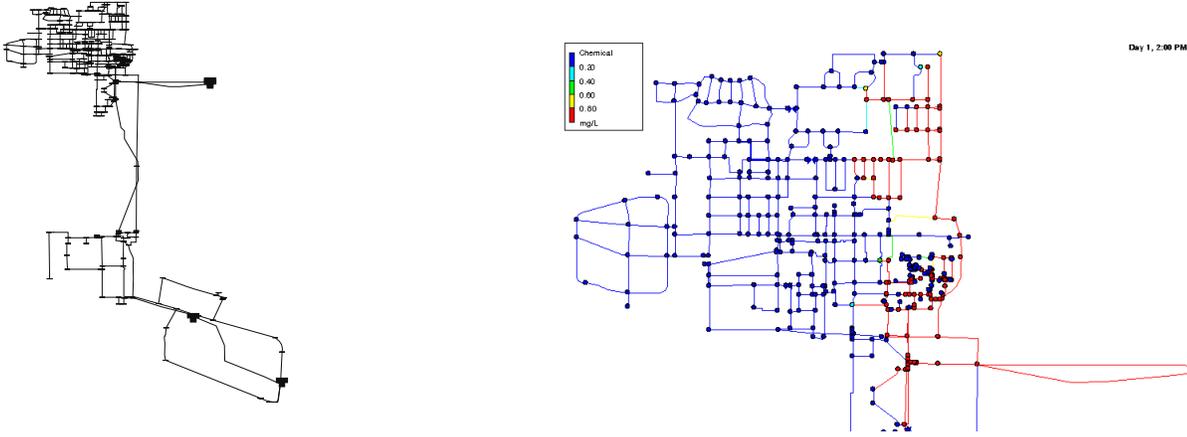


Figure 3: (a) LIA network (b) Attack Scenario (top half of LIA shown)

of these was measured to sense the original location of the attack. An SQP algorithm was able to find the location and magnitude of the source with two gradient calculations and 14 function evaluations, for a total of 954 function evaluations. For this attack scenario, a relatively small number of sensors are actually needed. The accuracy of the solution and the efficiency of the algorithm were not affected by reducing the number of sensors from 470 to 31 sensors.

At each iteration of the Level 1 optimization, the EPANET model is solved and a post-processor routine evaluates the nonlinear least squares objective function by summing over the sensor locations, $j \in I \cup S$. To test for a smaller number of sensors in the nonlinear least squares calculation, we selected summation elements of $I \cup S$ that were separated uniformly by a specified number of elements. These results are shown in Table 3. With this selection criterion, an incorrect source inversion was obtained only when fewer than 31 sensors were selected. On the other hand, a strategic placement of sensors, such as in pipes where the volumetric flow rates are high, may allow the use of even fewer sensors. Optimal sensor placement is currently being investigated [2].

Optimization of a 470 node and 470 parameter inversion model using a NAND gradient based method took 1544 secs on a single processor (Pentium 3, 500 Mhz). The majority of the work is in the calculation of the objective gradient through a finite difference method which is embarrassingly parallel. If sufficient computer resources are available and the overall serial work is minimal, substantial speedups should be possible through

Run No.	Number of Sensors	Error of source	Optimization Iter	Wall clock (sec)
1	470	0	5	1051
2	235	0	2	857
3	117	5.0E-3	5	1137
4	78	5.1E-1	5	1141
5	58	4.3E-2	7	1582
6	47	0	2	420
7	39	4E-2	6	1296
8	10	0.41	7	1491

Table 3: Inversion Numerical Results DAKOTA/EPANET

a parallel implementation. Experiments showed excellent scalability up to 4 processors (1 proc - 1544 secs, 2 procs - 854 secs, 4 procs - 401 secs). However, significant serial work is associated with Level 1 optimization in handling globalization through a line search method. Although the gradient calculations can be done in parallel, a single function evaluation is executed serially at the end of each optimization iteration, in addition to other serial tasks such as managing parallel processing and pre/post-processing. Even though additional experiments are necessary to quantify the serial overhead for large numbers of processors, we do not expect good parallel speedup with this approach.

The performance of the algorithm is also affected by changing the location of the source to an internal node. Additional optimization iterations are needed to invert for the source in this new attack scenario. The gradient calculation in the first iteration identifies the main feeder tank as a possible source location with some gradient, in addition to the internal node where the source is actually located. Eventually the algorithm finds the correct solution but requires more optimization iterations, gradient calculations and function evaluations.

For the internal node attack scenario the usage of different sensor locations produces somewhat unpredictable results (Table 3). It appears that the influence of the main feeder locations on the overall optimization algorithm is relatively large and affects the convergence efficiency. The most likely reason for the unpredictable behavior is inaccurate gradients. A relatively large finite difference step of 10% is required to produce a change in EPANET. In an attempt to calculate more accurate gradients, smaller finite difference steps were used but the smaller perturbations did not change the performance of EPANET. Although inversion for a water distribution network is possible with EPANET in its current form and with Level 1 optimization, difficult attack scenarios will be difficult to resolve and the convergence rates will be very poor. Exact gradient calculation through direct or adjoint sensitivities will greatly improve the performance of the optimization algorithm.

The Level 1 optimization interface has demonstrated an inversion capability and identified several issues associated with source inversion for large datasets. Even though parallel speedup appears capable of reducing the execution time for small numbers of processors, parallel speedup will be limited as a result of significant serial overhead associated with Level 1 optimization methods. Even if this serial overhead could be reduced, an all-pipe model for large cities will require a computer resource too large to be practical. The different behavior of the optimization algorithm caused by the more difficult inversion scenario, suggests the need for careful investigation of numerics associated with EPANET in the context of optimization. Both these factors potentially justify the use of a formulation for the chemical transport that would lend itself to the calculation of exact and more efficient intrusive inversion algorithms.

5 DAE and Adjoint Formulation

To develop a higher level optimization method intrusive sensitivities are needed, and a rewrite of at least the chemical transport formulation needs to be considered. Within a hydraulic time step, the chemical transport can be formulated as a DAE system, and by utilizing existing time integrators which also have sensitivity calculation capabilities such as CVODE or DASSPK [6], modifications can be made within the existing EPANET framework without a tremendous effort. This should allow not only more efficient gradient calculations but also more accurate calculations. We briefly describe the proposed formulation.

Equations (7), (8), (9) can be written as a general differential algebraic equation:

$$F(p, Y, \dot{Y}, t) = 0 \quad (11)$$

where $Y = [c, V_s c_s]^T$, $\dot{Y} = [\dot{c}, V_s \dot{c}_s]^T$, and p is a vector of inversion parameters. Given an objective function given by $G(p, Y) = \int_0^T g(p, Y, t) dt$ we are interested in finding $\frac{dG}{dp}$. In this form, an adjoint formulation can be derived as follows [13]:

$$F_{\dot{Y}}^T \dot{\lambda} + \left[\frac{\partial F_{\dot{Y}}^T}{\partial t} - F_Y^T \right] \lambda = -g_Y^T \quad (12)$$

$$(F_{\dot{Y}})_{t=T}^T \lambda = 0 \quad (13)$$

and the following equation for $\frac{dG}{dp}$ can be obtained:

$$\frac{dG}{dp} = \int_0^T (g_p - \lambda^T F_p) dt + (\lambda^T F_{\dot{Y}} Y_p)_{t=0} \quad (14)$$

As Figure 2 indicates, the adjoint approach can significantly improve the computational efficiencies and the modification to the current Lagrangian solution technique for the forward solution may be warranted.

6 Conclusions and Future Work

This initial investigation demonstrates the use of optimization techniques to locate potential chemical/biological attacks given a concentration and velocity profile for a time history. Performance measures for a source inversion on a prototype convection-diffusion system established computational advantages to intrusive methods. A relatively small number of sensors are needed to invert for the source location in a water distribution model; even fewer sensors may be allowed if they are placed strategically. The performance of the optimization algorithm and quality of the solution of the inversion problem is affected by source location. Parallel scalability was demonstrated in the Level 1 mode for the source inversion problem for 4 processors. Several shortcomings associated with the numerics of Level 1 optimization for a water network have been identified that suggest reformulating the current Lagrangian solution technique with an implicit DAE formulation.

References

- [1] R.J. Balling and J. Sobieszczanski-Sobieski. Optimization of couple systems: A critical overview of approaches. Technical Report 94-100, NASA ICASE Tech. Rep 94-100, Institute for Computer Applications in Science and Engineering, 1994.
- [2] Jonathan Berry, Lisa Fleischer, William Hart, and Cynthia Phillips. Optimal sensor placement in municipal water networks. In *Proceedings of EWRI*, Philadelphia, PA, June 22-26 2003.
- [3] L. T. Biegler, J. Nocedal, and C. Schmid. A reduced hessian method for large-scale constrained optimization. *SIAM J. Opt.*, 5:314, 1995.
- [4] G. Biros and O. Ghattas. Parallel preconditioners for KKT systems arising in optimal control of viscous incompressible flows. In *Proceedings of Parallel CFD '99, Williamsburg, VA, May 23-26, 1999*, Amsterdam, London, New-York, 1999. North Holland. <http://www.cs.cmu.edu/~oghattas/>.
- [5] M.S. Eldred, A.A. Giunta, B.G. van Bloemen Waanders, S.F. Wojtkiewicz, W.E. Hart, and M.P. Alleva. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis. version 3.0 users manual. Technical report SAND2001-3796, Sandia National Laboratories, Albuquerque, New Mexico 87185 and Livermore, California 94550, April 2001.
- [6] Shengtai Li and Linda Petzold. Software and algorithms for sensitivity analysis of large-scale differential algebraic systems. *Journal of Computational and Applied Mathematics*, 2000.
- [7] S. K. Nadarajah and A. Jameson J. Alonso. An adjoint method for the calculation of remote sensitivities in supersonic flow. In *AIAA-2002-0261*. AIAA, 2002.
- [8] P.A. Newman, G.J. WHou, and A.C. Taylor. Observations regarding use of advanced analysis, sensitivity analysis, and design codes in cfd. Technical Report 96-16, NASA ICASE, Institute for Computer Applications in Science and Engineering, 1996.
- [9] L.W. Rossman. Epanet 2. Technical report, EPA, 2000.
- [10] F. Shang, J. Uber, and M. Polycarpou. A particle backtracking algorithm for water distribution systems analysis. *A.S.C.E. Journal of Environmental Eng*, 2002.
- [11] M. Tryby, D. Boccelli, J. Uber, and L. Rossman. A facility location model for booster disinfection of water supply networks. *A.S.C.E. Journal of Water Resources Planning and Manag.*, 2002.
- [12] B.G van Bloemen Waanders, R.A. Bartlett, K.R. Long, P.T. Boggs, and A.G. Salinger. Large scale non-linear programming for pde constrained optimization. Technical report, Sandia National Laboroatories, 2002.
- [13] Y.Cao, S.Li, L. Petzold, and R.Serban. Adjoint sensitivity analysis for differential algebraic equations: Part i, the adjoint dae system. 2000.
- [14] M.L. Zierolf, M.M. Polycarpou, and J.G. Uber. Development and auto-calibration of an input-output model of chlorine transport in drinking water distribution systems. In *I.E.E.E. Trans. on Control Systems Technology*.