Optimization of Liquid-Liquid Extraction Column Using Genetic Algorithms

Ali D. Ali
Engineering Collage, University of Al-Nahrain/ Baghdad
Email: Ruaaad87@yahoo.com

Received on: 25/5/2011 & Accepted on: 1/3/2012

ABSTRACT

In the present study, liquid-liquid extraction column was optimized using Genetic Algorithms as a non-conventional optimization technique, which scores over conventional techniques. Genetic Algorithm (GA) is a stochastic search technique mimics the principle of natural genetics and natural selection to constitute search and optimization. Genetic Algorithm is applied to the optimal design of liquid-liquid extraction column to maximize the extraction rate using the superficial velocities of raffinate and extract phases, (υ_x, υ_y) respectively as design variables using Matlab GA toolbox. Different Genetic Algorithm strategies were used for optimization and the design parameters such as Population size, crossover rate and Mutation were studied. It was found that for constant distribution coefficient, *m* the convergence is obtained in a very few generations (51 generations). The effect of distribution coefficient, *m* was also studied on the optimization process and found that when increasing the distribution coefficient the optimum extraction rate increased. The best values for υ_x and υ_y were 0.142 and 0.059 respectively, and the objective function (maximum) was 0.2844187.

Keywords: Liquid-liquid extraction column design, Optimization, Genetic Algorithms.

تحسين عمل عمود الاستخلاص (سائل – سائل) باستخدام الخوارزميات الوراثية

الخلاصة

في هذا البحث تم استخدام تقنيات الخوارزميات الوراثية في ايجاد الامثل لبرج الستخلاص سائل - سائل كطريقة غير تقليدية تحاكي التطور الطبيعي للكائنات الحية. تم اعتماد سرعة الطور المستخلص وسرعة الطور المتبقي كمتغيرات تصميمية لايجاد افضل معدل استخلاص. استخدمت ستراتيجيات مختلفة للخوارزميات الوراثية في عملية ايجاد الافضل مثل حجم العينة, معدل الانقسام في الحلول المقترحة (الكروموسومات), الطفرات. عموما وجد بان افضل معدل استخلاص يحصل في الجيل رقم 51 بثبوت علاقة التوازن (m) كذلك تم دراسة تأثير علاقة التوازن على عملية ايجاد الافضل وبستراتيجيات مختلفة للبحث ووجد انه بزيادة m يزداد معدل الاستخلاص الامثل.

https://doi.org/10.30684/etj.30.10.10
2412-0758/University of Technology-Iraq, Baghdad, Iraq

INTRODUCTION

There are numerous methods available for optimization. The choice of any method depends on the suitability to the problem at hand. Conventional optimization techniques broadly consist of calculus based, enumerated and random techniques [1]. These techniques are based on well-established theories and work perfectly well to a case wherever applicable. But there are certain limitations to the above mentioned methods. For example, the steepest descent method starts its search from a single point and finally ends up with an optimal solution, but this method does not ensure that this optimum is the global optimum. Hence there is every possibility of these techniques getting trapped in local optima. Another great drawback of traditional methods is that these require complete information of objective function, its dependence on each variable and the nature of the function. These also make assumptions in realizing the function as a continuous one. All these characteristics of these traditional methods make them inapplicable to many real life problems where there is no sufficient information on mathematical model of the system, parameter dependence and other such information. This calls for non-conventional techniques to address many real life problems. Also, optimization algorithms in engineering design are becoming popular in recent times due to the availability and affordability of high-speed computers.

Genetic algorithm mimics the principle of natural genetics and natural selection to constitute search and optimization. Genetic algorithms are a part of evolutionary computing, which is a rapidly growing area of artificial intelligence. Genetic Algorithm (GA) is a stochastic search technique used in computer science to find approximate solutions for optimization and search problems that is based on natural selection; Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by Darwin's theory of evolutionary biology such as inheritance, mutation, natural selection, and recombination (or crossover). The genetic algorithm repeatedly modifies a population of individual solutions, at each step; the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution [2,3]. The Genetic algorithm can be applied to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear.

Genetic algorithm uses three main types of rules as shown in Figure 1 [4], at each step to create the next generation from the current population:

- ***Selection rules:*** select the individuals, called parents according to their level of fitness in the problem domain that contributes to the population at the next generation.
- ***Crossover rules:*** combine two parents to form children for the next generation.
- ***Mutation rules:*** apply random changes to individual parents to form children.

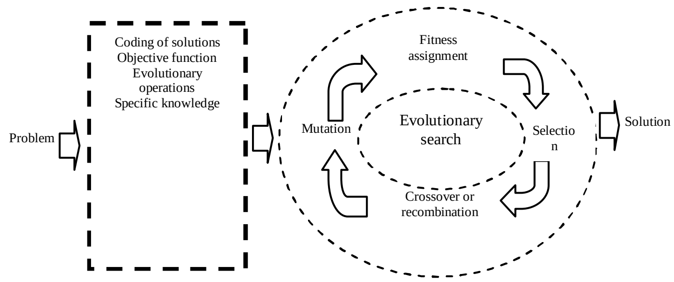

Figure (1) Problem solution using Genetic algorithms

Genetic algorithms are typically implemented as a computer simulation in which a population of abstract representations (called *chromosomes*) of candidate solutions (called *individuals*) to an optimization problem evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but different encodings are also possible. The evolution starts from a population of completely random individuals and happens in generations. In each generation, the fitness of the whole population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), modified (mutated or recombined) to form a new population, which becomes current in the next iteration of the algorithm [3].

GA OPERATIONS

Initialization

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions (*the search space*). Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found.

Selection

Selection is an operation to choose parent solutions. New solution vectors in the next generation are calculated from them. Individual solutions are selected through a *fitness-based* process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Since it is expected that better parents generate better offspring, parent solution vectors which have higher fitness values have a higher probability to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as this process may be very time-consuming.

Most functions (fitness assignments) are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Some of the more common methods include roulette wheel selection (the likelihood of picking an individual is proportional to the individual's score), tournament selection (a number of individuals are picked using roulette wheel selection, then the best of these is (are) chosen for mating), and rank selection (pick the best individual every time). Threshold selection can also be effective.

Crossover

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (or recombination), and mutation.

Crossover is an operation to combine multiple parents and make offspring. The crossover is the most essential operation in GA. There are several ways to combine parent chromosomes. The simplest crossover is called one-point crossover, the parent chromosomes are cut at one point, and the cut parts are exchanged. Crossover that uses two cut points is called two points crossover. Their natural expansion is called multipoint crossover or uniform crossover. Figure 2 shows example of these standard types of crossover [4].

There are several variations of crossover. One unique crossover is called simplex crossover [5]. The simplex crossover uses two better parents and one poor parent and makes one offspring. When both better parents have the same ‘0’ or ‘1’ at the certain bit position, the offspring copies the bit into the same bit position. When better parents have different bit at a certain bit position, then a complement bit of the poor parent is copied to the offspring. This is analogous to learning something from bad behavior.

For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents." New parents are selected for each child, and the process continues until a new population of solutions of appropriate size is generated.

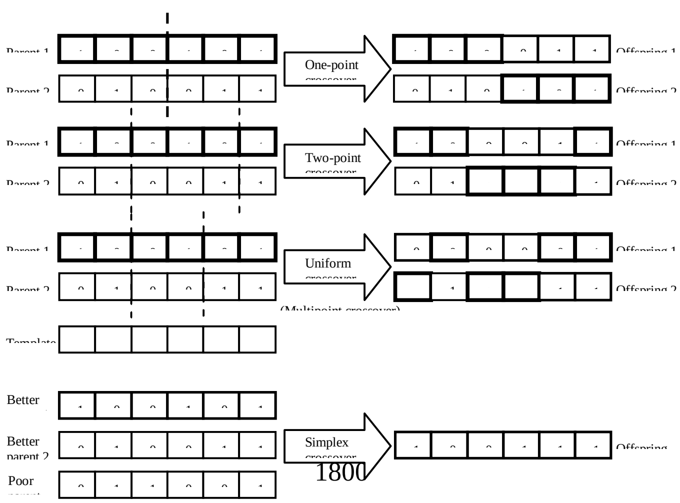

Figure (2) several variations of

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above.

Mutation

When parent chromosomes have similar bit patterns, the distance between the parents and offspring created by crossover is close in a genotype space. This means that the crossover cannot escape from the local minimum if individuals are concentrated near the local minimum. By mutation individuals are randomly altered. These variations (mutation steps) are mostly small. They will be applied to the variables of the individuals with a low probability (mutation probability or mutation rate). Normally, offspring are mutated after being created by recombination (crossover). Genetic algorithm uses an operator called mutation operator, which swaps a bit from either 0 to 1 or vice-versa as shown in the Fig. 3.

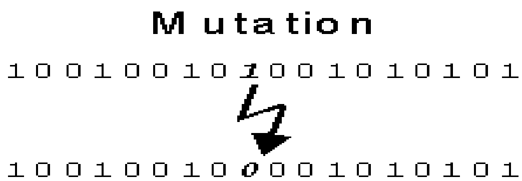

Figure (3) Mutation operator

TERMINATION

This generational process is repeated until a termination condition has been reached. Common terminating conditions are

- A solution is found that satisfies minimum criteria
- Fixed number of generations reached
- Allocated budget (computation time/money) reached
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
- Manual inspection
- Combinations of the above

OPTIMIZATION OF LIQUID-LIQUID EXTRACTION PROCESS

Liquid-liquid extraction is a mass transfer operation whereby a feed solution is contacted with a liquid solvent that is immiscible with one or more, but not all, of the components of the solution. During this contact, the material to be removed from the feed (the solute) is transferred from the feed phase to the solvent phase. The phases are then separated, generating an extract phase (solvent that has “picked up” the solute) and raffinate phase (original feed solution minus the solute). Essential to understanding the performance of an extraction column is the liquid-liquid equilibrium relationship, when the distribution of solute between two phases is linear or nearly so; i.e., $y = mx$ this means that the optimum point can be found mathematically.

Liquid-liquid extraction process is carried out either in a series of well-mixed vessels, such as well-mixed tanks or plate columns, so that the process can be treated as a staged process, or in continuous processes, such as spray columns, packed columns, and rotating disk columns. The concept of a column-type contactor is to allow the phases to flow countercurrently due to the density difference between the liquids. Staged processes are more difficult to optimize if the number of stages is treated as an integer variable. For either cocurrent or countercurrent flow in continuous process equipment, integer variables are avoided and optimization of any reasonable objective function can be carried out by several techniques. Table 1 summarizes a number of examples for staged and continuous models of extraction.

Table (1) Techniques of optimization applied to steady state Liquid extraction

Independent variables	Process model	Objective function	Optimization technique	Ref.
Number of stages Ratio of flow rate of entering solvent to retention rate of liquid or solids	Staged	Value of recovered solute less costs of equipment, raw materials, and solvent	Analytical differentiation and a graphical method	6
Number of stages extraction factor=(m)*(feed rate of Y divided by feed rate X)	Staged	Recovery factor	Analytical differentiation and equating derivatives to zero	7
Number of stages solvent flow rate	Staged	Value of product less cost of feed, solvent and fixed charges	Analytical differentiation and equating derivatives to zero	8
Scale up factor ratio of extract to raffinate flow rate	Staged	Capital costs, value of unextracted solvent,	Analytical differentiation and equating derivatives to zero	9

		operating costs, cost of lost solvent		
Velocity of the two continues phases, stirrer speed	**Continuous**	**Extraction rate**	**Modified gradient projection using the Broyden-Flecher-Shanno formula**	**10**
Extraction factor in one stage of column	**Continuous and staged**	**Overall separation factor**	**Analytical differentiation and equating derivatives to zero**	**11**
Feed rates, reflux rates, purge rates	**Staged**	**Cost of materials**	**Adaptive random search**	**12**

PROBLEM FORMULATION

The problem formulation of any optimization problem can be thought of as a sequence of steps. In the present study, extraction rate is to be optimized and the following steps are followed:

1. Choosing design variables (superficial velocities of Raffinate and extract)
2. Formulating constraints (given process conditions)
3. Formulating objective function (extraction rate)
4. Setting up variable bounds (discrete values for each variable)
5. Choosing an algorithm (genetic algorithm)
6. Obtaining the solution (arriving at maximum extraction rate)

PROCESS MODEL

Under certain conditions, the plug flow model for an extraction process has an analytical solution. Under other conditions, numerical solutions of the equation may be used. As a practical matter, specifying the model so that an analytical solution exists means assuming that the concentrations are expressed on a solute free mole basis, that the equilibrium relation between extract phase concentration, Y and raffinate phase concentration, X is straight line i.e. $Y^{*}=mX+B$, and the operating line is straight, i.e., the phases are insoluble. The model is then given as below: [13]:

$$\frac{dX}{dZ} - N_{OX}(X - Y) = 0 \quad \ldots\ldots\ldots\ldots (1)$$

$$\frac{dY}{dZ} - FN_{OX}(X - Y) = 0 \quad \ldots\ldots\ldots\ldots (2)$$

Where: F is extraction factor ($F=m\upsilon_x/\upsilon_y$)
Figure 4 shows the boundary conditions X_o and Y_1. Given values for *m*, N_{OX}, and the length of the column, a solution for Y_0 in term of υ_x and υ_y can be obtained; X_1 is related to Y_0 and F via a material balance: $X_1=1-(Y_0/F)$.

Hartland and Mecklenburgh [12] list the solutions for the plug flow model for a linear equilibrium relationship, in terms of *F* and N_{OX}:

$$Y_0 = \frac{F[1-\exp\{N_{OX}(1-F)\}]}{1-F\exp[N_{OX}(1-F)]} \quad \text{.........} \ (3)$$

In practice N_{OX} would be calculated from experimental data by least squares or from an explicit relation for the plug flow model [14].

$$N_{OX} = \left(\frac{1-X_1}{X_1+Y_0-1}\right)\ln\left(\frac{X_1}{1-Y_0}\right) \quad \text{.. ...} \ (4)$$

Jackson and Agnew [10] summarized a number of correlations for N_{OX} such as:

$$N_{OX} = 4.81\left(\frac{u_X}{u_Y}\right)^{0.24} \quad \text{................} \ (5)$$

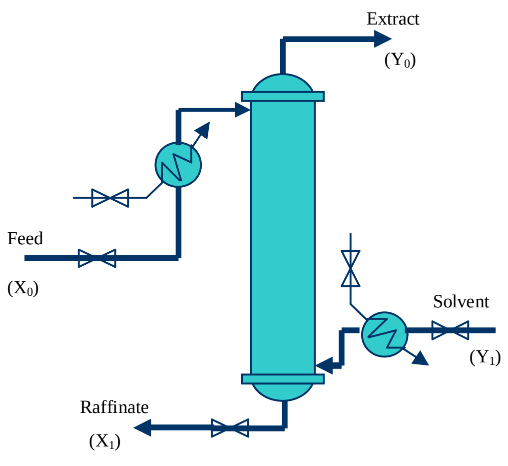

Figure (4) Schematic diagram of extraction column

INEQUALITY CONSTRAINTS

Implicit constraints exit because of the use of dimensionless variables:

$$X_0 \le X \le X_1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

$$Y_1 \le Y \le Y_0 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (7)$$

Constraints on υ_x and υ_y would be upper and lower bounds such as (for m= 1.5): [13]

$$0.05 < u_X < 0.25 \quad \ldots\ldots\ldots\ldots\ldots (8)$$

$$0.05 < u_Y < 0.3 \quad \ldots\ldots\ldots\ldots\ldots (9)$$

And the flooding constraint (for m= 1.5):

$$u_X + u_Y \le 0.2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (10)$$

Objective function

The objective function may be used is the same one proposed by Jackson and Agnew, namely to maximize the total extraction rate in RDC with constant disk rotating speed subject to the inequality and equality constraints:

$$\textbf{Maximize:} \quad f = \upsilon_y Y_0 \quad \ldots\ldots (11)$$

Therefore from equations 3, 5, and 11give:

$$f = u_Y\left[\frac{(mu_X/u_Y)[1-\exp\{4.81\left(\frac{u_X}{u_Y}\right)^{0.24}(1-mu_X/u_Y)\}]}{1-(mu_X/u_Y)\exp[4.81\left(\frac{u_X}{u_Y}\right)^{0.24}(1-mu_X/u_Y)]}\right] \quad \ldots\ldots (12)$$

Different values of distribution coefficients; *m* may be used according to the operating conditions (the values of *m* are taken as 1, 1.5, and 2) using Matlab GA toolbox .

RESULTS AND DISCUSSIONS

The design variables considered in the present case are superficial velocities in raffinate and extract phases, (υ_x, υ_y) respectively. Figures 5 and 6 show the effect of population size on the stopping generation and optimum extraction rate. The optimum extraction rate increase with increasing the population size.

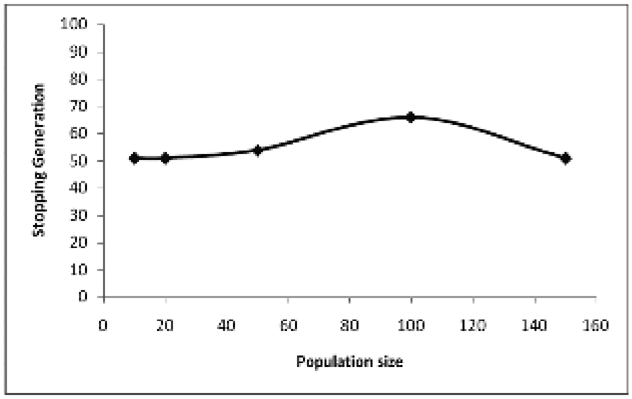

Figure (5) Effect of Population size on the stopping generation (m=1.5, Generation = 100, Crossover rate = 0.8, crossover function = Scattered)

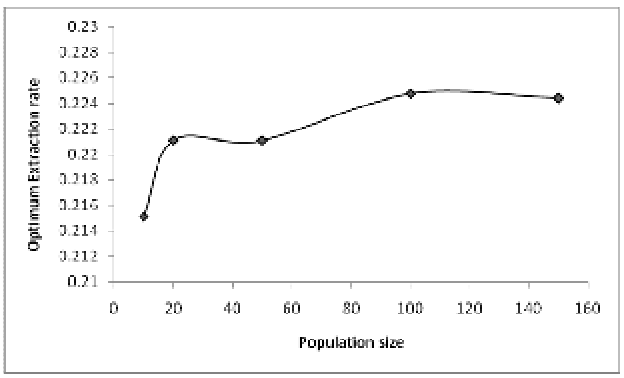

Figure (6) Effect of Population size on the optimum extraction rate (m=1.5, Generation = 100, Crossover rate =0.8, crossover function = Scattered)

Figures (7) and (8) show the effect of Crossover rate on the stopping generation and optimum extraction rate. The optimum extraction rate decrease with increasing the crossover rate.

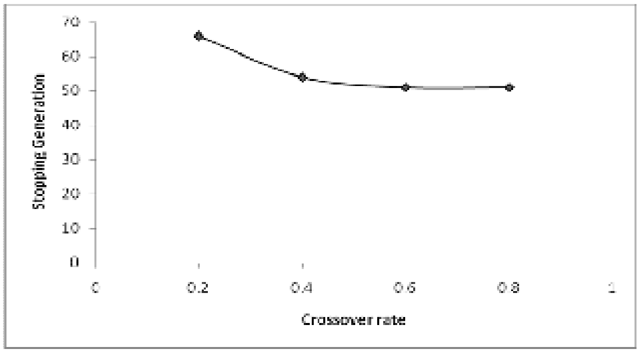

Figure (7) Effect of Crossover rate on the Stopping generation.

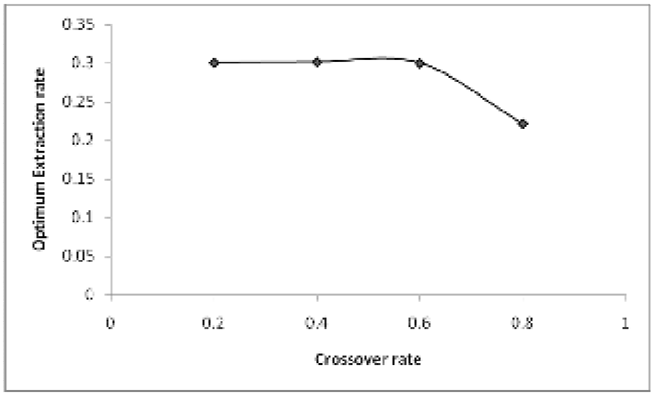

Figure (8) Effect of Crossover rate on the Optimum extraction rate.

Figure (9) show the optimization process for m=2, Population size=20, Generation = 100, Crossover rate =0.8, crossover function = Scattered. The obtained optimum conditions are (0.142, 0.059) and the objective function is 0.2844187. Figure 10 shows the effect of operating distribution ratio on the optimum extraction rate, the optimum extraction rate increase with increase the distribution ratio.

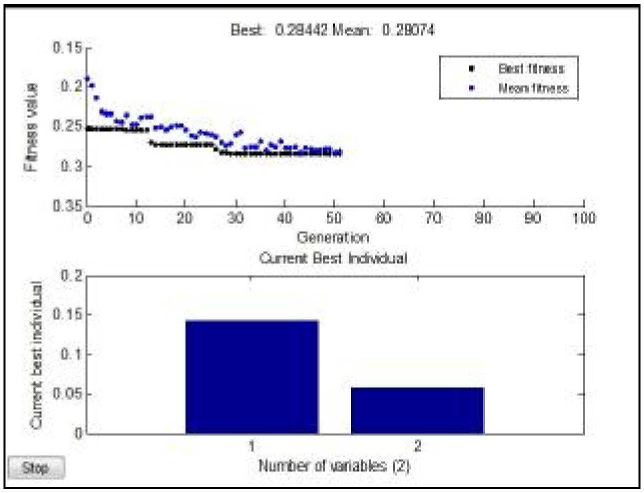

Figure (9) Optimization process

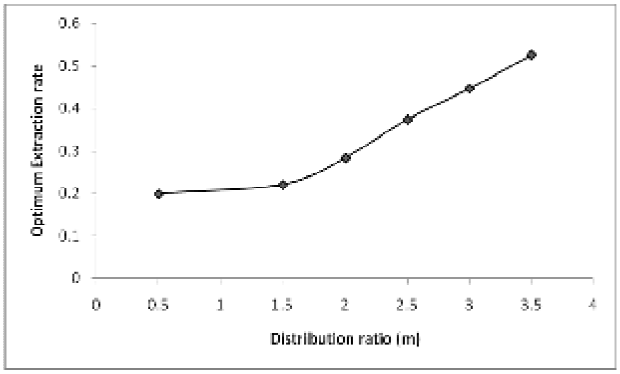

Figure (10) Optimization extraction rate vs. Distribution ratio (m).

CONCLUSIONS

In the present study, the problem of optimization of liquid extraction process has been solved using Genetic Algorithms. It is found that the performance of optimization process is enhanced with increasing the population size, decreasing the crossover rate, and increasing the operating distribution ratio (m). The obtained optimum conditions are (0.142, 0.059) and the objective function is 0.2844187.

NOMENCLATURE

f	Objective function (m/s)
m	Distribution coefficient (-)
F	Extraction factor ($m\upsilon_x/\upsilon_y$) (-)
X	Dimensionless raffinate phase concentration (-)
Y	Dimensionless extract phase concentration (-)
Z	Dimensionless contactor length (-)
N_{OX}	Number of transfer unit based on raffinate phase (-)
υ_x	Superficial velocity in raffinate phase (m/s)
υ_y	Superficial velocity in extract phase (m/s)

REFERENCES

[1]Rao S. S., "Optimization, Theory and applications", Wiley-Eastern, New Delhi (1984).

[2]Goldberg D. E., Genetic Algorithms in search, optimization and machine learning, Addison-Wesley (1989).

[3]Patrick H. Winston, "Artificial Intelligence", third edition, Addison Wesley Publishing Company, New York (1992).

[4]Hartmunt Pohlheim, 2005, "Genetic and Evolutionary Algorithms", at www.geatbx.com, last visit 22/2/2007.

[5]Davis L., "Handbook of Genetic Algorithms", New York, Van Nostand Reinhold (1991).

[6]Chen, N. H., "Optimal theoretical stages in countercurrent leaching" Chem. Eng. (Aug. 24, 1970) p. 71.

[7]Olson, R. S. "Optimum economic extraction", Chem. Eng. (Oct. 6, 1958) p. 143.

[8]Jensen, V. G., "The economic analysis of liquid-liquid extraction process", British Chem. Eng., 6:676 (1961).

[9]Treybal, R. E. "The economic design of mixer- settler extractors" AIChE J., 5:474 (1959).

[10]Jackson, P. J., "A model based scheme for the one-line optimization of a liquid-liquid extraction process" Comput. Chem. Eng., 4:241 (1983).

[11]Hartland S. "The optimum operationof an existing forward and backward extractor" Chem. Eng. Sci., 24: 1075 (1969).

[12]Hartland S. and Mecklenburgh J. C.,"The theory of back mixing", Wiley, New York (1975).

[13]Edgar T. F. and Himmelblau D. M., Optimization of chemical processes, McGraw-Hill, New York (1989).
[14]Treybal, R. E., Liquid extraction, McGraw-Hill, New York (1963).