Computers and Chemical Engineering 78 (2015) 1–9

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Efficient ant colony optimization for computer aided molecular design: Case study solvent selection problem

Berhane H. Gebreslassie, Urmila M. Diwekar*

Center for Uncertain Systems: Tools for Optimization & Management (CUSTOM), Vishwamitra Research Institute, Crystal Lake, IL 60012 United States

ARTICLE INFO

Article history:
Received 18 July 2014
Received in revised form 22 February 2015
Accepted 5 April 2015
Available online 11 April 2015

Keywords:
Ant colony optimization
Group contribution method
Computer aided molecular design
Hammersley sequence sampling
Oracle penalty function
UNIFAC

ABSTRACT

In this paper, we propose a novel computer-aided molecular design (CAMD) methodology for the design of optimal solvents based on an efficient ant colony optimization (EACO) algorithm. The molecular design problem is formulated as a mixed integer nonlinear programming (MINLP) model in which a solvent performance measure is maximized (solute distribution coefficient) subject to structural feasibility, property, and process constraints. In developing the EACO algorithm, the better uniformity property of Hammersley sequence sampling (HSS) is exploited. The capabilities of the proposed methodology are illustrated using a real world case study for the design of an optimal solvent for extraction of acetic acid from waste process stream using liquid–liquid extraction. The UNIFAC model based on the infinite dilution activity coefficient is used to estimate the mixture properties. New solvents with better targeted properties are proposed.

1. Introduction

Solvents are used for a variety of purposes in process industries. They are extensively used as process materials, as extracting agents, and as process liquids in process industries, pharmaceutical industries, and solvent based industries. Waste solvents are main source of pollution to air, water, and soil. Therefore, it is empirical to use environmentally benign solvents without compromising the process performance. Moreover, there are some solvents that must be eliminated because of environmental and health effects and regulatory requirements (Karunanithi et al., 2005; Kim and Diwekar, 2002b,c; Xu and Diwekar, 2005). The environmental awareness, the strict legislation and the need for high performance solvents have resulted in the search for less toxic and environmentally benign solvents and solvent formulations that have improved performance characteristics. Several methodologies have been developed for solvent selection over the years. The first approach uses traditional laboratory synthesis and test methodology to find promising solvents. This method can provide reliable and accurate results, but in many cases this approach is limited by cost, safety, and time constraints. The second approach is to screen the property database. Though, the screening of the database is the most common and simple method, it is limited by size and accuracy of the database. These methods are usually expensive and time-consuming.

Solvent selection based on computer aided molecular design (CAMD) is fast emerging systematic tool for efficient and reliable design of candidate solvents from their fundamental building blocks (Marrero and Gani, 2001; Karunanithi et al., 2005; Kim and Diwekar, 2002b,c; Xu and Diwekar, 2005). Beyond the solvent selection, the CAMD technique is practiced with great success in different disciplines such as pharmaceutical process designs (Gernaey and Gani, 2010), polymer design (Satyanarayana et al., 2009), and bioethanol production (Alvarado-Morales et al., 2009). CAMD is generating large number of structural molecules with desired properties from a small set of structural groups (building blocks). CAMD is the reverse use of the group contribution method. Different solution strategies are implemented to solve CAMD techniques: heuristic numeration (Hostrup et al., 1999; Li et al., 2002), knowledge based technique (Harper and Gani, 2000; Yamamoto and Tochigi, 2008), molecular property clusters with algebraic equations (Chemmangattuvalappil et al., 2009; Eljack and Eden, 2008; Kazantzi et al., 2007) and optimization-based methods (Karunanithi et al., 2005; Samudra and Sahinidis, 2013; Diwekar and Shastri, 2011; Ostrovsky et al., 2002).

In the optimization approaches, because of the nonlinearity behavior of the UNIFAC model, the CAMD for solvent selection is formulated as a mixed integer nonlinear programming problem (MINLP) that seeks to optimize the desired properties of the solvent molecules subject to molecular design feasibility rules. To solve the MINLP formulation of the CAMD problems, different optimization

* Corresponding author. Tel.: +1 6308863047.
E-mail addresses: berhane@vri-custom.org (B.H. Gebreslassie), Urmila@vri-custom.org (U.M. Diwekar).

http://dx.doi.org/10.1016/j.compchemeng.2015.04.004
0098-1354/

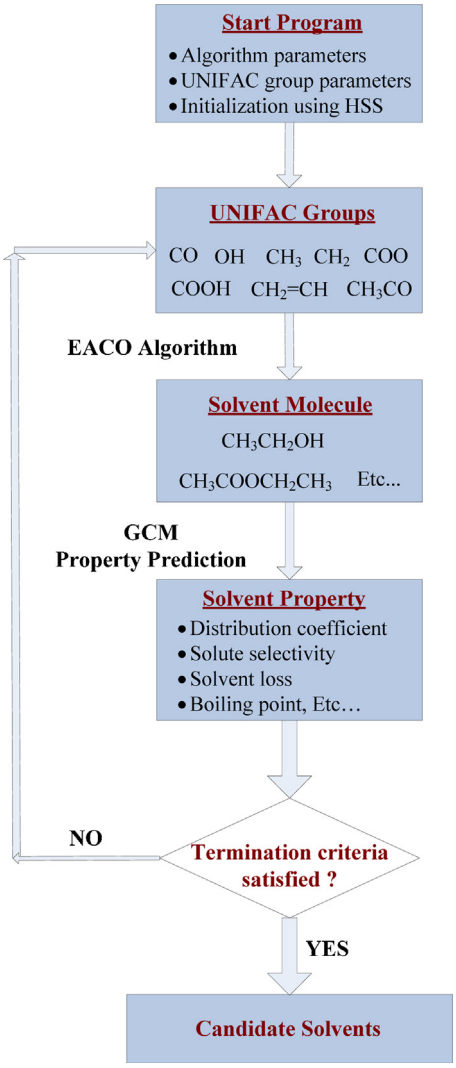

Fig. 1. A basic algorithm for solvent selection using EACO.

methods have been proposed: decomposition methods that use local optimizers for the NLP sub-problem (Odele and Macchietto, 1993; Harper et al., 1999; Karunanithi et al., 2005, 2006), global optimization (Ostrovsky et al., 2002; Samudra and Sahinidis, 2013), interval analysis (Achenie and Sinha, 2003), and dynamic optimization (Giovanoglou et al., 2003). Recently, probabilistic methods such as simulated annealing (Kim and Diwekar, 2002a,b,c), genetic algorithms (Cheng and Wang, 2008; Diwekar and Xu, 2005; Xu and Diwekar, 2005) are adopted as an alternative to the local optimization strategies to find better solutions.

Ant colony optimization (ACO) algorithm proposed by (Dorigo, 1992) is a metaheuristic optimization strategy that can provide a viable alternative to solve the MINLP of the CAMD problem. Although in recent years, there has been a significant research interest in developing and implementing for different applications, to the authors knowledge, there is no work in the literature that implement efficient ant colony (EACO) algorithm to solve the MINLP CAMD problems.

The algorithm implemented in this work to solve the solvent selection problem combines the CAMD and the EACO algorithm proposed by (Gebreslassie and Diwekar, 2015) as shown in Fig. 1. The algorithm parameters, the UNIFAC building block groups and their properties such as the volume and surface area parameters, and the interaction parameters between groups are first introduced. These UNIFAC groups are uniquely designed to generate all possible molecules by exploring all possible combinations. Using the group combinations from this set of groups, solvent molecules are generated. For example, as shown in the algorithm, ethanol (CH_3CH_2OH) is generated from the CH_3, CH_2, and OH groups. The number of combinations can be reduced by introducing constraints from physical and chemical properties, structural feasibility as well as those from regulatory restrictions. Once molecules are generated, the properties of the molecules are predicted based on the properties of their groups using the UNIFAC model. This method can generate candidate solvents with a reasonable accuracy. The rest of the paper is organized as follows. Section 2 describes the CAMD model formulation. The optimization problem of the solvent selection under study is formulated as an MINLP model in Section 3. Ant colony optimization and the proposed EACO algorithm to solve the MINLP optimization problem is presented in Section 4 followed by Section 5 that discusses the result. Finally, Section 6 presents the concluding remarks of the paper.

2. Solvent selection model formulation

To replace the current solvent or formulate a new one, there are several criteria and solvent properties that can be reviewed such as the solute distribution coefficient m, solvent selectivity β, solvent loss S_L, and physical properties like boiling point, density, viscosity, and so on.

2.1. Distribution coefficient (m)

It is a measure of solvent capacity and it represents the solute distribution between the solvent and the raffinate phases. A high value of m reduces the size of extraction equipment and the amount of recycling solvent. The solute distribution coefficient is estimated as shown in Eq. (1).

$$m = \frac{\text{concentration of solute in extractive phase}}{\text{concentration of solute in raffinate phase}} \cong \frac{\gamma^{\infty}_{BA}}{\gamma^{\infty}_{BS}} \frac{MW_A}{MW_S} \quad (1)$$

where A, B, and S represent the raffinate (a nonpolluting molecule e.g. water), solute (polluting molecule e.g. acetic acid), and solvent (e.g. ethanol) phases, respectively. MW denotes the molecular weight, and γ^{∞} is the infinite dilution activity coefficient.

2.2. Solvent selectivity (β)

It is the ratio between the solute distribution coefficient and raffinate. It estimates the ability of the solvent to selectively dissolve a solute (polluting molecule). A high solvent selectivity value thus can reduce the cost of solute recovery and it is defined as shown in Eq. (2).

$$\beta = \frac{\text{distribution coefficient of solute}}{\text{distribution coefficient of solvent}} = \frac{m_B}{m_A} \cong \frac{\gamma^{\infty}_{AS}}{\gamma^{\infty}_{BS}} \frac{MW_B}{MW_A} \quad (2)$$

2.3. Solvent loss (S_L)

It is a measure of the concentration of solvent in raffinate phase. It is the measure of the solvent loss tendency; a low solvent loss value means high selectivity toward solute and determines the immiscibility between the solvent and the raffinate. It is defined as shown in Eq. (3).

$$S_L = \text{concentration of solvent in raffinate phase} \cong \frac{1}{\gamma^{\infty}_{SA}} \frac{MW_S}{MW_A} \quad (3)$$

In Eqs. (1)–(3), the distribution coefficient, the solvent selectivity and the solvent loss properties are given as a function of the infinite dilution activity coefficients (γ^{∞}). It shows the non-ideality of the mixtures (A–B, A–S, and B–S). If the mixture is ideal, γ^{∞} is

close to 1. Otherwise, tends to be greater than one or close to zero. γ^{∞} is a function of groups, temperature T, pressure P, and concentration. There are several group contribution methods for the prediction of γ^{∞}. The most popular method is the UNIFAC group contribution method (Harper et al., 1999; Hostrup et al., 1999; Kim and Diwekar, 2002c; Xu and Diwekar, 2005).

In the UNIFAC model, the activity coefficient (γ_i) of a molecule i have two parts: the combinatorial and residual part.

$$\ln \gamma_i = \ln \gamma_i^C + \ln \gamma_i^R \tag{4}$$

The combinatorial part reflects the volume and surface area of each molecule; hence, the volume q_m and surface area r_m parameters of each group m in molecule i are involved in estimating the combinatorial part (γ_i^C). The residual part represents the interaction energies of the molecules; hence, the volume parameter and interaction parameters (a_{mn}; a_{nm}) between groups m and n in the mixture are required to predict γ_i^R. The interaction parameters can be obtained by regression of experimental data of the mixture. The infinite dilution activity coefficient of molecule i in mixture is limiting activity coefficient when the concentration of molecule i in the mixture tends to zero. That is, γ_i is function of the volume and surface area parameters, interaction parameters, temperature, pressure, and concentration. In this paper, the infinite dilution activity coefficients are calculated by the fifth revised original UNIFAC model (Hansen et al., 1991; Kim and Diwekar, 2002c) in which the original UNIFAC model is used with new groups and a revised data table. The UNIFAC model is summarized in Eqs. (5)–(8).

$$\ln \gamma_i^C = \ln\left(\frac{\Phi_i}{x_i}\right) + \frac{z}{2} q_i \ln\left(\frac{\theta_i}{\Phi_i}\right) + l_i - \frac{\Phi_i}{x_i}\sum_j x_j l_j \tag{5}$$

$$\ln \gamma_i^R = \sum_k \upsilon_k^i \left(\ln \Gamma_k - \ln \Gamma_k^i\right) \tag{6}$$

$$\ln \Gamma_k = Q_k \left[1 - \ln\left(\sum_m \theta_m \psi_{mk}\right) - \sum_m \frac{\theta_m \psi_{km}}{\sum_n \theta_n \psi_{nm}}\right] \tag{7}$$

where

$$\theta_m = \frac{Q_m X_m}{\sum_m Q_m X_m},\ \psi_{mn} = \exp\left(-\frac{a_{mn}}{T}\right),\ \theta_i = \frac{q_i x_i}{\sum_j q_j x_j},$$

$$\Phi_i = \frac{r_i x_i}{\sum_j r_j x_j} \tag{8}$$

In these equations, x_i is the mole fraction of component i, θ_i is the area fraction, Φ_i is the segment fraction, and r_i and q_i are measures of molecular van der Waals volume and molecular surface area, respectively. θ_m is the area fraction of group m, X_m the mole fraction of group m in the mixture, and a_{mn} the group interaction parameter. The combinatorial part, $\ln \gamma_i^C$, is dependent on the sizes and shapes of the molecules, whereas the residual part, $\ln \gamma_i^R$, is dependent on group areas and group interactions.

2.4. Normal boiling point (T_{bp})

It is estimated using the following linear prediction model (Joback and Reid, 1987).

$$T_{bp} = \sum_i^{N_1} t_a \left(N_2^i\right) + t_b \tag{9}$$

To estimate T_{bp} using Eq. (9), the boiling point group contribution parameters t_a and t_b are required and they are given in Table A2 of the Appendix.

For structural feasibility of a solvent configuration, the octet rule relates the total number of free attachments of groups within the solvent molecule and the number of groups in the solvent molecule. Therefore, to determine the chemical feasibility the octet rule is used for acyclic groups as shown in Eq. (10).

$$\sum_i^{N_1} b_i = 2(N_1 - 1) \tag{10}$$

where b_i is the number of free attachments in a group index i. Both high boiling point and low boiling point solvents are considered in the configuration.

3. Solvent selection optimization problem

The design task is finally posed as a mixed integer nonlinear programming (MINLP) problem that seeks to maximize the solute distribution coefficient of the candidate solvent subject to structural feasibility constraint (Eq. (10)), solvent performance property constraints (Eqs. (2), (3) and (9)), the UNIFAC mixture property constraints (Eqs. (4)–(8)). The lower and upper limit constraints related to the number of building block groups, type and the total number of groups making up the solvent molecule are also added. The problem formulation for an acetic acid extraction (one of the case studies widely studied for the applications of CAMD; Odele and Macchietto, 1993, Hostrup et al., 1999, Kim and Diwekar, 2002c), which is commonly used as a process solvent or produced as a byproduct, is given in Eq. (11). Because acetic acid can be a pollutant as well as a valuable solvent, it is desirable to minimize the discharge of acetic acid to the environment. To recycle or remove acetic acid from waste process streams, extraction process is commonly utilized. For the extraction process, one can either use high-boiling solvents or low boiling solvents (Joback and Reid, 1987; Kim and Diwekar, 2002c) depending on the process considered. Ethyl acetate, isoamyl acetate, and isopropyl acetate are widely used in industries to extract acetic acid. Ethyl acetate, which is one of the common solvents for acetic acid extraction, has high m (0.3156), but it unfortunately also has high S_L (0.0560).

In this work, the high-boiling and low-boiling solvent candidates are generated by EACO algorithm and compared with candidate solvents proposed from the literature solved using other heuristic optimization strategies and decomposition methods. The optimal high m and low S_L solvent can be easily separated from the extract stream and then recycled to the extraction equipment. As shown in Eqs. (4)–(8), the solvent selection optimization problem Eq. (11) is formulated based on γ^{∞}.

$$\begin{aligned} \min_{N_1 N_2^i} \quad & -m \\ s.t \quad & \beta \geq \beta^{\min} \\ & S_L \leq S_L^{\max} \\ & T_{bp}^{\min} \leq T_{bp} \leq T_{bp}^{\max} \\ & 1 \leq N_1 \leq 10 \\ & 1 \leq N_2^i \leq 24 \qquad \forall i \in N_1 \end{aligned} \tag{11}$$

The boundaries on the constraints are taken from the properties of the current practice of the solvent for acetic acid extraction (Kim and Diwekar, 2002c). In this optimization problem, the discrete decision variables are the number of groups N_1 involved in a solvent molecule and the group index (the type of building blocks) $N_2^i; i \in \{1, \ldots, N_1\}$ of that molecule. From the set of building blocks

a unique solvent molecule that has high solute distribution coefficient and satisfies the selectivity, solvent loss, normal boiling point, and structural feasibility constraints can be generated.

Due to the equality (Eqs. (2)–(10)) and inequalities such as the boundary constraints, the solvent selection problem is highly constrained optimization problem. However, the conventional ACO algorithm handles unconstrained optimization problems. In EACO algorithm, to handle constrained optimization problems Oracle penalty method (Schluter and Gerdts, 2010) is used. For details of this method please refer to Gebreslassie and Diwekar (2015).

4. Solution method

The solvent selection MINLP problem can be solved using different approaches but finding the optimal solution is not trivial. The equality constraints representing the property models are nonlinear and the gradient based methods such as Branch and Bound (BB), Generalized Bender's Decomposition (GBD), and Outer-Approximation (OA), are generally used for solving MINLP problems. However, these methods have limitations whenever, the optimization problems do not satisfy convexity conditions, the problems have large combinatorial explosion, or the search domain is discontinuous (Diwekar and Xu, 2005). Metaheuristic optimization strategies such as simulated annealing (SA) (Kirkpatrick et al., 1983), genetic algorithm (GA) (Holland, 1975) and ant colony optimization (ACO) (Dorigo, 1992) provide a viable alternative to the gradient based mathematical programming techniques. Although in recent years, there has been a significant research interest in developing ant colony optimization algorithms. To the best of the authors' knowledge, there has not been utilized to solve the CAMD problems.

The general perspective of the ACO algorithm is introduced below. The ACO is a metaheuristic class of optimization algorithm inspired by the foraging behavior of real ants (Dorigo and Stutzle, 2004). Natural ants randomly search food by exploring the area around their nest. If an ant locates a food source, while returning back to the nest, it lay down a chemical pheromone trail that marks its path. This pheromone trail will indirectly communicate with other members of the ant colony to follow the path. Over time, the pheromone will start to evaporate and therefore reduce the attraction of the path. The routes that are used frequently will have higher concentration of the pheromone trial and remain attractive. Thus, the shorter the route between the nest and food source imply short cycle time for the ants and these routes will have higher concentration of pheromone than the longer routes. Consequently, more ants are attracted by the shorter paths in the future. Finally, the shortest path will be discovered by the ant colony (Dorigo and Stutzle, 2004; Zecchin et al., 2006).

In ACO algorithms, artificial ants are stochastic candidate solution construction procedures that exploit a pheromone model and possibly available heuristic information of the mathematical model. The artificial pheromone trails (numeric values) are the sole means of communication among the artificial ants. Pheromone decay, a mechanism analogous to the evaporation of the pheromone trial of the real ant colony allows the artificial ants to forget the past history and focus on new promising search directions. Like the natural ants, by updating the pheromone values according to the information learned in each of the preceding iterations, the algorithmic procedure leads to very good and hopefully, a global optimal solution. It was originally introduced to solve combinatorial optimization problems, in which decision variables are characterized by a finite set of components. However, in recent years, its adaptation to solve continuous (Liao et al., 2011, 2014; Socha and Blum, 2007; Socha and Dorigo, 2008) and mixed variable (Schluter and Gerdts, 2010; Schluter et al., 2012; Socha, 2004) programming problems has received an increasing attention.

4.1. Efficient ant colony optimization (EACO) algorithm

One of the simplest and most widely used methods of random sampling is the Monte Carlo method. Monte Carlo method is a numerical method that provides approximate solution to a variety of physical and mathematical problems by random sampling. In crude Monte Carlo approach, a value is drawn at random from the probability distribution for each input, and the corresponding output value is computed. The entire process is repeated n times producing n corresponding output values. These output values constitute a random sample from the probability distribution over the output induced by the probability distributions over the inputs. The advantage of this approach is that the precision of the output distribution can be estimated using standard statistical techniques. The pseudorandom number generator produces samples that may be clustered in certain regions of the population and does not produce uniform samples. Therefore, in order to reach high accuracy, larger sample sizes are needed, which adversely affects the computational efficiency (Diwekar and Kalagnanam, 1997; Diwekar and Ulas, 2007). Gebreslassie and Diwekar (2015) proposed EACO algorithm that improves the performance of the conventional ACO algorithm for combinatorial, continuous and mixed variable optimization problems by introducing the Hammersley sequence sampling technique (HSS). The initial solution archive diversity for continuous and mixed-variable optimization problems plays an important role in the performance of ACO algorithm. The uniformity property of the HSS technique is exploited to avoid clustering of the initial solution archive in a small region of the potential solution space. Moreover, ACO algorithm is a probabilistic method; hence several random probability functions are involved in the algorithm procedure. For instance, for combinatorial ACO algorithm, the transition probability that help to choose the next solution component and for continuous and mixed-variable optimization problems, the probability of choosing ant guide from the solution archive to construct and sample the Gaussian kernel. The distribution of the random numbers generated for the acceptance probability of a solution component affects the performance of the ACO algorithm. At this stage, the multidimensional uniformity property of HSS is introduced to improve the computational efficiency of the ACO algorithm. The detail presentation of the EACO algorithms can be viewed (Gebreslassie and Diwekar, 2015). The major steps in EACO algorithm are shown in Table 1 and the algorithm proposed in this work that combines CAMD and EACO algorithm is given in Fig. 1.

5. Results and discussion

5.1. Solvent selection results

The building block group indexes used in the case study are summarized in Appendix A Table A2. The total number of groups used in this work is 24 and maximum of 10 groups per molecule are allowed. Therefore, the search space is composed of $24^{10}(6.34\times10^{13})$ combinations. The interaction parameter between the building groups and the three UNIFAC parameters: surface area, volume, and interaction parameters, as well as the boiling point parameters, the group free attachments and the molecular weights are tabulated in Appendix A Tables A1 and A2, respectively.

The algorithm terminate if it reaches maximum number of iterations (*MaxIter*), or if the tolerance (ϵ) that is the relative difference between solutions found in two consecutive iterations is lower than or equal to the parameter ϵ for a set of consecutive number of iterations *ICON*. The algorithm parameters used to solve the problems

Table 1
EACO algorithm for mixed integer nonlinear optimization problems.

Start program
- Set K, nAnts, NC, NO, NT, NOPT, ρ, q, ξ and termination criteria
- Initialize solution archive $T(K, \text{NC} + \text{NO})$ using HSS
- Initialize solution archive $T(K, \text{NT})$ randomly from the possible options
- Combine and evaluate the objective function of the K solutions $T(K, \text{NDIM})$
- Rank solutions based on the quality of the objective function ($T = \text{rank}(S_1, \ldots, S_K)$)
- For categorical optimization problems, introduce multidimensional random number generated using HSS rand (IterMax × nAnts × NT, NOPT)

While termination criterion is not satisfied
- Generate solutions equivalent to the number of ants

For all # nAnts
- Incremental solution construction

For all # NDIM
- ○ Probabilistically construct continuous decision variables
- ○ Probabilistically construct ordinal decision variables
- ○ Probabilistically construct categorical decision variables

End for # NDIM
- Store and evaluate the objective function of the newly generated solutions

End for # nAnts
- Combine, rank and select the best K solutions, $T = \text{Best}(\text{rank}(S_1, \ldots, S_K, \ldots, S_{K+n\text{Ant}}), K)$
- Update solution

End while
End program

T is solution archive and K is size of T. nAnts is the number of ants. NC, NO, NT and NDIM are the number of continuous, ordinal, categorical and the total number of decision variables, respectively. NOPT is the number of options in categorical variables.

are selected after performing a number of experimentations using different combination of the parameters. The algorithm parameters used for the EACO algorithm are the archive sizes $K = 1500$, the number of ants $n\text{Ants} = 30$, $q = 1\text{E}{-3}$ and the tolerance $\epsilon = 1\text{E}{-6}$, evaporation parameter $\rho = 0.75$.

An attractive solvent should have a high distribution coefficient, selectivity and low solvent loss. In this case, the specifications are set to the above values in order to screen out the least desirable candidates. The EACO algorithm generated more than 30 solvents which have higher solute distribution value than the current practice of solvent (Ethyl acetate) for acetic acid extraction, which also satisfy the given constraints as shown in Eq. (11). The first top 15 solvents for high boiling temperature solvents and the first top seven low boiling temperature solvents are summarized in Tables 2 and 3, respectively. As shown in the tables, most of the high ranked solvent molecules are ethers, alcohols and aldehydes. Hostrup et al. (1999) has enlisted seven candidate solvent molecules for the extraction of acetic acid from water and among these molecules based on the environmental and toxicological properties of the candidates, two of the solvents (diisobutylketone and heptyl acetate) with low hazard were proposed as promising solvent substitutes for the removal of acetic acid from water. The solute distribution co-efficient of these seven candidate solvents ranges from 0.11 (methyl decanoate) to 0.5 (1-nonanol), which is relatively low compared to the result from this work that are enlisted in Tables 2 and 3. Among, the seven candidate solvents only one (1-nonanol) would make the list. A high value of m reduces the size of extraction equipment and the amount of recycling solvent. In other words, these high m solvents will reduce the investment cost and the operational cost related to the energy consumption of the solvent recovery. From Tables 2 and 3, it can be concluded that the EACO algorithm can generate a set of candidate solvents with a better solute distribution coefficient and low solvent loss properties. These candidate solvents may need further investigation related to environmental and toxicological properties and cost.

Comparing the results from EACO algorithm shown in Tables 2 and 3 and the results reported by Diwekar and co-workers (Kim and Diwekar, 2002b; Xu and Diwekar, 2005) for similar solvent selection problem obtained from heuristic algorithm; Efficient Simulated Annealing (ESA), the best solvent molecule found by EACO algorithm with configuration ($2CH_2$, C, CH_2=CH, CH=CH, CH_3OH, CH_3O) has solute distribution $m = 0.691$, which exceeds the optimal solvent (CH_3, $6CH_2$, OH) found by ESA with $m = 0.6074$. Moreover, the EACO algorithm has found a set of candidate solvent

Table 2
High boiling point temperature candidate solvents.

Rank	Solvent	m	β	S_L	T_{bp}
1	$2CH_2$, C, CH_2=CH, CH=CH, CH_3OH, CH_3O	0.691	8.847	0.010	517.7
2	CH_2, CH, C, $2CH_2$=CH, CH_3OH, CH_3O	0.683	8.660	0.009	509.7
3	CH_3, CH, CH_2—CH, CH_2=C, C=C, CH_3OH, CH_3O	0.628	8.060	0.006	539.6
4	$2CH_3$, $3CH_2$, CH=C, OH	0.615	11.388	0.008	455.9
5	$2CH_3$, CH_2, CH=CH, CH=C, OH	0.613	7.755	0.009	460.1
6	$2CH_3$, CH, 2CH=CH, OH	0.609	7.585	0.008	459.7
7	CH_3, CH_2=CH, 2CH=C, CH_3OH, CH_2O	0.597	8.214	0.008	524.8
8	$2CH_2$=CH, CH=C, CH_3O, CH_2O	0.580	8.080	0.005	424.8
9	$2CH_2$=CH, CH_2=C, CH_3OH, 2CHO	0.566	7.966	0.005	531.5
10	$2CH_2$, CH=CH, CH=C, CH_3OH	0.517	9.129	0.009	452.6
11	CH_3, CH_2=CH, CH=CH, CH_3CO, CHO	0.451	7.241	0.006	459.3
12	CH_3, CH, $2CH_2$=C, $2CH_3CO$	0.448	12.003	0.009	528.7
13	$2CH_2$, CH=CH, CH_2=C, $2CH_3CO$	0.445	12.568	0.008	536.8
14	CH_3, 2CH=CH, CH_3CO	0.444	14.467	0.007	421.9
15	$3CH_2$=C, $2CH_3CO$	0.437	9.397	0.008	525.7

Table 3
Low boiling point temperature candidate solvents.

Rank	Solvent	m	β	S_L	T_{bp}
1	CH_2=CH, CH_2=C, CH_3O, CH_2O	0.751	7.79	0.033	374.9
2	CH_2=CH, CH=CH, CH_3O	0.709	9.25	0.019	337.2
3	CH_3, CH_2=CH, $2CH_2O$	0.666	7.58	0.044	355.4
4	CH_3, CH_2=C, CH_3CO	0.612	14.59	0.052	364.4
5	CH_3, CH=CH, CH_3CO	0.610	16.10	0.048	372.0
6	$2CH_3$, $2CH_2$, COO	0.344	17.88	0.033	372.1
7	$2CH_2$=CH, COO	0.294	7.57	0.021	365.5

Table 4
EACO algorithm compared to decomposition method of Karunanithi et al. (2005).

Method	Solvent	m	β	S_L	T_{bp}
Karunanithi et al. (2005)	$1CH_3$, $3CH_2$, $1CH_3CO$	0.491	11	0.0038	404.1
EACO	$3CH_3$, CH_2, C, CH_3COO	0.533	11.8	0.0036	463.2
EACO	$2CH_3$, $3CH_2$, CH_2O, CH_2COO	0.507	39.2	0.0037	414.7

molecules with better solute distribution coefficients than the first ranked solvent molecule found by the ESA. There are two reasons for this phenomenon: (1) the EACO algorithm finds a set of solutions equal to the number of ants used by the algorithm instead of one as in the simulated annealing, and this property enables EACO algorithm the ability to cover the search space easily and gives a better chance to find the global optimum; and (2) in simulated annealing, an infeasible solution is accepted or discarded randomly according to the Metropolis criterion. The disadvantage of this constraint violation handling strategy is that there is no standard for accepting or discarding solutions. Therefore, some infeasible solutions that have more chances to reach the optimum may be discarded and, consequently, some good patterns are lost in the process. In EACO algorithm the selection of q parameter is carefully chosen in such a way to focus initially at exploration and when the quality of the solution is improved to focus on exploitation and hence the probability of choosing the high ranked solutions in the solution archive become higher because of the accumulation of pheromone dominates the evaporation as the quality of the solution increases. These two properties of EACO algorithm improves the efficiency of the search process and makes it less susceptible to being trapped in local optima.

Before discussing the results of the low boiling temperature solvents, the authors would like to acknowledge data discrepancy related to HCOO— building block. Excluding solvents that include HCOO— functional group, comparing Table 2 that enlists the low-boiling point candidate solvents found by EACO algorithm, and the low-boiling point candidate solvents found by ESA, the EACO algorithm found a better optimal solvent configuration (CH_2═CH, CH_2═C, CH_3O, CH_2O) with $m = 0.751$ than the ESA with solvent configuration of (CH_3, CH_2═CH, CH_3O, CH—O) and $m = 0.66$ and the efficient genetic algorithm (EGA) with solvent configuration of ($2CH_3$, CH_2, CH_2CO) and $m = 0.5083$. We have observed that like EGA, the EACO algorithm does not find as many potential solvents as ESA. As explained by Xu and Diwekar (2005), this phenomena can be explained by the fact that in EACO algorithm, the q parameter focuses initially at exploration and as the quality of the solution improves, the algorithm focuses of the exploitation and hence the algorithm does not traverse as many local optima as ESA, which gives EACO a quicker convergence and a lesser probability of getting trapped in local optima. Moreover, in EACO algorithm, each ant constructs its own solution independently. Therefore, at each iteration, the different ant solution configurations leads to better solutions than only one solution configuration at each iteration like the ESA. This characteristics places EACO algorithm in a better position to cover the search space and find the global optimal solution than the ESA that may be trapped in local optima.

Moreover, the results from the EACO algorithm are also compared with the results of decomposition method proposed by Karunanithi et al. (2005). In the decomposition method, the original MINLP problem is first reformulated into MILP master problem and NLP sub problems and solved using the gradient method solvers. It couples the integer solution from an MILP master problem and solution of an inner NLP sub-problem to solve the original MINLP formulation of the of CAMD problem (Eq. (11)). Model parameters are first modified to adapt the solvent selection problem by Karunanithi et al. (2005). The total number of UNIFAC groups used for the comparison is 16. Similar to the original model, maximum of 10 building blocks per molecule are allowed. Therefore, in this case the search space is composed of 16^{10}(1.099E+12) combinations. The parameters used for the EACO algorithm are the archive sizes $K = 1500$, number of ants nAnt = 30, algorithm parameter $q = 1E{-}3$ and tolerance $\epsilon = 1E{-}6$. The property constraints such as the lower bound of the boiling point, solute distribution, and selectivity are 340 K, 0.49 and 11, respectively and the upper bound of the solvent loss is 0.0038 as given in Karunanithi et al. (2005). The results are tabulated in Table 4. As shown in the table, in terms of the desired thermodynamic properties of solvents, the solvents generated using the EACO algorithm are superior to the 2-hexananone solvent proposed from the decomposition method by Karunanithi et al. (2005). The proposed solution strategy provides solvents with better thermodynamic properties and from this it can be concluded that the proposed methodology can be a useful alternative to optimization of large scale CAMD problems.

5.2. EACO algorithms vs. conventional ACO for CAMD

The CAMD optimization problem results from EACO and the conventional ACO algorithms to the high and low boiling point temperature solvents are presented in Tables 5 and 6. The parameters used for the ACO algorithms are two archive sizes ($K = 500$, and $K = 1000$), nAnts = 30, $q = 1E{-}3$, $\epsilon = 1E{-}6$ and $\rho = 0.75$. As shown in the tables, on all cases, the performance of the EACO algorithm out performs the conventional ACO algorithm and the performance improvement ranges from 23.7% to 52.9%. The performance improvement is because of the multidimentional uniformity property of HSS, the EACO algorithm needs less iteration than the conventional ACO to find the optimal candidate solvents.

Sample of the convergence path of the EACO versus the conventional ACO for the CAMD problem is given in Fig. 2. The figure presents the trajectories of the solute distribution coefficient of the CAMD optimization problem as a function of the number of iterations to reach the optimal solution. As shown in the figure, the EACO algorithm found the optimal solution at 68th iteration. However, the conventional ACO needs 85 iterations to reach a local optimal value. Moreover, as shown in the figure, the EACO finds a feasible solution at 5th iteration while the conventional ACO finds the first

Table 5
High boiling point temperature solvent.

K	EACO		ACO		Improve (%)
	m	Iter	m	Iter	
500	0.61	108	0.62	153	29.4
1000	0.61	64	0.62	136	52.9

Table 6
Low boiling point temperature solvent.

K	EACO		ACO		Improve (%)
	m	Iter	m	Iter	
500	0.75	29	0.75	38	23.7
1000	0.71	62	0.71	87	28.7

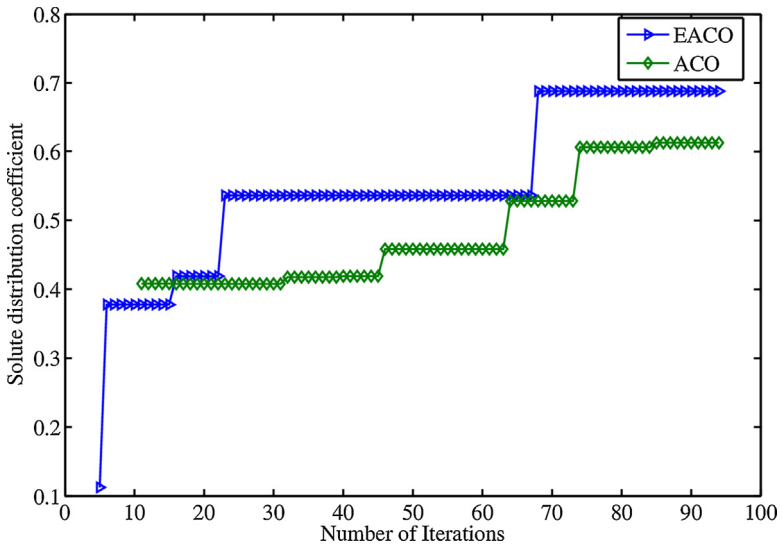

Fig. 2. EACO vs. ACO algorithm trajectory of solving CAMD optimization problem.

feasible solution at the 11th iteration. All the above observations prove that EACO algorithm benefits from the uniformity property of HSS.

5.3. Ordinal vs. categorical EACO algorithms to solve CAMD

Depending on the type of the discrete decision variables (i.e. ordinal and categorical variables), there are two approaches for solving the combinatorial optimization problems in the EACO algorithm (Gebreslassie and Diwekar, 2015). In the case of ordinal variables, the problem can be tackled through relaxation of the discrete variables, where in CAMD problem represents the index of the UNIFAC building block groups. Therefore, it can be solved by including a heuristic rule to the original algorithm for continuous variable optimization problem. The CAMD problem also can be solved using the EACO algorithm for optimization problems that involve categorical variables. To demonstrate the performance difference between these two approaches, results for high boiling point and low boiling point temperature solvents are presented in Tables 7 and 8. The parameters used for the EACO algorithm are archive sizes $K = 500$, $n\text{Ants} = 30$, $q = 1\text{E}{-3}$ and tolerance $\epsilon = 1\text{E}{-6}$, evaporation parameter $\rho = 0.75$. As shown in the tables, the performance of the EACO algorithm for categorical variables outperforms the EACO algorithm for ordinal variable interims of the number of iterations needed to reach the optimal solution. Iteration improves by 65.7% for low boiling temperature and 70% for high boiling temperature solvents. Moreover, the quality of the solution using the EACO algorithm for categorical variables is better for high boiling temperature solvents. As far as for the low boiling temperature solvents EACO algorithm for ordinal discrete variables performs better.

Socha (2009) reported that ACO algorithm for ordinal variables perform better on problems containing discrete variables that can be ordered and ACO algorithm for optimization problems that involve categorical variables perform better on problems where a proper ordering is not possible, or unknown (categorical variables). The CAMD optimization problem only involves discrete variables that can be ordered. However, the results show that in most case the performance of the EACO algorithm for categorical variables perform better than the algorithm for ordinal variables.

Table 7
High boiling point temperature: algorithm for ordinal vs. categorical.

	Iter	Improve (%)	m	Improve (%)
Ordinal	108	0.0	0.613	0.0
Categorical	37	65.7	0.615	0.3

Table 8
Low boiling point temperature: algorithm for ordinal vs. categorical.

	Iter	Improve (%)	m	Improve (%)
Ordinal	20	0.0	0.751	5.9
Categorical	6	70.0	0.709	0.0

6. Conclusions

This paper proposes a new alternative optimization strategy based on metaheuristic EACO algorithm to solve computer-aided molecular design problems. The proposed methodology involves formulating the solvent selection molecular design problem as an MINLP model and solution method for identifying candidate solvents and proposing the optimal solvent molecule.

The ACO algorithm is a simple to implement and yet an effective optimization framework for handling combinatorial, continuous and mixed-variable optimization problems. In this work, the EACO algorithm as an alternative to the gradient based, simulated annealing and genetic algorithm optimization is implemented to solve CAMD problems. A real world case study of solvent selection for acetic acid extraction from process waste stream is presented and discussed in this work. The GCM technique is implemented to estimate the mixture properties and using EACO algorithm, new solvents with better targeted properties are proposed. The EACO algorithm has found a set of candidate solvent molecules with better solute distribution than the first ranked solvent molecules proposed using ESA and the decomposition methods.

Acknowledgement

We gratefully acknowledge the funding from DOE National Energy Laboratory grant # DEFE0012451.

Appendix A. Parameters

Table A1
Interaction parameters between main group m and n (a_{mn}).

m	n											
	1	2	3	4	5	6	7	8	9	10	11	12
1	0	86.02	986.5	697.2	1318	476.4	677	232.1	507	251.5	663.5	387.1
2	−35.36	0	524.1	787.6	270.6	182.6	448.8	37.85	333.5	214.5	318.9	48.33
3	156.4	457	0	−137.1	353.5	84	−203.6	101.1	267.8	28.06	199	190.3
4	16.51	−12.52	249.1	0	−181	23.39	306.4	−10.72	179.7	−128.6	−202	165.7
5	300	496.1	−229.1	289.6	0	−195.4	−116	72.87	0	540.5	−14.09	−197.5
6	26.76	42.92	164.5	108.7	472.5	0	−37.36	−213.7	−190.4	−103.6	669.4	−18.8
7	505.7	56.3	529	−340.2	480.8	128	0	−110.3	766	304.1	497.5	0
8	114.8	132.1	245.4	249.6	200.8	372.2	185.1	0	−241.8	−235.7	660.2	560.2
9	329.3	110.4	139.4	227.8	0	385.4	−236.5	1167	0	−234	−268.1	−122.3
10	83.36	26.51	237.7	238.4	−314.7	191.1	−7.84	461.3	457.3	0	664.6	417
11	315.3	1264	−151	339.8	−66.17	−297.8	−165.5	−256.3	193.9	−338.5	0	−337
12	529	1397	88.63	171	284.4	123.4	577.5	−234.9	145.4	−247.8	1179	0

Table A2
The surface area R_k and volume Q_k values for the UNIFAC equation, boiling point parameters t_a, free attachments b_i and molecular weight MW.

MG index	Sup groups	SG index	R_k	Q_k	t_a	b_i	MW
1	CH_3—	1	0.9011	0.848	23.58	1	15
1	—CH_2—	2	0.6744	0.54	22.88	2	14
1	—CH⟨	3	0.4469	0.228	21.74	3	13
1	⟩C⟨	4	0.2195	0	18.25	4	12
2	CH_2=CH—	5	1.3454	1.176	43.14	1	27
2	—CH=CH—	6	1.1167	0.867	49.92	2	26
2	CH_2=C⟨	7	1.1173	0.988	42.32	2	26
2	—CH=C⟨	8	0.8886	0.676	49.1	3	25
2	⟩C=C⟨	9	0.6605	0.485	48.28	4	24
3	—OH	10	1	1.2	92.88	1	17
4	CH_3OH	11	1.4311	1.432	116.46	0	32
5	H_2O	12	0.92	1.4	175.03	0	18
6	CH_3CO—	13	1.6724	1.488	100.33	1	43
6	—CH_2CO—	14	1.4457	1.18	99.63	1	42
7	—CHO	15	0.998	0.948	74.74	1	29
8	CH_3COO—	16	1.9031	1.728	104.68	1	59
8	—CH_2COO—	17	1.6764	1.42	103.98	2	58
9	HCOO—	18	1.242	1.188	84.88	1	45
10	CH_3O—	19	1.145	1.088	46	1	31
10	—CH_2O—	20	0.9183	0.78	45.3	2	30
10	⟩CH—O—	21	0.6908	0.468	44.16	3	29
11	—COOH	22	1.3013	1.224	160.8	1	45
11	HCOOH	23	1.528	1.532	175.53	0	46
12	—COO—	24	1.38	1.2	81.1	2	44

References

Achenie LEK, Sinha M. Interval global optimization in solvent design. Reliable Comput 2003;9:317–38.

Alvarado-Morales M, Terra J, Gernaey KV, Woodley JM, Gani RR. Biorefining: computer aided tools for sustainable design and analysis of bioethanol production. Chem Eng Res Des 2009;87:1171–83.

Chemmangattuvalappil NG, Eljack FT, Solvason CC, Eden MR. A novel algorithm for molecular synthesis using enhanced property operators. Comput Chem Eng 2009;33:636–43.

Cheng HC, Wang FS. Optimal biocompatible solvent design for a two-stage extractive fermentation process with cell recycling. Comput Chem Eng 2008;32:1385–96.

Diwekar U, Xu W. Improved genetic algorithms for deterministic optimization and optimization under uncertainty. Part I. algorithms development. Ind Eng Chem Res 2005;44:7132–7.

Diwekar UM, Kalagnanam JR. Efficient sampling technique for optimization under uncertainty. AIChE J 1997;43:440–7.

Diwekar UM, Shastri Y. Design for environment: a state-of-the-art review. Clean Technol Environ Policy 2011;13:227–40.

Diwekar UM, Ulas S. Sampling techniques, vol. 26. Encyclopedia of Chemical Technology; 2007.

Dorigo M. Optimization, learning and natural algorithms. Italy: Dept. of Electronics, Politecnico di Milano; 1992, PhD Thesis.

Dorigo M, Stutzle T. Ant colony optimization theory. A Brandford Book. Cambridge, Massachusetts: The MIT Press; 2004.

Eljack FT, Eden MR. Systematic visual approach to molecular design via property clusters and group contribution methods. Comput Chem Eng 2008;32:3002–10.

Gebreslassie BH, Diwekar UM. Efficient ant colony optimization (EACO) algorithm for deterministic optimization. Swarm Evol Comput 2015 (submitted).

Gernaey KV, Gani R. A model-based systems approach to pharmaceutical product process design and analysis. Chem Eng Sci 2010;65:5757–69.

Giovanoglou A, Barlatier J, Adjiman CS, Pistikopoulos EN, Cordiner JL. Optimal solvent design for batch separation based on economic performance. AIChE J 2003;49:3095–109.

Hansen HK, Rasmussen P, Fredenslund A, Schiller M, Gmehling J. Vapor–liquid equilibria by UNIFAC group contribution. Ind Eng Chem Res 1991;30:2352–5.

Harper P, Gani R, Kolar P, Ishikawa T. Computer-aided molecular design with combined molecular modeling and group contribution. Fluid Phase Equilib 1999;158–160:337–47.

Harper PM, Gani R. A multi-step and multi-level approach for computer aided molecular design. Comput Chem Eng 2000;24:677–83.

Holland JH. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. Ann Arbor, MI: University of Michigan Press; 1975.

Hostrup M, Harper P, Gani R. Design of environmentally benign processes: integration of solvent design and separation process synthesis. Comput Chem Eng 1999;23:1395–414.

Joback KG, Reid RC. Estimation of pure–component properties from group-contributions. Chem Eng Commun 1987;57:233–43.

Karunanithi AT, Achenie LEK, Gani R. A new decomposition-based computer-aided molecular/mixture design methodology for the design of optimal solvents and solvent mixtures. Ind Eng Chem Res 2005;4:4785–97.
Karunanithi AT, Achenie LEK, Gani R. A computer-aided molecular design framework for crystallization solvent design. Chem Eng Sci 2006;61:1247–60.
Kazantzi V, Qin X, El-Halwagi M, Eljack FT, Eden MR. Simultaneous process and molecular design through property clustering. Ind Eng Chem Res 2007;46:3400–9.
Kim K, Diwekar U. Efficient combinatorial optimization under uncertainty. 1. Algorithmic development. Ind Eng Chem Res 2002a;41:1276–84.
Kim K, Diwekar U. Efficient combinatorial optimization under uncertainty. 2. Application to stochastic solvent selection. Ind Eng Chem Res 2002b;41:1285–96.
Kim K, Diwekar U. Hammersley stochastic annealing: efficiency improvement for combinatorial optimization under uncertainty. IIE Trans Inst Ind Eng 2002c;34:761–77.
Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science 1983;220:671–80.
Li M, Harten PF, Cabezas H. Experiences in designing solvents for the environment. Ind Eng Chem Res 2002;41:5867–77.
Liao T, Montes De Oca MA, Aydin D, Stutzle T, Dorigo M. An incremental ant colony algorithm with local search for continuous optimization. In: Proceedings of the 13th annual conference on Genetic and evolutionary computation; 2011. p. 125–32.
Liao T, Stutzle T, Montes De Oca MA, Dorigo M. A unified ant colony optimization algorithm for continuous optimization. Eur J Oper Res 2014;234:597–609.
Marrero J, Gani R. Group – contribution based estimation of pure component properties. Fluid Phase Equilib 2001;184:183–208.
Odele O, Macchietto S. Computer aided molecular design: a novel method for optimal solvent selection. Fluid Phase Equilib 1993;82:47–54.
Ostrovsky GM, Achenie LEK, Sinha M. On the solution of mixed-integer nonlinear programming models for computer aided molecular design. Comput Chem 2002;26:645–60.
Samudra AP, Sahinidis NV. Optimization-based framework for computer-aided molecular design. AIChE J 2013;59(10):3686–701.
Satyanarayana KC, Abildskov J, Gani R. Computer-aided polymer design using group contribution plus property models. Comput Chem Eng 2009;33:1004–13.
Schluter M, Gerdts M. The oracle penalty method. J Global Optim 2010;47:293–325.
Schluter M, Gerdts M, Ruckmann JJ. A numerical study of MIDACO on 100 MINLP benchmarks. Optimization 2012;61:873–900.
Socha K. ACO for continuous and mixed-variable optimization. Lect Notes Comput Sci 2004:25–36.
Socha K. Ant Colony optimization for continuous and mixed-variable domains. IRIDIA, CoDE, Université Libre de Bruxelles; 2009 (PhD thesis, CP 194/6).
Socha K, Blum C. An ant colony optimization algorithm for continuous optimization: application to feed-forward neural network training. Neural Comput Appl 2007;16:235–47.
Socha K, Dorigo M. Ant colony optimization for continuous domains. Eur J Oper Res 2008;185:1155–73.
Xu W, Diwekar U. Improved genetic algorithms for deterministic optimization and optimization under uncertainty. Part II. Solvent selection under uncertainty. Ind Eng Chem Res 2005;44:7138–46.
Yamamoto H, Tochigi K. Computer-aided molecular design to select foaming agents using a neural network method. Ind Eng Chem Res 2008;47:5152–6.
Zecchin A, Simpson A, Maier H, Leonard M, Roberts A, Berrisford M. Application of two ant colony optimization algorithms to water distribution system optimization. Math Comput Model 2006;44:451–68.