CHEMICAL ENGINEERING TRANSACTIONS

VOL. 61, 2017

A publication of AIDIC
The Italian Association of Chemical Engineering
Online at www.aidic.it/cet

Guest Editors: Petar S Varbanov, Rongxin Su, Hon Loong Lam, Xia Liu, Jiří J Klemeš

ISBN 978-88-95608-51-8; **ISSN** 2283-9216

DOI: 10.3303/CET1761116

A Mixed Discrete PSO Algorithm for Synthesis of Mass Exchange Network with Incompatible Multicomponent

Shaohui Tao*, Mengmeng Jiang, Mingyue Liu, Li Ding
College of Chemical Engineering, Qingdao University of Science and Technology, Qingdao, 266042, China
tau@qust.edu.cn

A new mixed discrete particle swarm optimization (PSO) algorithm, i.e. DCH-PSO, was presented to solve mixed integer nonlinear programming (MINLP) problems. The main contribution of DCH-PSO is to produce integer speed for discrete variable by randomly choosing it from an integer speed range, which is determined by analysing the velocity updating formula of original PSO, and the influence of historical velocity to the current one was implemented in a probabilistic way. At last, DCH-PSO was applied to synthesize mass exchange network (MEN) involving incompatible multicomponent. The design of orthogonal experiments was applied to study the effect of algorithm's tuning parameters, and the results show that DCH-PSO is efficient for solving MEN synthesis problems.

1. Introduction

Energy-saving and emission reduction are the prerequisites for sustainable development of chemical industry, so process integration has attracted a great deal of attention in the past decade (Alcamis et al., 2015). Mass exchange network (MEN) synthesis (El-Halwagi and Manousiouthakis, 1990), which can reduce harmful emissions in a cost-saving way, is an important technology of process integration. To rigorously synthesize MEN involving incompatible multicomponent, the numbers of columns' trays shall be taken as decision variables and a mixed integer nonlinear programming (MINLP) problem (Chang et al., 2015) with strong non-linearity and non-convexity has to be solved (Liu, 2013).

There are several types of deterministic algorithm for solving MINLP problems, e.g. branch and bound, outer approximation and extended cutting plane, et al (Grossmann and Trespalacios, 2014). The deterministic algorithms can find solutions for convex MINLP problems efficiently and exactly, but it's usually difficult or even impossible for them to solve non-convex ones (Burer and Letchford, 2012). So, evolutionary algorithm which is not limited to the problem's continuity or convexity and can find a near-optimal solution quickly is attractive in the research field of process synthesis.

As one of the simplest evolutionary algorithms, particle swarm optimization (PSO) algorithm (Kennedy and Eberhart, 1995) can also be applied to solve the MINLP problems (Khare and Rangnekar, 2013). For the existed mixed discrete versions of PSO, there are two ways to handle discrete variables, i.e. binary-coding (Shokrian et al, 2014) and rounding (Pal et al., 2011). For the binary-coding method, discrete variables are coded with 0-1 string, which will increase the dimension of solution and lead to early convergence, especially for large integer variable, e.g. numbers of columns' trays in MEN synthesis problem. For the rounding method, the discrete variable is taken as a continuous variable when updating the solution and then rounded to integer, which will increase the swarm's redundancy and decrease the algorithm's exploration ability, and a random rounding method was proposed to reduce the redundancy of swarm (He and Chen, 2008).

To overcome the disadvantages of discrete variable handling methods mentioned above, the theory base of velocity updating equation, i.e. principle of bird flocking proposed by (Eberhart and Kennedy, 1995), was analyzed in-depth and extended to discrete space in this work, so a discrete version of velocity updating equation was obtained, where the influence of historical speed was implemented in a probabilistic way, then PSO in discrete-continuous hybrid space, i.e. DCH-PSO, was presented.

The aim of this research is to develop a mixed discrete PSO algorithm for solving MINLP problems with high nonlinearity, non-convexity and integers as decision variables.

Please cite this article as: Tao S., Jiang M., Liu M., Ding L., 2017, A mixed discrete pso algorithm for synthesis of mass exchange network with incompatible multicomponent, Chemical Engineering Transactions, 61, 709-714 DOI:10.3303/CET1761116

In this paper, the following results will be presented: (i) the whole flowchart of DCH-PSO algorithm, (ii) the effects of the tuning parameters of DCH-PSO obtained with design of orthogonal experiments, and (iii) the result of DCH-PSO applied to a simplified MEN synthesis problem.

2. Development of DCH-PSO algorithm

2.1 The Original PSO algorithm

To solve a *D*-dimensional optimization problem, the solution is represented by a particle which has two *D*-dimensional vectors representing position and velocity, and solutions pool is called the swarm. For a swarm with *N* particles during the t^{th} iteration, the i^{th} particle's position vector is noted as $x_i^t=(x^t_{i,1}, x^t_{i,2}, \dots, x^t_{i,j}, \dots, x^t_{i,D})$ where $x^t_{i,j}$ means the j^{th} decision variable of the i^{th} particle at iteration *t*, and the velocity vector is noted as $v_i^t = (v^t_{i,1}, v^t_{i,2}, \dots, v^t_{i,j}, \dots, v^t_{i,D})$, the optimization process of PSO is iteratively updating the velocity vector and moving the particle's position according to Eq(1) and Eq(2).

$$v_{i,j}^{t+1} = w^t \cdot v_{i,j}^t + c_1 \cdot r_1 \cdot \left(P_{i,j}^t - x_{i,j}^t\right) + c_2 \cdot r_2 \cdot \left(G_j^t - x_{i,j}^t\right) \tag{1}$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1} \tag{2}$$

In Eq(1) and Eq(2), P_i^t is the best solution experienced by the i^{th} particle and G^t is the best solution experienced by the swarm until the t^{th} iteration, a fitness function is defined according to the optimization problem for comparing particle's performance and determining P_i^t and G^t during search process; w^t is the inertia weight reflecting the influence of v_i^t to v_i^{t+1}; c_1 , i.e. cognitive coefficient, and c_2, i.e. social coefficient, are positive acceleration factors in the range [0, 4] and reflects the impact of P_i^t and G^t to velocity; r_1 and r_2 are uniform random numbers in the range [0, 1]. To control the global exploration ability of the particle, there is a maximum velocity, i.e. v_j^{max}, which is set by k_v $(x_j^U - x_j^L)$, where k_v is a tuning parameter, x_j^U and x_j^L is the upper and lower bound for the j^{th} decision variable of the optimization problem, $|v^t_{i,j}| \le v_j^{max}$ has to be satisfied during iteration.
The inertia weight w^t decreases during the search process, as shown in Eq(3), which means w^t is large initially to emphasize particle's exploration ability and small in the later period to emphasize particle's exploitation ability. Generally, w_0 = 0.7 and w_1 = 0.4 according to the literature (Marini and Walczak, 2015).

$$w^t = w_0 + \frac{w_1 - w_0}{t} \tag{3}$$

Although there are some rules for tuning these parameters (Marini and Walczak, 2015), their values need to be set by trial for a given problem. Obviously, with particle's position $x^t_{i,j}$ being integer initially, they will keep as integer if its speed $v^{t+1}_{i,j}$ kept as integer during the search process, so Eq(1) is analysed in the next section to realize this.

2.2 Analysis of velocity updating equation

It's obvious that the updated velocity vector is composed with three vectors according to Eq(1), as shown in Figure 1 for a 2-dimensional particle, where *a*, *b* and *c* represent $c_1r_1(x_i^t - P_i^t)$, $c_2r_2(x_i^t - G_i^t)$, and $w^tv_i^t$.

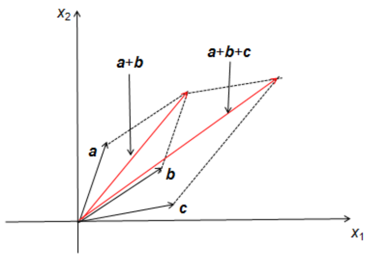

Figure 1: The composed vectors of updated velocity vector.

Since r_1 and r_2 are uniform random numbers in the range [0, 1], the lower bound $l^t_{i,j}$ and upper bound $u^t_{i,j}$ for the j^{th} element in vector *a*+*b* can be obtained with Eq(4) and Eq(5).

$$l_{i,j}^t = \min\left\{0,\ c_1 \cdot \left(P_{i,j}^t - x_{i,j}^t\right)\right\} + \min\left\{0,\ c_2 \cdot \left(G_j^t - x_{i,j}^t\right)\right\} \tag{4}$$

$$u_{i,j}^t = \max\left\{0,\ c_1 \cdot \left(P_{i,j}^t - x_{i,j}^t\right)\right\} + \max\left\{0,\ c_2 \cdot \left(G_j^t - x_{i,j}^t\right)\right\} \tag{5}$$

Eq(4) and Eq(5) provide the possible minimum and maximum value for the j^{th} element of $a+b$, so $v^{t+1}_{i,j}$ must lie in the range $[l^t_{i,j}, u^t_{i,j}]$ and the bounds will be integer with c_1, c_2, $P^t_{i,j}$, G^t_j and $x^t_{i,j}$ being integers initially, which is easy to be satisfied. For continuous variable, there are infinite choices for element of vector $a+b$ with equal probability and the latter two items in right hand side of Eq(1) promises this. While for discrete variable there are only limited choices since both $l^t_{i,j}$ and $u^t_{i,j}$ are limited integers, to randomly select a discrete element of vector $a+b$ from the range, the roulette wheel method, which is used in genetic algorithm (GA) for selecting individual, can be used and the probability for any integer in the range equals to $1/(u^t_{i,j} - l^t_{i,j+1})$, so it's obvious that the generation of integer element of vector $a+b$ is consistent with continuous ones inherently.
Now the first item of Eq(1), i.e. $w^t v^t_{i,j}$, must be considered for discrete variable, which is not as easy as the latter two items in the equation because w^t is not an integer. The essence of $w^t v^t_{i,j}$ is the effect of historical speed to the current one, which decreases during the search process. An equivalent way to reflect the effect of historical speed is proposed here, as shown in Eq(6) and Eq(7).

$$p_w^t = p_w^0 + \frac{p_w^1 - p_w^0}{t} \tag{6}$$

$$w^t = \begin{cases} 1, \ if \ \ rand \geq p_w^t \\ 0, \qquad\quad else \end{cases} \tag{7}$$

In Eq(6), $p_w^1 > p_w^0$, so it provides a decreasing probability for $w^t = 1$, which means the effect of historical speed to current one decreases during the search process and p_w^t is called probability of historical effect here. At last, by choosing an integer in the range provided with Eq(4) and Eq(5) and determining w^t with Eq(6) and Eq(7), a discrete element of $a+b+c$ can be obtained, with $x^0_{i,j}$ and $v^0_{i,j}$ being integers $x^t_{i,j}$ will keep as integers during the search process, and the PSO algorithm handling discrete variables in this way is noted as DCH-PSO here, because the discrete variables move in the discrete space with inherently the same principle as continuous ones in the continuous space.

2.3 Deb's method of handling constraints

To solve an optimization problem with constraints as described with Eq(8), the constraints must be handled and the optimal solution must satisfy them.

$$\begin{aligned} &\min \ \ f(\boldsymbol{x}) \\ &s.t. \ \ \ c_k(\boldsymbol{x}) \leq 0, k = 1, 2, ..., K \\ &\qquad\ \ c_l(\boldsymbol{x}) = 0, l = 1, 2, ..., L \end{aligned} \tag{8}$$

Deb (2000) has proposed a simple method to handle constraints for GA, which is noted as Deb's method here and can also be applied to other evolutionary algorithms with the advantage of simplicity and efficiency. The principle of Deb's method is satisfying constraints first, as shown in Eq(9) for problem (8).

$$F(\boldsymbol{x}) = \begin{cases} f(\boldsymbol{x}), if \ \ c_k(\boldsymbol{x}) \leq 0 \ \forall k \ and \ c_l(\boldsymbol{x}) = 0 \ \forall l \\ f^{\max} + \sum_{k=1}^{K} c_k(\boldsymbol{x}) + \sum_{l=1}^{L} c_l(\boldsymbol{x}), else \end{cases} \tag{9}$$

In Eq(9), $F(x)$ is the fitness of particle, $f(x)$ is the value of objective function and f^{max} is the objective function of the worst feasible solution in the swarm, so it means that the infeasible particle is worse than any feasible particle and the infeasible particle with less constraints violation is better than the infeasible particle with larger one.

2.4 The DCH-PSO algorithm

Combined the velocity updating and constraints handling method with the particle updating process of original PSO, the DCH-PSO steps can be described in the following, where the continuous variables are updated in the same way of original PSO.
Step 1. Set values of algorithm's tuning parameters: swarm size N, acceleration factors c_1 and c_2, initial and final inertia weights for continuous variable, i.e. w_0 and w_1, initial and final probability of historical effect, i.e. p_w^0 and p_w^1, scale factor k_v for maximum velocity.
Step 2. Initialize particles' position and velocity. The roulette wheel method is used to produce initial discrete position in its feasible bound and uniform random generator is used in the feasible bound for continuous variable. The initial discrete velocity for each particle is -1 or 1 according to the random number generated in the range [-1, 1], a negative random number corresponds to -1 and vice versa, while the continuous velocity is just a uniform random number in [-1, 1]. Set iteration number t=0.

Step 3. Calculate each particle's fitness with Eq(9) and select P_i^t and G^t according to the fitness, update the continuous speed of particle with Eq(1), calculate the bounds of discrete speed with Eq(4) and Eq(5), and produce discrete velocity with Eqs(4) - (7) as described in Section 2.2, and then update the particle's position according to Eq(2).
Step 4. Set t = t+1 and update the inertia weights and historical effect probability with Eq(3), Eq(6) and Eq(7). Check whether G^t has stagnates, if not, returns to Step 3, else exit the algorithm.

3. MEN synthesis involving incompatible multicomponent

3.1 Problem description

The MEN synthesis problem to be solved in this work is on removing H_2S and CO_2 from two coke-oven gas streams with two optional lean streams: dilute ammonia solution which is a process lean stream with limited flow-rate and methanol which is a utility lean stream that can be used without limit, the data for the rich and lean streams are listed in Table 1 and 2, where S_1 means dilute ammonia solution and S_2 means methanol.

Table 1: Data of rich streams

No.	G_i(kg s^{-1})	y^s_{H2S}	y^s_{CO2}	y^t_{H2S}	y^t_{CO2}
R_1	0.9	0.07	0.06	0.0003	0.005
R_2	0.1	0.051	0.115	0.0001	0.01

Table 2: Data of lean streams

No.	L_i^{up}(kg·s^{-1})	x^s_{H2S}	x^s_{CO2}	x^t_{H2S}	x^t_{CO2}	Annual cost($·kg^{-1}·y^{-1})
S_1	2.3	0.0006	0	0.031	0.171	117,360
S_2	∞	0.0002	0	0.0035	0.103	176,040

The lean stream price for MEN can be calculated with Ann. cost in Table 2 and the flow-rates of lean streams, the equipment cost is calculated with the tray price, which is 4,552 $·y^{-1} per tray, then the total annual cost (TAC) of MEN can be obtained by the sum of lean stream price and tray price.
The absorption equilibrium functions for H_2S and CO_2 in the two lean streams are described as Eq(10) to Eq(13).

$$y_{H_2S} = 1.45 \cdot x^1_{H_2S} \tag{10}$$

$$y_{CO_2} = 0.35 \cdot x^1_{CO_2} \tag{11}$$

$$y_{H_2S} = 0.26 \cdot x^2_{H_2S} \tag{12}$$

$$y_{CO_2} = 0.58 \cdot x^2_{CO_2} \tag{13}$$

In the above equations, y_{H2S} and y_{CO2} are weight concentration of H_2S and CO_2 in rich stream, either R_1 or R_2; x^1_{H2S} and x^1_{CO2} are weight concentration of H_2S and CO_2 in S_1, x^2_{H2S} and x^2_{CO2} are weight concentration of H_2S and CO_2 in S_2.
This problem is taken from (El-Halwagi and Manousiouthakis, 1990) and solved by (Liu, 2013) and other researchers, which can be solved with a general super structure method. According to the optimization results from literatures (Liu, 2013), the optimal network structure can be simplified as shown in Figure 2.

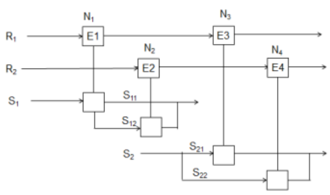

Figure 2: Simplified structure of mass exchange network

Figure 2 means the cheaper lean stream S_1 shall be used first to decrease stream price, then the utility lean stream S_2 is used to exchange with unqualified rich streams. The objective function for synthesis is the TAC of the MEN; the discrete decision variables here are number of trays for each mass exchanger, i.e. N_1, N_2, N_3 and N_4, the continuous decision variables for the problem are the flow-rate of S_{11}, S_{12}, S_{21} and S_{22}; the constraints for the problem are described in (Liu, 2013).

3.2 Synthesis with DCH-PSO

Obviously, the MEN synthesis problem described above is an MINLP problem, since the mass exchange for each exchanger is calculated via Kremser equation (Shenoy and Fraser, 2003) with high non-linearity. To solve this problem with DCH-PSO, the values of algorithm's tuning parameters must be set and the sensitivity of optimization result to the tuning parameters shall be studied too. Since there are 7 tuning parameters for DCH-PSO, a $L_{18}(3^7)$ orthogonal experiments table was used here, and the values for each level of the tuning parameters are listed in Table 3.

Table 3: Orthogonal experiments table for DCH-PSO tuning parameters

Level	Factors						
	N	c_1	c_2	p_w^1	w_0	w_1	k_v
1	40	1	1	0.4	0.6	0.3	0.05
2	50	2	2	0.5	0.7	0.4	0.1
3	60	3	3	0.6	0.8	0.5	0.2

In Table 3, only one parameter for historical effect probability, i.e. p_w^1, was considered and $p_w^0 = 0$ was set, because particles are expected to be exploratory at the beginning.

To show the performance enhancement of DCH-PSO, a rounding version PSO was applied to the synthesis problem too, whose optimal tuning parameters were found according to a $L_{27}(3^6)$ orthogonal experiments and the levels are shown in Table 3. The result of orthogonal experiments is shown in next section.

4. Results

Setting tuning parameters according to Table 3 and running DCH-PSO 50 times for solving the simplified MEN synthesis problem shown in Figure 2 under each experimental condition, with TAC as performance index, Figure 3 was obtained, whose vertical axis is TAC / (10^5 \$·y^{-1}) and horizontal axis is the value for each tuning parameter.

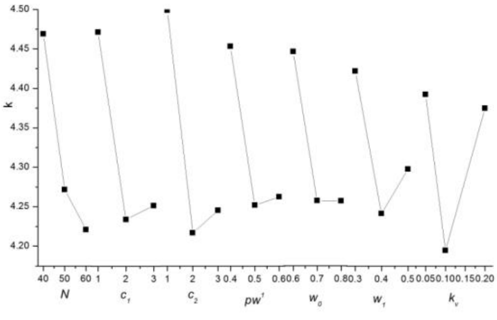

Figure 3: The effects of tuning parameters to DCH-PSO's performance.

As can be seen from Figure 3, each tuning parameter has an optimal value except N, which shows that the algorithm's performance improved with increased swarm size while other tuning parameters' value must be specified according to the given problem. The decreasing order of tuning parameter's influence, i.e. extreme deviation, is c_2, N, c_1, p_w^1, k_v, w_0 and w_1, so c_2 and c_1 are the most important parameters except N, which shows that the global and local best solution affects the search direction seriously, so the algorithm may fall into the local optimal solution easily and the diversity of the swarm must be considered during the search process. The history effect probability parameter p_w^1 for discrete variable is more important than inertia weight parameter w_0 and w_1, which can be explained as the variation of discrete variables has a greater effect than continuous ones. The scale parameter k_v is more important than w_0 and w_1 shows the balance of particle's exploration and exploitation ability depends on the maximum velocity of the particle more, while the influence of inertia weight for continuous variables can not be ignored.

According to the result of orthogonal experiments, the optimal values of tuning parameters for rounding version of PSO are: N = 60, $c_1 = c_2$ = 1, w_0 = 0.7, w_1 = 0.3 and k_v = 0.05, after 50 times running of this version of PSO, the TAC for the worst and best solution is 1,691,627 $\$\cdot y^{-1}$ and 414,855 $\$\cdot y^{-1}$, with average TAC 513,610 $\$\cdot y^{-1}$, the data for the best MEN is N_1 = 11, $N_2 = N_3$ = 4, N_4 =3, F_{S11} = 1.9205 $kg\cdot s^{-1}$, F_{S12} = 0.2753 $kg\cdot s^{-1}$, F_{S21} = 0.2239 $kg\cdot s^{-1}$ and F_{S22} = 0.1 $kg\cdot s^{-1}$. For DCH-PSO, the optimal values for tuning parameters are: N = 60, $c_1 = c_2$ = 2, $p^1{}_w$ =0.5, w_0 = 0.8, w_1 = 0.4 and k_v = 0.1, running DCH-PSO with theses parameters for 50 times, the TAC for the worst and best solution are 417,720 $\$\cdot y^{-1}$ and 413,160 $\$\cdot y^{-1}$, the average TAC is 414,530 $\$\cdot y^{-1}$ and the best result of literature is 413,200 $\$\cdot y^{-1}$ (Liu, 2013). The data for best MEN is N_1 = 11, $N_2 = N_3 = N_4$ = 3, F_{S11} = 1.8615 $kg\cdot s^{-1}$, F_{S12} = 0.3294 $kg\cdot s^{-1}$, F_{S21} = 0.2692 $kg\cdot s^{-1}$ and F_{S22} = 0.1 $kg\cdot s^{-1}$. Obvious performance enhancement of DCH-PSO compared to rounding version can be seen from the results, especially for integer variables, i.e. N_2 and N_3.

5. Conclusions

A new mixed discrete PSO algorithm, i.e. DCH-PSO, is proposed in this research by analysis of velocity updating equation of original PSO algorithm, the most important feature of DCH-PSO is that discrete variables are updated with the same principle of continuous ones, i.e. bird flocking, and the disadvantages of binary-coding and rounding versions of mixed discrete PSO are avoided. A simplified MEN involving incompatible multicomponent is synthesized by DCH-PSO and the result is a little better than the best literature result, and the effects of tuning parameters are also studied with design of orthogonal experiments, which shows that the swarm's diversity may be one of the most important factors for global optimization.

References

Alcamis E., Matino I., Colla V., Maddaloni A., Romaniello L., Rosito F., 2015, Process integration solutions for water networks in integrated steel making plants, Chemical Engineering Transactions, 45, 37-42.

Burer S., Letchford A.N., 2012, Non-convex mixed-integer nonlinear programming: a survey, Surveys in Operations Research and Management Science, 17, 97-106.

Chang C., Wang Y., Feng X., Zhang P., 2015, Efficient solution strategy for stage-wise MINLP model of interplant heat integration using heat recovery loop, Chemical Engineering Transactions, 45, 67-72.

Deb K., 2000, An efficient constraint handling method for genetic algorithms, Computer Methods in Applied Mechanics and Engineering, 186, 311-338.

El-Halwagi M.M., Manousiouthakis V., 1990, Simultaneous synthesis of mass exchange and regeneration networks, America Institute of Chemical Engineering Journal, 36(8), 1209-1219.

Grossmann I.E., Trespalacios F., 2014, Review of Mixed-Integer Nonlinear and Generalized Disjunctive Programming methods in Process Systems Engineering, Chemie Ingenieur Technik, 86(7), 991-1012.

He Y., Chen D., 2008, Hybrid particle swarm optimization algorithm for mixed-integer nonlinear programming, Journal of Zhejiang University (Engineering Science), 42(5), 747-751. (in Chinese)

Kennedy J., Eberhart R.C., 1995, Particle swarm optimization, In Proceedings of the 1995 IEEE International Conference on Neural Networks, 4, 1942-1948.

Liu L., 2013, Study on combined mass and heat exchange networks synthesis for multi-component systems, Dissertation, Dalian University of Science and Technology. (in Chinese)

Marini F., Walczak B., 2015, Particle swarm optimization (PSO), A Tutorial, Chemometrics and Intelligent Laboratory Systems, 149, 153-165.

Pal A., Singh S.B., Deep K., 2011, Use of particle swarm optimization algorithm for solving integer and mixed integer optimization problems, Journal of Computing Science and Communication Technologies, 4(1), 663-667.

Shenoy U. V., Fraser D. M. , 2003, A new formulation of the Kremser equation for sizing mass exchangers, Chemical Engineering Science, 58(22), 5121-5124.

Shokrian M., High K. A., 2014, Application of a multi objective multi-leader particle swarm optimization algorithm on NLP and MINLP problems, Computers and Chemical Engineering, 60, 57-75.