International Conference on Control, Engineering & Information Technology (CEIT'13)
Proceedings Engineering & Technology - Vol.2, pp. 206-210, 2013

A PSO Approach for Optimum Design of Multivariable PID Controller for nonlinear systems

Taeib Adel
Research Unit on Control,
Monitoring and Safety of Systems (C3S)
High School ESSTT
Email: taeibadel@live.fr

Ltaeif Ali
Research Unit on Control,
Monitoring and Safety of Systems (C3S)
High School ESSTT
Email: ltaief24@yahoo.fr

Chaari Abdelkader
Research Unit on Control,
Monitoring and Safety of Systems (C3S)
High School ESSTT
Email: nabile.chaari@yahoo.fr

***Abstract*—The aim of this research is to design a PID Controller using particle swarm optimization (PSO) algorithm for multiple-input multiple output (MIMO) Takagi-Sugeno fuzzy model. The conventional gain tuning of PID controller (such as Ziegler-Nichols (ZN) method) usually produces a big overshoot, and therefore modern heuristics approach such as PSO are employed to enhance the capability of traditional techniques. However, due to the computational efficiency, only PSO will be used in this paper. The results show the advantage of the PID tuning using PSO-based optimization approach.**

I. Introduction

PID control, which is usually known as a classical output feedback control for SISO systems, has been widely used in the industrial world [1] and [2]. The tuning methods of PID control are adjusting the proportional, the integral and the derivative gains to make an output of a controlled system track a target value properly. several researchers focus on multiple-inputmultiple output MIMO control systems. Because many industrial processes are MIMO systems which need MIMO control techniques to improve performance, though they are naturally more difficult than SISO systems. As we know, MIMO PID controller design has developed over a number of years. Luyben (1986) proposed a simple tuning method for decentralized PID controllers in MIMO system from single-loop relay tests [3]. Yusof and Omatu (1993) presented a multivariable self-tuning PID controller based on estimation strategies [4]. Wang et al. (1997) proposed a tuning method for fully cross-coupled multivariable PID controller from decentralized relay feedback test to find the critical oscillation frequency of the system by first designing the diagonal elements of multivariable PID controller independent of off-diagonal ones [5]. Recently, the computational intelligence has proposed particle swarm optimization (PSO) [6,7] as opened paths to a new generation of advanced process control. The PSO algorithm, proposed by Kennedy and Eberhart [6] in 1995, was an evolution computation technology based on population intelligent methods. In comparison with genetic algorithm, PSO is simple,easy to realize and has very deep intelligent background. It is not only suitable for scientific research, but also suitable for engineering applications in particular. Thus, PSO received widely attentions from evolution computation field and other fields. Now the PSO has become a hotspot of research. Various objective functions based on error performance criterion are used to evaluate the performance of PSO algorithms.

In this paper, a scheduling PID tuning parameters using particle swarm optimization strategy for MIMO nonlinear systems . This paper has been organized as follows: In section 2, a brief review of the TS fuzzy model formulation is given. Estimation method of recursive weighted least-squares (RWLS)in section 3. In section 4, PID control systems of multivariable processes. Finally, some conclusions are made in section 5.

II. Takagi-Sugeno fuzzy model of a MIMO process

Generally, modeling process consists to obtain a parametric model with the same dynamic behavior of the real process. In this section, we are interested to the problem of the MIMO process identification [8]. We consider a MIMO system with$n_i$nputs and n_0 outputs. This system can be approximated by a set of discrete time fuzzy MISO models. We consider also:

- two polynomials A and B defined by:

$$\begin{array}{c} A = a_0 + a_1 q + a_2 q^2 + ... + a_{n_A} q_{n_A}. \\ B = b_0 + b_1 q + b_2 q^2 + ... + b_{n_B} q_{n_B}. \end{array} \quad (1)$$

  q is a backward shift operator $(q^n y(k) = y(k-n))$.
- two integers m and n, m n which define a delayed sample of a discrete time signal as:

$$y(k)_m^n = [y(k-m), y(k-m-1), ..., y(k-n)] \quad (2)$$

The MISO models are a input-output NARX (Non linear Auto Regressive with eXogenous input) defined by:

$$y_l(k+1) = f_l(x_l(k)) \qquad l = 1, 2, ..., n_0 \quad (3)$$

With the regression vector represented by:

$$\begin{array}{c} x_l(k) = [\{y_1(k)\}_0^{n_{yl1}}, \{y_2(k)\}_0^{n_{yl2}}, ..., \{y_{n0}(k)\}_0^{ln_{yln_0}}, \\ \{u_1(k)\}_{n_{dl1}}^{n_{ul1}}, \{u_2(k)\}_{n_{dl2}}^{n_{ul2}}, ..., \{u_{n_i}(k)\}_{n_{dln_i}}^{n_{udln_i}}] \end{array} \quad (4)$$

n_y and n_u define the number of delayed outputs and inputs respectively. n_d is the number of pure delays. n_y is a $n_0 * n_0$ matrix and n_u,nd are no ni matrices. f_l are unknown non linear functions. MISO models are estimated independly [9], so, to simplify the notation, the output index l is omitted and we will be interested only in the multi-input, mono-output case. The Takagi-Sugeno MISO rules are estimated from the system input- output data (Babuska, 1998). The base rule contains r rules of the following form:

$$\mathbf{R}_i : If y_1(k) \ is \ \mathbf{A}_{i1} \ and \ if \ u_{nu}(k-n+1) \ is \ \mathbf{A}_{ino}$$
$$\textbf{then} \ y_i(k+1) = \sum_{j=1}^{n} a_{ir} y_i(k-j+1) + \sum_{j=1}^{n} b_{ij} u_i(k-j+1)$$
$$+ \sum_{l=1}^{nu} \sum_{j=1}^{n} b_{ilj} u_l(k+j-1) + c_i \quad \text{i=1,2..,r} \tag{5}$$

III. Estimation method of recursive weighted least-squares (RWLS)

For nonlinear systems the online adaptation is necessary to obtain a model able to continue the system in its evolution. The system described by relation (4) can also be rewritten as:

$$y_i(k) = \theta_i^t \varphi_i(k-1) \tag{6}$$

with θ being a system parameter vector and φ a regression vector. It should be noted that the system (5) is in general nonlinear but it is linear with respect to its unknown parameter vectors. Based on parameterizations (6), the identification algorithm giving estimates $\widehat{\theta}(k)$ of $\theta(k)$ can be obtained using the RWLS.
We define:

$$\begin{aligned}\varphi_i(k-1) = [\mu_i \, y_i(k-1) ... \, \mu_{ij} y_i(k-n) \\ \mu_i u_i(k-1) ... \, \mu_i u_i(k-n) \ \ \mu_i]\end{aligned} \tag{7}$$

$$\theta_i = [a_{i1} ... a_{in} \, b_{i1} ... b_{in} c_i] \tag{8}$$

$$\varphi_i \left(k-1\right) = \left[\varphi_{i1}^t \left(k-1\right) \, \varphi_{i2}^t \left(k-1\right) ... \varphi_{ir}^t \left(k-1\right)\right]^t \tag{9}$$

$$\theta_i(k) = \theta_i(k-1) + L_i(k)[y_i(k) - \varphi^t(k)\theta_i^t(k-1)] \tag{10}$$

$$L_i(k) = \frac{P(k-1) * \varphi^t(k)}{1/\mu_{ik} + \varphi(k) P(k-1) \varphi^t} \tag{11}$$

$$P(k) = P_i(k-1 - L_i(k)\varphi(k)P_i(k-1) \tag{12}$$

for $k = 1, ..., N, P(k-1)$ is a covariance matrix and L(k) referred to the estimator gain vector. A common choice of initial value is to take $\theta_i(0) = 0$ and $P_i(0) = \alpha I$ where α is a large number.

IV. PID control systems of multivariable processes

Consider a multivariable PID control structure as shown in Fig. 1, where:
Desired output vector : $\mathbf{Y}_d = [y_{d1}, y_{d2}, ..., y_{dn}]^T$.
Actual output vector: $\mathbf{Y} = [y_1, y_2, ..., y_n]^T$.
Error vector : $\mathbf{E} = \mathbf{Y}_d - \mathbf{Y} = [y_{d1} - y_1, y_{dn} - y_2, ..., y_{dn} - y_n] = [e_1, e_2, ..., e_n]^T$.
Control input vector : $\mathbf{U} = [u_1, u_2, ..., u_n]^T$.
$n * n$ multivariable processes:

$$\mathbf{H}(z) = \begin{vmatrix} h_{11}(z) & ... & h_{1n}(z) \\ ... & ... & ... \\ h_{n1}(z) & ... & h_{nn}(z) \end{vmatrix} \tag{13}$$

$n * n$multivariable PID controller:

$$\mathbf{C}(z) = \begin{vmatrix} c_{11}(z) & ... & c_{1n}(z) \\ ... & ... & ... \\ c_{n1}(z) & ... & c_{nn}(z) \end{vmatrix} \tag{14}$$

The form of $kij(z)$, for $i, j \in \underline{n}$ and $\underline{n} = \{1, 2, ..., n\}$, is given by:

$$\mathbf{C}(z) = kp_{ij}(1 + \frac{z}{Ti_{ij}(z-1)} + \frac{(z-1)}{Td_{ij}z}) \tag{15}$$

where kp is the proportional gain, Ti is the integral time constant, and Td is the derivative time constant. It can be also rewritten (9) as:

$$\mathbf{C}(z) = kp_{ij} + \frac{ki_{ij}z}{(z-1)} + \frac{kd_{ij}(z-1)}{z}) \tag{16}$$

where $ki = kp/Ti$ is the integral gain and $kd = kp * Td$ is the derivative gain. For convenience, let $K = [kp_{ij}; ki_{ij} \ ; kd_{ij}]^T$ represent the gains vector of i^{th}-row and j^{th}-column sub- PID controller in $\mathbf{C}(z)$ [12]. In the design of a PID controller, the performance criterion or objective function is first defined based on our desired specifications and constraints under input testing signal. Typical output specifications in the time domain are peak overshooting, rise time, settling time, and steady-state error, to name a few. Three kinds of performance criteria

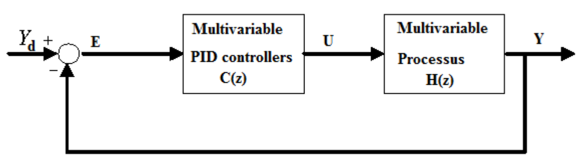

Fig. 1. A multivariable PID control system.

usually considered in the control design are integral of the Absolute Error (**I**AE), integral of Square Error (**I**SE) and integral of Time weighted Square Error (**I**TSE) which are given as:

$$\textbf{IAE} = |e_1(k) + e_2(k) + ... + e_n(k)| \tag{17}$$

$$\textbf{ISE} = \sum e_1(k)^2 + \sum e_2(k)^2 + ... + \sum e_n(k)^2 \tag{18}$$

$$\textbf{ITSE} = \sum k * e_1(k)^2 + \sum k * e_2(k)^2 + ... + \sum k * e_n(k)^2 \tag{19}$$

Therefore, for the PSO-based PID tuning, these performance indexes (Eqs. (17)-(19)) will be used as the objective function. In other word, the objective in the PSO-based optimization is to seek a set of PID parameters such that the feedback control system has minimum performance index.

A. Tuning of PID uzing Z-N method

The first method of Z-N tuning is based on the open-loop step response of the system. The open-loop systems Shaped response is characterized by the parameters, namely the process time constant T and L. These parameters are used to determine the controllers tuning parameters (see TABLE.1). The second method of Z-N tuning is closed-loop tuning

TABLE I
ZIEGLER-NICHOLS OPEN-LOOP TUNING PARAMETER

Controller	kp	Ti=kp/ki	Td=kd/kp
P	T/L	-	0
PI	0.9(T/L)	L/0.3	0
PID	1.2(T/L)	2L	0.5L

method that requires the determination of the ultimate gain and ultimate period. The method can be interpreted as a technique of positioning one point on the Nyquist curve [13]. This can be achieved by adjusting the controller gain (Ku) till the system undergoes sustained oscillations (at the ultimate gain or critical gain), whilst maintaining the integral time constant (Ti) at infinity and the derivative time constant (Td) at zero (see TABLE.2).

TABLE II
ZIEGLER-NICHOLS CLOSED-LOOP TUNING PARAMETER

Controller	kp	Ti=kp/ki	Td=kd/kp
P	0.5ku	-	0
PI	0.45ki	1.5kp/Pu	0
PID	0.6ku	2kp/Pu	kpPu/8

B. Implementation of PSO-Based PID Tuning

1) Particle swarm optimization (PSO): Particle swarm optimization was introduced by Kennedy and Eberhart by simulating social behavior of birds flocks in (199[10]. The PSO algorithm has been successfully applied to solve various optimization problems [14]. The PSO works by having a group of m particles. Each particle can be considered as a candidate solution to an optimization problem and it can be represented by a point or a position vector $\mathbf{X}_{ij} = [X_{i1}, ..., X_{id}]$ in a d dimensional search space which keeps on moving toward new points in the search space with the addition of a velocity vector $\mathbf{V}_{ij} = [V_{i1}, ..., V_{id}]$ to further facilitate the search procedure. The initial positions and velocities of particles are random from a normal population in the interval [0, 1]. All particles move in the search space to optimize an objective function **f**. Each member of the group gets a score after evaluation on objective function **f**. The score is regarded as a fitness value. The member with the highest score is called global best. Each particle memorizes its previous best positions. During the search process all particles move toward the areas of potential solutions by utilizing the cognitive and social learning components. The process is repeated until any prescribed stopping criterion is reached. After any iteration, all particles update their positions and velocities to achieve better fitness values according to the following:

$$\begin{aligned} \mathbf{V}_{pd}^{t+1} &= \omega \mathbf{V}_{pd}^{t} + c_1 r_1 (\mathbf{pbest}^t - \mathbf{X}_{pd}^t) \quad (20) \\ &+ c_2 r_2 (\mathbf{gbest}^t - \mathbf{X}_{id}^k), \\ \mathbf{X}_{pd}^{t+1} &= \mathbf{X}_{pd}^{t} + \mathbf{V}_{pd}^{t+1}, \quad (21) \end{aligned}$$

where:
t is the current iteration number, $\mathbf{pbest}_i$ is $pbest$ of particle i, $\mathbf{gbest}_g$ is $gbest$ of the group, r_1, r_2 are two random numbers in the interval [0, 1], c_1, c_2 are positive constants and w is the inertia weight,is a parameter used to control the impact of the previous velocities on the current velocity. It influences the tradeoff between the global and local exploitation abilities of the particles. Weight is updated as:

$$\omega = \omega_{\max} - \left(\frac{\omega \max - \omega_{\min}}{iter_{\max}} \right) iter \quad (22)$$

where $\omega_{\min}$, $\omega_{\max}$, $iter$, and $iter_{\max}$ are minimum, maximum values of ω , the current iteration number and pre-specified maximum number of iteration cycles, respectively.

2) Proposed PSO-PID Controller: This paper presents a PSO-PID controller for searching the optimal controllers parameters of MIMO nonlinear system ,kp_i , ki_i and , kd_i with the PSO algorithm. Each individual K_i contains 3*m members kp_i , ki_i and , kd_i. Its dimension is n*3*m. The searching procedures of the proposed PSO-PID controller were shown as below [12]. Optimal design for both conventional PID controllers can be fulfilled using PSO technique. Based on the PSO technique, the PID controller can be tuned to some parameters values that minimize those fitness functions given in (17), (18) and(19). The algorithmic steps for the PSO is as follows:

- **Step** 1: Select the number of particles, generations, tuning accelerating coefficients c_1 and c_2 and random numbers r_1, and r_2 to start the optimal solution searching.
- **Step** 2: Initialize the particle position and velocity.
- **Step** 3: Select the particles individual best value for each generation.
- **Step** 4: Select the particles global best value, particle near the target among all the particles, is obtained by comparing all the individual best values.
- **Step** 5: Update particle individual best pbest , global best gbest , in the velocity equation (20) and obtain the new velocity.
- **Step** 6: Update the new velocity value in Eq. (21) and obtain the position of the particle.
- **Step** 7: Find the optimal solution with a minimum (IAE, ISE, ITSE) from the updated new velocity and position values.

V. SIMULATION RESULTS

This section presents a simulation example to shown an application of the proposed control algorithm and its satisfactory performance. The MIMO nonlinear system is characterized by the equation(16),[Song, 2006], [Petlenkov, 2007].

$$\begin{cases} y1(k) = \frac{a_1 y_1(k-1) y_2(k-1)}{1 + a_2 y_1^2(k-1) + a_3 y_2^2(k-1)} + \\ \quad a_4 u_1(k-2) + a_5 u_1(k-1) + a_6 u_2(k-1) \\ y_2(k) = \frac{b_1 y_2(k-1) sin(y_2(k-2))}{1 + b_2 y_2^2(k-1) + b_3 y_1^2(k-1)} + \\ \quad b_4 u_2(k-2) + b_5 u_2(k-1) + b_6 u_1(k-1) \end{cases} \quad (23)$$

The system parameters are: $a_1 = 0.7; a_2 = 1; a_3 = 1; a_4 = 0.3; a_5 = 1eta_6 = 0.2; b_1 = 0.5; b_2 = 1; b_3 = 1; b_4 = 0.5; b_5 = 1etb_6 = 0.2$. which is used as a test for control techniques introduced in this paper, to demonstrate the effectiveness of the proposed algorithms. Here y_1 and y_2 are the outputs, u_1 and u_2 are the inputs which is uniformly bounded in the region $[-2, 2]$.
We choose $[y_1(k-1), y_1(k-2), u_1(k-1), u_1(k-2), u_2(k-1)]$ and $[y_2(k-1), y_2(k-2), u_2(k-1), u_2(k-2), , u_2(k-1)]$ as inputs variables, and the number of fuzzy rules is four. The setup applied in this work was the following: the population size was 20, the stopping criterion was 30 generations, ω_{min} was 0.5, ω_{max} was 0.9, and $c_1 1 = c_2 = 2$.

In the conventionally Z-N tuned PID controller, the systems response produces high overshoot, but a better performance obtained with the implementation of PSO-based PID controller tuning. In the PSO-based PID controllers (PSO-PID), different performance index gives different results. These are shown in TABLE.III and TABLE.IV. Comparative results for the PID controllers are given below in TABLE.V and TABLE VI where the step response performance is evaluated based on the overshoot, settling time and Rise time. The corresponding plot for the step responses are shown in Fig. 2 and Fig.3.

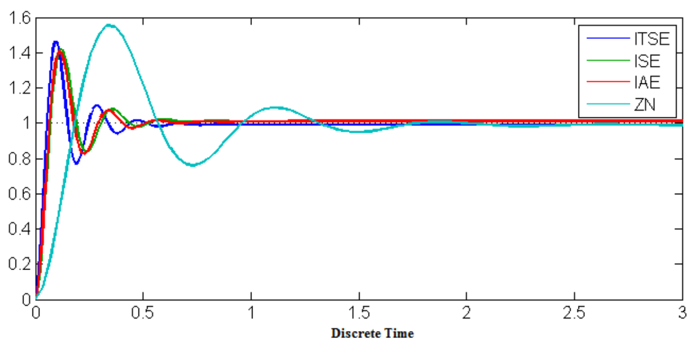

Fig. 2. system response (y1)

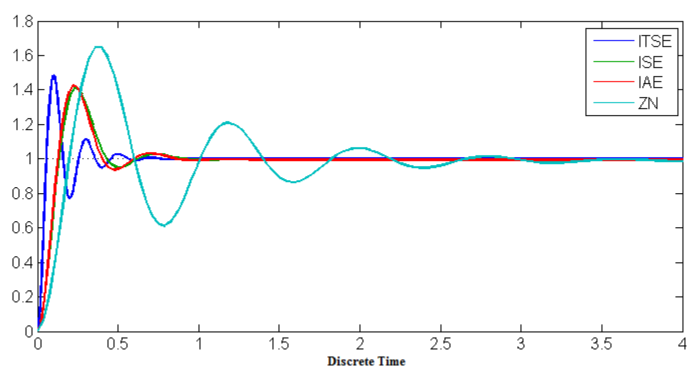

Fig. 3. system response (y2)

TABLE III
OPTIMIZED PID PARAMETERS (Y1)

Tuning Method	kp	ki	kd
Z-N-PID	32.3431	3.2943	4.4829
PSO-PID1 (ISE)	26.7236	46.2192	30.136
PSO-PID1 (IAE)	8.7263	46.2192	33.1360
PSO-PID1 (ITSE)	5.3487	15.7988	45.4369

TABLE IV
OPTIMIZED PID PARAMETERS (Y2)

Tuning Method	kp	ki	kd
Z-N PID	38.0743	4.6739	5.6734
PSO-PID1 (ISE)	35.7236	16.2192	9.136
PSO-PID2 (IAE)	39.5246	17.2514	9.7983
PSO-PID3 (ITSE)	45.3487	15.7988	41.4369

TABLE V
STEP RESPONSE PERFORMANCE FOR PID CONTROLLERS (Y1)

Tuning Method	**Overshoot(%)**	**Rise Time**	**Setting Time**
Z-N PID	55.3483	0.1264	1.6733
SPSO-PID1 (ISE)	41.6977	0.0474	0.6182
PSO-PID2 (IAE)	40.5825	0.0453	0.4791
PSO-PID3 (ITSE)t	46.1849	0.0374	0.4246

TABLE VI
STEP RESPONSE PERFORMANCE FOR PID CONTROLLERS (Y2)

Tuning Method	**Overshoot(%)**	**Rise Time**	**Setting Time**
Z-N PID	64.8174	0.1352	3.2941
PSO-PID1 (ISE)	40.8062	0.0929	0.7970
PSO-PID2 (IAE)	42.3434	0.0884	0.7775
PSO-PID3 (ITSE)	48.5050	0.0389	0.5265

VI. CONCLUSIONS

This paper presents a design method for determining the PID controller parameters using the PSO method for MIMO nonlinear systems. The proposed method integrates the PSO algorithm with performance criterions into a PSO-PID controller. The comparison between PSO-based PID (PSO-PID) performance and the ZN-PID is presented. The results show the advantage of the PID tuning using PSO-based optimization approach.

REFERENCES

[1] Astrom, K.J. Hgglund, T. (1995), PID Controllers: Theory, Design, and Tuning, 2nd Edition, ISA, ISBN 978-1-55617-516-9, North Carolina.
[2] Suda, N. (1992), PID Control. Asakura Publishing Co., Ltd., ISBN 978-4-254-20966-2, Tokyo.
[3] Luyben, W. L. (1986), A simple method for tuning SISO controllers in a multivariable system, Industrial Engineering Chemistry Product Research and Development, 25, 654660.
[4] Yusof, R., Omatu, S. (1993), A multivariable self-tuning PID controller, Internal Journal of Control, 57(6), 13871403.
[5] Wang, Q. G., Zou, B., Lee, T. H., Qiang, B. (1997), Auto-tuning of multivariable PID controllers from decentralized relay feedback, Automatica, 33(3), 319330.

[6] Kennedy J. and Eberhart C, Particle Swarm Optimization, Proceedings of the IEEE International Conference on Neural Networks, Australia, pp. 1942-1948,1995.

[7] Oliveira, P. M, Cunha, J. B, and Coelho,J.o.P, Design of PID controllers using the Particle Swarm Algorithm, Twenty-First IASTED International Conference: Modeling, Identification, and Control (MIC 2002), Innsbruck, Austria. 2002.

[8] I. Lagrat, H. Ouakka and I. Boumhidi LESSI, Fuzzy clustring for identification of Takagi-Sugeno fuzzy models of a class of nonlinear multivariable systems, Facult Des Sciences, B. P 1796 Atlas, 30000 Fez- MOROCCO.

[9] T. Takagi, M.Sugeno, Fuzzy Identification of Systems and its Application to Modeling and Control, IEEE Transactions on Systems, Man and Cybernetics, vol. 15 (1), 116-132, 1985.

[10] S.J.Qin and T.A.Badgwell, A survey of industrial model predictive control technology, Control Engineering Practice, No. 11, pp733- 764, 2003.

[11] Wei-Der Chang, A multi-crossover genetic approach to multivariable PID controllers tuning, Department of Computer and Communication, Shu-Te University, Kaohsiung 824, Taiwan, Expert Systems with Applications 33 (2007) 620626.

[12] S. Morkos, H. Kamal, Optimal Tuning of PID Controller using Adaptive Hybrid Particle Swarm Optimization Algorithm, Int. J. of Computers, Communications and Control, ISSN 1841-9836, 2012.

[13] Astrom, K. J. and T., Hagglund, PID Controllers: Theory, Design and Tuning, ISA, Research Triangle, Par, NC, (1995).

[14] Soltani, M., Chaari, A. and Benhmida, F. (2012a), A novel fuzzy c regression model algorithm using new measure of error and based on particle swarm optimization, International Journal of Applied Mathematics and Computer Science, Vol. 22 No. 3, pp. 617-28.