International Journal of Computational Intelligence and Applications
Vol. X, No. X (200X) 1-14

OPTIMAL CONTROL OF A HYSTERESIS SYSTEM BY MEANS OF CO-OPERATIVE CO-EVOLUTION

KITTIPONG BOONLONG*

Department of Mechanical Engineering,
Chulalongkorn University, Thailand

NACHOL CHAIYARATANA†

Research and Development Center for Intelligent Systems,
King Mongkut's Institute of Technology North Bangkok, Thailand

SUWAT KUNTANAPREEDA‡

Research and Development Center for Intelligent Systems,
King Mongkut's Institute of Technology North Bangkok, Thailand

This paper presents the use of a co-operative co-evolutionary genetic algorithm (CCGA) for solving optimal control problems in a hysteresis system. The hysteresis system is a hybrid control system which can be described by a continuous multivalued state-space representation that can switch between two possible discrete modes. The problems investigated cover the optimal control of the hysteresis system with fixed and free final state/time requirements. With the use of the Pontryagin maximum principle, the optimal control problems can be formulated as optimisation problems. In this case, the decision variables consist of the value of control signal when a switch between discrete modes occurs while the objective value is calculated from an energy cost function. The simulation results indicate that the use of the CCGA is proven to be highly efficient in terms of the minimal energy cost obtained in comparison to the results given by the searches using a standard genetic algorithm and a dynamic programming technique. This helps to confirm that the CCGA can handle complex optimal control problems by exploiting a co-evolutionary effect in an efficient manner.

Keywords: Co-operative Co-evolutionary Genetic Algorithm, Hybrid Control System, Hysteresis System, Optimal Control

1. Introduction

Traditionally, an automatic control system can be classified as either a continuous system or a discrete system. However, there is also an attempt in building systems which incorporate continuous and discrete characteristics. Such system is generally referred to as a hybrid system[1, 2]. In a hybrid system, it is essential that there is an interaction between continuous-valued and discrete-valued state variables.

* E-mail: kittipong_toy@hotmail.com
† E-mail: nchl@kmitnb.ac.th
‡ E-mail: suwat@kmitnb.ac.th

The continuous part of the system can be described by a differential equation while the discrete part is explained via the use of an automaton.

Various control systems can be described as hybrid systems. This includes a control system known as a hysteresis system. The hysteresis system can be described by a continuous multivalued state-space representation that can switch between two possible discrete modes. As the name stated, the characteristic governing the switch from one discrete mode to the other can be described as a hysteresis loop. One particular aspect of research works which involve the finding of appropriate methods for controlling a hysteresis system concerns the use of an optimal control strategy[3]. Riedinger and Iung[3] have derived a mathematical expression that can be used to describe the profiles of continuous-valued state and control signal during the time period between two consecutive switches of the state-space representation of the system. With the use of such mathematical expression, the optimal control problem of a hysteresis system has been reduced to a single objective optimisation problem. In this case, the decision variables are made up from the control signal value at the beginning of each time period described above and the objective is the energy used during the execution of the control scheme. The optimal control problem investigated by Riedinger and Iung[3] covers the projection of the control signal over an infinite time period where the objective of the problem decays exponentially over time. This helps to eliminate the interaction between each decision variable. Thus the problem can be easily solved using a dynamic programming technique.

Although a number of impressive results have been reported in Riedinger and Iung[3], the optimisation problem would be more complicated if the projection of control signal over a fixed and finite period of time is considered instead. The change in the objective function would lead to an inevitably increase in the interaction between the decision variables. Hence, the use of a dynamic programming method would be inappropriate in this latter case. In this paper, a particular evolutionary computation technique that will be utilised in an attempt to find an optimal solution to the above problem is the co-operative co-evolutionary genetic algorithm (CCGA)[4]. The reason that the CCGA is chosen for this particular problem is because the CCGA has a unique characteristic in being able to deal with a problem by assembling a solution to the overall problem from a number of sub-components of the complete solution. For this particular problem, the complete time profiles of the state variable and control signal are made up from a set of short time profiles. Each short time profile covers the time period that begins at the point where the state-space representation is switched from one mode to the other and ends at the point where the representation is switched back. Since the state variable and control signal during each short time profile is dictated by a single value of control signal at the beginning of the profile, this value of control signal can be treated as the decision variable of the solution to the corresponding sub-problem. With this arrangement, the solution to each sub-problem can be viewed as a sub-component that can be assembled into a complete solution.

The organisation of this paper is as follows. In section 2, the explanation on the hysteresis system and the associated optimal control problems will be given. This also includes the description about the difference between the optimal control problems considered over infinite and finite time periods. In section 3, the background on the CCGA will be discussed. The application of the CCGA on the optimal control problems in the hysteresis system will be explained in section 4. The simulation results obtained after applying the CCGA to the problems and the related discussions are given in section 5. The results from using a standard genetic algorithm and a dynamic programming technique to solve the optimal control problems are also included in this section in order to compare with the CCGA search results. Finally, the conclusions are drawn in section 6.

2. Optimal Control of a Hysteresis System

Firstly, consider a hysteresis system defined by

$$\dot{x} = f(x) + u \tag{1}$$

where x is the state variable, $\dot{x}$ is the first time derivative of the state variable, u is the command input and $f(x)$ is a multivalued function shown in Fig. 1. From Fig. 1, the hysteresis system displays a discontinuity in the vector field at the interior points $-\Delta$ and Δ. In the first discrete mode, when the continuous trajectory x meets the boundary Δ from the left hand side, a switch from $f(x) = 1$ to $f(x) = -1$ takes place. On the other hand, in the second discrete mode when the continuous trajectory x meets the boundary $-\Delta$ from the right hand side, a switch from $f(x) = -1$ to $f(x) = 1$ occurs. The mode switching characteristic of the system representation is autonomous and can be described as

$$\dot{x} = f_i(x, u) = i + u \tag{2}$$

where i can takes the value 1 or –1. The resulting automata of the hysteresis system is schematically displayed in Fig. 2.

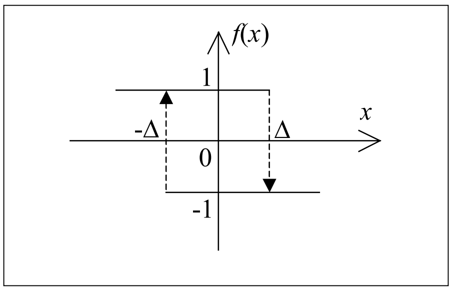

Fig. 1. Multivalued function describing the behaviour of a hysteresis system.

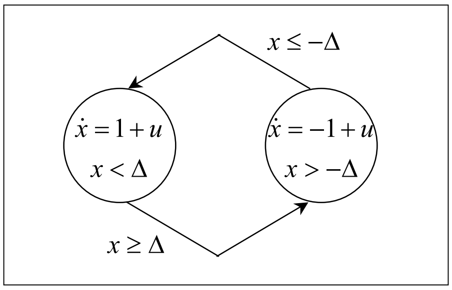

Fig. 2. Automata of the hysteresis system.

The optimal control of the hysteresis system can be achieved by identifying the control command input u which can minimise an energy cost function given by

$$J = \frac{1}{2}\int_0^{t_f} (qx^2 + u^2)e^{-t}\,dt \tag{3}$$

where J is the energy cost, q is the weighting factor and t_f is the final time. In this paper, two types of t_f are considered: infinite and finite. In the case of free final time or $t_f \to \infty$, the energy cost is calculated subject to (a) the conditions from the system representation in equation (2), (b) $x(0) = x_0$ (initial state) and (c) $u(t) \in R$. In contrast, in the case of fixed final time or when t_f is finite, the energy cost is obtained subject to the same constraints as in the case of free final time and one additional constraint: $x(t_f) = x_f$ (final state). With the use of the Pontryagin maximum principle[5], Riedinger and Iung[3] have derived an expression which can be used to identify the values of control signal during any time period at which the system is in one of the two discrete modes. Let $[0, \tau_1, \tau_2, \ldots, \tau_k, \ldots]$ for $k = 0, 1, 2, \ldots$ be the sequence of time at which a switch from one discrete mode to the other mode occurs. The mathematical expression derived by Riedinger and Iung[3] which describes the state variable and control signal during each time interval $[\tau_k, \tau_{k+1}]$ is given by

$$x(t) = \frac{i}{q} + A_k e^{r_1(t-\tau_k)} + B_k e^{r_2(t-\tau_k)} \tag{4}$$

and

$$u(t) = -i + r_1 A_k e^{r_1(t-\tau_k)} + r_2 B_k e^{r_2(t-\tau_k)} \tag{5}$$

where

$$r_1 = \frac{1-\sqrt{1+4q}}{2}, \tag{6}$$

$$r_2 = \frac{1+\sqrt{1+4q}}{2}, \tag{7}$$

$$A_k = \frac{1}{r_1\sqrt{1+4q}}(ir_2 - qx(\tau_k) - r_1 u(\tau_k)) \tag{8}$$

and

$$B_k = \frac{1}{r_2\sqrt{1+4q}}(-ir_1 + qx(\tau_k) + r_2 u(\tau_k)). \tag{9}$$

Equations (4)-(9) can be interpreted as follows. At the beginning of the trajectory or at $t = 0$ and $\tau_0 = 0$, the initial state of the system is dictated by the initial condition $x(0) = x_0$. For a given value of the control signal $u(0)$, equations (8) and (9) can be used to obtain the constants A_0 and B_0 where i is set to -1 or 1 depends upon the initial state. The values A_0 and B_0 are then used in equations (4) and (5) where the profiles of $x(t)$ and $u(t)$ can be obtained until $x(t)$ reaches the value of $-\Delta$ or Δ depends whether the system is in which discrete mode. At this point the system will switch to the other discrete mode and the value τ_1 is obtained. For a given value of control signal $u(\tau_1)$, equations (8) and (9) are then used again to obtain the constants A_1 and B_1. Notice that i is now changed to the negation of i used during the time period $[0, \tau_1]$. Equations (4) and (5) are subsequently used to generate the profiles of $x(t)$ and $u(t)$ during the time period $[\tau_1, \tau_2]$ where τ_2 is the next mode switching time. This process will continue repeatedly until the final time is reached. It can be seen that the variables which dictate the optimality of the control performance are $u(\tau_k)$ for $k = 0, 1, 2, \ldots$. In other words, from an optimisation viewpoint the set of $u(\tau_k)$ will be the decision variable set while the energy cost function in equation (3) will be the corresponding objective function. Riedinger and Iung[3] have divided the original energy cost function into a set of cost functions that can be described by

$$J = \sum_{k \geq 0} \int_{\tau_k}^{\tau_{k+1}} (qx^2 + u^2) e^{-t} dt \,. \tag{10}$$

With this arrangement, the optimal values of $u(\tau_k)$ for $k = 0, 1, 2, \ldots$ are then identified sequentially via the use of a dynamic programming technique. Note that the optimal value of each $u(\tau_k)$ will lead to the optimal use of energy during the time period $[\tau_k, \tau_{k+1}]$.

In addition to the use in the optimal control problem with a free final state/time requirement, the expression derived by Riedinger and Iung[3] can also be applied to the optimal control problem with a fixed final state/time requirement. Recall that in this latter case at $t = t_f$, $x(t_f)$ is required to be at the state x_f. This makes the values of $u(\tau_k)$ for all k become interdependent with one another. In this paper, the use of a co-operative co-evolutionary genetic algorithm (CCGA) to identify the optimal value of $u(\tau_k)$ is proposed.

3. Co-operative Co-evolutionary Genetic Algorithm

The co-operative co-evolutionary genetic algorithm (CCGA) was first introduced by Potter and De Jong[4]. The CCGA functions by introducing an explicit notion of modularity to the optimisation process. This is done in order to provide reasonable opportunity for complex solutions to evolve in the form of interacting co-adapted sub-components. In brief, the CCGA explores the search space by utilising a population which contains a number of species or sub-populations. In contrast to other types of genetic algorithm, each species in the CCGA represents a variable or a part of the problem which is needed to be optimised. A combination of an individual from each

```
gen = 0;
for (s = 0; s < number of species; ++s) {
    pop[s][gen] = randomly initialised population;
    evaluate fitness of each individual in pop[s][gen];
}
termination condition = FALSE;
while (termination condition == FALSE) {
    gen = gen + 1;
    for (s = 0; s < number of species; ++s) {
        select pop[s][gen] from pop[s][gen-1] based on fitness;
        apply genetic operators to pop[s][gen];
        evaluate fitness of each individual in pop[s][gen];
    }
    if (gen == maximum number of generations ||
        the solution has converged) {
       termination condition = TRUE;
    }
}
```

Fig. 3. Pseudo code of the CCGA.

species will lead to a complete solution to the problem where the fitness value of the complete solution can then be identified. This value of fitness will be assigned to the individual of interest that participates in the formation of the solution. After the fitness values have been assigned to all individuals, the evolution of each species is then commenced using a standard genetic algorithm. A pseudo code describing the mechanism of the CCGA is displayed in Fig. 3. In Fig. 3, the CCGA begins by initializing the population of individuals for each decision variable. The initial fitness of each species member is calculated by combining it with a random individual from each of the other species and applying the resulting set of decision variables to the fitness function. After this initialisation stage, each sub-population in the CCGA is evolved using a traditional genetic algorithm. The fitness of a species member is then obtained by combining it with the current best individuals or the randomly selected individuals from the remaining species depending upon whether which combination yields a higher fitness value. Potter and De Jong[4] have explained that this fitness assignment strategy helps to increase a chance for each individual to achieve a fitness value which is appropriate to its contribution to the solution produced especially when there are high interdependencies between decision variables. The CCGA has been successfully used in a number of problem domains including multivariable function optimisation[4], sequential rule evolution[6, 7], design of a cascade neural network[8] and container loading optimisation[9]. A comprehensive description of the CCGA and the summary of its applications can be found in Potter and De Jong[10].

4. Application of the CCGA on the Optimal Control Problems

Two optimal control problems are considered in this paper: optimal control with free and fixed final state/time requirements. Details on the problem formulation and genetic operators used are discussed as follows.

4.1. *Decision Variables*

Referring to section 2, the decision variables of the optimal control problems in this case are made up from $u(\tau_k)$ – the control signal at the time τ_k which is responsible for the trajectories of the state variable and control signal during the time interval $[\tau_k, \tau_{k+1}]$. In order to achieve optimal control, it is crucial that the state variable must remain within the hysteresis loop. This means that the range of possible value of $u(\tau_k)$ is governed by the bound $-\Delta \leq x(t) \leq \Delta$ where Δ defines the hysteresis boundary. Note that at $t = \tau_k$, $x(\tau_k)$ is equal to $-\Delta$ and Δ for the system being at the discrete mode $i = 1$ and $i = -1$, respectively. From equations (5)-(9), different value of $u(\tau_k)$ will lead to a different value of τ_{k+1} at which $x(\tau_{k+1})$ reaches the value of $-\Delta$ or Δ. By varying the value of $u(\tau_k)$, it is found that while the system is at the discrete mode $i = 1$, there is a minimum possible value of $u(\tau_k)$ which make the state $x(t)$ remains within the hysteresis boundary; this provides the lower limit for the search range. On the other hand, when the system is at the discrete mode $i = -1$, the maximum possible value of $u(\tau_k)$ gives the upper limit for the search range.

Two distinct optimal control problems are investigated here: optimal control with free and fixed final state/time requirements. In order to cover the projection of energy cost function over an infinite time period, a relatively large number of decision variables are required in the case of optimal control with a free final state/time requirement. However, the number of decision variables is also dictated by the exponential decay characteristic of the cost function. In contrast, the number of decision variables is dictated by the value of the final time (t_f) in the case of optimal control with a fixed final state/time requirement. In this paper, the value of t_f is chosen such that the effect of the exponential decay component would not have yet achieved a full effect on the cost function. Hence, the number of decision variables in this case will be less than that in the case of optimal control with a free final state/time requirement. Furthermore, given a specific requirement that once the trajectory time reaches the value of t_f the state of the system must equal to a pre-defined final state value (x_f), the value of $u(\tau_k)$ which can cause this situation can be determined directly. Using equations (4) and (6)-(9) with a setting of $x(t_f) = x_f$, three values of unknown variables – A_k, B_k and $u(\tau_k)$ – can be determined numerically from the simultaneous equation set for a given value of τ_k provided that $x(\tau_k)$ is set to either -Δ or Δ and $\tau_k < t_f$. This means that the decision variables in the optimisation does not need to include the value of $u(\tau_k)$ which brings the system to the required final state at the specific time. Also recall from the description of equations (4)-(9) given in section 2 that each value of τ_k is dictated by the control signal $u(\tau_{k-1})$. This implies that if the total number of decision variables available is large enough, different combinations of some decision variables extracted from the whole variable set and the directly determined $u(\tau_k)$ can lead to different objective values. For example, if the total number of decision variables is set to 15, using only the first 13 decision variables ($u(0)$, $u(\tau_1)$, …, $u(\tau_{12})$) and the value of $u(\tau_{13})$ which is directly determined in the energy cost calculation can result in a different cost from that obtained using all decision variables and $u(\tau_{15})$. Note also that these two candidate sets of decision variables in the given example will make the system reaches the final state while the system is in the same discrete mode. Hence, the objective value of each solution in the optimal control problem with a fixed final state/time requirement is the minimum cost obtained after considering all possible subsets of decision variables extracted by shortening the number of variables by two at a time.

Within each optimal control problem, five case studies involving different values of weighting factor (q) in the energy cost function are investigated: q = 0.5, 50, 400, 500 and 2,000. In the first two case studies (q = 0.5 and q = 50), the energy cost function emphasises more on optimising the energy cost obtained from the control signal part. Riedinger and Iung[3] have illustrated that the system representation will periodically switch between the two discrete modes while maintaining the optimal usage of energy. In the third case study (q = 400), the energy cost function indicates an equal concentration on optimising both the energy cost from the control signal part and that from the state variable part. The system representation still exhibits a periodical switching characteristic in this case. However, the profiles of the control signal and state variable trajectories are different from those in the first two case studies due to the change in the weighting factor. Finally, in the last two case studies (q = 500 and q = 2,000), the energy cost function emphasises more on optimising the energy cost obtained from the state variable part. In contrast to the first three case studies, the system representation will either switch between the two discrete modes for a few times before settling in one discrete mode or exhibit no switching characteristics at all depending upon the restriction on the final values of the state and time. The profiles of optimal state and control signal described by Riedinger and Iung[3] for the case of optimal control with a free final state/time requirement are shown in Figs. 4 and 5, respectively. The number of decision variables and the range of $u(\tau_k)$ for each case study are summarised in Table 1. Each decision variable will be coded into an individual in a sub-population of the CCGA. Note that in all case studies the hysteresis boundary Δ is set to 0.1 and the initial state x_0 is set to -0.2 (discrete mode i = 1). In addition, the final state (x_f) is set to 0 where the system must

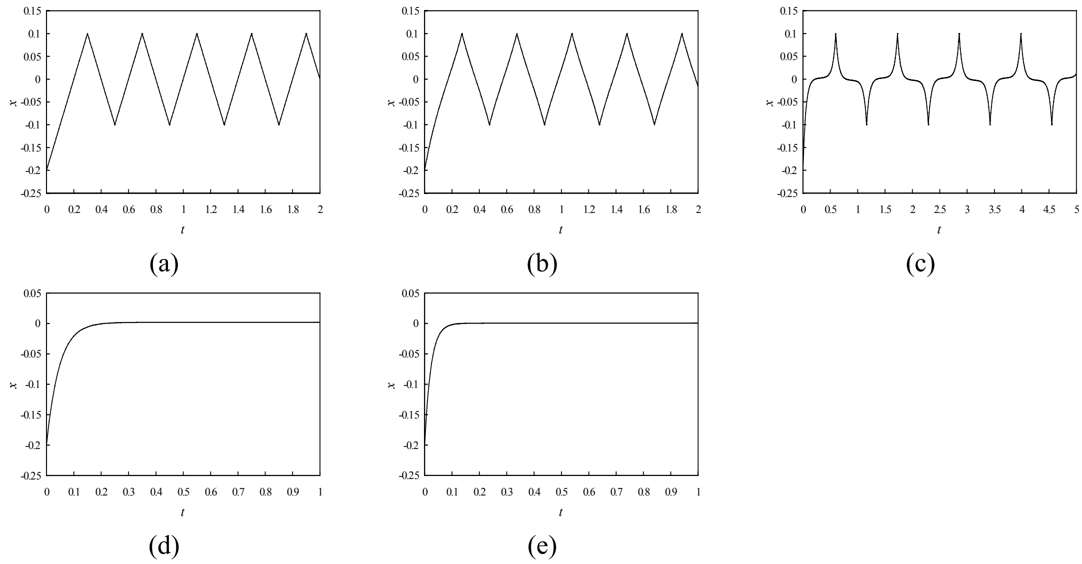

Fig. 4. Profiles of optimal state $x(t)$ where $x(0)$ = -0.2 and Δ = 0.1
(a) q = 0.5 (b) q = 50 (c) q = 400 (d) q = 500 (e) q = 2,000.

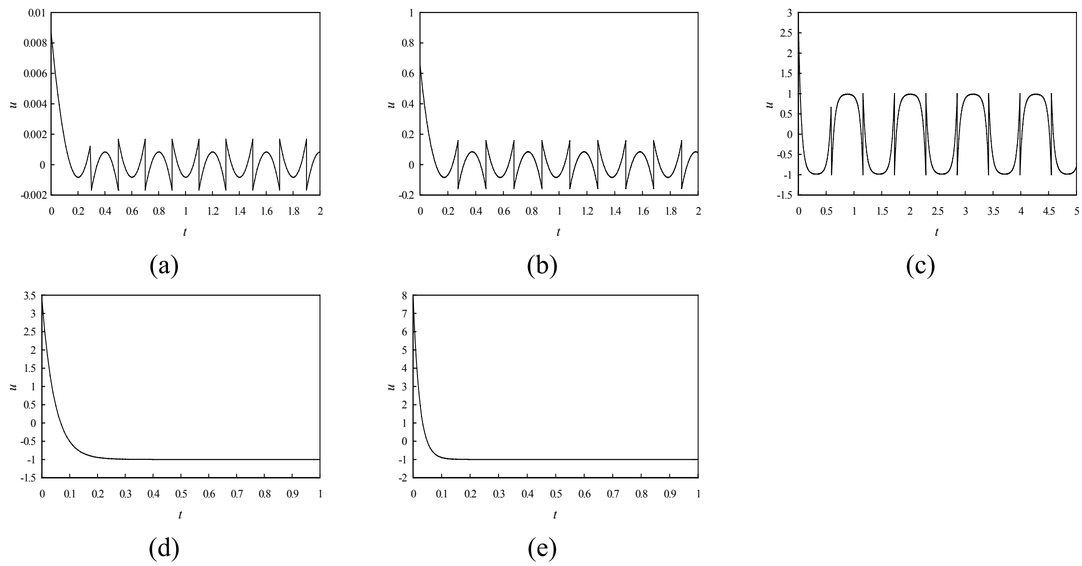

Fig. 5. Profiles of optimal control signal $u(t)$ where $x(0)$ = -0.2 and Δ = 0.1
(a) q = 0.5 (b) q = 50 (c) q = 400 (d) q = 500 (e) q = 2,000.

be in discrete mode i = -1 at the final time in the case of optimal control problem with a fixed final state/time requirement. Furthermore, five values of final time (t_f) are explored in the problem: t_f = 0.2, 0.65, 1.59, 2.5 and 12.5.

4.2. *Objective Function*

The objective of the search is to find the set of control signals $u(\tau_k)$ at the mode switching time τ_k that gives the minimum energy cost. The energy cost function displayed in equation (10) will be used as the objective function.

4.3. *Chromosome Coding*

The chromosome of each individual represents the control signal $u(\tau_k)$ at the mode switching time τ_k. In this investigation, a binary coding scheme is utilised. Each decision variable is coded using a binary code of length 26. Since each decision variable is represented by an individual from a sub-population, there will be 300 and 15 sub-populations in the cases of optimal control with free and fixed final state/time requirements, respectively.

4.4. *Fitness Scaling and Selection Methods*

A linear fitness scaling technique[11] is used in the CCGA in order to improve the search performance. After the scaling procedure, a stochastic universal sampling[12] is used in the fitness selection.

4.5. *Crossover and Mutation Methods*

The standard uniform crossover is used in the recombination. In addition, a standard bit-flip operation is used for the mutation.

Table 1. Number and range of $u(\tau_k)$ for the optimal control problems.

Case	q	Number of Variables		Range of $u(\tau_k)$
		Free x_f/t_f	Fixed x_f/t_f	
1	0.5	300	15	$-0.50 \le u(\tau_k) \le 0.50$, i =1 $-0.50 \le u(\tau_k) \le 0.50$, i =-1
2	50	300	15	$-0.21 \le u(\tau_k) \le 4.00$, i =1 $-4.00 \le u(\tau_k) \le 0.21$, i =-1
3	400	300	15	$1.00 \le u(\tau_k) \le 5.00$, i =1 $-5.00 \le u(\tau_k) \le -1.00$, i =-1
4	500	300	15	$1.23 \le u(\tau_k) \le 6.00$, i =1 $-6.00 \le u(\tau_k) \le -1.23$, i =-1
5	2,000	300	15	$3.44 \le u(\tau_k) \le 9.00$, i =1 $-9.00 \le u(\tau_k) \le -3.44$, i =-1

4.6. *Diversity Control*

The parent sub-population and the offspring sub-population of the same species are merged together and the appropriate individuals of the new generation are then extracted from this merged sub-population. Note that the number of individuals extracted is equal to the size of the sub-population. The strategy used in the extraction process is based on the mechanisms of a diversity control oriented genetic algorithm or DCGA[13, 14]. In brief, the process starts with the elimination of duplicate individuals in the merged sub-population. The remaining individuals are then sorted according to their fitness values in descending order. Following that the best individual from the remaining individuals is determined and kept for passing on to the next generation. Then a cross-generational probabilistic survival selection (CPSS) method is applied to the remaining non-elite individuals where a probability value is assigned to each individual according to its similarity to the best individual in that particular sub-population. The survival probability of an individual is given by

$$p_s = \{(1-c)h/L + c\}^{\alpha} \tag{11}$$

where p_s denotes the survival probability, h is the Hamming distance between the interested individual and the best individual, L is the chromosome length, c is the shape coefficient and α is the exponent coefficient. With this form of survival probability assignment, an individual which its genomic structure is quite different from that of the best individual will be given a high chance of survival. Each individual will then be selected according to the assigned survival probability. If the total number of all selected individuals including the pre-selected elite individual does not reach the required sub-population size after the survival selection loop, randomly

Table 2. Parameter settings for the CCGA.

Parameter	Value	
	Free x_f/t_f	Fixed x_f/t_f
Number of sub-populations	300	15
Sub-population size	50	50
Chromosome length (1 sub-population)	26	26
Crossover probability	0.75	0.75
Mutation probability	0.1	0.1
Diversity control		
Shape coefficient, c	0.235	0.235
Exponent coefficient, α	0.51	0.51
Number of generations	250	250

Table 3. Parameter settings for the simple genetic algorithm.

Parameter	Value
Population size	50
Chromosome length	390
Number of elite solutions	2
Crossover probability	0.75
Mutation probability	0.1
Number of generations	7,500

generated individuals will be added to the individual array until the required number is met. Note that the process regarding the diversity control scheme will be repeated for all sub-populations.

The parameter settings for the CCGA in the cases of optimal control with free and fixed final state/time requirements are summarised in Table 2. Note that since there are no interdependencies between decision variables in the case of optimal control with a free final state/time requirement[3], the fitness of a species member is obtained after only combining it with the current best individuals from the remaining species. In contrast, since there are interdependencies between decision variables in the case of optimal control with a fixed final state/time requirement, the fitness assignment strategy[4] explained earlier in section 3 will be used.

For the purpose of comparison, a standard genetic algorithm and a dynamic programming technique are also used to solve the optimal control problems. In the standard genetic algorithm search, the strategies involving the fitness scaling, selection, crossover and mutation operations described above are also utilised. However, the standard genetic algorithm will not be used in the optimal control problem with a free final state/time requirement. As mentioned earlier, there are no interdependencies between decision variables in this problem. Potter and De Jong[4] have illustrated that both the CCGA and standard genetic algorithm can easily locate the solution of this kind of problem. The only observable difference in terms of the search performance is that the CCGA will converge to the optimal solution at a faster rate. Hence for this optimal control problem the search using a standard genetic algorithm is omitted. The parameter settings for the standard genetic algorithm are summarised in Table 3. The simulation results from using the CCGA, the standard genetic algorithm and the dynamic programming technique will be presented in the following section.

5. Simulation Results and Discussions

Two optimal control problems are investigated in this paper: problems with free and fixed final state/time requirements. Within each problem, five case studies involving different values of weighting factor (q) in the energy cost function are explored: q = 0.5, 50, 400, 500 and 2,000. In the case of optimal control problem with a fixed final state/time requirement, the required final state (x_f) is equal to 0 and five values of final time (t_f) are used in each case study: t_f = 0.2, 0.65, 1.59, 2.5 and 12.5. In addition, it is also required that the system reaches the desired final state and time while the system is in the discrete mode i = -1. The CCGA and the dynamic programming technique are used to solve the optimal control problem with a free final state/time requirement. In contrast, the CCGA, the standard genetic algorithm and the dynamic programming technique are used in the optimal control problem with a fixed final state/time requirement. Each simulation is repeated 10 times where the best results in terms of energy cost from all case studies mentioned above are summarised in Table 4.

Firstly, consider the results from the optimal control problem with a free final state/time requirement. It can be seen that there are no differences between the solutions identified by the CCGA and those generated by the dynamic programming technique in all case studies. As mentioned earlier, for this optimal control problem there are no interactions between decision variables. Hence, it comes as no surprise that solutions identified by optimising each decision variable one by one as carried out by the dynamic programming technique are the same as those generated by solving for all decision variables at the same time as done by the CCGA. Recall that in this problem, the fitness value of an individual in the CCGA is obtained after only combining it with the current best individuals from other species.

Moving onto the second optimal control problem: a fixed final state/time requirement has been imposed on the problem. This means that there will certainly be interactions between decision variables in this problem. However, by increasing the value of the required final trajectory time, the level of interaction between decision variables will decrease. This is because the effect of the exponential decay term in the

Table 4. Summary of simulation results.

Simulation	Energy Cost		
	Dynamic Program.	SGA	CCGA
Free x_f/t_f			
$q = 0.5$	0.001315	-	0.001315
$q = 50$	0.120506	-	0.120506
$q = 400$	0.698618	-	0.698618
$q = 500$	0.745116	-	0.745116
$q = 2{,}000$	1.188661	-	1.188661
Fixed x_f/t_f			
$q = 0.5$			
$t_f = 0.20$	No solutions	0.090740	0.090740
$t_f = 0.65$	0.056248	0.013101	0.013101
$t_f = 1.59$	0.001252	0.001589	0.001156
$t_f = 2.50$	0.003527	0.001993	0.001698
$t_f = 12.5$	0.001315	0.002044	0.001315
$q = 50$			
$t_f = 0.20$	No solutions	0.129537	0.129537
$t_f = 0.65$	0.135730	0.087213	0.087209
$t_f = 1.59$	0.104172	0.104276	0.104088
$t_f = 2.50$	0.116569	0.126270	0.114480
$t_f = 12.5$	0.120505	0.125180	0.120505
$q = 400$			
$t_f = 0.20$	No solutions	0.395608	0.395608
$t_f = 0.65$	0.483926	0.439164	0.439164
$t_f = 1.59$	0.596964	0.596964	0.596964
$t_f = 2.50$	0.657696	0.657723	0.657696
$t_f = 12.5$	0.698616	0.698659	0.698616
$q = 500$			
$t_f = 0.20$	No solutions	0.444205	0.444205
$t_f = 0.65$	No solutions	0.500868	0.500868
$t_f = 1.59$	0.687331	0.653342	0.653833
$t_f = 2.50$	0.709081	0.709081	0.709081
$t_f = 12.5$	0.750035	0.750035	0.750035
$q = 2{,}000$			
$t_f = 0.20$	No solutions	0.998299	0.998299
$t_f = 0.65$	1.089460	1.089460	1.089460
$t_f = 1.59$	1.248336	1.248336	1.248336
$t_f = 2.50$	1.309286	1.309286	1.309286
$t_f = 12.5$	1.350348	1.350348	1.350348

cost function will be high when the required final trajectory time is large. This is confirmed by the results generated by the dynamic programming technique. It can be observed from all five case studies that the dynamic programming technique has difficulties in locating the solution when the value of the final time is small while the solution can be easily located when the value of final time is high. Although the interaction between decision variables decreases as the final trajectory time increases, an additional factor that only affects the search performance of the genetic algorithms is the number of decision variables. As mentioned earlier in section 4.1, a different number of times that a mode switching occurs can lead to a different energy cost. In addition, by varying the value of the required final time, the number of times that a switch between the two discrete modes takes place will inevitably also be varied. Nonetheless, it is sufficed to say that the required number of times that a switch between two modes occurs will increase when the required final time is increased. This means that while the interaction between decision variables may reduce as the value of the final time is increased, the number of generations required by the genetic algorithms to locate the optimal solution may increase because of the change in the number of decision variables. The results from the case studies with q = 0.5 and 50 indicate that as the value of the final time increases, the performance of the standard genetic algorithm is worsen. This implies that the standard genetic algorithm has not yet converged to the optimal solution within the setting of the maximum number of generations. However, in the case studies with q = 400, 500 and 2,000, the solutions found by the standard genetic algorithm and the CCGA are the same. This helps to identify the effect of the weighting factor (q) on the conflicts between the two effects described above. Note also that the convergence rate of the CCGA seems to be high enough to cope with the same conflicting effects in all five case studies.

6. Conclusions

In this paper, the optimal control of a hysteresis system is considered. Two optimal control problems are investigated: problems with free and fixed final state/time requirements. With the use of the Pontryagin maximum principle, the optimal control problems can be formulated as optimisation problems[3]. In this case, the decision variables are composed of the value of control signal when a switch between discrete modes occurs while the search objective is expressed in terms of energy cost. The cost function is calculated from a weighted-sum between the energy from the control signal part and that from the state variable part. The use of different weighting factors results in different profiles of control signal and state trajectories. Three search techniques have been used to solve the optimal control problems: a co-operative co-evolutionary genetic algorithm (CCGA), a standard genetic algorithm and a dynamic programming technique. The simulation results indicate that the CCGA can locate the solutions in all case studies while the dynamic programming technique fails to identify the solutions in some cases. In addition, the CCGA can also find superior solutions to those located by the standard genetic algorithm and the dynamic programming technique is some cases as well. The key factor that contributes to the search performance of the CCGA is the way in which the algorithm exploits the co-evolutionary effect among the searches carried out by all sub-populations. With such strategy, the CCGA can handle a complex optimisation problem that has a reasonably large number of decision variables quite well.

References

1. M. S. Branicky, V. S. Borkar and S. K. Mitter, A unified framework for hybrid control: Model and optimal control theory, *IEEE Trans. Automatic Control* **43**(1) (1998) 31-45.
2. R. A. Decarlo, M. S. Branicky, S. Pettersson and B. Lennartson, Perspectives and results on the stability and stabilizability of hybrid systems, *Proc. IEEE* **88**(7) (2000) 1069-1082.
3. P. Riedinger and C. Iung, Optimal control for hybrid systems: An hysteresis example, *Proc. 1999 IEEE Int. Conf. Systems, Man, and Cybernetics* (1999) 188-193.
4. M. A. Potter and K. A. De Jong, A cooperative coevolutionary approach to function optimization, *Lect. Notes in Comput. Sci.* **866** (1994) 249-257.
5. L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze and E. F. Mischenko, *The Mathematical Theory of Optimal Processes* (Interscience, New York, 1962).
6. K. A. De Jong and M. A. Potter, Evolving complex structures via cooperative coevolution, *Proc. Fourth Annual Conf. Evolutionary Programming* (1995) 307-318.
7. M. A. Potter, K. A. De Jong and J. J. Grefenstette, A coevolutionary approach to learning sequential decision rules, *Proc. Sixth Int. Conf. Genetic Algorithms* (1995) 366-372.
8. M. A. Potter and K. A. De Jong, Evolving neural networks with collaborative species, *Proc. 1995 Summer Computer Simulation Conf.* (1995) 340-345.
9. C. Pimpawat and N. Chaiyaratana, Using a co-operative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem, *Proc. 2001 Congr. on Evolutionary Computation* (2001) 1197-1204.
10. M. A. Potter and K. A. De Jong, Cooperative coevolution: An architecture for evolving coadapted subcomponents, *Evolutionary Computation* **8**(1) (2000) 1-29.
11. D. E. Goldberg, *Genetic Algorithms: In Search, Optimization and Machine Learning* (Addison-Wesley, Reading, 1989).
12. J. E. Baker, *An Analysis of the Effects of Selection in Genetic Algorithms* (Ph.D. Thesis, Vanderbilt University, Tennessee, 1989).
13. H. Shimodaira, DCGA: A diversity control oriented genetic algorithm, *Proc. Second Int. Conf. Genetic Algorithms in Engineering Systems: Innovations and Applications* (1997) 444-449.
14. H. Shimodaira, A diversity-control-oriented genetic algorithm (DCGA): Performance in function optimization, *Proc. 2001 Congr. on Evolutionary Computation* (2001) 44-51.