INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL.16,NO.1,MARCH 2011,8-18

Optimal Switching Instants of Linear Switched Systems Based on Meta-heuristic Approaches

Mouadh SAKLY, Anis SAKLY, Nesrine MAJDOUB and Mohamed BENREJEB

Abstract- **In this paper, optimal control problems for switched systems are studied. In particular, we focus on such problems given a prespecified sequence of active subsystems and propose two approaches based on meta-heuristics to find the optimal switching instants. The first one is based on the Genetic Algorithm (GA) and the second on the Particle Swarm Optimization (PSO) algorithm. The objective is to minimize the performance index, depending on these instants, over a finite time horizon. We assume that a pre-assigned sequence is given and that at each switching instant, a jump in the state space variable may occur and that an additional cost is then associated with it.**

***Index Terms*— Hybrid systems, Linear switched systems, Optimal control, Genetic Algorithm, Particle Swarm Optimization, switching instants, Autonomous and controlled systems.**

1. INTRODUCTION

Hybrid control [1] has become a hot research topic since it combines the standard control, where dynamic systems are typically described by differential or difference equations, with discrete logic or discrete events [2]. When digital computers, digital networks, and embedded systems involved in control systems become ubiquitous and increasingly complex, understanding the coupling between logic-based components and continuous physical systems becomes important [3][4]. Moreover, purposely making use of hybrid control strategies to achieve the control objective unachievable via conventional control methods has become practically feasible [5][6]. As a special type of hybrid systems, a switched system consists of several subsystems and a switching law specifying the active subsystem at each time instant.

Many real processes such as robotic systems [7], [8], air traffic control [9], computer processes [10], automotive systems [11], chemical processes, and manufacturing processes, can be modeled as such systems. Recently, optimal control problems for switched systems have attracted many researchers.

Manuscript received Septembre 23, 2010; revised Avril 25, 2011.
*Mouadh Sakly is currently a PhD candidate in Electrical Engineering in the National School of Engineers of Monastir (ENIM)
(email: s.mouadh@gmail.com)
**Anis sakly is currently a professor at departement of Electrical Engineering at National School of Engineers of Monastir (email: sakly_anis@yahoo.fr)
***Nesrine Majdoub is currently a PhD candidate in Electrical Engineering in the National School of Engineers of Tunis (ENIT) (email: majnesrine@yahoo.fr)
**** Mohamed BENREJEB is a full professor at the National School of Engineers of Tunis (ENIT)
(email: mohamed.benrejeb@enit.rnu.tn)

A solution for a such problem consists in determining not only the continuous control inputs, as in a conventional optimal control problem but also on the computation of the discrete control inputs defined in terms of switching instants and their corresponding active subsystems [12].

The approach proposed in [12- 14] is to decompose the problem into two stages. Stage (a) is a conventional optimal control problem that finds the optimal cost giving the sequence of active subsystems and the switching instants. Stage (b) is a nonlinear optimization problem that finds the local optimal switching instants.

In particular, stage (b) performed for nonlinear optimization problem finds optimal switching instants based on gradient methods [12][15][16]. The proposed approach transforms the problem set by the equivalence of the switching instants. The gradient method is used to generate a formula for the first and second order derivative of the cost, the meaning of continuous control, in relation to switching instants. The conceptual algorithm, applied for this optimization methodology, allows at each instant to:

- find a value of cost by solving an optimal control problem (stage(a)),
- find first order derivative of the cost (and second order derivative if it is to be used),
- use gradient information of cost to update switching instants.

When a smaller gradient term than a given smaller number is reached, the optimal switching instants correspond to the last update instants. The difficulty with this approach is the unavailability of an explicit form of the first and second order derivative of the cost formula. Being presented with such difficulties, some evolutionary meta-heuristic algorithms, such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are proposed and have been applied to solve General Switched Linear Quadratic (GSLQ) optimal control problems. GA and PSO offer a high degree of flexibility and robustness in dynamic environments.

The rest of this paper is organized as follows: after briefly introducing the GSLQ problem formulation in Section 2, we present the meta-heuristic optimization approaches based on GA and PSO, respectively, in Sections 3 and 4. In Section 5, we propose a solution for linear switched systems for both autonomous and controlled systems cases, and we compare the two methods with the gradient method through several numerical examples, Section 6 concludes this paper.

2. PROBLEM FORMULATION

2.1. Linear Switched Systems

In this paper, we consider two types of linear switched systems consisting of the subsystems $\dot{x} = A_i x$ for the autonomous case and $\dot{x} = A_i x + B_i u$ for the controlled case with $i \in I = \{1, 2, 3, ..., M\}$. In order to control a switched system, one needs to choose not only a continuous input but also a switching sequence. A switching sequence in $t \in [t_0, t_f]$ regulates the sequence of active subsystems and is defined as follow:

$$\sigma = ((t_0, i_0), (t_1, i_1), ..., (t_k, i_k), ..., (t_K, i_K)) \qquad (1)$$

where $K \geq 0$; $t_0 \leq t_1 \leq ... \leq t_K$ and $i_k \in I = \{1, 2, 3, ...K+1\}$ for $k = 1, 2, 3, ...,$ and K. Note that (t_k, i_k) indicates that at instant t_k the system switches from subsystems i_{k-1} to i_k, and subsystem i_k is active during the time interval $[t_k, t_{k+1}]$. For a switched system to be well-behaved, we only consider sequences which switch at most a finite number of times in $[t_0, t_f]$. If we regard σ as a discrete input, then the overall control input to the system is a pair (σ, u). Finally, we note that the feature distinguishing a switched system from a general hybrid system is that its continuous state does not exhibit jumps at the switching instants. Such a feature makes the computation of continuous inputs amenable via the usage of conventional optimal control methods.

2.2. Optimal Control Problem

Consider a controlled linear switched system with subsystems [15][17]
$\dot{x} = A_i x + B_i u$ with $i \in I = \{1, 2, 3....M\}$. Assume that a perspecified sequence of active subsystems $(1, 2, 3, ..., k, ..., K+1)$ is given. Find optimal switching instants $t_1, ..., t_K$, $t_0 \leq t_1 \leq ... \leq t_K \leq t_f$ such that the corresponding continuous state trajectory x departs from a given initial state $x(t_0) = x_0$, and the cost

$$J(t_1, ..., t_k) = \psi(x(t_f)) + \int_{t_0}^{t_f} L(x(t), u(t))dt \qquad (2)$$

is minimized, where ψ and L will be considered later in the general quadratic problem.

We observe that a small disturbance of $t_1, ...,$ and t_K will only cause a small disturbance of J value. Furthermore, it can be shown that J is a continuously differentiable function of $t_1, ...,$ and t_K.

We propose in this paper to solve the problem of GSLQ problems. We consider a special class of optimal control problems for switched systems, i.e., general switched linear quadratic problems.

First of all, we state the problem as follows:

For a switched system S with linear subsystems $\dot{x} = A_i x + B_i u$, $i \in I$ and a given switching sequence $\sigma = ((t_0, i_0), (t_1, i_1) ... (t_K, i_K))$ find the optimal switching instants $t_1, ..., t_K$ such that the cost function in a general quadratic form

$$\begin{pmatrix} J = \frac{1}{2} x(t_f)^T Q_f x(t_f) + M_f x(t_f) + L_f \\ + \int_{t_0}^{t_f} \left(\frac{1}{2} x^T Q x + x^T V u + \frac{1}{2} u^T R u + M x + N u + W \right) dt \end{pmatrix} \qquad (3)$$

is minimized where t_0, t_f and $x(t_0) = x_0$ are given, $Q_f, Q \geq 0$ and $R > 0$. Note that for general quadratic control of a linear system $\dot{x} = A_i x + B_i u$, we can use the dynamic programming approach to obtain the following results.

The optimal value function

$$\dot{V}^*(x, t) = \frac{1}{2} x P^T(t) x + S(t) x + T(t) \qquad (4)$$

satisfy the following general Ricatti equations:

$$-\dot{P}(t) = \begin{pmatrix} Q + P(t) A_i + A_i^T P(t) \\ -(P(t) B_i + V) R^{-1} (B_i^T P(t) + V^T) \end{pmatrix} \qquad (5)$$

$$-\dot{S}(t) = \begin{pmatrix} M + S(t) A_i \\ -(S(t) B_i + N) R^{-1} (B_i^T P(t) + V^T) \end{pmatrix} \qquad (6)$$

$$-\dot{T}(t) = W - \frac{1}{2}(S(t) B_i + N) R^{-1} (B_i^T S^T(t) + N^T) \qquad (7)$$

where

$$P(t) = P^T(t) \qquad (8)$$

The optimal control is of the feedback form

$$u(t) = -K(t) x(t) - E(t) \qquad (9)$$

with

$$K(t) = R^{-1} (B^T P(t) + V^T) \qquad (10)$$

$$E(t) = R^{-1} (B^T S^T(t) + N^T) \qquad (11)$$

Focusing now on the general switched linear quadratic problem, assume that for any nominal switching instants, we always choose $u(.)$ in the form (9) which is optimal in this case. We can choose the nominal $K(.)$ and $E(.)$ rather than $u(.)$ to be fixed at each iteration of optimization (but be updated after the iteration). This can give us the flexibility of letting $u(.)$ vary as a function of x since u depends on x now.

3. GENETIC ALGORITHM OVERVIEW

3.1. Introduction

The concept of the Genetic Algorithm, first formalized by Holland and extended to functional optimization by De Jong [18], involves the use of optimization search strategies patterned after the Darwinian notion of natural selection and evolution.

During a GA optimization, a set of trial solutions is chosen and evolves toward an optimal solution under the selective pressure of the objective function [19].

In general, a GA optimizer must be able to perform five basic tasks

- encode the solution parameters in the form of chromosomes,
- initialize a starting population,
- evaluate and assign fitness values to individuals in the population,
- perform reproduction through the fitness weighted selection of individuals from the population,
- perform recombination and mutation to produce members of the next generation.

This algorithm is represented in Fig. 1.

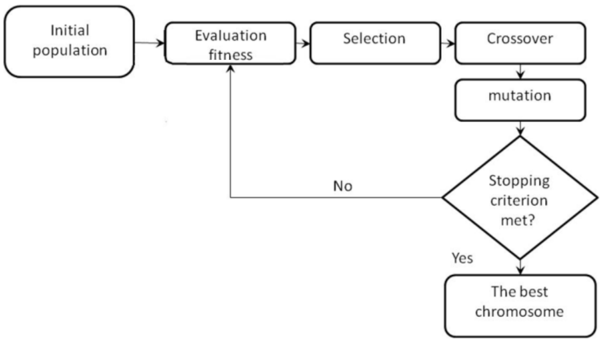


Fig. 1. Genetic Algorithm

3.2. Selection

Each of the individuals is selected by its fitness value. There are many different methods of selection. We are interested in the elitism method that arranges the most promising individuals on the top of population and crosses them to obtain an improvement of the population. In fact, the individuals that give the best fitness have a good probability to be selected for the next generation.

3.3. Crossover

After the selection step, a crossover allows a generation of new individuals. The crossover step cuts two chromosomes, named parents, at a random place. Then these two individual strings are reversed and two chromosomes are created and named children.

3.4. Mutation

Mutation modifies in a random way and with a small probability the value of a chromosome.

3.5. Optimization with GA

Our approach takes a random population of switching instants. Each individual corresponds to a chromosome representing the switching instants from t_1 to t_K where $K+1$ is the number of subsystems. Then, the algorithm evaluates this population using selection, crossover and mutation operations to obtain after a fixed number of generations of the optimal switching instants. We note that GA uses a real type encoding and not a binary one.

The proposed optimization algorithm consists of:

Step 1:

- Define the parameters of the algorithm: size of population, maximal number of generations, crossover probability and mutation probability, and bounds of search space.

Step 2:

- Generate randomly the initial population of switching sequence instants.
- Initialize the iteration index I .

Step 3:

- Evaluate and assign fitness values $J_1(t)$ to individuals in the population by solving an optimal control problem (stage (a)).
- Identify the individuals corresponding to the minimum cost and re-arrange them using the elitism method.
- Produce children of parents selected by crossing.
- Mute children.
- Evaluate and assign fitness values $J_1(t)$ to individuals by solving an optimal control problem (stage (a)).
- Identify the individual corresponding to the minimum cost.
- Insert the individual, corresponding to the initial minimal cost, to the new population.

Step 4:

- Increment I .
- If $I < \max\_I$ the maximal number of generations then return to **step 3**, else go to **step 5**.

Step 5:

- Return the best individual for the minimum cost.

4. PARTICLE SWARM OVERVIEW

4.1. Introduction

Particle swarm optimization is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995 [20], inspired from social behavior of bird flocking or fish schooling.

PSO shares many similarities with evolutionary computation techniques such as GA.

The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles [21- 23].

Each particle is characterized by a position p and velocity v. During flight, each particle updates its own velocity and position by taking benefit from its best experience and the best experience of the entire population [24].

Let k be the iteration index. The new particle velocity and position are updated according to the move equations [25] [26]:

$$v_{k+1} = w_k v_k + b_1 r_1 \left(pbest_k - p_k \right) + b_2 r_2 \left(pgbest_k - p_k \right) \quad (12)$$

$$p_{k+1} = p_k + v_{k+1} \quad (13)$$

with :

- p_k : position of each particle at iteration k
- v_k : velocity of each particle at iteration k
- b_1 and b_2 : strength of attraction, fixed positive coefficients of acceleration
- r_1 and r_2 : two random numbers drawn uniformly in the interval [0,1]
- $pbest_k$: best position discovered by the particle until the iteration index k.
- $pgbest_k$: global best particle position of the entire population.

Inertia weight w controls the impact of the previous velocities on the current velocity. It influences the tradeoff between the global and local exploitation abilities of the particles. For initial stages of the search process, large inertia weight to enhance the global exploitation is recommended while for last stages, the inertia weight is reduced for better local exploration [26].

Weight is updated as:

$$w_k = w_{max} - \left(\frac{w_{max} - w_{min}}{max\_I} \right) . k \quad (14)$$

where w_{min} and w_{max} are minimum and maximum values of w and *max_I* represents the number of maximal iterations.

At each iteration, the behavior of a given particle is a compromise among three possible choices (see Fig. 2):

- to follow its own way,
- to go toward its best previous position,
- to go toward the best neighbor.

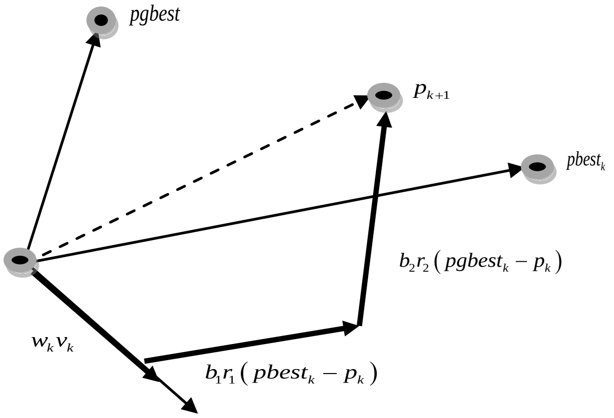


Fig. 2. The particule flight illustration

4.2. Optimization with PSO

As the same approach of GA, we take a random population of switching instants. Hence, each particle represents the switching instants from t_1 to t_K where $K+1$ is the number of subsystems.

The proposed optimization algorithm is composed of five steps:

Step 1:

- Define and initialize the parameters of the algorithm: number of particles, the coefficients of confidence b_1, b_2, the maximal number of iterations *max_it*, the bounds of inertia weight w_{max} and w_{min} and the bounds of search space.

Step 2:

- Initialize randomly the positions and velocities of particles into the search space.
- By solving an optimal control problem (stage (a)) evaluate the fitness values of each particle.
- Identify the switching instants corresponding to the best fitness (minimum fitness).
- Affect the best instant to $pbest$: $pgbest = pbest$.
- Initialize iteration index k.

Step 3:

- Update v_K and p_K.
- Evaluate fitness values to each sequence by solving stage (a).
- Find the switching instants corresponding to the best fitness, $pbest_k$.
- Compare initial $pgbest_k$ and actual $pbest_k$:

  If $pbest_k < pgbest_k$ then $pgbest_k \leftarrow pbest_k$

Step 4:

- Increment iteration index k.
- Compare the iteration index k and *max_I* :

If $k < max\_I$ then return to **Step 3**, else go to **Step 5**.

Step 5:

- Return the best switching instants for the minimum cost.

5. NUMERICAL EXAMPLES

In this section, we illustrate some numerical examples to validate the two proposed approaches.

For the first approach, based on GA, we keep the same parameters for all examples :

- Population size : 30
- Maximal number of generations : 20
- Crossover probability : 0.9
- Mutation probability : 0.2

All the same for the second approach based on PSO, we take the following parameters:

- Swarm size : 30
- Maximal number of iterations : 20
- $b_1 = b_2 = 2$
- $w_{max} = 0.9$ and $w_{min} = 0.4$

5.1. Example 1 [27]:

We consider the following autonomous linear switched system:

- subsystem 1:

$$\dot{x} = A_1 x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} x$$

- subsystem 2:

$$\dot{x} = A_2 x = \begin{bmatrix} 0.1 & -0.5 \\ 0.5 & 0.1 \end{bmatrix} x$$

- subsystem 3:

$$\dot{x} = A_3 x = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} x$$

with $t_0 = 0$ and $t_f = 3$ s. The system switches at $t = t_1$ from subsystems 1 to 2 and at $t = t_2$ from 2 to 3, such that $0 \le t_1 \le t_2 \le t_f$.

The criterion to minimize is:

$$\left(\begin{array}{l} J = \frac{1}{2}\left(x_1\left(t_f\right)\right)^2 + \frac{1}{2}\left(x_2\left(t_f\right)\right)^2 \\ + \frac{1}{2}\int_0^{t_f}\left(\left(x_1(t)\right)^2 + \left(x_2(t)\right)^2\right)dt \end{array} \right)$$

such that $x_1(0) = -1$ and $x_2(0) = 3$.

We present now the different results obtained during the optimization of this system using the two methods mentioned previously.

Figs. 3 and 4 present the results obtained by using a GA-based approach while Figs. 5 and 6 give results obtained by a PSO-based approach.

Figs. 3 and 5 show the state trajectory while Figs. 4 and 6 show the evolution of the cost fitness J .

In order to compare the performance of the two approaches we present in Table 1 the obtained numerical results.

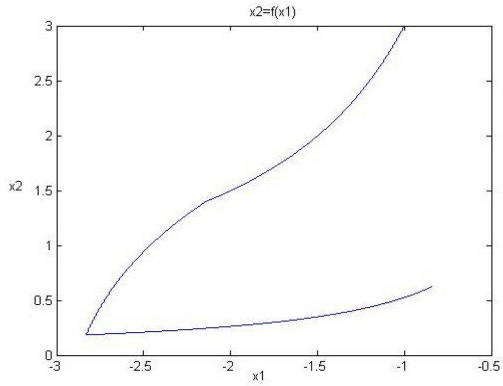


Fig. 3. State trajectory using GA-based approach for Example 1

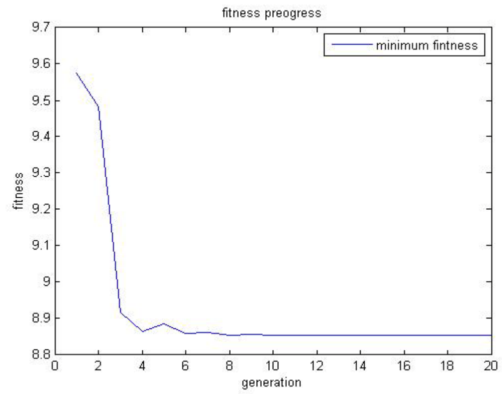


Fig. 4. Fitness progress J using GA-based approach for Example 1

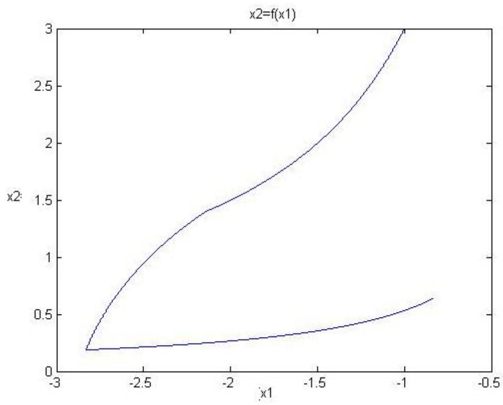


Fig. 5. State trajectory using PSO-based approach for Example 1

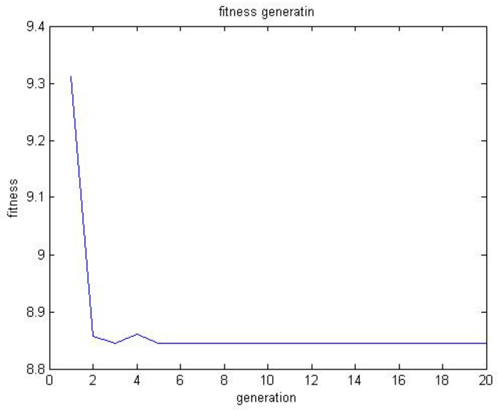


Fig. 6. Fitness progress J using PSO-based approach for Example 1

Table 1. Results of optimization for example 1

	t_1 (s)	t_2 (s)	Minimum fitness
GA-based approach	0.7435	1.7989	8.8537
PSO-based approach	0.7609	1.8020	8.8453

We can remark that we have obtained nearly the same cost by using the two approaches

5.2. Example 2 [27]:

The system 2 corresponds to system 1 but by introducing jumps between the subsystems.

The switching jumps are presented as follow :

$$x\left(t_1^+\right)=\begin{bmatrix}1.1 & 0\\ 0 & 0.9\end{bmatrix}x\left(t_1^-\right)+\begin{bmatrix}-0.2\\ 0.2\end{bmatrix}$$

$$x\left(t_2^+\right)=\begin{bmatrix}0.9 & 0\\ 0 & 1.1\end{bmatrix}x\left(t_2^-\right)+\begin{bmatrix}0.2\\ -0.2\end{bmatrix}$$

The criterion to minimize is:

$$\left(\begin{array}{c} J=\frac{1}{2}\left(x_1\left(t_f\right)\right)^2+\frac{1}{2}\left(x_2\left(t_f\right)\right)^2+\frac{1}{2}\int_0^{t_f}\left(\left(x_1(t)\right)^2+\left(x_2(t)\right)^2\right)dt \\ +\sum_{k=1}^{2}\left(\frac{1}{2}\left(x_1\left(t_k^-\right)\right)^2+\frac{1}{2}\left(x_2\left(t_k^-\right)\right)^2\right) \end{array}\right)$$

Such that $x_1(0)=-1$ and $x_2(0)=3$.

Figs. 7 and 8, present the results obtained by using the GA-based approach while Fig. 9 and Fig. 10 present those obtained by the PSO-based approach. Figs. 7 and 9 show the state trajectory while Figs. 8 and 10 show the evolution of the cost fitness J.

Table 2 summarizes the numerical results obtained. We note that in [27] using the gradient method the optimal switching instants obtained are t_1 = 0.6167s, t_2 = 1.6724s and the corresponding optimal cost J is 16.6518.

Even though, for this example and the others, the gradient method converges to the optimum instants, the proposed approach based on meta-heuristics is better. Indeed, this approach can find those instants regardless of the choice of optimal initial instants and is extendable to the case of nonlinear switching systems and therefore non-convex cost functions (fitness) by obtaining the global optima.

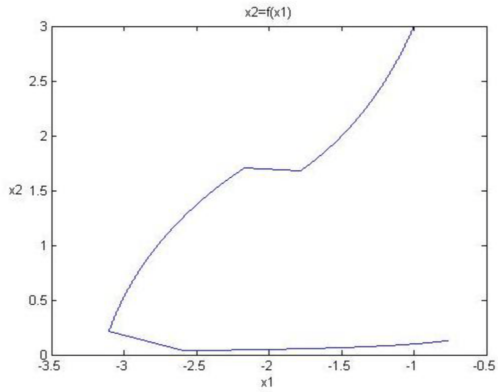


Fig. 7. State trajectory using GA-based approach for Example 2

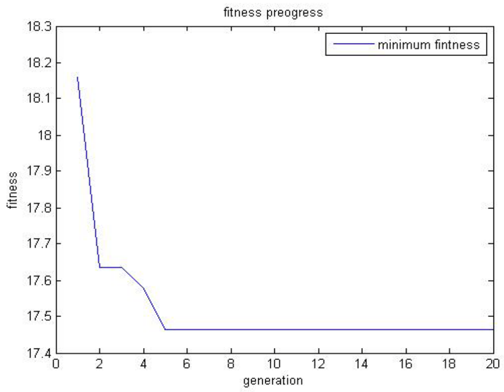


Fig. 8. Fitness progress J using GA-based approach for Example 2

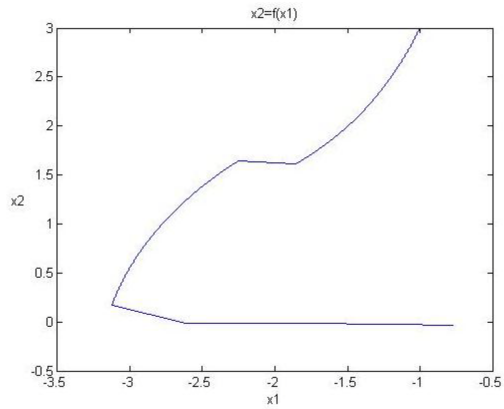


Fig. 9. State trajectory using PSO-based approach for Example 2

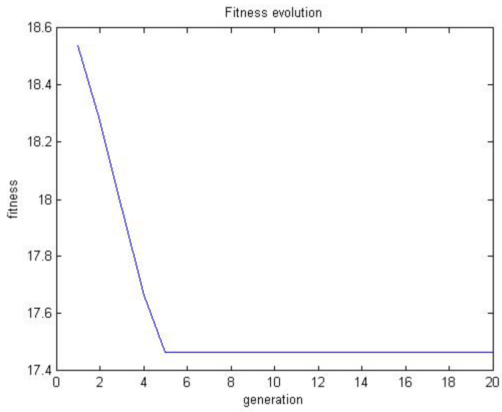


Fig. 10. Fitness progress J using PSO-based approach for Example 2

Table 2. Results of optimization for example 2

	t_1 (s)	t_2 (s)	Minimum fitness
GA-based approach	0.6159	1.7794	17.463
PSO-based approach	0.6214	1.7786	17.463

We can remark that we have obtained exactly the same cost by using the two approaches

5.3. Example 3 [12]:

We consider now a controlled linear switched system consisting of :

- subsystem 1:

$$\dot{x} = A_1 x + B_1 u = \begin{bmatrix} 0.6 & 1.2 \\ 0.8 & 3.4 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u$$

- subsystem 2:

$$\dot{x} = A_2 x + B_2 u = \begin{bmatrix} 4 & 3 \\ -1 & 0 \end{bmatrix} x + \begin{bmatrix} 2 \\ -1 \end{bmatrix} u$$

We assume that $t_0 = 0$ and $t_f = 2$ s. The system switched at $t = t_1$ from subsystems 1 to 2, such that $0 \le t_1 \le t_f$. We want to find an optimal switching instant t_1 that minimizes the criterion :

$$J = \frac{1}{2}\left(x_1\left(t_f\right)-4\right)^2 + \frac{1}{2}\left(x_2\left(t_f\right)-2\right)^2 + \frac{1}{2}\int_0^{t_f}\left(\left(x_2(t)-2\right)^2 + u^2(t)\right)dt$$

such that:

$x_1(0) = 0$ and $x_2(0) = 2$.

We present now the different results obtained. Figs. 11-13 present the results obtained by using the GA-based approach while Figs. 14-16 present results obtained by the PSO-based approach.

Figs. 11 and 14 present the control input evolution, Figs. 12 and 15 present the state trajectory and Figs. 13 and 16 present the evolution of the cost J .

Table 3 summarizes the numerical results obtained. We note that in [12] using the gradient method, the optimal switching instant obtained is t_1 = 0.1897s and the corresponding optimal cost J is 9.7667.

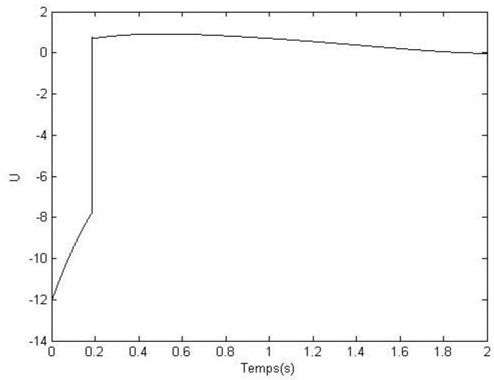


Fig. 11. Control input u evolution using GA-based approach for Example 3

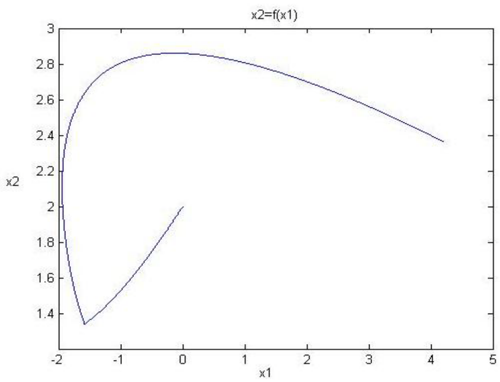


Fig. 12. State trajectory using GA-based approach for Example 3

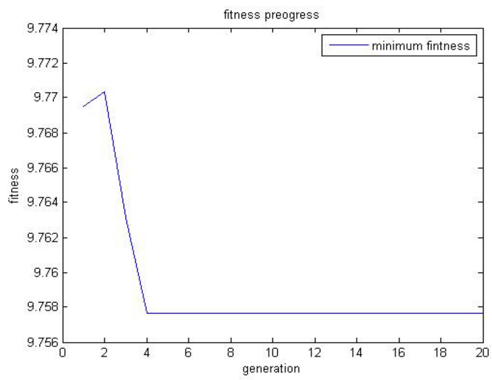


Fig. 13. Fitness evolution J using GA-based approach for Example 3

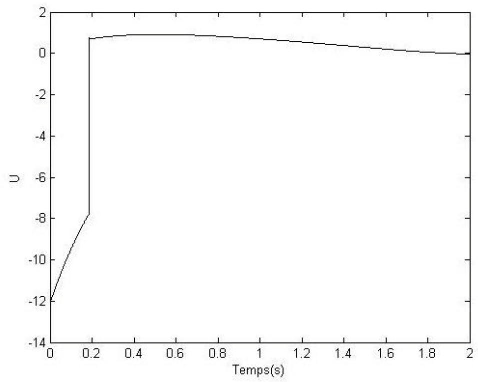


Fig. 14. Control input u evolution using PSO-based approach for Example 3

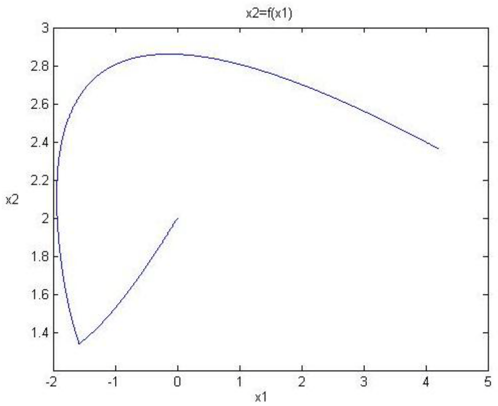


Fig. 15. State trajectory using PSO-based approach for Example 3

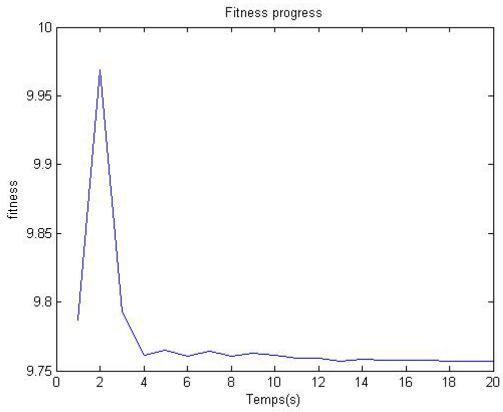


Fig. 16. Fitness evolution J using PSO-based approach for Example 3

Table 3. Results of optimization for example 3

	t_1 (s)	Minimum fitness
GA-based approach	0.1870	9.7577
PSO-based approach	0.1890	9.7651

We remark that we have obtained nearly the same cost by using the two approaches and it is similar to the cost obtained by using the gradient method

5.4. Example 4 [17]:

We consider the controlled linear switched system consisting of:

- subsystem 1:

$$\dot{x} = A_1 x + B_1 u = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$

- subsystem 2:

$$\dot{x} = A_2 x + B_2 u = \begin{bmatrix} 0.5 & 5.3 \\ -5.3 & 0.5 \end{bmatrix} x + \begin{bmatrix} 1 \\ -1 \end{bmatrix} u$$

- subsystem 3:

$$\dot{x} = A_3 x + B_3 u = \begin{bmatrix} 1 & 0 \\ 0 & 1.5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

Assume that $t_0 = 0$, $t_f = 3$ s and the system switches at $t = t_1$ from subsystem 1 to 2 and at $t = t_2$ from 2 to 3, such that $0 \le t_1 \le t_2 \le t_f$. We want to find an optimal switching instants t_1 and t_2 which minimize the criterion:

$$J = \left(\begin{matrix} \left(x_1\left(t_f\right) + 4.1437\right)^2 + \left(x_2\left(t_f\right) - 9.3569\right)^2 \\ + \frac{1}{2}\int_0^{t_f} \left(u^2\left(t\right)\right) dt \end{matrix} \right)$$

such that:

$x_1(0) = 4$ and $x_2(0) = 4$

Figs. 17-19 presents the results obtained by using GA-based approach while Figs. 20-22 present those obtained by the PSO-based approach.

Figs. 17 and 20 show the control input evolution, Figs. 18 and 21 present the state trajectory while Figs. 19 and 22 show the evolution of the cost fitness J .

Table 4 presents the numerical results obtained. We note that in [18] using the gradient method the optimal switching instants obtained are t_1 = 1s, t_2 = 2s and the corresponding optimal cost J is 5.0487×10^{-29} .

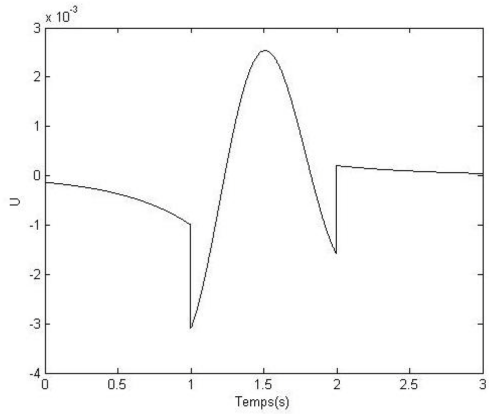


Fig. 17. Control input u evolution using GA-based approach for Example 4

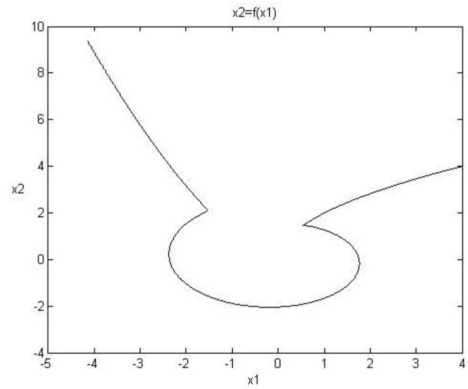


Fig. 18. State trajectory using GA-based approach for Example 4

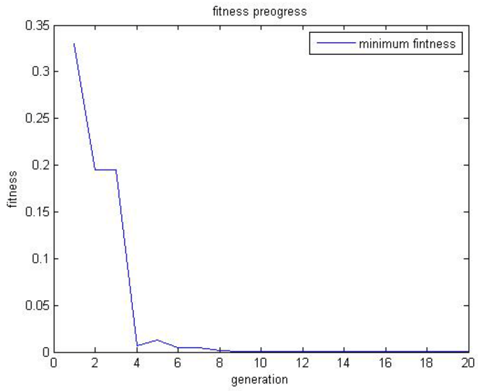


Fig. 19. Fitness evolution J using GA-based approach for Example 4

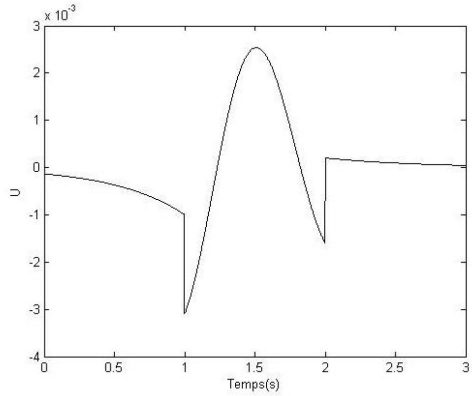


Fig. 20. Control input u evolution using PSO-based approach for Example 4

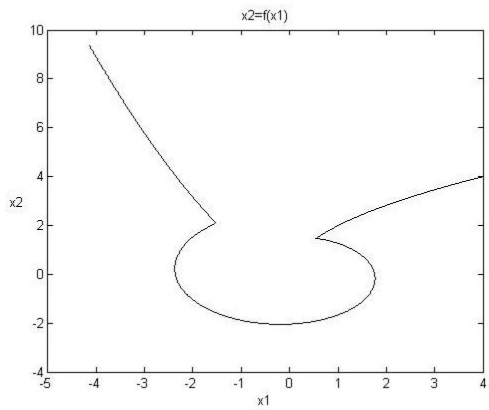


Fig. 21. State trajectory using PSO-based approach for Example 4

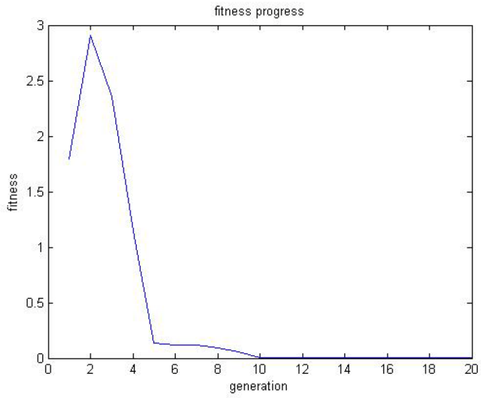


Fig. 22. Fitness evolution J using PSO-based approach for Example 4

Table 4. Results of optimization of example 4

	t_1 (s)	t_2 (s)	Minimum fitness
GA-based approach	0.9977	1.9975	0
PSO-based approach	1	2	0

We can remark that we have obtained a better cost by using the two approaches than the cost obtained by using gradient method.

To detect the more difference between the performances of the two proposed approaches, we present in Table 5 the time of running programs for each example. The programs have been developed under the MATLAB 7.1 on a PC whose features are:

- RAM : 3 GB
- Processor : Intel ® Pentium ® Dual CPU 3 GHz
- Operating system : Windows Vista.
- We remark that PSO is faster than a GA-based approach.

Table 5. Time of running programs in minute

	Example 1	Example 2	Example 3	Example 4
GA-based approach	120.471	180.056	360.287	360.145
PSO-based approach	45.150	47.894	120.301	120.138

We can remark that PSO algorithm converges faster than GA one

6. CONCLUSION

In this paper, we proposed two meta-heuristic-based solutions for optimal control problems, one based on Genetic Algorithm and another on Particle Swarm Optimization. We are interested in both autonomous and controlled linear switched systems and we demonstrated the effectiveness of these approaches to solve optimal control problems. However, these meta-heuristic-based approaches can be extended to nonlinear switched systems just by changing the stage (a) of the proposed algorithm with its equivalent in the nonlinear approach. In this case, the proposed approaches can yield better results compared to those obtained by the gradient method because the cost function is not convex for nonlinear systems.

REFERENCES

[1] A. van der Schaft and H. Schumacher. "An Introduction to Hybrid Dynamical Systems". *Lecture Notes in Control and Information* Sciences,Vol 251, Springer-Verlag, 2000.

[2] P. Ramadge, W. Wonham. "The control of discrete event systems". *Proceedings of the IEEE*, 77(1):81-98, 1989.

[3] M.S. Branicky, V. S. Borkar, S. K. Mitter, "A unified framework for hybrid control: model and optimal control theory", *IEEE Trans. on Auto. Contr.*, 43 (1), pp. 31-45, 1998.

[4] K. Gokbayrak, C. G. Cassandras, "A hierarchical decomposition method for optimal control of hybrid systems", In *Proc. of the 39th IEEE CDC*, pp.1816-1821, Sydney,Australia,2000.

[5] K. J. Astrom, K. Puruta. "Swinging up a pendulum by energy control". Automatica. 36:287-295, 2000.

[6] J. Hespanha, D. Liberzon, A. S. Morse. "Overcoming the limitations of adaptive control by means of logic-based switching. " *Syst. & Contr. Lett.* 2003, pp 49-6, vol. 49.

[7] T. Bak, J. Bendtsen, and A.P. Ravn, "Hybrid control design for a wheeled mobile robot, Hybrid Systems: Computation and Control", O. Maler, Amir Pnueli (Eds), no. 2623 in LNCS, pp.50-65, Spinger, 2003.

[8] A. Back, J. Guckenheirmer, and M. Myers, "A Dynamical Simulation Facility for Hybrid Systems", *in Hybrid systems, Lecture Notes in Computer Science*, vol. 736, R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, Eds, pp. 255-267, New York: Springer, 1993.

[9] C. Tomlin, G.J. Pappas, J. Lygeros, D.N. Godbole, and S. Sastry, "Hybrid control models of next generation air traffic management", Hybrid Systems IV, no 1273 in LNCS, Springer 1997.

[10] A.S. Matveev, and A.V. Savkin, "Qualitative Theory of hybrid dynamical systems", 348 p. 45 illus, BirkhÄauser, 2000.

[11] A. Bemporad, P. Borodani, and M. Manneli, "Hybrid control of an automotive robotized gearbox for reduction of consumptions and emissions, Hybrid Systems :Computation and Control", O. Maler, Amir Pnueli (Eds), no. 2623 in LNCS,pp 81-96, Spinger, 2003.

[12] X. Xu and P. J. Antsaklis, "Optimal Control of Switching Systems Based on Parameterization of the Switching Instants" *IEEE Trans. Automat. Contr*. pp 2-16 vol. 49, 2004.

[13] X. Xu and P. J. Antsaklis, "Optimal control of switched systems: new results and open problems*,"* in *Proc. 2000 Amer. Control Conf*., pp. 2683–2687, Chicago, 2000.

[14] X. Xu and P. J. Antsaklis,, "Results and perspectives on computational methods for optimal control of switched systems" *in Hybrid Systems: Computation and Control* (HSCC 2003), Prague, Czech Republic, pp. 540– 555, vol. 2623, 2003.

[15] X. Xu and P. J. Antsaklis, "A Dynamic Programming Approach for Optimal Control of Switched Systems," in *Proceedings of the 39th IEEE Conference on Decision and Control* Sydney, pp.1822-1827, Australia, 2000.

[16] R. Luus, Y. Q. Chen, "Optimal switching control via direct search optimization," in *Proceedings of the 2003 IEEE International Symposium on Intelligent Control*, Houston, Texas, pp.302-306, 2003.

[17] X. Xu' and P J. Antsaklis "An approach of solving General switched linear quadratic optimal control problems" in *Proceedings of the 40th IEEE conference on decision and control,* pp 2478-2483,Orlando- Florida, USA, December 2001.

[18] D. E. Goldberg, "Genetic algorithm in search, optimization and machine learning",Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989.

[19] M. Sakly, A. Sakly, N. Majdoub and M. Benrjeb, "Optimization of Switching Instants for Optimal Control of Linear Switched Systems based on Genetic Algorithms", *The 2ndIFAC International Conference on Intelligent control Systems and signal Processing*, pp.1-5, ICONS *2009, Istanbul, Turkey.*

[20] J. Kennedy and R.C. Eberhart "Particle Swarm Optimization", *Proc IEEE international conference on Neural Networks* (Perth, Australia 1995), IEEE service center, Piscataway, NJ, pp.1942- 1948.

[21] M. Clerc and P. Siarry "une nouvelle métaheuristique pour l'optimisation difficile: la méthode des essaims particulaires", EDP SCIENCE 2004.

[22] M. Shahid Shaikh and P. E. Caines "On the Optimal Control.of Hybrid Systems: Optimization of Switching Times and Combinatoric Location Schedules" in *Proc American control conference*, Denver, Calorado,2003.

[23] S. Boubaker and F. M'sahli "Solutions Based on Particle Swarm Optimization for Optimal Control Problems of Hybrid Autonomous Switched Systems". *Int. J. of Intelligent Control and Systems,* pp. 128-135, Vol. 13, n. 2, 2008.

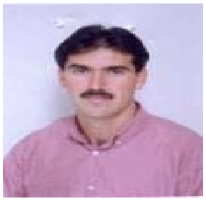
[24] S. Boubaker and F. Msahli, "Synthese d'une loi de commande optimale d'un systeme hybride linéaire à deux dynamiques par essaims particulaires", *Fifth Conference of Electrical Engineering and Automatic*, Hammamet Tunisia, 02-04 May 2006, pp.254-258.

[25] O. Chao and L.Weining, "Comparison between PSO and GA for parameters optimization of PID controller", in *Proceedings of the IEEE, International Conference on Mechatronics and Automation*, Luoyang China, 25-26 June 2006.

[26] H. Wan Jun, Q. Yan-Hui, L. Guo-Liang and Q. Wen-Yi, "Fuzzy controller design and parameter optimization", *Procceding of the fifth international conference on machine learning and cybernetics*, Dalian, 13-16 August 2006.

[27] X. Xu. And P. J. Antsaklis, "Quadratic Optimal Control Problems for Hybrid Linear Autonomous Systems with State Jumps", in *Proceedings of the American Control Conference*, pp 3393-3398, *Denver*, Colorado,*USA*, Juin 4-6, 2003.

Mouadh Sakly was born in Tunisia in 1983. He received the Electrical Engineering diploma from the National School of Engineers of Gabes (ENIG) in 2007. He obtained his master degree in electric engineering from the National School of Engineers of Monastir (ENIM) in 2009. His research interests include optimal control of dynamic hybrid systems based on meta-heuristic approaches.



Anis Sakly was born in Tunisia in 1970. He received the Electrical Engineering diploma in 1994 from National School of Engineers of Monastir (ENIM), and then the PhD degree in Electrical Engineering in 2005 from National School of Engineers of Tunis (ENIT). His is currently Professor at ENIM. His research interests are in analysis and synthesis of intelligent control systems, particularly soft computing-based control approaches.

Nesrine Majdoub was born in Tunisia in 1979. She received the Electrical Engineering diploma in 2004 from National School of Engineers of Monastir (ENIM). She is currently a PhD candidate in Electrical Engineering in the National School of Engineers of Tunis (ENIT) and perform modeling and synthesis of intelligent control of Hybrid Dynamic Systems.

Mohamed Benrejeb was born in Tunisia in 1950. He obtained the diploma of "IDN Engineer" in 1973, the Master degree of Automatic Control in 1974, the PhD degree in Automatic Control from the University (USTL) of Lille in 1976, and the DSc degree from the same university in 1980. He has been a full professor at the National School of Engineers of Tunis (ENIT) since 1985 and at Central school of Lille since 2003. His research interests are in the area of analysis and synthesis of complex systems based on classical and non conventional approaches and recently in discrete event system domain.