# Application of Genetic Immune Algorithm on Solving Disinfection Problem of Water Distribution Network System

Chien-Wei Chu, Shuang-Fu Yeh, and Min-Der Lin*

***Abstract*—This paper presents the methodology and application of genetic immune algorithms (GIA) on solving optimal scheduling of booster disinfection problems in a water distribution network system (WDNS). The objective of this study is to initiate a total chlorination dose to satisfy the minimum and maximum required chlorine residual at every demand node in a WDNS while minimizing the chlorine consumption as much as possible. The performance of GIA is evaluated using a well-known benchmark WDNS booster disinfection case study. The comparison results indicate that GIA is able to find solutions comparable to the best solutions published in the literature.**

***Index Terms*—immune algorithm; genetic algorithm; genetic immune algorithm; water distribution network system; booster disinfection.**

## I. Introduction

A water distribution network system (WDNS) is an important composition of a water supply system in any city and it carries out the task of transporting drinking water to the consumers. However, the quality of drinking water in a WDNS usually becomes worse or at risk for disease because of the long transport time which can bring about the process of physical, chemical, and microorganism change in the water. For this reason, drinking water must be dealt with a supply source by water purification plants. Chlorination disinfection is the most popular treatment technology because residual chlorine concentrations can be maintained in a WDNS to restrain microbes. The conventional management practice for residual chlorine maintenance is to add large quantities of chlorine at the source to guarantee a sufficient residual throughout the remote parts of a WDNS. Nevertheless, the high residual chlorine may not only cause the consumers to complain about the taste and odor, but also result in the formation of disinfectant by-products (DBPs) which may have harmful effects on human health [1]-[2]. Since the chlorine concentration must be avoided overdose, the chlorine concentration should be increased either at the source or at some critical locations, e.g., boosters, in the network. Booster chlorination, a strategy where disinfectant is reapplied within the network, make use of scheduling in space and time, then assure more uniform disinfectant residual and less risk of DBPs formation throughout the WDNS.

Over the past decades, few attempts have been made to study booster disinfection. Boccelli *et al.* (1998) were the first to solve a linear programming problem which optimizes the locations and operation schedules of booster disinfection stations [3]. Tryby *et al.* (2002) extended the above model as a mixed-integer linear programming problem that the number of booster stations becomes a decision variable [4]. Propato and Uber (2004) employed a least-squares technique to solve the optimal injection rates of disinfection boosters to minimize the space-time variation of the residual disinfectant distributed in the system [5]. Probabilistic heuristic algorithms, such as genetic algorithm (GA), have recently been utilized for optimal booster disinfection problems. Several studies have adopted a GA approach with a focus on minimizing the chlorine injection scheduling while maintaining a bounded chlorine concentration at the consumer node [6]-[9]. Another effective heuristic algorithm in simulating biological processes is the immune algorithm (IA), a relatively new optimization algorithm that imitates the immune system defending against invaders in a biological body [10]-[13]. IA has been successfully applied to various benchmarking and real-world optimization problems including allocation problems, synchronous motors with parameter correction, VLSI floor plan design problems, and structural optimization [14]-[17]. IA also has successfully solved scheduling of booster disinfection optimization problem and the evaluation results confirm the efficacy of IA [18]. Therefore, this study combines GA with IA into a genetic immune algorithm (GIA), which employs GA to briefly screen initial antibody repertoires for IA, to enhance IA efficacy for solving the optimal scheduling of booster disinfection problems. A benchmark case study described by [3] is evaluated to validate the effectiveness and efficiency of the proposed GIA.

## II. Optimization Model Formulation

A WDNS quality model simulates how the concentration of chlorine decay varies with time throughout the network under a known set of hydraulic conditions and source input patterns.

The principles of conservation of mass and appropriate reaction kinetics for the constituent being modeled are used in formulating the WDNS quality model. The water flow of WDNS is assumed as one dimension incompressible liquid. Water quality of constituents transported along the *i*th pipe is given by the following classical advection equation:

$$\frac{\partial C_i}{\partial t} = -u_i \frac{\partial C_i}{\partial x} + r(C_i) \qquad (1)$$

C-W. Chu is assistant research fellow with Department of Environmental Engineering, National Chung Hsing University, Taiwan, R.O.C (e-mail: yoyopig@sys.ev.nchu.edu.tw).

S-F. Yeh is Ph.D. candidate with Department of Environmental Engineering, National Chung Hsing University, Taiwan, R.O.C. (e-mail: d2117@tccg.gov.tw ).

*Corresponding author: M. D. Lin is professor with Department of Environmental Engineering, National Chung Hsing University, Taiwan, R.O.C (phone: +886-4-22840441 ext 521; fax: +886-4-22862587; e-mail: mdlin@nchu.edu.tw ).

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

where $C_i$ = concentration (mg/L) of chlorine in pipe $i$ as a function of distance $x$ and time $t$; $u_i$ = flow velocity (m/s) in pipe $i$; $r(C_i)$ = reaction rate relationship. The reaction rate relationship $r(C_i)$ for chlorine is adopted from work by Rossman (2000).

EPANET2, which combines the hydraulic and water quality modules, is employed in this study to simulate the chlorine concentration of WDNS. The water quality is numerically solved using a Lagrangian time-driven method (LTDM) [19]. A more detailed review of EPANET2 is presented by [20].

In general terms, a water quality optimization problem aims to minimize chlorine mass injection dosage and maintain the chlorine concentration profiles at monitoring nodes within the specified bounds. The optimal design formulation used in this study is given by (2)-(4) [8]

$$AC_i = 0.06BC_i \tag{2}$$

$$Minimize \quad \mathrm{DAC} = \sum_{i=1}^{n_b}\sum_{t=0}^{24}\left[AC_i\right]_t \tag{3}$$

subject to

$$C_{\min} \le C_m \le C_{\max},\ m = 1,...,n_m \tag{4}$$

where $AC_i$ = hourly added chlorine (g/hour); $BC_i$ = chlorine added (booster chlorination) for an hour at a node with a booster station in mg/min (obtained from EPANET2 Toolkit functions); $DAC$ = daily added chlorine (g/day); $n_b$ = number of booster stations (injection points), $t$ = time (hour), $C_m$ = disinfectant concentration at monitoring location $m$ (mg/L), $C_{min}$ and $C_{max}$ are specified minimum and maximum chlorine concentrations (mg/L), respectively, and $n_m$ = number of monitoring locations.

## III. Genetic Immune Algorithms

In addition to IA model, a novel GIA, which adopts a GA as a pre-processor to screen briefly the initial antibody repertoires, was also developed in this study. GAs, developed by Holland and coworkers at the University of Michigan [21]-[22], are heuristic optimization methods that search for solutions of complex problems using an analogy between optimization and natural selection. Unlike gradient-based methods, GAs utilize random search procedures inspired by biological evolution and cross-breeding trial designs, and allows only the "fittest" designs to survive and propagate to successive generations. When using GAs to solve optimization problems, decision variable are encoded as substrings of binary digits or real numbers. These substrings are concatenated to form "chromosomes" representing a particular design or solution. A population of randomly generated chromosomes (trial solutions) breeds the subsequent offspring via crossover, mutation and selection processes. Theoretically, trial solutions are optimized through generations until a termination criterion is satisfied.

IA is a set of computational systems inspired by theoretical immunology and observed immune functions, principles, and mechanisms. It has been applied for solving various complicated optimization problems [10]-[13], [15], [17]. The aims of the immune system are to protect the body from disease-causing agents (pathogens) and eliminate malfunctioning cells [11]. The complex immune system discriminates between "self" cells and foreign "non-self" pathogens. Each cell in an organism is comprised of molecules characterized as self genes. Conversely, molecules constituting alien organisms are characterized as non-self genes. Immune responses can rapidly eliminate foreign non-self pathogens, while adaptive immunity targets particular pathogens. The immune system, which is the first line of defense against foreign pathogens, includes anatomical barriers (skin and mucous membrane), physiological barriers (temperature and pH), and endocytic and phagocytic barriers (macrophages). Humoral immunity and cell-mediated immunity are the second line of defense, and they comprise the immune response of immunocompetent cells that include B lymphocytes (or B cells) and T lymphocytes (or T cell) [23]. Both cell types have surface receptor molecules (the B cell receptor molecule is also called an antibody). When foreign pathogens (antigens) invade an organism, an immune response is stimulated, generating immune cells that recognize the antigens. Once the antigens are recognized by the immune cell receptors, a "clone selection" process causes the immune cells that recognize the antigens to proliferate and secrete antibodies [24]. Some proliferated immune cells become plasma cells, while others are maintained as memory cells [25]. Memory cells circulate through the blood, lymphatic system, and tissues. When exposed to a second antigenic stimulus, the memory cells begin differentiating into plasma cells capable of producing high-affinity antibodies. Affinity is the degree of binding between an antibody receptor and antigen. As affinity increases, binding increases; thus, the immune recognition and response increase [11]. For a detailed review of immunology and IA, see [11].

Therefore, when solving the scheduling of booster disinfection optimization problems using GIA, the antibodies and antigen can be regarded as trial solutions and the optimal solution, respectively. GIA achieves the optimal solution by iteratively searching for the antibody with the highest affinity. The computational procedure of the proposed GIA for solving the scheduling of booster disinfection optimization problems is as follows:

*Step 1:* Define antigen.

When applying GIA to solve optimization problems, the objective function and constraints are represented by antigens [25]. Antigens in the immune system are recognized by antibody receptors in a manner similar to the relationship between a lock and key [11]. The antigen represents the configuration of variables in the optimal solution of the optimization problem, while the corresponding segment of the antibody represents a trial solution for the variable. Therefore, the antigen in this study is defined as the WDNS with minimum daily added chlorine.

*Step 2:* Generate an initial repertoire of antibodies.

Since GA has good global search capability, this study employed a simple GA to concisely and comprehensively examine the solution space and locate high-quality trial solutions to enhance the affinities of the initial IA repertoire. The number of generations, population size, and number of offspring generated in each generation for the simple GA

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

were 200, 50 and 10, respectively. The variables used in the GA were real-number coded for consistency with those in IA.

*Step 3:* Evaluate the affinity of antibodies to the antigen.

The affinity $Ag_i$ of each antibody to antigen (called "antibody-antigen affinity" in this study) in the current repertoire is calculated based on its objective function value and potential constraint violations. In evaluating the affinity of the individual antibodies, constraint requirements are also examined. When a constraint is violated, the degree of violation is weighted to penalize the antibody's affinity. Antibodies with high affinity represent good individuals. In solving the minimum added chlorine problem for a WDNS, the $Ag_i$ of antibody $i$ is calculated by

$$Ag_i = \frac{1}{vio_i + DAC_i} \tag{5}$$

where $DAC_i$ is the daily added chlorine dosage of the WDNS identified by antibody $i$, as indicated in (3), and $vio_i$ is the weighted penalty for constraint violations.

*Step 4:* Select the $n$ best antibodies in the current repertoires based on their affinities.

*Step 5:* Generate clone set $C$.

If the procedure follows *Step 4*, then these n best antibodies are cloned to generate a temporary repertoire of clone set $C$. If the procedure follows *Step 9*, then $C$ is generated by cloning the antibodies in the memory set $M$, which is described in *Step 8*. Clone set $C$ has possession of the better antibodies, thereby increasing the affinities to the antigen.

*Step 6:* Generate new antibodies.

According to [26], genetic operations can enhance the IA in producing solutions and perturbations for selected solutions to avoid the local optimum. In this step, genetic operations such as crossovers and mutations — resembling those in GA — are performed by the clone set $C$ to generate new and generally improved antibodies. The crossover operation generates new antibodies by mixing genetic material in the chromosomes of the original antibodies in the current repertoires. Since the variables are real-number coded in this study, arithmetical crossover is applied to interpolate the values of two elements at selected crossover points instead of exchanging them. This approach can maintain the elements of newly generated vectors within the original domain, and it can be expressed as [27]:

$$\begin{aligned} x_u^{k+1} &= c \cdot x_v^k + (1-c) \cdot x_u^k \\ x_v^{k+1} &= c \cdot x_u^k + (1-c) \cdot x_v^k \end{aligned} \tag{6}$$

where $x_u^k$ and $x_v^k$ are the two decision variables to be crossed, $x_u^{k+1}$ and $x_v^{k+1}$ are the newly generated variables, and $c$ is a constant between 0 and 1.

Conversely, mutation randomly changes the antibody elements, but it introduces diversities so that the algorithm does not get stuck at the local optima. For an antibody $V_i = (x_1, \ldots, x_m, \ldots, x_n)$, each decision variable $x_m$, $1 \le m \le n$ has the same probability of mutation. Let $V_i' = (x_1, \ldots, x'_m, \ldots, x_n)$ be the mutated antibody $V_i$, then the mutated element $x'_m$ can be defined as

$$x'_m = \begin{cases} x_m + \Delta(k, b_m - x_m) \\ x_m - \Delta(k, x_m - a_m) \end{cases} \tag{7}$$

$$\Delta(k, y) = y \cdot \left( 1 - \gamma^{(1-\frac{k}{T})^b} \right) \tag{8}$$

where $\gamma$ is a random number uniformly distributed on [0, 1], $k$ is the evaluation number, $T$ is the maximal evaluation number, and $b$ is a system parameter determining the degree of dependency on the evaluation number.

The two expressions in (7) have an equal chance to be selected. The perturbation function $\Delta(k, y)$ returns a value in the range [0, $y$], so that the probability of $\Delta(k, y)$ close to 0 increases as k increases. The computational results from (7) and (8) are dependent upon the antibody repertoire age. Equation (8) causes the search to cover the decision space uniformly during the early search stages (small $k$) and locally during the late stages (large $k$). In other words, mutation performs a global search of the solution space at the beginning of the iterative process; when the regions likely containing the global optimum are located, fine local tuning is performed.

*Step 7:* Survey newly generated antibodies.

The antibody-antigen affinities $Ag_i$ of the antibodies generated in *Step 6* are evaluated. Moreover, to retain the antibody diversity in the current repertoire, affinity $Ab_j$ (antibody-antibody affinity hereafter) between the antibodies $j$ and the best antibody in memory set $M$ are also investigated. The antibody-antibody affinity $Ab_j$ in this study can be expressed as

$$Ab_j = \frac{1}{(1 + D_j)} \tag{9}$$

$$D_j = \sum_{i=1}^{L} \delta_i, \quad \text{where } \delta_i = \begin{cases} 1 & \text{if } antibody_j^i \neq antibody_{best}^i \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $D_j$ is the Hamming distance between the antibodies $j$ and the best antibody, $antibody_j^i$ is the $i$th element of antibody $j$, and $antibody_{best}^i$ is the $i$th element of the best antibody.

*Step 8:* Generate memory set $M$.

Memory set $M$, which is analogous to the memory cells in biological immune systems, is a group of antibodies with high antibody-antigen affinities. Memory set $M$ is used to provide-based on its experience memory-antibodies (trial solutions) most likely to recognize antigens (optimal solutions).

If the newly generated antibodies possess higher antibody-antigen affinity $Ag_i$ than the current members of $M$, the inferior antibodies in the current memory set $M$ are replaced [26]. However, the antibody-antibody affinities $Ab_j$ of superior antibodies are also analyzed. Only the antibodies with low $Ab_j$, are included in memory set $M$ to retain the repertoire diversity.

*Step 9:* Examine the termination criterion.

If the termination criterion is satisfied, the computation procedure stops. Otherwise, the procedure returns to *Step 5*. The termination criterion in this work is the maximum number of evaluations. Fig. 1 presents a flow chart of the computational procedure of the proposed GIA.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

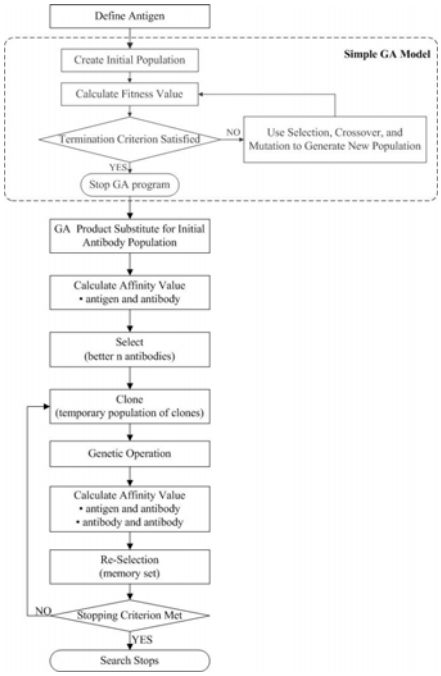


Fig. 1 Flowchart of genetic immune algorithm.

## IV. CASE STUDY

The WDNS of the Cherry Hill-Brushy Plains portion of the South Central Connecticut Regional Water Authority (SCCRWA) has been the subject of numerous studies to validate and test network water quality models. The network configuration is adopted from the study of Boccelli *et al.* (1998) that includes 34 consumer nodes, 1 source node representing a pump station, 1 storage tank, and 47 pipes [3]. The 24-hour cycle of hydraulic analysis for the supply source (pump station) and storage tank of the system was based on four periods. During period 1 (0 to 6 h) and period 3 (12 to 18 h), the supply source controls the flow, while period 2 (6 to 12 h) and period 4 (18 to 24 h) the pump is off and the storage tank controls the flow into network. The chlorine concentrations of the WDNS need to be maintained between 0.2 mg/L and 4.0 mg/L to ensure pathogen control and avoid producing DBPs. This study investigated three scenarios published in the literature, and the results obtained by GIA were compared with previous studies. The scenario I considers the 6 booster stations A-F and the locations are adopted from the study of [6], as shown in Fig. 2.


Fig. 2 Network sketch of scenario I

The first-order bulk and wall decay coefficients used in this case are 0.55 $d^{-1}$ and 0.36 m/d, respectively [6]. The booster station numbers and locations of scenarios II and III are adopted from the study of [3], as shown in Fig. 3. In scenario II, the disinfectant can only be added at node A, while all of the 6 booster stations are considered in scenario III. The bulk decay coefficient of both scenarios II and III is 0.5 $d^{-1}$, and the wall decay is ignored.

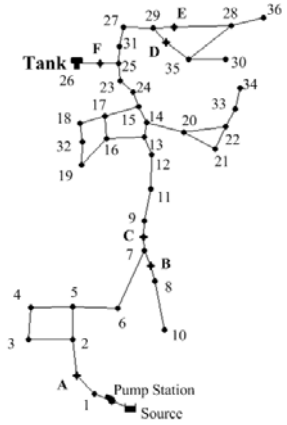


Fig. 3 Network sketch of scenarios II and III

In this study, network hydraulics and water quality are simulated using EPANET 2.0 [20], and the optimization problem is solved by GIA. The performances of three different sets of GIA parameters in scenario I, as shown in Table 1 and Table 2, were evaluated to obtain the optimal parameter configuration.

Table 2 Parameter configurations of GIA used in this study

| GIA parameters | | GIA run1 | GIA run 2 | GIA run3 |
|---|---|---|---|---|
| simple GA parameters | population sizes | 50 | | |
| | Number of Generations | 200 | | |
| Antibody repertoire sizes | | 100 | 150 | 200 |
| Clone sizes | | 30 | 20 | 10 |
| Probability of crossover | | 0.5 | 0.7 | 0.9 |
| Probability of mutation | | 0.06 | 0.04 | 0.02 |

Fig. 4 shows the plot of the evolution of objective function value solving by GIA with different parameter configurations. Since the GIA run 2 possess the best performance in comparison with the other runs, this study respectively employed these two configurations for the parameters of GIA to solve all the three scenarios.

It can be found that a medium size of antibody repertoire 150, clone size 20, crossover rate 0.7, and mutation rate 0.04 is the most suitable parameter configuration for GIA.

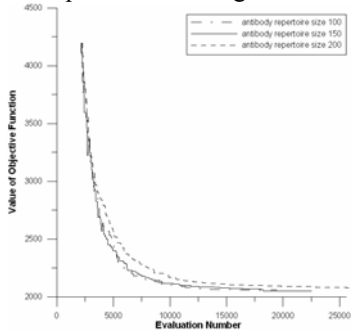


Fig. 4 Evolution of objective function value for scenario I solving by GIA with different

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Tables 3 to 5 present the optimal schedules of the booster disinfectant obtained by GIA for the three scenarios, and the results obtained by LP and GA are also illustrated [3],[6].

As the results of scenario I (Table 3) indicate, the minimum chlorine mass rate obtained by GIA (2,050 mg/day) outperforms GA (2,145 mg/L). For scenario II (Table 4), GIA (2,826 mg/day) still outperforms GA (2,980 mg/L) and approximately equals to LP (2,824 mg/day). As the results of scenario III indicate (Table 5), the mass rate obtained by GIA(1,193 mg/day) is approximately better than GA (1,205 mg/day). However, they are both greater than LP (1,120 mg/day). It can be concluded that GIA is capable of finding solutions better than GA. As for the comparisons with LP, there is an important issue needed to be mentioned. The LP solutions were obtained via EPANET 1.0 which applies the Eulerian discrete volume element method in its water quality module [3]. However, the solutions of GA [6] and this study (GIA) were all obtained by EPANET 2.0 which applies the Lagrangian time driven method to simulate the water quality. Therefore, the difference between the LP solutions and others may be due to the different numerical methods employed in EPANET 1.0 and 2.0.

Table 3 Optimal injection rates of scenario I obtained by GA and GIA

| Booster location | Period | Injection rate for GA[a] (mg/min) | Injection rate for GIA[b] (mg/min) |
|---|---|---|---|
| A | 1 | 3.14 | 34.54 |
| | 2 | 6.25 | 88.20 |
| | 3 | 0.85 | 288.56 |
| | 4 | 2.36 | 243.87 |
| B | 1 | 629.13 | 346.93 |
| | 2 | 15.59 | 7.12 |
| | 3 | 531.73 | 53.74 |
| | 4 | 10.25 | 1.32 |
| C | 1 | 699.70 | 427.15 |
| | 2 | 601.26 | 110.15 |
| | 3 | 526.20 | 112.39 |
| | 4 | 250.14 | 489.00 |
| D | 1 | 16.16 | 1.76 |
| | 2 | 12.52 | 988.11 |
| | 3 | 12.04 | 0.00 |
| | 4 | 13.67 | 431.06 |
| E | 1 | 1,250.02 | 980.24 |
| | 2 | 0.00 | 0.00 |
| | 3 | 1,394.98 | 1,124.40 |
| | 4 | 0.00 | 0.00 |
| F | 1 | 11.18 | 0.69 |
| | 2 | 3.93 | 1.00 |
| | 3 | 9.38 | 24.35 |
| | 4 | 4.06 | 0.02 |
| Total mass rate (g/day) | | 2,145.20 | 2,050.06 |

[a]Munavalli and Mohan Kumar (2003), [b]this study

The case study has found optimal booster chlorination disinfection injection rates at both the source and disinfection boosters in the WDNS. The evaluation results confirm the potential of GIA in solving the scheduling of booster disinfection optimization problems.

Table 4 Optimal injection rates of scenario II obtained by LP, GA, and GIA

| Booster location | Period | Injection rate for LP[a] (mg/min) | Injection rate for GA[b] (mg/min) | Injection rate for GIA[c] (mg/min) |
|---|---|---|---|---|
| A | 1 | 5,678.9 | 5,772.7 | 6,592.0 |
| | 2 | 0.0 | 0.0 | 0.0 |
| | 3 | 2,166.0 | 2,507.5 | 1,257.0 |
| | 4 | 0.0 | 0.0 | 0.0 |
| Total mass rate (g/day) | | 2,824 | 2,980 | 2,825.6 |

[a]Boccelli et al. (1998), [b]Munavalli and Mohan Kumar (2003), c this study

Table 5 Optimal injection rates of scenario III obtained by LP, GA, IA, and GIA

| Booster location | Period | Injection rate for LP[a] (mg/min) | Injection rate for GA[b] (mg/min) | Injection rate for GIA[c] (mg/min) |
|---|---|---|---|---|
| A | 1 | 587.7 | 599.3 | 835.5 |
| | 2 | 0.0 | 0.0 | 0.0 |
| | 3 | 634.7 | 680.9 | 916.4 |
| | 4 | 0.0 | 0.0 | 0.0 |
| B | 1 | 4.4 | 9.8 | 0.0 |
| | 2 | 0.0 | 0.7 | 5.0 |
| | 3 | 4.5 | 4.3 | 0.0 |
| | 4 | 0.0 | 0.0 | 0.0 |
| C | 1 | 354.0 | 473.6 | 22.0 |
| | 2 | 0.0 | 0.0 | 0.1 |
| | 3 | 459.3 | 413.0 | 96.1 |
| | 4 | 0.0 | 0.3 | 4.0 |
| D | 1 | 0.0 | 0.7 | 0.7 |
| | 2 | 0.3 | 0.3 | 2.7 |
| | 3 | 0.0 | 0.0 | 1.0 |
| | 4 | 0.2 | 0.7 | 20.0 |
| E | 1 | 0.1 | 0.4 | 0.0 |
| | 2 | 0.4 | 0.4 | 1.0 |
| | 3 | 0.2 | 0.3 | 1.0 |
| | 4 | 0.2 | 1.0 | 2.0 |
| F | 1 | 0.0 | 0.0 | 16.1 |
| | 2 | 671.1 | 713.5 | 554.9 |
| | 3 | 15.0 | 47.1 | 363.7 |
| | 4 | 377.9 | 400.7 | 470.5 |
| Total mass rate (g/day) | | 1,120 | 1,205 | 1,192.6 |

[a]Boccelli et al. (1998), [b]Munavalli and Mohan Kumar (2003), [c] this study

## V. CONCLUSION

This study provides the first experimental analysis using a GIA to solve the optimal scheduling of booster disinfection problems of a WDNS. The chlorine injection pattern was formulated as a single objective problem. The results specify that booster disinfection can significantly increase desired residual concentration above the reasonable limit while helping to reduce variability in monitoring nodal concentrations. The results obtained from a benchmark case study show that the GIA is able to obtain optimal solution (chlorine injection dosage) more effectively and efficiently than GA.. Application of the GIA to the WDNS quality optimization problem remains in its infancy and further improvements are necessary.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

## ACKNOWLEDGMENT

This research was funded by a grant from the National Science Council of the Republic of China (NSC-96-2628-E-005-001-MY3). Part of the materials compiled in this paper have been presented previously at the 3rd International Conference on Convergence and Hybrid Information Technology (November 11-13, 2008) held in Korea.

## REFERENCES

[1] Bielmeier, S. R., Best, D. S., Guidici, D. L., and Narotsky, M. G., "Pregnancy loss in the rat caused by bromodichloromethane." Toxicol. Sci., 2001, 59(2), pp. 309-315

[2] Waller, K., Swan, S. H., DeLorenze, G., and Hopkins, B., "Trihalomethanes in drinking water and spontaneous abortion." Epidemiology, 1998, 9(2), pp. 134-140.

[3] Boccelli, D. L., Tryby, M. E., Uber, J. G., Rossman, L. A., Zierolf, M. L., and Polycarpou, M. M., "Optimal scheduling of booster disinfection in water distribution systems." J. Water Resour. Plann. Manage., 1998, 124(2), pp. 99-110.

[4] Tryby, M. E., Boccelli, D. L., Uber, J. G., and Rossman, L. A., "Facility location model for booster disinfection of water supply networks." Journal of Water Resources Planning and Management, 2002, 128(5), pp. 322-333.

[5] Propato, M., and Uber, J. G., "Linear least-squares formulation for operation of booster disinfection systems." J. Water Resour. Plann. Manage., 2004, 130(1), pp. 53-62.

[6] Munavalli, G. R., and Mohan Kumar, M. S., "Optimal scheduling of multiple chlorine sources in water distribution systems." J. Water Resour. Plann. Manage, 2003, 129(6), pp.493-504.

[7] Ostfeld, A., and Salomons, E., "Optimal operation of multiquality water distribution systems: unsteady conditions." Eng. Optimiz, 2004 , 36(3), pp. 337 - 359.

[8] Ozdemir, O. N., and Ucaner, M. E., "Success of booster chlorination for water supply networks with genetic algorithms." J. Hydraul. Res, 2005, 43(3), pp. 267-275.

[9] Ucaner, M. E., and Ozdemir, O. N., "Application of genetic algorithms for booster chlorination in water supply networks." World Water and Environmental Resources Congress Philadelphia, PA, United States, 2003, pp. 1629-1638.

[10] Bezerra, G. B., de Castro, L. N., and Von Zuben, F. J., "A hierarchical immune network applied to gene expression data." Proc., Lect. Notes Comput. Sci., Catania, Italy, 2004, pp. 3239, 14-27.

[11] Castro, L. N. d., and Timmis, J. I., "Artificial immune systems as a novel soft computing paradigm." Soft Comput., 2003, 7(8), pp. 526-544.

[12] Harmer, P. K., Williams, P. D., Gunsch, G. H., and Lamont, G. B., "An artificial immune system architecture for computer security applications." IEEE Trans. Evol. Comput., 2002, 6(3), pp. 252-280.

[13] Koster, M., Grauel, A., Klene, G., and Convey, H., "A new paradigm of optimisation by using artificial immune reactions." Knowledge-Based Intelligent Information and Engineering Systems, Pt 1, Proceedings, 2003, pp. 287-292.

[14] Chun, J.-S., Lim, J.-P., Jung, H.-K., and Yoon, J.-S., "Optimal design of synchronous motor with parameter correction using immune algorithm." IEEE Trans. Energy Convers., 1999, 14(3), pp. 610-615.

[15] Hajela, P., Yoo, J., and Lee, J., "GA based simulation of immune networks applications in structural optimization." Eng. Optimiz., 1997, 29(1-4), pp. 131-149.

[16] Mori, K., Tsukiyama, M., and Fukuda, T. "Immune algorithm with searching diversity and its application to resource allocation problem." T. IEE Japan, 1993, 113-C(10), pp. 872-878.

[17] Tazawa, I., Koakutsu, S., and Hirata, H. "Immunity based genetic algorithm and its application to the VLSI floorplan design problem." Proc., IEEE Conf. on Evolutionary Computation, ICEC, Nagoya, 1996, pp. 417-421.

[18] Chu, C-W., M-D. Lin, and K-T. Tsai, "Optimal Scheduling of Booster Chlorination with Immune Algorithm, 3rd International Conference on Convergence and Hybrid Information Technology, ICCIT 2008, V2, 2008, pp. 1226-1232.

[19] Liou, C. P., and Kroon, J. R. "Modeling the propagation of waterborne substances in distribution networks." J. Am. Water Work Assoc., 1987, 79(11), pp. 54-58.

[20] Rossman, L. A., *EPANET 2 users manual*, Risk Reduction Engineering Laboratory, U. S. Environmental Protection Agency, Cincinnati, Ohio, 2000.

[21] Goldberg, D. E., *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley Publishing Co., Reading, Mass, 1989.

[22] Holland, J. H., *Adaptation in natural and artificial systems*, MIT Press, Cambridge, Mass, 1975.

[23] Boukerche, A., Juca, K. R. L., Sobral, J. B., and Notare, M. S. M. A., "An artificial immune based intrusion detection model for computer and telecommunication systems." Parallel Comput., 2004, 30(5-6), pp. 629.

[24] Engin, O., and Doyen, A., "A new approach to solve hybrid flow shop scheduling problems by artificial immune system." Future Gener. Comput. Syst., 2004, 20(6), pp. 1083-1095.

[25] Wen, X., and Song, A., "An immune evolutionary algorithm for sphericity error evaluation." Int. J. Mach. Tools Manuf., 2004, 44(10), pp. 1077.

[26] Chen, T.-C., and You, P.-S., "Immune algorithms-based approach for redundant reliability problems with multiple component choices." Comput. Ind., 2005, 56(2), pp. 195-205.

[27] Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, New York, 1992.