Available online at www.sciencedirect.com
ScienceDirect
Computers and Chemical Engineering 31 (2007) 1565–1575
Computers & Chemical Engineering
www.elsevier.com/locate/compchemeng

Pinch multi-agent genetic algorithm for optimizing water-using networks

Kai Cao, Xiao Feng *, Hang Ma

Department of Chemical Engineering, Xi'an Jiaotong University, Xi'an 710049, People's Republic of China

Received 27 March 2006; received in revised form 11 January 2007; accepted 17 January 2007
Available online 24 January 2007

Abstract

A new genetic algorithm, pinch multi-agent genetic algorithm (PMAGA), is developed for optimizing water-using networks. All agents are fixed on a lattice, and they will compete or cooperate to increase their energy. On the other hand, agents can also increase their energy with knowledge. For single contaminant water-using systems, PMAGA and other algorithms as well as Lingo can find the same minimum freshwater consumption. For multiple contaminants systems, it can find the same or better results in case studies. Moreover, PMAGA is more efficient for much shorter computational time compared with other algorithms, and furthermore, it can yield many water-using networks consuming the same minimum freshwater but with different configurations, whereas Lingo yields only one network. These alternative configurations give more options for industries.

Keywords: Wastewater minimization; Genetic algorithm; Water network

1. Introduction

Many industrial processes need water as either raw material or utility, discharging it into the environment after the usage. The economic growth and industrialization demand results in more and more fresh water consumption, so that freshwater becomes a limited resource. As a result, the price of freshwater rises. Because of environmental impact, wastewater should be minimized before being discharged. The optimum water-using networks can reduce the freshwater consumption and wastewater discharge. This is very important, especially in developing countries. During the past 20 years, many new analysis and design techniques have been developed to improve water-using systems. These techniques include graphical methods, often involving pinch analysis (El-Halwagi, Gabriel, & Harell, 2003; Hallale, 2002; Kuo & Smith, 1998; Linhoff & Smith, 1994; Manan, Wan Alwi, & Ujang, 2006; Wang & Smith, 1994, 1995), mathematical programming (Bagajewicz, Rivas, & Savelski, 2001; Doyle & Smith, 1997; Feng & Seider, 2001; Huang, Chang, Ling, & Chang, 1999; Prakotpol & Srinophakun, 2004; Savelski & Bagajewicz, 2003) and mass-exchange networks synthesis (El-Halwagi, 1997; Hallale & Fraser, 2000). In addition, progress has been made in rule-based design methodology (Prakash & Shenoy, 2005).

Graphical methods and mass-exchange networks synthesis methods provide simple methods and beneficial results when applied to water-using networks. However, they suffer from a major drawback, since, as the number of contaminants dealt with increases, they become increasingly difficult to be applied.

Mathematical optimization is a more suitable approach for optimum water-using networks, for both grassroot and retrofit applications. In wastewater minimization problem involving single key contaminant, the optimization programming can be formulated as a linear programming (LP) problem. For water systems with multiple contaminants, the mathematical representation is in the class of non-linear programming (NLP) (Mann & Liu, 1999). In addition, non-convex generalized disjunctive programming (GDP) models for process networks, including water-using networks, with bilinear equalities and a global optimization algorithm were proposed over the last few years (Karuppiah & Grossmann, 2006; Lee & Grossmann, 2000, 2003; Raman & Grossmann, 1994; Vecchietti & Grossmann, 1999). However, as far as NLP problem is concerned, because this class of problem has many local optimums, a globally optimal solution cannot be guaranteed. The accuracy and efficiency of conventional techniques for finding the global optimum usually

Abbreviations: Lingo, optimization software package; NLP, non-linear program; LP, linear program; MAGA, multi-agent genetic algorithm; PMAGA, pinch multi-agent genetic algorithm

* Corresponding author.
E-mail address: xfeng@mail.xjtu.edu.cn (X. Feng).

0098-1354/$ – see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compchemeng.2007.01.004

Nomenclature

a, b, c, d, e, h	penalty coefficients
C	set of contaminants
$C_{i,k}^{\text{in}}$	inlet concentration of contaminant k to unit i (ppm)
$C_{i,k}^{\text{in,max}}$	maximum allowable inlet concentration of contaminant k to unit i (ppm)
$C_{i,k}^{\text{out}}$	outlet concentration of contaminant k from unit i (ppm)
$C_{i,k}^{\text{out,max}}$	maximum allowable outlet concentration of contaminant k from unit i (ppm)
$C_{\text{w},k}$	concentration of contaminant k in freshwater (ppm)
$F_{i,j}$	flowrate of reused water from unit i to unit j (t/h)
Fd_i	flowrate of wastewater from unit i (t/h)
Fw_i	flowrate of freshwater demanded by unit i (t/h)
K	number of contaminants
Lsize, P_0, P_c, P_m, sLsize, sRadius, sP_m, sGen	PMAGA parameters
$M_{i,k}$	mass load of contaminant k to be removed in unit i (g/h)
N	number of reused water streams which can be canceled by Pinch rules
P	set of water-using process units
P	number of process units
y_i^d	binary variable = 0 when $\text{Fd}_i \le 10^{-4}$; otherwise = 1
y_i^w	binary variable = 0 when $\text{Fw}_i \le 10^{-4}$; otherwise = 1
$y_{i,j}$	binary variable = 0 when $F_{i,j} \le 10^{-4}$; otherwise = 1

depend on the initial guess. The inappropriate initial set leads to the solution getting stuck at the local optimum. This issue is particularly challenging when the dimension is high.

Therefore, developing a new algorithm with high efficiency and suitable for complex water-using systems is a challenging work.

2. Genetic algorithm and multi-agent genetic algorithm

2.1. Genetic algorithm

A well-known technique for avoiding local optima in improving search is genetic algorithm (GA) (Keedwell & Khu, 2006).

Genetic algorithm is a search algorithm based on the mechanics of natural selection and natural genetics. GA has been successfully applied to a wide variety of problems including non-convex function optimization. The GA approach uses a population of individual solutions that iterate from one generation to the next as the search progresses. This is motivated by a hope that the new population will be better than an old population. The performance of each of the solutions is evaluated by an "objective function" which relates the solution variables to the problem at hand. In order to form a new population, GA uses genetic operators and selection process. Genetic operators including crossover and mutation are used to generate the new solutions (offspring) from the current solutions (parents). Selection is the process of keeping and deleting some solutions from both parents and offspring for the same number of the next population. Moreover, selection is the process of choosing some parents to generate offspring also. In the selection process, the solutions are selected according to their values of objective function (fitness). The more fitness has, the more chance to be selected. The algorithm will repeat until a termination condition is satisfied. The best solution is returned to represent the optimum solution. The classical GA has the following basic characteristics:

1. it works in the representation space after encoding the parameters by alphabets or real values, especially binary symbols, instead of parameter space;
2. it searches from a population of points instead of a single point;
3. it uses only the fitness function values which may be model-independent in the optimization instead of any model knowledge;
4. it uses probabilistic operation rules.

The main drawback of classical GA is the computation time consuming (Keedwell & Khu, 2005, 2006; Lavric, Iancu, & Plesu, 2005; Prakotpol & Srinophakun, 2004), which can be very large in the case of thorough investigation. Moreover, GA neglects to include a variety of heuristics (implicit knowledge of the problem) that are abundant in optimum water-using networks (Keedwell & Khu, 2006). Keedwell and Khu (2006) proposed a novel heuristic-based and cellular automata-inspired approach to the optimal design of water distribution networks. The proposed approach uses a parallel, localist and heuristic-based algorithm to optimally design water distribution networks requiring only a limited number of model evaluations. However, this approach has no notion of global optimality, but is concerned only with making local changes to nodes in the network according to the rules specified above. Nevertheless, three key properties – parallelism, localist representation and homogeneity – embodied in this approach are very helpful to optimize process networks.

2.2. Multi-agent genetic algorithm

In 2003, Zhong, Liu, Xue, and Jiao combined multi-agent system with genetic algorithm to form a new algorithm, multi-agent genetic algorithm (MAGA), to solve global numerical optimization problem. This algorithm inspired by multi-agent system in artificial intelligence overcomes the limitation of computation time to some extent. MAGA can efficiently solve the optimization problem in the following form:

$$\min f(\underset{\sim}{\mathbf{X}}), \quad \underset{\sim}{\mathbf{X}} = (x_1, \ldots, x_n) \in S \tag{1}$$

where $S \subseteq R^n$ defines the search space which is an n-dimensional space bounded by the parametric constraints $\underline{x}_i \le x_i \le \bar{x}_i$, $i = 1, \ldots, n$.

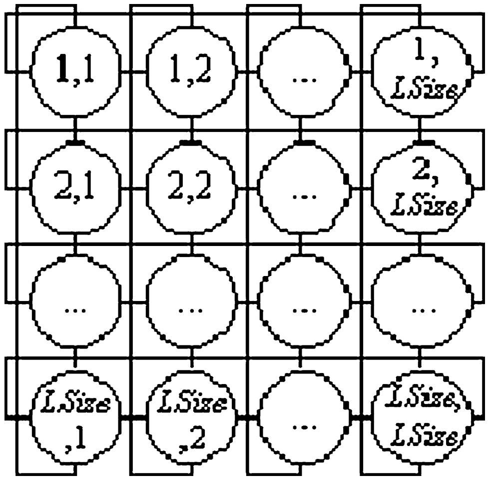

Fig. 1. Agent lattice.

According to Russell and Norvig (1995) and Zhong et al. (2003), an agent is any entity that can be viewed as perceiving its environment through sensors and acting upon its environment through effectors, and a multi-agent system can be used for all types of systems composed of multiple autonomous components showing the following characteristics: (1) each agent has incomplete information or capabilities for solving problems; (2) no system global control exists; (3) data are decentralized; and (4) computation is asynchronous.

In MAGA each agent stands for a solution, a real-valued vector $(x_1, \ldots, x_n)$, and all agents are put in a living environment, i.e., an Lsize × Lsize lattice. Every agent is fixed on a lattice-point and only has local perceptivity, so it can only interact with the agents around it. As a result, the agent lattice can be represented as the form in Fig. 1, where each circle stands for an agent, the data in circle stands for the position, and two agents can interact with each other to increase energy if and only if there is a line connecting them. The energy of an agent is equal to the negative value of objective function.

$$\text{Energy}(\underset{\sim}{\mathbf{X}}) = -f(\underset{\sim}{\mathbf{X}}) \quad (2)$$

Because the purpose of each agent is increasing its energy as possible as it can, there is fierce competition among agents, but the competition only exists between an agent and its neighbors. As a result, the agent with less energy will die and its neighbors will occupy the lattice-point freed by them. Certainly, cooperation may also take place between agents and their neighbors. Since agents have intelligence, they can also increase its own energy using their knowledge. Four evolutionary operators realize all these intelligent behaviors of agents and guide the evolutionary process: neighborhood competition operator, neighborhood orthogonal crossover operator (Leung & Wang, 2001), mutation operator and agent self-learning operator. Neighborhood competition operator and neighborhood orthogonal crossover operator realize competition and cooperation among agents. Mutation operator and agent self-learning operator increase the energy of agents by knowledge. The detailed information on these four evolutionary operators has been given by Zhong et al. (2003).

MAGA can be described in the following steps (Zhong et al., 2003).

P_c and P_m are predefined parameters. Best^t and CBest^t stand for the best agent found until tth generation and the best agent in tth generation, respectively. $U(0, 1)$ is a uniform random number generator.

Step 1 Initialize Lattice^0, updating Best^0, $t \leftarrow 0$.
Step 2 Performing neighborhood competition operator on each agent in Lattice^t, then obtaining $\text{Lattice}^{t+(1/3)}$.
Step 3 For each agent in $\text{Lattice}^{t+(1/3)}$, if $U(0, 1) < P_c$, performing neighborhood orthogonal crossover operator on it, then obtaining $\text{Lattice}^{t+(2/3)}$.
Step 4 For each agent in $\text{Lattice}^{t+(2/3)}$, if $U(0, 1) < P_m$, performing mutation operator on it, then obtaining Lattice^{t+1}.
Step 5 Finding CBest^{t+1} in Lattice^{t+1}, performing agent self-learning operator on CBest^{t+1}.
Step 6 If $\text{Energy}(\text{CBest}^{t+1}) > \text{Energy}(\text{Best}^t)$, then $\text{Best}^{t+1} \leftarrow \text{CBest}^{t+1}$; otherwise $\text{Best}^{t+1} \leftarrow \text{Best}^t$, $\text{CBest}^{t+1} \leftarrow \text{CBest}^t$.
Step 7 If termination criteria are reached, return, otherwise $t \leftarrow t+1$, go to Step 2.

3. Modeling the water-using networks

A schematic of a water-using process unit is shown in Fig. 2. Mass transfer of the contaminants is from the rich process stream to the water stream. Consequently, the concentrations in the rich process stream decrease as those in the water stream increase. The contaminants may correspond to suspended solid (SS), chemical oxygen demand (COD) or similar quantities whose concentration levels limit the reuse of the effluent water in the other operations.

To simplify the mathematical model of water-using networks, these following assumptions are made to the process:

1. Each water-using process unit operates at steady state with a constant mass load of each contaminant to be removed.
2. Only physical processes occur, not involving chemical reactions.

Let P and C denote the sets of water-using process units and contaminants,

$$\text{P} = \{i | i \text{ is the index of water-using process units}; \quad i = 1, \ldots, P\} \quad (3)$$

$$\text{C} = \{k | k \text{ is the contaminant index}; \quad k = 1, \ldots, K\} \quad (4)$$

where P is the number of process units and K is the number of contaminants.

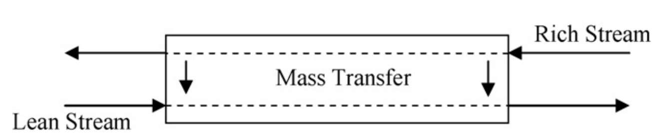

Fig. 2. Schematic for water-using process unit.

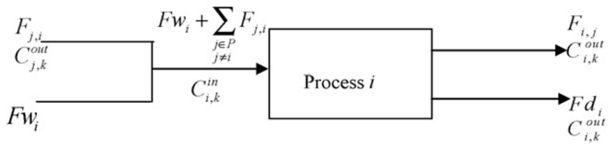

Fig. 3. Superstructure model of any operation in water system involving water reuse.

Fig. 3 (Mann & Liu, 1999) illustrates the superstructure model of any water-using process unit in a water system concerning water reuse.

For each water-using process unit *i*, the overall water balance, assuming no water loss, is

$$\mathrm{Fw}_i + \sum_{\substack{j\in P\\ j\neq i}} F_{j,i} - \mathrm{Fd}_i - \sum_{\substack{j\in P\\ j\neq i}} F_{i,j} = 0, \quad i\in \mathrm{P} \tag{5}$$

where Fw_i, $F_{j,i}$, $F_{i,j}$ and Fd_i are the flowrate of freshwater demanded by unit *i*, water from unit *j* to unit *i*, water from unit *i* to unit *j* and wastewater from unit *i*, respectively. Similarly, the species balances for contaminant *k*, are

$$\mathrm{Fw}_i\, C_{\mathrm{w},k} + \sum_{\substack{j\in P\\ j\neq i}} (F_{j,i} C^{\mathrm{out}}_{j,k}) + M_{i,k} = C^{\mathrm{out}}_{i,k}\left(\mathrm{Fw}_i + \sum_{\substack{j\in P\\ j\neq i}} F_{j,i}\right), \quad i\in \mathrm{P},\quad k\in \mathrm{C} \tag{6}$$

where $C_{\mathrm{w},k}$, $C^{\mathrm{in}}_{i,k}$ and $C^{\mathrm{out}}_{i,k}$ are the concentrations of contaminant *k* in freshwater, the inlet and outlet concentrations of contaminant *k*. $M_{i,k}$ is the mass load of contaminant *k* to be removed. In addition to the mass balances, concentration limits are necessary:

$$C^{\mathrm{in}}_{i,k} = \frac{\sum_{\substack{j\in P\\ j\neq i}} (F_{j,i} C^{\mathrm{out}}_{j,k}) + \mathrm{Fw}_i\, C_{\mathrm{w},k}}{\left(\mathrm{Fw}_i + \sum_{\substack{j\in P\\ j\neq i}} F_{j,i}\right)} \le C^{\mathrm{in,max}}_{i,k}, \quad i\in \mathrm{P},\quad k\in \mathrm{C} \tag{7}$$

$$C^{\mathrm{out}}_{i,k} \le C^{\mathrm{out,max}}_{i,k}, \quad i\in \mathrm{P},\quad k\in \mathrm{C} \tag{8}$$

where $C^{\mathrm{in,max}}_{i,k}$ and $C^{\mathrm{out,max}}_{i,k}$ are the maximum allowable inlet and outlet concentrations, respectively.

Finally, all variables are non-negative:

$$\mathrm{Fw}_i,\, \mathrm{Fd}_i,\, F_{i,j},\, F_{j,i},\, C^{\mathrm{in}}_{i,k},\, C^{\mathrm{out}}_{i,k},\, C_{\mathrm{w},k} \ge 0 \tag{9}$$

For the single contaminant system, the outlet concentration is forced to reach the maximum allowable outlet concentration to linearize the model. Then, Eqs. (6) and (7) become Eqs. (10) and (11), respectively (subscript *k* is deleted because there is a single contaminant).

$$\mathrm{Fw}_i C_{\mathrm{w}} + \sum_{\substack{j\in P\\ j\neq i}} (F_{j,i} C^{\mathrm{out,max}}_{j}) + M_i = C^{\mathrm{out,max}}_{i}\left(\mathrm{Fw}_i + \sum_{\substack{j\in P\\ j\neq i}} F_{j,i}\right), \quad i\in \mathrm{P} \tag{10}$$

$$C^{\mathrm{in}}_{i} = \frac{\sum_{\substack{j\in P\\ j\neq i}} (F_{j,i} C^{\mathrm{out,max}}_{j}) + \mathrm{Fw}_i\, C_{\mathrm{w}}}{\left(\mathrm{Fw}_i + \sum_{\substack{j\in P\\ j\neq i}} (F_{j,i})\right)} \le C^{\mathrm{in,max}}_{i}, \quad i\in \mathrm{P} \tag{11}$$

For the multiple contaminants system, the outlet concentration of each contaminant cannot be fixed.

4. Minimizing freshwater water consumption

In this paper, we seek to find the optimum water-using network with minimum freshwater consumption:

$$\min \sum_i \mathrm{Fw}_i, \quad i\in \mathrm{P} \tag{12}$$

The objective function expressed by (12) is subject to constraints (5) and (9)–(11) for single contaminant systems, or to constraints (5)–(9) for multiple contaminants systems.

4.1. Simplifying the superstructure by water pinch

According to Lavric et al. (2005), Li and Yao (2004), and Mann and Liu (1999), the approach of optimization of the water-using networks by the combination of Water Pinch analysis with mathematical programming not only can discover the operational bottlenecks so as to identify the minimum freshwater consumption and determine the elementary rules for water-using networks design based on the understanding for water-using processes, but also can prevents the case that the superstructure is so large that it is difficult to solve. Thus, it is very helpful to analyze the water-using process by Water Pinch rules before solving the optimization problem.

These rules can be demonstrated as follows (Li & Yao, 2004):

(1) For single contaminant systems, if a water-using process unit is below the pinch, it does not discharge wastewater. For multiple contaminants systems, if a water-using process unit is below the pinch for each contaminant, it does not discharge wastewater.
(2) For single contaminant systems, if a water-using process unit is above the pinch, it does not use freshwater. For multiple contaminants system, if a water-using process unit

is above the pinch for each contaminant, it does not use freshwater.

(3) If water-using process unit i is above the pinch (including the pinch) for each contaminant, and process unit j is below the pinch for each contaminant, the water reuse flowrate from i to j is zero.

(4) For single contaminant systems, if the maximum allowable outlet concentration of unit i is greater than that of unit j, the water reuse flowrate from i to j is zero.

4.2. Techniques for handling constraints

Because MAGA is a stochastic search algorithm that generates solutions randomly, each equality constraint is a bother. MAGA cannot produce solutions that satisfy the equality constraints randomly, and optimization of the water-using networks confronts such problem because of mass balances. That means the set of equality constraints drives MAGA from addressing the feasible solution. Therefore, MAGA must be modified to solve the problem. In this paper, the modified MAGA is named pinch multi-agent algorithm (PMAGA).

In this paper, all variables involved in the model have to be divided into three groups: random, dependent and determined variables. The values of random variables come from randomization, whereas the values of dependent variables come from simultaneous solutions of a linear system of equations composed of these equality constraints after assigning the values of random variables. Then, the values of determined variables come from the random and dependent variables. Although this method can solve the equality constraints problem, it cannot guarantee satisfying inequality constraints, such as (7)–(9) and (11). Major available techniques for GA to handle the constrained problems are based on the penalty function concept,

$$F(\underset{\sim}{\mathbf{X}}) = f(\underset{\sim}{\mathbf{X}}) + \text{Penalty}(\underset{\sim}{\mathbf{X}}) \tag{13}$$

where f is an objective function that is to be minimized and is subject to some equality and inequality constraints. $\text{Penalty}(\underset{\sim}{\mathbf{X}}) = 0$ if no violation occurs at $\underset{\sim}{\mathbf{X}}$ and is positive otherwise. Therefore, this penalty function will be adopted to solve the inequality constraints problem.

In the next two sections, the random variables, dependent variables, determined variables and penalty function of single contaminant and multiple contaminants systems will be explained in detail, respectively.

4.2.1. Single contaminant systems

In a single contaminant system that contains P water-using process units, the possible number of freshwater and wastewater stream is equal to the number of the units. Thus, there are $2P$ variables. There are $P^2 - P$ variables that represent the reused water streams. Hence, the total number of variables is $2P + P^2 - P = P^2 + P$ variables. The optimization model of this water-using system contains $2P$ equality constraints coming from Eqs. (5) and (10). If the values of $P^2 - P$ variables are fixed, the rest $2P$ variables can be determined by solving the $2P$ equations simultaneously. In the other words, the value of $2P$ variables depends on the others. Thus, the flowrates of reused water streams are selected as the random variables from randomization by PMAGA. The flowrates of freshwater streams and wastewater streams are dependent variables. Their values come from the solution of the linear system of equations after assigned the values of random variables. At last, each inlet concentration can be calculated by Eq. (11), so it is determined variable. Furthermore, if N reused water streams can be canceled by Pinch rules, the number of random variables becomes $P^2 - P - N$. Thus, the dimensionality reduction of the search space is realized. The relationship between random and dependent variables can be formulated as the following matrix equation:

$$\underset{\sim}{\mathbf{A}}\underset{\sim}{\mathbf{F}} = \underset{\sim}{\mathbf{B}} \tag{14}$$

$$\underset{\sim}{\mathbf{F}} = (\text{Fw}_1, \ldots, \text{Fw}_P, \quad \text{Fd}_1, \ldots, \text{Fd}_P)^{\text{T}} \tag{15}$$

$$\underset{\sim}{\mathbf{A}} = \begin{pmatrix} 1 & & & -1 & & \\ & \ddots & & & \ddots & \\ & & 1 & & & -1 \\ C_1^{\text{out,max}} - C_{\text{w}} & & & & & \\ & \ddots & & & & \\ & & C_P^{\text{out,max}} - C_{\text{w}} & & & \end{pmatrix} \begin{matrix} 1 \\ \vdots \\ P \\ P+1 \\ \vdots \\ 2P \end{matrix}$$
$$\begin{matrix} 1 & \cdots & P & P+1 & \cdots & 2P \end{matrix} \tag{16}$$

$$\underset{\sim}{\mathbf{B}} = \begin{pmatrix} \sum_{j \in \text{P}, j \neq 1} F_{1,j} - \sum_{j \in \text{P}, j \neq 1} F_{j,1} \\ \vdots \\ \sum_{j \in \text{P}, j \neq P} F_{P,j} - \sum_{j \in \text{P}, j \neq P} F_{j,P} \\ \sum_{j \in \text{P}, j \neq 1} [F_{j,1}(C_j^{\text{out,max}} - C_1^{\text{out,max}})] + M_1 \\ \vdots \\ \sum_{j \in \text{P}, j \neq P} [F_{j,P}(C_j^{\text{out,max}} - C_P^{\text{out,max}})] + M_P \end{pmatrix} \tag{17}$$

The objective function for a single contaminant system may be defined as:

$$\min F(\underset{\sim}{\mathbf{X}}) = \sum_{i \in \text{P}} |\text{Fw}_i| + P(\underset{\sim}{\mathbf{X}}) \tag{18}$$

where $\underset{\sim}{\mathbf{X}}$ is a vector composed of the random variables in the search space, i.e., an agent. $P(\underset{\sim}{\mathbf{X}})$ is the penalty function which can be calculated by using the following equation:

$$P(\underset{\sim}{\mathbf{X}}) = a\sqrt{\sum_{i \in \text{P}} \min(\text{Fw}_i, 0)^2 + \sum_{i \in \text{P}} \min(\text{Fd}_i, 0)^2} + b\sqrt{\sum_{i \in \text{P}} \min((C_i^{\text{in,max}} - C_i^{\text{in}}), 0)^2} \tag{19}$$

where a and b are penalty coefficients. $P(\underset{\sim}{\mathbf{X}})$ may be regarded as the distance between $\underset{\sim}{\mathbf{X}}$ and the feasible region and Fw_i, Fd_i and C_i^{in} come from $\underset{\sim}{\mathbf{X}}$. Apparently, the farther the distance is, the greater the $F(\underset{\sim}{\mathbf{X}})$ is.

4.2.2. Multiple contaminants systems

In a multiple contaminants system that contains P units and K contaminants, there are $KP+P$ equality constraints coming from P water balance constraints and KP contaminant balance constraints around each unit. Eq. (7) permits to explicitly define the set of restrictions in term of the relationship between $C_{i,k}^{\mathrm{in}}$ and maximum allowable inlet concentrations of the pollutants.

There are P^2+P variables representing the flowrates of freshwater, wastewater and reused water stream and PK variables representing outlet concentrations. Therefore, the total number of variables for a multiple contaminants system is P^2+P+PK. Hence, the number of random variables is equal to P^2. In this work, the flowrates of freshwater and reused water streams are selected to be the random variables. The flowrates of wastewater streams and outlet concentrations $C_{i,k}^{\mathrm{out}}$ are dependent variables, and the inlet concentrations $C_{i,k}^{\mathrm{in}}$ are determined variables. Furthermore, if N streams including freshwater and reused water can be canceled by Pinch rules, the number of random variables becomes P^2-N. The relationship between random and dependent variables can be formulated as the following matrix equation:

$$\underset{\sim}{\mathbf{A}}\underset{\sim}{\mathbf{R}} = \underset{\sim}{\mathbf{B}} \tag{20}$$

$$\underset{\sim}{\mathbf{R}} = (\mathrm{Fd}_1, \ldots, \mathrm{Fd}_P, \quad C_{1,1}^{\mathrm{out}}, \ldots, C_{P,K}^{\mathrm{out}})^{\mathrm{T}} \tag{21}$$

$$\underset{\sim}{\mathbf{B}} = \begin{pmatrix} \mathrm{Fw}_1 + \sum_{j\in \mathrm{P}, j\neq 1} F_{j,1} - \sum_{j\in \mathrm{P}, j\neq 1} F_{1,j} \\ \vdots \\ \mathrm{Fw}_P + \sum_{j\in \mathrm{P}, j\neq P} F_{j,P} - \sum_{j\in \mathrm{P}, j\neq P} F_{P,j} \\ M_{1,1} + \mathrm{Fw}_1\, C_{\mathrm{w},1} \\ \vdots \\ M_{P,K} + \mathrm{Fw}_P\, C_{\mathrm{w},K} \end{pmatrix} \tag{22}$$

$\underset{\sim}{\mathbf{A}}$, the coefficient matrix, can be get from Eqs. (5) and (6).

The objective function for a multiple contaminants system is the same as that for a single contaminant system, but the penalty function should be redesigned in the following form:

$$P(\underset{\sim}{\mathbf{X}}) = c\sqrt{\sum_{i\in \mathrm{P}} \min(\mathrm{Fd}_i, 0)^2 + \sum_{i\in \mathrm{P}, k\in \mathrm{C}} \min(C_{i,k}^{\mathrm{out}}, 0)^2} + d\sqrt{\sum_{i\in \mathrm{P}, k\in \mathrm{C}} \min((C_{i,k}^{\mathrm{out,max}} - C_{i,k}^{\mathrm{out}}), 0)^2} + e\sqrt{\sum_{i\in \mathrm{P}, k\in \mathrm{C}} \min((C_{i,k}^{\mathrm{in,max}} - C_{i,k}^{\mathrm{in}}), 0)^2} \tag{23}$$

where c, d and e are penalty coefficients.

5. Simplifying the structure of the networks

The mathematical model can also consider other objectives or constraints. For example, consider to simplify the structure of the networks.

In water-using networks design, there might be many different networks that satisfy the fresh water target, so it is necessary to simplify the structure of the networks by reducing the number of connections between units, which will directly affect the capital cost of the network. The method for simplifying the structure of the networks is the same as that for finding the minimum freshwater consumption, except the objective function should be formulated as follows:

$$\min\left(\sum_{i\in \mathrm{P}} y_i^{\mathrm{w}} + \sum_{i\in \mathrm{P}}\sum_{j\in \mathrm{P}, j\neq i} y_{i,j} + \sum_{i\in \mathrm{P}} y_i^{\mathrm{d}} + h\left|\min\sum_{i\in \mathrm{P}}\mathrm{Fw}_i - \sum_{i\in \mathrm{P}}\mathrm{Fw}_i\right| + P(\underset{\sim}{\boldsymbol{X}})\right) \tag{24}$$

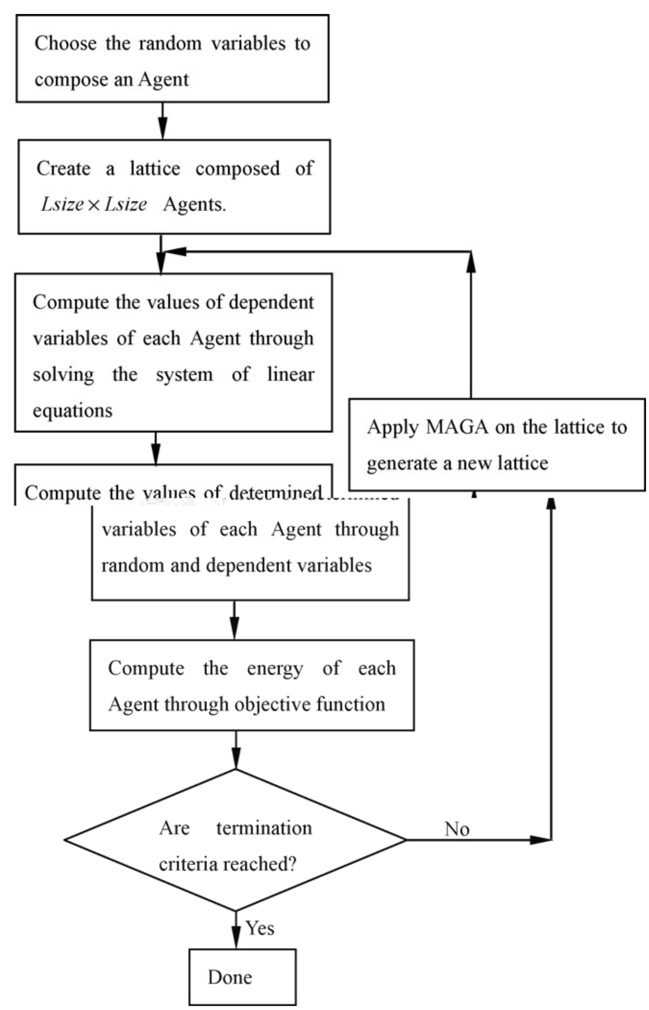

Fig. 4. The procedure of PMAGA for optimizing water-using networks.

Table 1
Limiting water data for case study 1

Process unit	M (g/h)	$C^{\text{in,max}}$ (ppm)	$C^{\text{in,out}}$ (ppm)
1	1000	0	100
2	1000	0	50
3	500	50	100
4	625	75	200
5	4000	100	300

$$\begin{aligned} &y_i^{\mathrm{w}} = 0, \quad \text{if } |\mathrm{Fw}_i| \le 10^{-4}, \quad \text{otherwise } y_i^{\mathrm{w}} = 1 \\ &y_{i,j} = 0, \quad \text{if } |F_{i,j}| \le 10^{-4}, \quad \text{otherwise } y_{i,j} = 1 \\ &y_i^{\mathrm{d}} = 0, \quad \text{if } |\mathrm{Fd}_i| \le 10^{-4}, \quad \text{otherwise } y_i^{\mathrm{d}} = 1 \end{aligned} \qquad (25)$$

where $\min\sum_{i\in \mathrm{P}}\mathrm{Fw}_i$ is the minimum freshwater consumption, $P(\underset{\sim}{\mathbf{X}})$ the penalty function and h is the penalty coefficient.

6. Case study

Before case study, it is necessary to summarize the PMAGA. Fig. 4 illustrates its procedure.

In this work, the concentration of contaminant in freshwater is zero and the PMAGA parameters are set as follows:

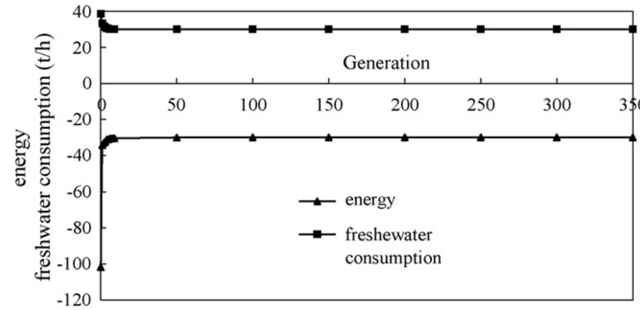

Fig. 5. Agent energy and freshwater consumption trace of Best^t for case study 1.

$$\mathrm{Lsize} = 5, \quad P_0 = 0.2, \quad P_{\mathrm{c}} = 0.2, \quad P_{\mathrm{m}} = 0.2,$$
$$\mathrm{sLsize} = 5, \quad \mathrm{sRadius} = 0.2, \quad \mathrm{sP_m} = 0.15, \quad \mathrm{sGen} = 10,$$
$$a = 1, \quad b = 1, \quad c = 1, \quad d = 1, \quad e = 1, \quad h = 1$$

This program is written in JAVA-language and runs on an Intel Celeron 1.1 GHz computer which is equivalent roughly to 1000 MHz computer used by Prakotpol and Srinophakun (2004).

6.1. Single contaminant system

This system, involving five process units, is taken from Mann and Liu (1999), with the limiting water data for the process units shown in Table 1.

Table 2
The PMAGA performance on an agent across generations for case study 1

Generation	Random variables								
	$F_{1,3}$	$F_{1,4}$	$F_{1,5}$	$F_{2,3}$	$F_{2,4}$	$F_{2,5}$	$F_{3,4}$	$F_{3,5}$	$F_{4,5}$
0	6.83	5.93	9.98	6.64	4.52	4.88	5.93	5.54	8.74
1	5.83	0.41	3.51	1.95	3.86	10.00	0.00	0.61	5.42
2	1.91	0.28	7.20	0.00	3.31	7.88	1.33	0.33	4.95
3	1.37	0.28	3.40	6.82	4.29	10.00	0.29	3.90	0.00
4	1.57	0.26	7.75	6.18	3.90	6.76	0.26	3.79	0.00
5	1.57	0.26	7.75	6.18	3.90	6.76	0.26	3.79	0.00
...					...				

Generation	Dependent variables									
	Fw_1	Fw_2	Fw_3	Fw_4	Fw_5	Fd_1	Fd_2	Fd_3	Fd_4	Fd_5
0	10.00	20.00	1.68	−6.19	−3.99	−12.73	3.97	3.67	1.45	25.15
1	10.00	20.00	4.02	0.02	0.45	0.25	4.19	11.20	−1.12	19.98
2	10.00	20.00	5.00	−0.17	0.10	0.61	8.81	5.25	−0.19	20.46
3	10.00	20.00	1.59	−0.38	0.13	4.94	−1.11	5.59	4.48	17.44
4	10.00	20.00	1.91	−0.06	0.01	0.43	3.15	5.61	4.36	18.30
5	10.00	20.00	1.91	−0.06	0.01	0.43	3.15	5.61	4.36	18.30
...						...				

Generation	Determined variables					Energy
	C_1^{in}	C_2^{in}	C_3^{in}	C_4^{in}	C_5^{in}	
0	0.00	0.00	66.99	138.63	140.93	−170.96
1	0.00	0.00	57.66	54.54	99.83	−45.53
2	0.00	0.00	27.66	68.65	104.47	−40.94
3	0.00	0.00	48.90	60.55	70.59	−35.26
4	0.00	0.00	48.23	56.71	81.47	−32.36
5	0.00	0.00	48.23	56.71	81.47	−32.36
...				...		

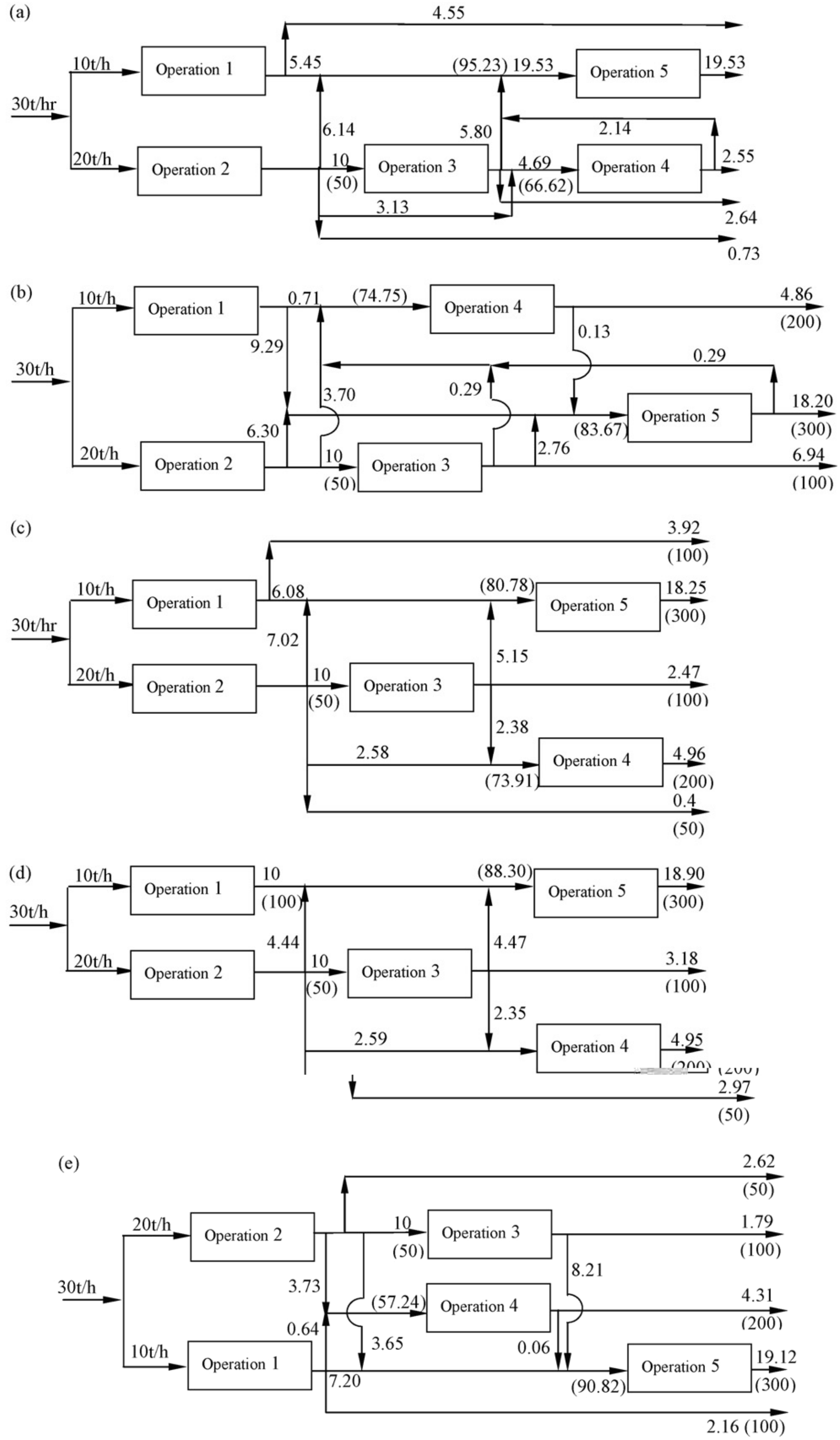

Fig. 6. Optimum water-using network for an example problem with single contaminant: (a) results from Prakotpol and Srinophakun (2004); (b) results from Lingo; and (c–e) results from PMAGA.

The number of the total variables of this process is $P^2+P=30$. It is readily observed that 2 freshwater and 11 reused water do not exist after the limiting data are analyzed by Water Pinch method:

$$\mathrm{Fw}_4, \mathrm{Fw}_5, F_{1,2}, F_{2,1}, F_{3,1}, F_{4,1}, F_{5,1}, F_{3,2}, F_{4,2}, F_{5,2}, F_{4,3}, F_{5,3}, F_{5,4}$$

Thus, the number of the total variables of this process is decreased to 17. The number of random variables for this system is $P^2-P-N=9$ and the random variables are:

$$F_{1,3}, F_{1,4}, F_{1,5}, F_{2,3}, F_{2,4}, F_{2,5}, F_{3,4}, F_{3,5}, F_{4,5}$$

Because PMAGA is a stochastic search algorithm, it can produce different networks that satisfy the fresh water target every time the program runs. The program runs three times, and three different water-using networks are constructed. An agent whose position is (2, 3) is randomly selected to show the current state of an agent at (t) and its future state ($t+1$), which is shown in Table 2. Fig. 5 shows the agent energy and freshwater consumption trace of Best^t. Note that the agent energy increases and freshwater consumption decreases as the generations increase. The networks are shown in Fig. 6 and compared to the results from Prakotpol and Srinophakun (2004) and from the commercial software Lingo. Note that the number in brackets is concentration in ppm.

As can be seen that the minimum freshwater consumption and minimum wastewater generated are the same at 30 t/h for the five networks, but the configurations are different. Some of the networks are simple while others are complex, which gives engineers more options to choose. Moreover, the computation time of PMAGA is about 0.3 min, which is 98% shorter than that of Prakotpol and Srinophakun (2004), 14.58 min. In order to get the simplest network, the number of connections between the process units is optimized by PMAGA. Fig. 7 shows the agent energy and the number of connections trace of Best^t. As can be seen that the number of connections of the simplest network, shown in Fig. 8, is 9.

6.2. Multiple contaminants system

This system, involving three process units and three contaminants, is taken from Mann and Liu (1999), with the limiting water data for the process units shown in Table 3.

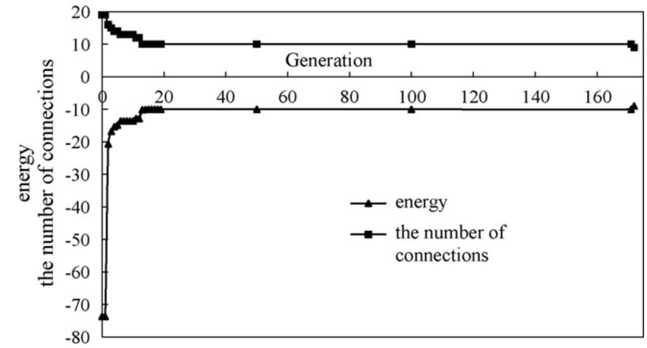

Fig. 7. Agent energy and the number of connections trace of Best^t.

Table 3
Limiting water data for case study 2

Process unit	Contaminant	M (g/h)	$C^{\mathrm{in,max}}$ (ppm)	$C^{\mathrm{out,max}}$ (ppm)
1	A	3000	0	100
	B	2400	0	80
	C	1800	0	60
2	A	4000	50	150
	B	3000	40	115
	C	3600	15	105
3	A	1500	50	125
	B	600	50	80
	C	2000	30	130

The number of the total variables and random variables is $P^2+P+PK=9+3+9=21$ and $P^2=9$, respectively. The random variables are: Fw_1, Fw_2, Fw_3, $F_{1,2}$, $F_{1,3}$, $F_{2,1}$, $F_{2,3}$, $F_{3,1}$, $F_{3,2}$. The dependent variables are: Fd_1, Fd_2, Fd_3, $C^{\mathrm{out}}_{1,1}$, $C^{\mathrm{out}}_{1,2}$, $C^{\mathrm{out}}_{1,3}$, $C^{\mathrm{out}}_{2,1}$, $C^{\mathrm{out}}_{2,2}$, $C^{\mathrm{out}}_{2,3}$, $C^{\mathrm{out}}_{3,1}$, $C^{\mathrm{out}}_{3,2}$, $C^{\mathrm{out}}_{3,3}$. It is readily observed that $F_{2,1}$ does not exist after the limiting data is analyzed by Water Pinch method. Because maximum allowable inlet concentration of unit 1 is zero, $F_{3,1}$ does not exist. Thus, the number of the random variables is decreased to 7.

Table 4 shows the current state of a randomly selected agent whose position is (2, 3) at (t) and its future state ($t+1$). Fig. 9 shows the agent energy and freshwater consumption trace of Best^t. The network by using PMAGA is shown in Fig. 10. As can be seen that the freshwater consumption and minimum wastewater generated are the same at 70 t/h, which is 9.67 t/h less than that of by using the commercial software Lingo, 79.67 t/h

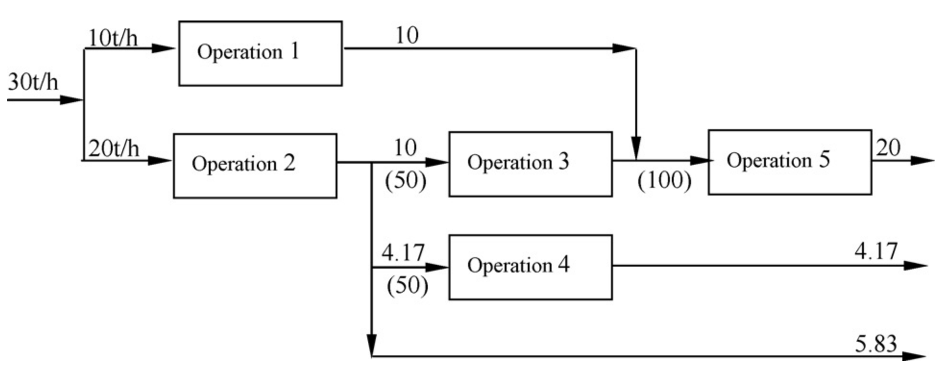

Fig. 8. The simplest water-using network.

Table 4
The PMAGA performance on an agent across generations for case study 2

Generation	Random variables						
	Fw_1	Fw_2	Fw_3	$F_{1,2}$	$F_{1,3}$	$F_{2,3}$	$F_{3,2}$
0	19.70	33.31	27.57	32.84	5.50	4.17	8.75
1	19.70	33.31	27.57	32.84	5.50	4.17	8.75
2	29.99	27.10	14.46	3.84	11.54	0.00	0.00
3	31.97	27.32	10.25	15.51	10.31	0.00	0.00
4	31.97	27.32	10.25	15.51	10.31	0.00	0.00
5	31.07	28.81	9.54	13.88	10.62	0.00	0.00
...				...			

Generation	Dependent variables											
	Fd_1	Fd_2	Fd_3	$C^{out}_{1,1}$	$C^{out}_{1,2}$	$C^{out}_{1,3}$	$C^{out}_{2,1}$	$C^{out}_{2,2}$	$C^{out}_{2,3}$	$C^{out}_{3,1}$	$C^{out}_{3,2}$	$C^{out}_{3,3}$
0	−18.64	70.72	28.49	152.28	121.83	91.37	129.21	98.75	97.26	77.24	45.16	78.09
1	−18.64	70.72	28.49	152.28	121.83	91.37	129.21	98.75	97.26	77.24	45.16	78.09
2	14.61	30.94	26.00	100.03	80.03	60.02	141.70	106.90	123.81	102.10	58.60	103.57
3	6.15	42.83	20.56	93.84	75.07	56.30	127.38	97.23	104.44	120.02	66.84	125.52
4	6.15	42.83	20.56	93.84	75.07	56.30	127.38	97.23	104.44	120.02	66.84	125.52
5	6.57	42.70	20.16	96.55	77.24	57.93	125.08	95.38	103.15	125.29	70.47	129.75
...							...					

Generation	Determined variables									Energy
	$C^{in}_{1,1}$	$C^{in}_{1,2}$	$C^{in}_{1,3}$	$C^{in}_{2,1}$	$C^{in}_{2,2}$	$C^{in}_{2,3}$	$C^{in}_{3,1}$	$C^{in}_{3,2}$	$C^{in}_{3,3}$	
0	0.00	0.00	0.00	75.80	58.70	49.19	36.96	29.05	24.39	−260.18
1	0.00	0.00	0.00	75.80	58.70	49.19	36.96	29.05	24.39	−260.18
2	0.00	0.00	0.00	12.43	9.94	7.46	44.40	35.52	26.64	−126.12
3	0.00	0.00	0.00	33.99	27.19	20.39	47.07	37.65	28.24	−109.70
4	0.00	0.00	0.00	33.99	27.19	20.39	47.07	37.65	28.24	−109.70
5	0.00	0.00	0.00	31.40	25.12	18.84	50.87	40.70	30.52	−108.39
...						...				

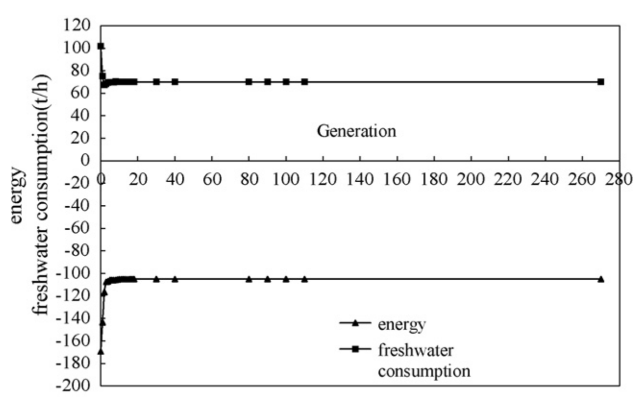

Fig. 9. Agent energy and freshwater consumption trace of Best^t for case study 2.

(Prakotpol & Srinophakun, 2004). Although the result is the same as Prakotpol and Srinophakun (2004), the computation time of PMAGA is about 0.8 min, which is much shorter than that of Prakotpol and Srinophakun (2004), 206.18 min. Because there are only three process units, only one network can be obtained.

7. Conclusion

In this work, a modified multi-agent genetic algorithm, PMAGA, is developed to optimize the water-using networks. The optimization model can be formulated as NLP. This NLP targets to minimize the total freshwater feed to the process subject to water and contaminant mass balance around each operation and maximum inlet and outlet concentration of each

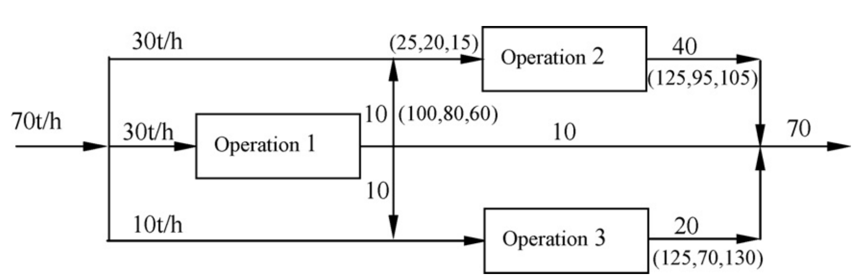

Fig. 10. Optimum water-using network for an example with multiple contaminants.

operation. Water Pinch rules are applied to decrease the dimensional of the search space. The problem of equality constraints is handled by dividing all variables into three groups: random, dependent and determined variables. The values of random variables come from randomization according to genetic methods and the values of dependent variables come from solving a linear system of equations simultaneously after assigning the values of random variables into these equations. Then the values of determined variables come from the random and dependent variables. Penalty function is adopted to solve the inequality constraints problem.

PMAGA is tested and the results are compared to the ones given by using the commercial software Lingo and from literatures. For single contaminant, the results from PMAGA, literatures and Lingo reach the same value of minimum freshwater consumption, but PMAGA can present many different configurations, while other algorithms including Lingo can present only one configuration. PMAGA can also optimize the structure of networks by reducing the number of connections between the unit operations. For multiple contaminants, PMAGA gives better or equal minimum freshwater consumption when compared to Lingo because this software package could stuck in a local optimum while PMAGA has an ability to escape from there. Furthermore, the computational time of PMAGA is much shorter than other GAs, which means it is more effective.

Acknowledgements

Financial support provided by the National Natural Science Foundation of China under Grant no. 20436040 and the National Key Basic Research Development Program of China (no. 2003CB214500) are gratefully acknowledged.

References

Bagajewicz, M. J., Rivas, M., & Savelski, M. J. (2001). Algorithmic procedure to design water utilization systems featuring a single contaminant in process plants. *Chemical Engineering Science*, *56*, 1897–1911.

Doyle, S. J., & Smith, R. (1997). Targeting water reuse with multiple contaminants. *Transactions of the Institute of Chemical Engineering B*, *73*(3), 181–189.

El-Halwagi, M. M. (1997). *Pollution prevention through process integration—Systematic design tools*. CA, USA: Academic Press.

El-Halwagi, M. M., Gabriel, F., & Harell, D. (2003). Rigorous graphical targeting for resource conservation via material recycle/reuse networks. *Process Design and Control*, *42*, 4319–4328.

Feng, X., & Seider, W. D. (2001). A new structure and design methodology for water networks. *Industrial Engineering Chemistry Research*, *40*, 6140–6146.

Hallale, N. (2002). A new graphical targeting method for water minimization. *Advances in Environmental Research*, *6*, 377–390.

Hallale, N., & Fraser, D. M. (2000). Super targeting for mass exchange networks. *Transactions of the Institution of Chemical Engineers, Part A*, *78*, 202–216.

Huang, C. H., Chang, C. T., Ling, H. C., & Chang, C. C. (1999). A mathematical model for water usage and treatment network design. *Industrial and Engineering Chemistry Research*, *38*, 2666–2679.

Karuppiah, R., & Grossmann, I. E. (2006). Global optimization for the synthesis of integrated water systems in chemical processes. *Computers and Chemical Engineering*, *30*, 650–673.

Keedwell, E., & Khu, S. T. (2005). A hybrid genetic algorithm for the design of water distribution networks. *Engineering Applications of Artificial Intelligence*, *18*, 461–472.

Keedwell, E., & Khu, S. T. (2006). Novel cellular automata approach to optimal water distribution network design. *Journal of Computing in Civil Engineering*, (January/February), 49–56.

Kuo, W. J., & Smith, R. (1998). Designing for the interactions between water-use and effluent treatment. *Transactions of the Institute of Chemical Engineering A*, *76*, 287–301.

Lavric, V., Iancu, P., & Plesu, V. (2005). Genetic algorithm optimization of water consumption and waste consumption and wastewater network topology. *Journal of Cleaner Production*, *13*, 1405–1415.

Lee, S., & Grossmann, I. E. (2000). New algorithms for nonlinear generalized disjunctive programming. *Computers and Chemical Engineering*, *24*(9–10), 2125–2141.

Lee, S., & Grossmann, I. E. (2003). Global optimization of nonlinear generalized disjunctive programming with bilinear equality constraints: Applications to process networks. *Computers and Chemical Engineering*, *27*, 1557–1575.

Leung, Y. W., & Wang, Y. (2001). An orthogonal genetic algorithm with quantization for global numerical optimization. *IEEE Transactions on Evolutionary Computation*, *5*(1), 41–53.

Li, Y., & Yao, P. J. (2004). Optimal design of water utilization networks by combination of water pinch analysis and mathematical programming. *Journal of Chemical Industry and Engineering (China)*, *55*, 220–225.

Linhoff, B., & Smith, R. (1994). Recent development in pinch analysis. In *Paper presented at process system engineering*.

Manan, Z. A., Wan Alwi, S. R., & Ujang, Z. (2006). Water pinch analysis for an urban system: A case study on the Sultan Ismail Mosque at the Universiti Teknologi Malaysia (UTM). *Desalination*, *194*, 52–68.

Mann, J. G., & Liu, Y. A. (1999). *Industrial water reuse and wastewater minimization*. NY, USA: McGraw-Hill Companies.

Prakash, R., & Shenoy, U. V. (2005). Design and evolution of water networks by source shifts. *Chemical Engineering Science*, *60*, 2089–2093.

Prakotpol, D., & Srinophakun, T. (2004). GAPinch: Genetic algorithm toolbox for water pinch technology. *Chemical Engineering and Processing*, *43*, 203–217.

Raman, R., & Grossmann, I. E. (1994). Modelling and computational techniques for logic based integer programming. *Computers and Chemical Engineering*, *18*(7), 563–578.

Russell, S., & Norvig, P. (1995). *Artificial intelligence: A modern approach*. NY, USA: Prentice-Hall.

Savelski, M., & Bagajewicz, M. (2003). On the necessary conditions of optimality of water utilizations systems in process plants with multiple contaminants. *Chemical Engineering Science*, *58*, 5349–5362.

Vecchietti, A., & Grossmann, I. E. (1999). LOGMIP: A disjunctive 0–1 nonlinear optimizer for process systems models. *Computers and Chemical Engineering*, *23*, 555–565.

Wang, Y. P., & Smith, R. (1994). Wastewater minimization. *Chemical Engineering Science*, *49*, 981–1006.

Wang, Y. P., & Smith, R. (1995). Wastewater minimization with flowrate constrains. *Transactions of the Institute of Chemical Engineering A*, *73*, 889–904.

Zhong, W., Liu, J., Xue, M., & Jiao, L. (2003). Global numerical optimization using multi-agent genetic algorithm. In *Paper presented at the proceedings of the fifth international conference on computational intelligence and multimedia applications* (pp. 165–170). Xi'an, China: IEEE Computer Society.