Asian Journal of Chemistry Vol. 22, No. 3 (2010), 2275-2284

Classical Approaches of Genetic Algorithms and Their Suitability

LORENTZ JÄNTSCHI†‡, SORANA D. BOLBOACA*‡ and RADU E. SESTRAS‡
"Iuliu Hatieganu" University of Medicine and Pharmacy Cluj-Napoca,
13 E. Isac, 400023 Cluj-Napoca, Romania
Fax: (4)(264)593847; Tel: (4)(264)431697; E-mail: sbolboaca@umfcluj.ro

Genetic algorithms derived from observations of nature and simulation of artificial selection of organisms with multiple loci that control a measurable trait. To date, genetic algorithms evolved into complex and strong informatics tools able to deal with hard problems of decision, classification, optimization and simulation. A series of studies reported biotechnology hard problems solved using genetic algorithms. In this context, the aim of the present article is to introduce genetic algorithms and to present their suitability for biotechnology hard problems. Important results are reported in the available literature that deal with the application of genetic algorithms for biotechnology process modelling.

Key Words: Genetic algorithms, Hard problems, Processes modelling, Kinetic models.

INTRODUCTION

Any problem which has its complexity and the complexity of algorithms applied to find the optimum solution differ in terms of time (time complexity ***e.g.***, the number of transitions from start to the end, hopefully with the correct answer) and space (space complexity ***e.g.***, amount of random access memory required to the program for run) from one approach to another. A hard problem is one for which all algorithms that solve it are of high complexity. The problems with exponential complexity are also considering hard because even the best algorithm is used, it will probably be unusable on real-world instances[1]. If a problem is hard, then the search for the optimum solution often goes into out-of-time for real world applications. Besides, a large set of problems encountered in practice do not necessarily call for the optimum. Because most of the hard problems subsist from many years, for some of them one or several heuristics have already been formulated. These are rules of thumb recipes for solving a particular problem, usually based on common sense and avoiding obvious mistakes. Three heuristics applicable to a wide range of hard problems, known as meta-heuristics, were developed. All three are stochastic in nature and two of them are based on natural processes that have been taking place such as tabu search[2], simulated annealing[3] and genetic algorithms[4].

†Technical University of Cluj-Napoca, 103-105 Muncii Bvd, 400641 Cluj-Napoca, Romania.
‡University of Agricultural Sciences and Veterinary Medicine Cluj-Napoca, 3-5 Manastur, 400372 Cluj-Napoca, Romania.

In regards of genetic algorithms (GAs) history, the computer simulations of evolution started with the work of Nils Aall Barricelli[5]. Shortly later, the Australian quantitative geneticist Alex Fraser published a series of papers on simulation of artificial selection of organisms with multiple loci controlling a measurable trait[6]. Fraser's simulations included all of the essential elements of modern genetic algorithms.

Genetic algorithms (GAs) are adaptive heuristic search algorithm developed to mimic some of the processes observed in natural evolution with the idea to use this power of evolution to solve optimization problems. Genetic algorithms are designed to simulate processes in natural systems necessary for evolution, specially those follow the principles first laid down by Charles Darwin of survival of the fittest. Since in nature, competition among individuals for scanty resources results in the fittest individuals dominating over the weaker ones. Genetic algorithms are implemented as computer simulations in which a population of abstract representations (called chromosomes or the genotype of the genome) of a candidate solutions (called individuals, creatures, or phenotypes) subject to an optimization problem which evolves toward better solutions. Genetic algorithms simulates the survival of the fittest among individuals over consecutive generation for solving the problem. Each generation consists of a population of character strings analogous to the DNA chromosomes. Each individual represents a point in a search space and a possible solution. The individuals in the population are then made to go through a process of evolution. Genetic algorithms is based on an analogy with the genetic structure and behaviour of chromosomes within a population of individuals.

The genetic algorithms were applied to the hard problems from many scientific fields since popular in the early 1970s[7]: computer science and engineering[8-12], agriculture[13,14], medicine and chemistry[15-19].

The aim of this study was to illustrate how genetic algorithm can be used to solve hard problems from the biotechnology field. The study includes a classical approach, which is minted to prepare the arena for genetic algorithms. Since is no unique approach to solve a problem, the comparison between different approaches were revealed.

EXPERIMENTAL

Natural as well as controlled processes evolve through a mechanism. If it refers to chemistry, then the mechanism is about explaining the pathway of a reaction, whilst in biology is about explaining how a feature is created. Starting from a biochemistry approach[20], the fast reaction between copper and thiosulfate ions was investigated[21]. The formulation of the kinetic problem is as follows: (i) two reactants (A and B) through forming of an intermediary X lead to the products (P), (ii) the intermediary X concentration can be correlated with an observed absorption of light intensity at a given wavelength (430 nm) using the well known Lambert-Beer relationship[22],(iii) as it is in any general kinetics study, of particular interest is to obtain the reaction rates and partial orders.

Model formulation starts assigning of unknown reaction rates and orders for every elementary reaction of the process and writing of the reaction kinetics. Thus, for a reaction with pre-equilibrium, following chemical reactions and mathematical equations applied,: $A + B \rightarrow X$ ($d[X]/dt = k_0 \cdot [A]^{y0}[B]^{y1}$); $X \rightarrow A + B$ ($d[X]/dt = -k_1 \cdot [X]^{y2}$); $X \rightarrow P$ ($d[X]/dt = -k_2 \cdot [X]^{y3}$), where $[\cdot]$ denote concentration in the aqueous solution. A mass conservation principle is applied if the process evolve in a controlled environment and starts from two initial concentration of reactants A and B, without addition of the reactants during the reaction. The following relationships between two instantaneous concentrations $[\cdot]_1$ and $[\cdot]_2$ are observed: $[A]_1-[A]_2 = [B]_1-[B]_2 = [P]_2 - [P]_1 + [X]_2 - [X]_1$.

The kinetic model (KM) can be state as: (i) chemical reaction: $A+B \underset{k_0}{\overset{k_1}{\rightleftharpoons}} X \xrightarrow{k_2} P$; (ii) a fundamental assumption: A = A(t), B = B(t), X = X(t), P = P(t) and mathematical equations (using A in place of [A] and so on): $dA/dt = -k_0A^{y0}B^{y1} + k_1X^{y2}$, $dB/dt = - k_0A^{y0}B^{y1} + k_1X^{y2}$, $dX/dt = k_0A^{y0}B^{y1} - k_1X^{y2} - k_2X^{y3}$ and $E \sim X \rightleftharpoons \hat{E} = aX+b$ (from Lambert-Beer); (iii) mathematical inequalities: A, B, X, P ≥ 0 (concentrations); $y_0, y_1, y_2, y_3 \geq 0$ (partial orders); $k_0, k_1, k_2 \geq 0$ (reaction rates); (iv) experimental data: E = E(t), experimental extinctions; (v) another fundamental assumption: during the reaction, the values $k_0, k_1, k_2, y_0, y_1, y_2, y_3$ remains constant; (vi) the objective function: $(aX+b-E)^2 = \min$.

The kinetic model problem is a hard optimization problem, having the mathematical equations and inequalities, an objective function to be minimized and nine parameters to be determined. Fig. 1 presents a classical approach of solving algorithm[23].

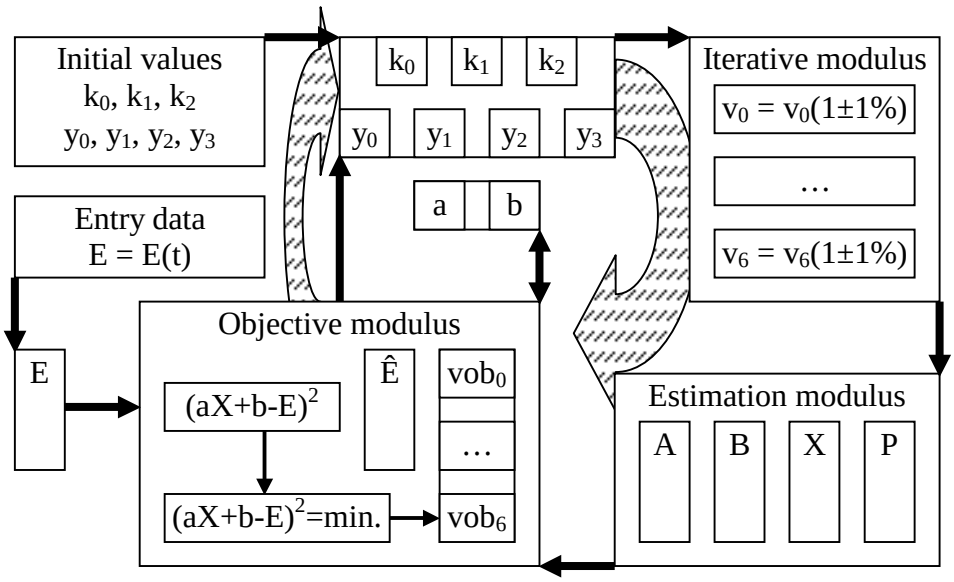

Fig. 1. Classical approach of a solving algorithm [Ref. 23]

The explanation of the classical approach of solving algorithm from Fig. 1 is as follows: (a) The initial values of the constants are stored into an array of data (which is optimized iteratively). (b) The iterative module uses the $k_0...k_2$, $y_0...y_3$ values to generate a table with 3^7 ($3 = |\{-1;0;1\}|$) rows and 7 columns ($7 = |\{k_0, k_1, k_2, y_0, y_1, y_2, y_3\}|$). (c) Every row of the table generated in iterative module is used into the estimative module in order to estimated the compounds' concentration variation during time (reactants A and B, intermediary X and product P). (d) The objective module uses the {A, B, X, P} estimations to obtain the coefficients of regression {a, b}. (e) The values $\{v_0...v_6\}$ which provide the smallest value of the objective function $\Sigma(aX+b-E)^2$ are saved into $\{vob_0...vob_6\}$ array. (f) At the end of a complete iteration of the iterative module (3^7 iterations) the $\{vob_0...vob_6\}$ values replace the old $k_0...k_2$, $y_0...y_3$ values. (g) The cycle ***Iterative module*** $\rightarrow$ ***Estimative module*** $\rightarrow$ ***Objective module*** are repeated until the estimation of the unknown parameters $k_0...k_2$, $y_0...y_3$ produce a stable approximation (after a iteration cycle their values does not vary significant).

Method: The genetic algorithms approach: In terms of a genetic algorithm, the kinetic model problem has: (a) The genetic code of a solution composed from seven genes $|k_0|$ $|k_1|$ $|k_2|$ $|y_0|$ $|y_1|$ $|y_2|$ $|y_3|$ (Fig. 2). (b) An initial population that may represent a chosen number of individuals (let be for example of 100) randomly selected. (c) Selection process may give the chance to survival of the fittest for a given per cent of the individuals (let's say 50 %), allowing them to pass to the next generation based on the goodness of each individual that depends on its fitness, assessed by the objective function, which simultaneously obtain the ‘a’ and ‘b’ values (of remaining two parameters, Fig. 3). (d) Crossover. Individuals (two for example) are chose from the population using the selection operator. A double crossover along the bit strings is chose (usually randomly) then the values of the genes are exchanged up to this point and the two new offspring created from this mating are put into the next generation of the population. If there are recombine portions of good individuals, this process is likely to create even better individuals (Fig. 4). (e) Mutation. Introduces random modifications; with some low probability, a portion of the genes of the new individuals will have their values flipped, with the purpose of maintaining diversity within the population and to inhibit premature convergence (Fig. 5).

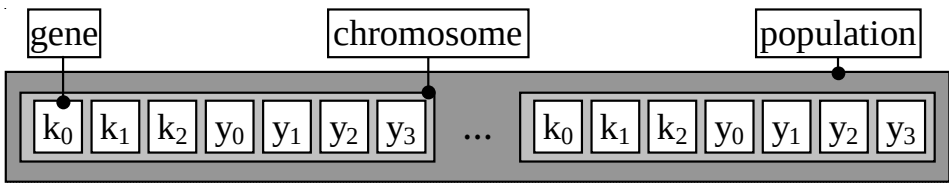

Fig. 2. Genetic code for a solution of kinetic model problem

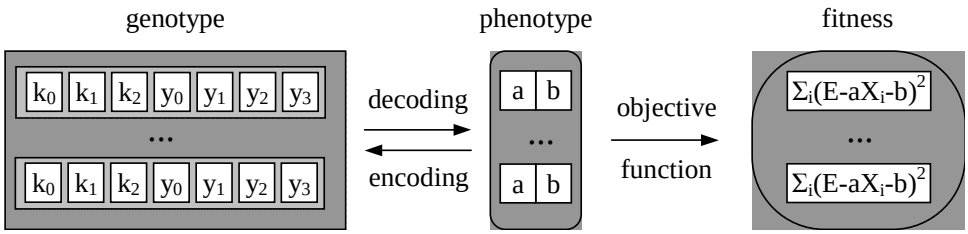

Fig. 3. Natural selection for kinetic model problem: genotype, phenotype and fitness

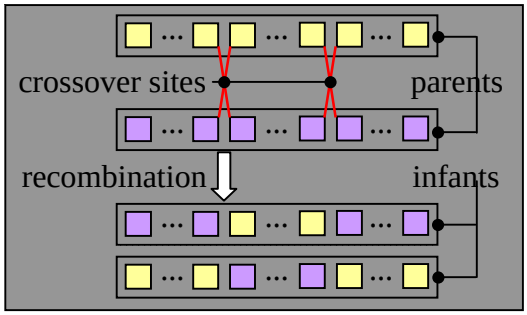

Fig. 4. A double crossing over (involves the breakage and rejoining of parental chromosomes)

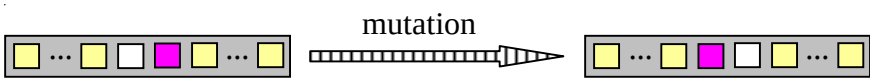

Fig. 5. Mutation of a genotype in kinetic model problem

RESULTS AND DISCUSSION

Biotechnology processes modeled using genetic algorithmss: As can be seen from the previous section, the genetic algorithm is easier to digest and implement and does not implies as many as classical algorithm do computations for iteration. A series of hard problems in biotechnology were solved using genetic algorithms, process kinetic modelling being just one of the genetic algorithm applications.

Lee *et al.*[24] reported parameter estimation using a hybrid of simplex and genetic algorithm by introducing the simplex method as an additional operator in the genetic algorithm. During the reproduction of each iteration step, the hybrid approach applies the simplex method to a top percentage of the population to produce new candidate solutions in the next generation. The remaining of the new population is generated using the genetic algorithm reproduction scheme (*i.e.*, selection, crossover and mutation). The genetic algorithm was applied for optimization of three kinetics reactions, when significant improvements of algebraic methods were obtained: (a) Carboxylation of phosphoenolpyruvate (PEP) to oxaloacetate (OAA) catalyzed by

PPC (P-enolpyruvate), when CO_2 is transformed to phosphate (Pi): CO_2 + PEP → OAA + Pi; (b) Transformation of adenosine tri-phosphate (ATP) to adenosine di-phosphate (ADP) in the presence of OAA transformed to PEP catalyzed by PCK (phosphoenolpyruvate carboxy-Kinase): OAA + ATP → PEP + ADP + CO_2; (c) Transformation of phosphoenolpyruvate (PEP) to pyruvate (Pyr) in presence of ADP (transformed to ATP) catalyzed by pyruvate kinase (PyKi): PEP + ADP → Pyr + ATP.

Pizarro *et al.*[25] reported an evolution of the basic genetic algorithm adapted to the features of the model to explain the industrial fermentation growth rate of acetic fermentation. Each chromosome represents in their approach a possible combination of values of the five parameters to optimize, in binary code. There was imposed an allowed range of values for each parameter to adopt (as many values as allowed by the binary codification and the number of significant figures).The initial population was composed of randomly selected values for the parameters within the allowed ranges and codified into binary code. The evaluation program decodes the values of the parameters for each chromosome and then uses them to simulate a batch process with each sequence of parameters. The simulation algorithm solves a system of differential equations giving overall rates, viable biomass concentration and relationships between the product formation, substrate consumption and the cell growth using the Runge-Kutta algorithm. The initial concentrations were those of the representative sequence of the process and the initial viable biomass/total biomass ratio represents the parameters of the chromosome. There was an oxygen control in the simulation, because in the real process the oxygenation conditions are enough to satisfy the demanded amount of oxygen. The algorithm had two important stop conditions: when no real positive values for one concentration are obtained and when the process time in the simulation has reached the final process time of the representative sequence. A new generation with the same number of chromosomes is formed by applying reproduction, crossover and mutation operators. The chromosomes with the best fitting ability obtained the best value in the desirability function (*i.e.* closer to 1) and have more chances of being selected and copied into the next generation. Uniform crossover is used and the five best chromosomes of each generation pass unchanged to the next generation. These chromosomes are called elitist chromosomes. Twins and out-of-range chromosomes are disallowed by using a 'while' loop with filters. When some of these chromosomes are discovered after crossover, chromosomes also obtained by crossover substitute them and if they are discovered after mutation, they are replaced by the original chromosomes in the same positions but mutated again with the same chances of mutation. With this process, the mutability is not increased and the number of chromosomes remains constant. The process stops after five generations without changes higher than a fixed percentage of the mean response of the elitist chromosomes. The algorithm is completed five times each time when the program is run. A final run when the initial population is composed of the best chromosomes found in each of the previous

runs is performed. The acetic concentrations in the fermentators of the industrial plant of Vinagrerias Riojanas SA (Logrono, Spain), obtained by NIR, were studied. The data were obtained for a period of 4 months without changes in the industrial parameters of the process, *i.e.* oxygenation conditions and temperature. The average temperature was 29.5 °C and the oxygenation conditions were enough to satisfy the oxygen demand and thus the oxygen became a non-limiting substrate. Nowadays, the fermentators of the industrial plant work discontinuously with charges. The batch bioreactors studied were always fed with white wine of the same origin. The process time was about 30-31 h and 218 complete sequences were obtained. An average concentration sequence was calculated by analyzing the data. This sequence is representative of the process to be modelled. The variability in the concentrations among the sequences is due to analytical errors and to factors that cannot be controlled in an industrial process (*i.e.* differences in the ethanol concentration of the wine among batch processes). Therefore, the model obtained with this sequence does not model this variance.

Guangzhu *et al.*[26] improved a simple genetic algorithm developing a hybrid genetic algorithm, which was used to estimate the kinetic parameters of polyesterification between dimer fatty acid and ethylene glycol. The work proved that the model developed by authors is useful for the polyesterification of dimer acid and ethylene glycol catalyzed by *p*-toluene sulfonic acid. The authors used 28.1 g (0.05 mol) of dimer fatty acid, 3.11 g (0.05 mol) of ethylene glycol and 0.5 % of *p*-toluene sulfonic acid as the catalyst. All were placed into a round bottom flask (three necks), which was equipped with a dephlegmator and a pipe for the nitrogen. Nitrogen was introduced into the flask to remove the oxygen and to prevent the oxidation of the materials. The flask was placed into an oil bath at 170 °C. After 0.5 h reaction, the nitrogen was stopped and vacuum pumping was used to remove the water from reactant. The reaction continued 8-10 h in vacuum. The acid value of the reactant was measured at certain reaction times during the progress of the reaction. The estimation of the parameters was carried out in three steps. First, the order of reaction was confirmed using the assumption of equal activity. Second, experiments were designed to estimate the parameters of rate constant of the reaction between carboxylic group on the monomer and hydroxyl group of the polymers. Excess monomer was supplemented into the reactant after it had reacted for several hours with the materials proportion of 1:1 and the reactions could be ignored except for the added monomer and the polymers. Finally, obtained values were introduced into the rate equations to obtain the values of reaction rates between carboxylic end of the monomer and hydroxyl group of the monomer and carboxyl on the polymers and hydroxyl on the polymers.

Moscovitch *et al.*[27] used the genetic algorithm in kinetic analysis of multiple proton transfer reactions. They demonstrated that the search of the rate constants can be fully automated using the genetic algorithm approach leading to a detailed kinetic analysis due to ability of identification of multiequilibria systems. A system

with inherent complexity made of seven independent variables consisting of a proton emitter (a pyranine molecule that eject a proton when excited by a photon), indicator (fluorescein, a pH indicator with two proton-binding sites represented by the oxyanion of the xanthene ring and the carboxylate of the benzene) and bicarbonate anion (HCO_3^-, a buffer molecule with unknown concentration that reacts with the proton but not generate a measurable signal) was investigated. The complexity of the studied system is comparable to the studying the protonation dynamics of an indicator attached to a protein[28-31]. The authors used a 100 mM NaCl aqueous solution supplemented with pyranine (20 mM) and fluorescein (10 mM), equilibrated with the air at two pH values (6.8 and 7.3) and subjected to a train of laser pulses (1-1.5 mJ/pulse; 10 Hz, 355 nm, 3 ns full-width half maximum). The absorption transients were recorded at 458 and 496 nm, where pyranine and fluorescein are, respectively, absorbing (the time resolutions were either 30 ns or 300 ns per data point and were converted into concentration units using the extinction coefficients 24000 M^{-1} cm^{-2} for pyranine and 50000 M^{-1} cm^{-2} for fluorescein). Two reactants of the system are directly observable. The rate constants of protonation of these reactants were measured (manual analysis) and the complexity of the system was increased by taking into consideration the concentration of the bicarbonate. The parameter's space was used in order to search for the uniqueness. The values of adjustable parameters were used to reconstruct the signal and to calculate the fittest. The first generation consisted of 100 phenotypes (randomly selected values for the adjustable parameters). The best-fit phenotype was cloned and replaces the worst-fitting one. The genetic algorithm was searching for the minimum of the fitness function in a seven-dimensional space. Each runs last 3000 generations from 2 to 6 h depending on the processor of the computer used. A stable solution in terms of no tendency to drift into a new set of adjustable parameters was search. The genetic algorithm was considered unique when the target signals were noiseless (the values derived by the program were identical to those used to create the signal). The results revealed that genetic algorithm is a reliable method for searching a solution of kinetics equations when the rate constants must be determines proving that the system work even if the concentration of one reactant is unknown. The utility of the genetic algorithm in solving chemical kinetic problems have also been proved by several researchers[32-35].

Popelier *et al.*[36] modelled the mutagenic activity of 23 triazines and 24 halogenated hydroxyfuranones in order to generate significantly statistic valid quantitative models and to identify the active centre of the investigated compounds. A genetic algorithm was used to optimize the number of descriptors of the best model expressed as best possible coefficient of determination and leave-one-out cross validation coefficient. The genetic algorithm introduced a population of 256 randomly selected models and the cross-validation error was the fitness function at a mutation rate of 0.003 and for a maximum number of generations equal to 200. The analysis suggested a preferred mechanistic pathway for the initial hydroxylation of the triazines and elucidates the mechanistic ambivalence of hydroxyfuranones. A similar approach

was applied in investigation of steroid binding affinity and antibacterial activity of nitrofuran derivatives[37]. The genetic algorithm was used in order to select variable in the best performing model. Similar approach was used by Matsuda *et al.*[38] when the authors were able through their design system cyclopaedically to generated ion liquids structures corresponding to particular physical properties.

Conclusion

Having a population of abstract representations (the genotype of the genome) of candidate solutions (phenotypes) genetic algorithm optimization problem evolves toward better solutions simulating the survival of the fittest among individuals over consecutive generation for solving the problem as living organisms evolve in nature.

The usage of genetic algorithm may reduce the algorithm complexity as was shown in kinetic model hard optimization problem.

Key hard problems solved using genetic genetic algorithms include modelling industrial fermentation growth rate of acetic fermentation, heavy oil thermal cracking 3-lumping, fluid catalytic cracking unit main fractionator, reverse engineering of molecular mechanical machines, chemical kinetic problems and selection of descriptors used in quantitative structure-property/activity relationship models.

ACKNOWLEDGEMENT

This work was in part supported by grants (ID-206/2007, ID-202/1.10.2007) of the National University Research Council, Romania.

REFERENCES

1. E. Falkenauer, Genetic Algorithms and Grouping Problems, Wiley, New York (1998).
2. F. Glover, *Decision Sci.*, **8**, 156 (1977).
3. L. Davis, Genetic Algorithms and Simulated Annealing, San Francisco: M. Kaufmann (1987).
4. J. Bosworth, F. Norman, B.P. Zeigler, Comparison of Genetic Algorithms with Conjugate Gradient Methods, NASA Contractor Reports, CR-2093 (1972).
5. N.A. Barricelli, *Methodos*, 45 (1954).
6. A. Fraser, *Aust. J. Biol. Sci.*, **10**, 484 (1957).
7. J.H. Holland, In Proceedings of the 1970 IEEE Symposium on Adaptive Processes Decision and Control: XVII, ACM PRESS, New York (1970).
8. A.M. Barros, H.S. Lopes and A.L. Stelle, In Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Image and Signal Processing, CIISP 2007, pp. 151-156 (2007).
9. J. Northern III and M. Ribeiro, In Proceeding of the 2007 IEEE Region 5 Technical Conference, TPS, Art. no. 4380385, pp. 223-227 (2007).
10. M.E. Aydemir, T. Gunel, S. Kargin, I. Erer and S. Kurnaz, In Proceedings of 2nd International Conference on Recent Advances in Space Technologies, Art. no. 1512654, pp. 684-688 (2005).
11. M. Sato, S. Matsumoto, Y. Teramoto and N. Adachi, *Trans. Japan. Soc. Artif. Intel.*, **16**, 324 (2001).
12. R. Neruda, *Neural Network World*, **11**, 267 (2001).
13. H. Muhlenbein and D. Schlierkamp-Voosen, *Evol. Comput.*, **1**, 335 (1993).
14. R. Salomon, *BioSystems*, **39**, 263 (1996).
15. P. Willett, *Trends Biotechnol.*, **13**, 516 (1995).
16. G. Jones, P. Willett, R.C. Glen, A.R. Leach and R. Taylor, *J. Mol. Biol.*, **267**, 727 (1997).

17. S. Kikuchi, D. Tominaga, M. Arita, K. Takahashi and M. Tomita, *Bioinformatics*, **19**, 643 (2003).
18. R. Leardi, *J. Chemometr.*, **15**, 559 (2001).
19. O. Roeva, T. Pencheva, S. Tzonkov, M. Arndt, B. Hitzmann, S. Kleist, G. Miksch, K. Friehs and E. Flaschel, *E. J. Biotechnol.*, **10**, 593 (2007).
20. L. Jäntschi, *Leonardo J. Sci.*, **2**, 1 (2003).
21. C.E. Stoenoiu, S.D. Bolboaca and L. Jäntschi, In Proceeding of the 4th ICAMC, Plovdiv, Bulgaria, p. 511 (2007).
22. P.W. Atkins, Physical Chemistry, Oxford: Oxford University Press, edn. 5 (1994).
23. L. Jäntschi and S.D. Bolboaca, Processes at Phases Interface: Mathematical Modelling, Numerical Optimization, Web Implementation, with Applications for Separation and Characterization of Biological Active Compounds Series (MEC/5892/18.09.2006)-Research Synthesis for 2008 year (in Romanian), UEFISCSU-ET108 (2008).
24. B. Lee, J. Yen, L. Yang and J.C. Liao, *Biotechnol. Bioeng.*, **62**, 722 (1999).
25. C. Pizarro, J.M. Gonzalez-Saiz and D. Garrido-Vidal, *J. Chemometr.*, **17**, 453 (2003).
26. F. Guangzhu, L. Fasong, L. Heping, Q. Hai and C. Yingde, *Chem. Eng. Technol.*, **29**, 740 (2006).
27. D. Moscovitch, O. Noivirt, A. Mezer, E. Nachliel, T. Mark, M. Gutman and G. Fibich, *Biophys. J.*, **87**, 47 (2004).
28. E. Nachliel, M. Gutman, S. Kiryati and N.A. Dencher, *Proc. Natl. Acad. Sci. (USA)*, **93**, 10747 (1996).
29. E. Nachliel and M. Gutman, *Biochim. Biophys. Acta*, **1514**, 33 (2001).
30. E. Shimoni, E. Nachliel and M. Gutman, *Biophys. J.*, **64**, 480 (1993).
31. R. Yam, E. Nachliel and M. Gutman, *J. Am. Chem. Soc.*, **110**, 2636 (1988).
32. B. Filipic, I. Zun and M. Perpar, *Int. J. Hum. Comput. Stud.*, **53**, 517 (2000).
33. C. Hongqing, Y. Jingxian, K. Lishan, C. Yuping and C. Yongyan, *Comput. Chem.*, **23**, 143 (1999).
34. D.B. Terry and M. Messina, *J. Chem. Inf. Comput. Sci.*, **38**, 1232 (1998).
35. C. Viappiani, G. Bonetti, M. Carelli, F. Ferrati and A. Sternieni, *Rev. Sci. Instrum.*, **69**, 270 (1998).
36. P.L.A. Popelier, P.J. Smith and U.A. Chaudry, *J. Comput.-Aided. Mol. Des.*, **18**, 709 (2004).
37. P.J. Smith and P.L.A. Popelier, *J. Comput.-Aided. Mol. Des.*, **18**, 135 (2004).
38. H. Matsuda, H. Yamamoto, K. Kurihara and K. Tochigi, *Fluid Phase Equilibr.*, **261**, 434 (2007).

(*Received*: 30 June 2009; *Accepted*: 1 December 2009) AJC-8119