Differential evolution-based receding horizon control design for multi-UAVs formation reconfiguration

Zhang Xiang-Yin and Duan Hai-Bin

School of Automation Science and Electrical Engineering, Beijing University of Aeronautic and Astronautic, Beijing, 100191, China

Formation reconfiguration is one of the most complicated problems on multiple uninhabited aerial vehicles (multi-UAVs) co-ordinated control. Based on the differential evolution (DE) algorithm, a novel type of control method using receding horizon control (RHC) is proposed aiming at driving multi-UAVs to form a new flying formation. The global control problem of multi-UAVs formation reconfiguration is transformed into several online local optimization problems within a series of receding horizons, while the DE algorithm is adopted to optimize the control sequences for each UAV. The working process of the RHC controller is presented in detail. Finally, two simulation experiments are performed, and simulation results show the feasibility and effectiveness of our proposed control approach.

Key words: differential evolution (DE); formation reconfiguration; multiple uninhabited aerial vehicles (multi-UAVs); receding horizon control (RHC).

1. Introduction

Recent world events have highlighted the utility of uninhabited aerial vehicles (UAVs) for both military and potential civilian applications (Vachtsevanos *et al.*, 2005). Multi-UAVs formation can distribute the equipment, necessary for a specific mission, to all vehicles in the swarm and offer a huge increase of performance and robustness compared to a single operating vehicle (Paul *et al.*, 2008). UAVs are often required to change its relative formation from one to another if any change takes places. The need

© 2010 The Institute of Measurement and Control   10.1177/0142331210366643

for formation reconfiguration is well recognized when dealing with changes in the dynamic environment or flight mission, and it has become one of the most important steps in field operation of UAV systems. Reconfiguration of UAV formation can bring low cost and high efficiency (Shannon *et al.*, 2003).

The formation reconfiguration problem for multi-UAVs, as shown in Figure 1 (Raffard *et al.*, 2004), can be described as follows: given a group of UAVs with an initial configuration, a final configuration, and a set of inter- and intra-UAV constraints, the goal is to determine a nominal control input for each vehicle such that the multi-UAVs group can start from the initial configuration and reach its final configuration, while satisfying the set of constraints. The formation reconfiguration problem can be recognized as an optimal control problem with dynamical constraints (Zelinski *et al.*, 2003). Several theoretical techniques such as graph theory (Hendrickx *et al.*, 2008), reconfiguration maps, Dijkstra algorithm (Giuletti *et al.*, 2000; Ueno and Kwon, 2007), or functional optimization (Chichka *et al.*, 1999) have been developed to define the new/optimal positions to be occupied by the UAVs in the formation.

As a large-scale centralized control problem, formation reconfiguration aims to obtain the control input actions (such as steering angle, throttle/thrust, etc.) for each UAV, through complex calculation, to drive each UAV in a complicated flight manoeuvre. In this process, multi-UAVs must satisfy several constraints, eg, the distance between two UAVs must be greater than the safety collision distance, and also should not be too much greater than the communication distance.

The control parameterization and time discretization (Furuknwa *et al.*, 2003) (CPTD) method, which treats the time-optimality as one of the main objectives in the control of dynamical systems, is widely used in solving the reconfiguration problem where a vehicle or system is required to move from an initial state to a specified terminal state under some fixed terminal constraint. Based on the CPTD method, Furukawa (2002) discussed the time-optimal control of formation reconfiguration with fixed terminal state constraints in detail; Furukawa *et al.* (2003), Simeon *et al.* (2002) and
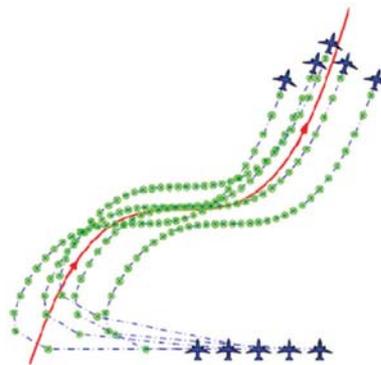

Figure 1 Formation reconfiguration of five UAVs

Saber *et al*. (2003) discussed the free terminal state of the time-optimal control. However, these studies do not consider the distance constraints among UAVs (Duan *et al*., 2008). Xiong *et al*. (2008) and Duan *et al*. (2008) used bio-inspired computational methods, such as genetic algorithm and particle swarm optimization, to compute control inputs for multi-UAVs, and took into account the communication distance and safety collision distance between each two UAVs. Considering the complicated combating environment with moving threats, Bai *et al*. (2009) proposed the hybrid diversity-PSO and CPTD approach to solve the reconfiguration problem with the free terminal state. However, the calculation of the CPTD method is much complex. For example, given a flight formation with five UAVs, 51 parameters need to be determined in one calculation process (Bai *et al*., 2009; Duan *et al*., 2008; Xiong *et al*., 2008). Even though the bio-inspired optimization can remarkably decrease the computing complexity, it is still difficult to fit the rapidly changing air battle.

Receding horizon control (RHC), or model predictive control (MPC), is an optimization-based control method originating in the process industry in early 1970s. With the development of theories about MPC in the 1990s and the development of faster algorithms and computers, application areas of RHC have been extended during the following decades. Recently, RHC has been utilized to achieve formation flight and other co-operative tasks (Keviczky *et al*., 2006), and the RHC-based method proved more successful in online optimizing the airport capacity profile in a dynamic environment (Hu *et al*., 2007). RHC is based on the simple idea of repetitive solution of an optimal control problem and state updating after the first input of the optimal command sequence (Bhattacharya *et al*., 2001). The main idea of RHC is the online receding/moving optimization, which breaks the global control problem into several local optimization problems of smaller sizes, and thus can decrease the computing complexity and computational expense significantly (Mayne *et al*., 2000). Compared with the CPTD method, the RHC method needs much fewer parameters to be identified in each pre-computing process. For the complicated non-linear control problems like multi-UAVs formation reconfiguration, the RHC-based method requires a practical optimization technique to yield the optimal control sequences. In this work, the differential evolution (DE) algorithm is adopted to optimize the control actions in the RHC process. DE is one of the recent population-based techniques; it was first proposed by Price and Storn in 1995 (Storn and Price 1995) as a heuristic method for minimizing non-linear and non-differentiable continuous space functions. This optimization approach has successfully found its application in many mechanical fields, and proved its superiority to other optimization techniques.

This paper is organized as follows. In Section 2, the description of multi-UAVs formation reconfiguration problem basing on RHC is presented, including the dynamical model of multi-UAVs formation, the basic idea of RHC and the mathematic description of the formation reconfiguration. Subsequently, the principle of DE algorithm is presented in Section 3. Section 4 describes the DE-based RHC controller design for the problem of formation reconfiguration, and the working process is also

presented in detail. Then, simulation results are given to verify the feasibility and effectiveness of our proposed method in Section 5. Finally, the concluding remarks and future work are contained in Section 6.

2.   Description for multi-UAVs formation reconfiguration based on RHC

2.1.   Model of multi-UAVs

In this paper, we consider a group of N UAVs are flying at the same altitude without sideslip, and they turn through the co-ordinated-turn. For each UAV in the flight formation, its state variable is set as $x_i = (v_i, \psi_i, x_i, y_i)^T$, $i = 1, \ldots, N$, and the dynamics of the single UAV can be written as:

$$
\begin{aligned}
\dot{v} &= (T - D)W \\
\dot{\psi} &= g \cdot \sin\phi v \\
\dot{x} &= v \cdot \cos\psi \\
\dot{y} &= v \cdot \sin\psi
\end{aligned}
\tag{1}
$$

where v is the horizontal flying velocity of each UAV in the flight formation, w denotes the heading angle and the UAV's horizontal location is represented by (x, y) in the earth-surface inertial reference frame. The control inputs of each UAV's autopilot are represented by $u = (T, \phi)^T$, which contain the thrust T and the roll angle ϕ. D represents the aerodynamic drag, which is simply regarded as a constant in this paper. The weight of each UAV is W, and gravity acceleration $g = 98\,m/s^2$. Dynamics of the ith UAV can be described as:

$$
\dot{x}_i(t) = f(x_i(t), u_i(t)), i = 1, \ldots, N
\tag{2}
$$

Set one UAV in the formation as the leader, which is treated as the reference point, and its state vector is represented by x_L. Relative to the leader UAV, $(x_i - x_L)$ denotes the relative state of the ith UAV. Assume the initial time of the reconfiguration process is $t_0 = 0$, and the terminal time is $t = T$. In this paper, the process of the formation reconfiguration is regarded as a control optimization problem, and so, the goal is to find the continuous control action $U = (u_1, \ldots, u_N)$ that minimize a cost function, which enables the terminal positions of every UAV to reach their desired value. Set the cost function as the quadratic form, which is described in the following equation:

$$
\min J(U) = \sum_{i=1}^{N} (x_{ref,i} - (x_i(T\,|\,u_i) - x_L(T\,|\,u_L)))^T Q(x_{ref,i} - (x_i(T\,|\,u_i) - x_L(T\,|\,u_L)))
$$

$$
st \quad x_i(t\,|\,u_i) = x_i(0) + \int_0^t f(x_i(\tau), u_i(\tau))d\tau
\tag{3}
$$

$$
U_{\min} \leqslant U \leqslant U_{\max}
$$

where $\boldsymbol{X}_{ref} = [\boldsymbol{x}_{ref,1}, \ldots, \boldsymbol{x}_{ref,N}]$ represents the terminal reference state of the multi-UAV system, which defines the terminal configuration of the multi-UAVs. $\boldsymbol{x}_i(T|\boldsymbol{u}_i)$ denotes the ith UAV's terminal state driven by the control input $\boldsymbol{u}_i$. For each UAV, given the initial state $\boldsymbol{x}_i(0)$, its state $\boldsymbol{x}_i(t)$ at any time t can be uniquely determined by $\boldsymbol{u}_i$. The control inputs are constrained by the performance of UAV. $\boldsymbol{Q} = diag\{q_1, q_2, q_3, q_4\}$ is a positive definite matrix.

Denote the distance between any two UAV as $d_{i,j}(t)$, $i, j = 1, \ldots, N$, which is computed by:

$$d_{i,j}(t) = \sqrt{(x_i(t) - x_j(t))^2 - (y_i(t) - y_j(t))^2}$$

In order to avoid collision between two UAVs, $d_{i,j}(t)$ must be greater than the safe anti-collision distance D_{safe}, ie,

$$d_{i,j}(t) \geqslant D_{safe}, \forall t \in [0, T], \quad \forall_{i \neq j} i, j \in \{1, \ldots, N\} \tag{4}$$

In order to ensure the real-time communication to achieve the information sharing, $d_{i,j}(t)$ must be less than the communication distance D_{safe}, ie,

$$d_{i,j}(t) \leqslant D_{com}, \forall t \in [0, T], \quad \forall_{i \neq j} i, j \in \{1, \ldots, N\} \tag{5}$$

Comprehensively considering the distance restrictive conditions as shown in Equations (4) and (5), the extended cost function can be rewritten as

$$\min J_{extend}(\boldsymbol{U}) = J(\boldsymbol{U}) + \omega \cdot \int_0^T \sum_{i \neq j} [\max(0, D_{safe} - d_{i,j}(t)) + \max(0, d_{i,j}(t) - D_{com})] dt$$

$$\tag{6}$$

where ω denotes the distance punishment constant coefficient, and it should be great enough so that the distance restricts of multi-UAVs formation can be satisfied. In this paper, $\omega = 10^{10}$.

2.2 Principle of RHC

Receding optimization is the most important idea of RHC, which is also the typical difference between RHC and optimum control, as shown in Figure 2. The whole control process is divided into a series of optimizing intervals called the receding horizon. In RHC, the current control action is obtained by solving a finite horizon optimal control problem at each sampling instant, using the current state of the system as the initial state; the optimization yields an optimal control sequence and only the top several
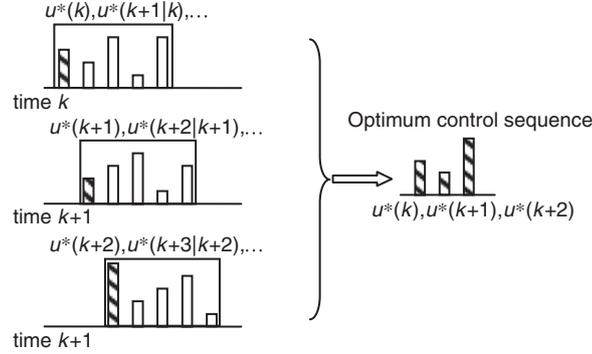
control actions in this sequence are implemented to the system. For each prediction horizon, the local optimization process has the same cost function as the global control optimization problem.

RHC method forms the closed-loop rolling mechanism, including observation, planning, implementation and re-observation. In fact, RHC is a *p*-step-ahead online optimization strategy (Hu *et al.*, 2007). Within every time interval, using the current information, RHC optimizes the specific problem for the following *p* intervals.

2.3   Description of the RHC-based formation reconfiguration

RHC divides the global control problem into some local optimization problems in receding time horizons. These local optimization problems have the same optimization objective as the global control problem. In the *k*th sampling instant, the dynamic of the *i*th UAV of multi-UAVs formation can be written as:

$$x_i(k + 1) = x_i(k) + \int_{kT}^{(k+1)T} f(x_i(k), u_i(k))dt \tag{7}$$

The control inputs are subject to the following constraints:

$$U = \{u_i(k) | | u_{min} \leqslant u_i(k) \leqslant u_{max}\} \tag{8}$$

where $x_i(k) = [v_i(k), \psi_i(k), x_i(k), y_i(k)] \in R^4$ represents the *i*th UAV's state at the *k*th sampling time, and the control input of the *i*th UAV, staying constant until next predictive horizon, is represented by $u_i(k) = [T_i(k), \phi_i(k)] \in R^3$. *T* denotes the span of one time horizon, or sampling interval.

At time *k*, the RHC controller computes predictive control sequences of the current and future *p* predictive time horizons for each UAV according to multi-UAVs' current state and the constraints described in Equation (8), and these control sequences can be
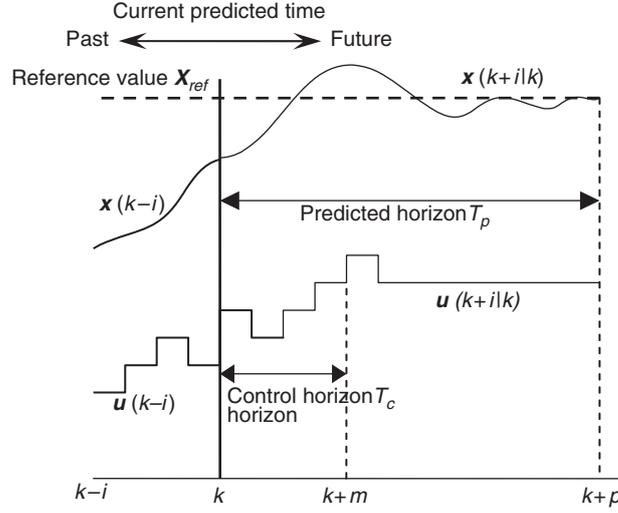
Figure 3 Basic ideas of RHC

represented by $u_i(k\,|\,k)$, $u_i(k+1\,|\,k),\ldots,u_i(k+p-1\,|\,k)$. Then, the predictive states of each UAV in the next p time horizons can be obtained, which are represented by $x_i(k+1\,|\,k)$, $x_i(k+2\,|\,k),\ldots,x_i(k+p\,|\,k)$. The next p time horizons are called the predictive time horizon.

Denote the quadratic cost by the following fitness function at the kth time:

$$\min J(k) = \sum_{j=1}^{p}\sum_{i=1}^{N}\left(x_{ref,i} - \left(x_i(k+j\,|\,k) - x_L(k+j\,|\,k)\right)\right)^T$$

$$\times Q\left(x_{ref,i} - \left(x_i(k+j\,|\,k) - x_L(k+j\,|\,k)\right)\right)$$

$$st \quad x_i(k+1\,|\,k) = x_i(k) + \int_0^T f\left(x_i(k), u_i(k\,|\,k)\right)dt$$

$$x_i(k+j+1\,|\,k) = x_i(k+j\,|\,k) + \int_0^T f\left(x_i(k+j\,|\,k), u_i(k+j\,|\,k)\right)dt$$

$$u_{\min} \leqslant u_i(k+j\,|\,k) \leqslant u_{\max} \tag{9}$$

Minimizing fitness function (9), the optimal control solution to the local optimization problem at time k can be obtained, which is represented by $u_i^*(k+j-1\,|\,k), j=1,\ldots,p$. Apply the preceding m m control actions $u^*(k\,|\,k)$, $u^*(k+1\,|\,k),\ldots,u^*(k+m-1\,|\,k)$, $(0 \leqslant m \leqslant p)$ to each UAV's autopilot successively in current and following $m-1$ time horizons. Subsequently, at time $k+m$, repeat sampling, predicting, optimization and implementation. By using this receding technique, the multi-UAVs formation's state can approximate to the reference value finally. This process can be described as Figure 3.

RHC treats the global control problem as a series of online local optimization problems. However, multi-UAVs formation reconfiguration problem is actually a constrained non-linear problem, and is difficult solve using the traditional approaches. How to compute the control law is the key technique to RHC. Numerous population-based optimization approaches provide good solutions to these complicated problems. In this paper, DE algorithm is utilized to optimize the fitness function, and thus the RHC control law can be worked out directly.

3.   Principle of DE algorithm

DE algorithm mainly has three evolutionary operations, namely mutation, recombination and selection. The positions of individuals are represented as real-coded vectors, which are randomly initialized inside the limits of the given search space in the beginning of an optimization process (Zielinski and Laur, 2008). The individuals are evolved during the optimization process by applying mutation, recombination and selection operators to each individual in every generation. A stopping criterion determines after the building of every new generation if the optimization process should be terminated.

Like other evolutionary algorithms, DE also deals with a population of solutions. Suppose that the initial solution population has NP individuals, and the search space is D-dimensional, the solution vector in continuous space can be represented by $x_i = [x_{i1}, x_{i2}, \ldots, x_{iD}], (i = 1, \ldots, NP)$. Let there be some criterions of optimization, usually named fitness or cost function. Then the optimization goal of DE algorithm is to find the values of the variables that minimize the fitness, ie, to find

$$x^* : f(x^*) = \min_x f(x)$$

In the mutation process, for every target vector x_i, a randomly chosen population member x_{r_1} is added to one vector difference (also built from two randomly chosen members x_{r_2} and x_{r_3} of the current population), and generate new individual called mutated vector v_i as follows:

$$v_i = x_{r_1} + F \times (x_{r_2} - x_{r_3}) \tag{10}$$

where $x_{r_1}, x_{r_2}, x_{r_3}$ and the so-called target vector x_i are mutually different. F is a control parameter that is usually chosen from the interval [0,2]. Best values are usually within the range [0.5, 0.9] (Zielinski *et al.*, 2006). Figure 4 shows the mutation process (Storn and Price, 1997).

Then, the individuals of the population are updated by means of the recombination operation. By coping components from the mutation vector v_i and the target vector x_i in dependence, the trial vector u_i was generated. This process can be written as the
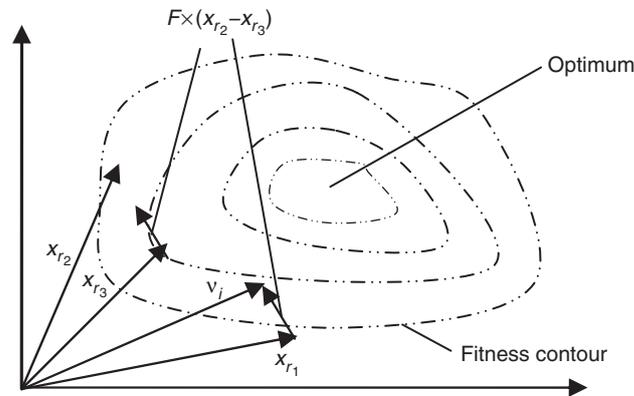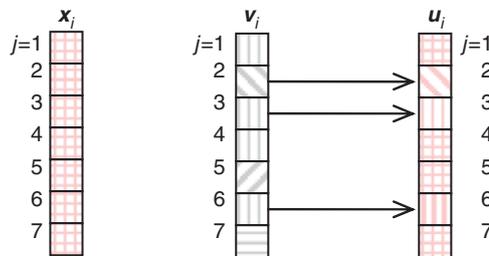
Figure 4 Mutation process of DE

Figure 5 Recombination process of DE

following equation:

$$u_{ji} = \begin{cases} v_{ji}, \text{ if } randb \leqslant CR & or & j = randr, & i = 1, \dots, NP \\ x_{ji}, \text{ if } randb \leqslant CR & or & j \neq randr, & j = 1, \dots, D \end{cases} \quad (11)$$

where the random number $randb \in [0,1]$, the recombination control parameter CR is a constant in the interval $[0,1]$. $randr$ is an integer randomly chosen from $[1,D]$. The recombination process is described in Figure 5 (Storn and Price, 1997).

Next, a selection operator, as a deterministic process in the DE algorithm (Zielinski and Laur, 2008), is implemented to choose the better individuals with lower fitness function value from the target vector and the trial vector, which is inherited by the next generation, expressed as:

$$x_i(t+1) = \begin{cases} u_i, & if & f(u_i) \leqslant f(x_i) \\ x_i, & else \end{cases} \quad (12)$$

This selection scheme allows only improvement but not deterioration of the fitness function value; it is called 'greedy' (Storn and Price, 1997). Selection operator ensures

that the best fitness function value cannot get lost when moving from one generation to the next, which usually results in fast convergence behaviour.

4.   DE-based RHC controller for formation reconfiguration

In this paper, the DE algorithm is utilized to work out the predictive control law directly. The block diagram of the DE-based RHC controller for multi-UAVs formation reconfiguration process is shown in Figure 6.

In the online reconfiguration process, set Equation (9) as the fitness function of DE, and set predicted control sequence $u(k+i-1\,|\,k), i=1,\ldots,p$ as the individual vector, which is just the objective of the DE operators. For the flight formation of N UAVs, the length of the predicted horizon is p and the control input has two actions: thrust and roll angle, so the DE's search region is a $D=2\cdot N\cdot p$-dimensional space. At time k, the ath individual of DE is represented by $x_a=[T_1(k\,|\,k),\phi_1(k\,|\,k),\ldots,T_i(k\,|\,k),\phi_i(k\,|\,k),\ldots,$ $T_N(k\,|\,k),\phi_N(k\,|\,k),\ldots,T_i(k+j-1\,|\,k),\phi_i(k+j-1\,|\,k),\ldots,]$,   where $a=1,\ldots NP$, $i=1,\ldots N$, $j=1,\ldots p$. Apply DE's evolutionary operators to x_a until the terminal criterion is satisfied, and then choose the individual with the lowest fitness function value as the optimal control sequence. After that, implement the preceding $2\cdot N\cdot m$ control actions to corresponding UAV at each time horizon respectively.

In order to decrease the computing complexity, we adopt the one-step predicted control, ie, $m=p=1$.

When multi-UAV is flying in a formation, at the kth time horizon, the flight formation receives the command that a new formation is inevitable, and thus, the DE-based RHC controller implements the process online in the following steps:

Step 1:   Set the parameters of RHC and DE.
Step 2:   Input each UAV's current state $X(k)=[x_1,\ldots,x_N]$ as well as the desired formation configuration and then carry on the optimization process.
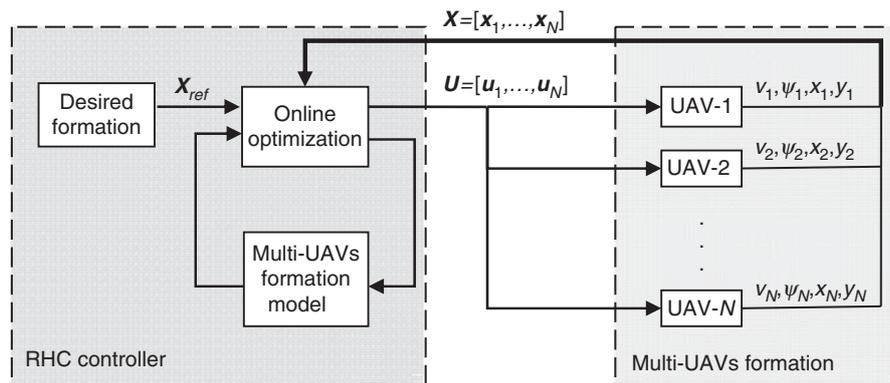

Figure 6 Block diagram of formation reconfiguration

Step 3: Initialize the DE population (each individual of the population is a candidate solution to $U(k|k)$). In order to improve the online searching efficiency and make full use of all aspects of information, half the individuals of the population are chosen randomly, and others are set as the control actions $U(K-1)$ at the former time horizon.

Step 4: Compute the fitness function value $f(x_a)$ according to Equation (9).

Step 5: Apply DE's mutation and recombination operators to x_a and generate trial vector u_a according to Equations (10) and (11), and then compute its fitness function value $f(u_a)$.

Step 6: Compare $f(u_a)$ and $f(x_a)$, implement DE's selection operator according to Equation (12) and then preserve the individual with the lower fitness value in the next generation. Go to Step 4 until the stopping criterion is satisfied.

Step 7: The best individual of DE population is just the optimal control sequence $U^*(k|k)$, output and apply them to each UAV respectively.

Step 8: Go to Step 2 and move forward into the following $(k+1)$th time horizon.

The above-mentioned formation reconfiguration process of the DE-based RHC controller can also be described in Figure 7.

5.   Simulation results

In order to investigate the feasibility and effectiveness of our proposed DE-based RHC approach to the problem of the multi-UAVs formation reconfiguration control, a series of experiments were conducted under the complicated combating environment. The proposed approach was coded in MATLAB language and implemented on PC-compatible with 2 GB of RAM under Microsoft Windows Vista.

In all experiments, parameters of DE and RHC are set as follows: $F = 0.9$, $CR = 0.5$, $NP = 100$, the number of iteration $NC = 100$, $m = p = 1$, $Q = diag\{1,1,1,1\}$, $T = 1$ s. In all experiments, for each UAV, aerodynamic drag $D = 2000$, the thrust $T \in [0,6000]$, the roll angle $\phi \in [-\pi/3, \pi/3]$, and UAV's weight $W = 10000$; $D_{safe} = 5$, $D_{com} = 50$.

Given the flight formation has five UAVs, they fly in an initial formation shape '|', and the terminal formation is a V-shape formation. The initial states of each UAV are $[2,0,0,20]$, $[2,0,0,10]$, $[2,0,0,0]$, $[2,0,0,-10]$ and $[2,0,0,-20]$. The desired formation can be described as $\mathbf{X}_{ref} = [0,0,-20,20;0,0,-10,10;0,0,0,0;0,0,-10,-10;0,0,-20,-20]$; choose UAV-3 as the leader of the flight formation. In the experiment, after four time horizons, multi-UAVs manoeuvre to the desired V-shape successfully, and Figure 8 shows the reconfiguration trajectory of multi-UAVs by the DE-based RHC controller. Figure 9(a)–(d) show the evolution curve of DE algorithm within each time horizon. Figure 10 describes the optimal fitness function value in each optimization process. Figures 11 and 12 show the optimal control actions implemented to each UAV, including the thrust and the roll angle. Figure 13 shows the distance between each two
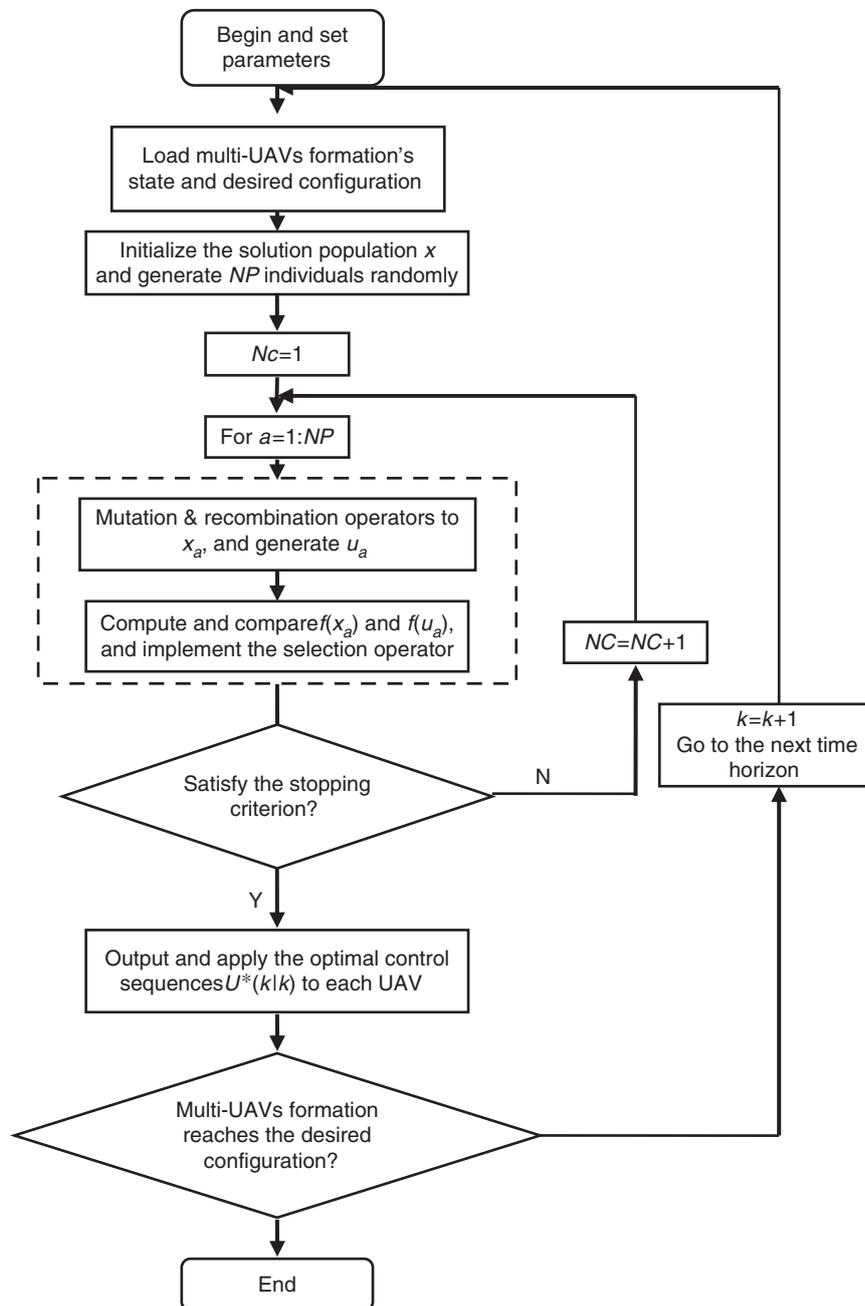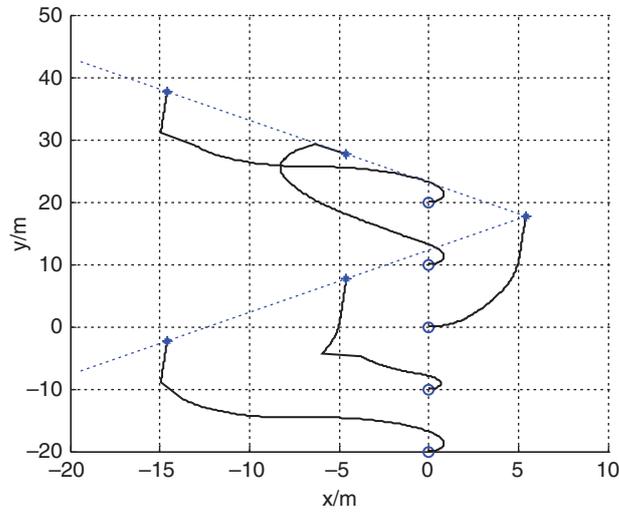
Figure 7 Flow chart of the proposed RHC controller

Figure 8 Reconfiguration trajectory of multi-UAVs; 'o' is the initial position, and '•' is the terminal position
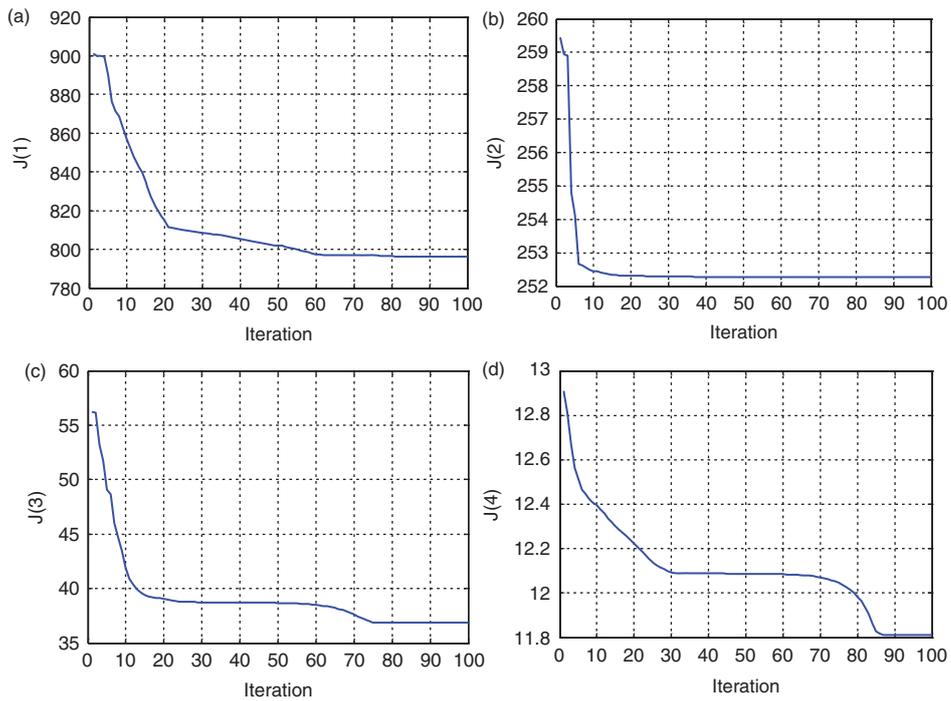

Figure 9 (a) Evolution curve at the 1st time horizon; (b) evolution curve at the 2nd time horizon; (c) evolution curve at the 3rd time horizon; (d) evolution curve at the 4th time horizon
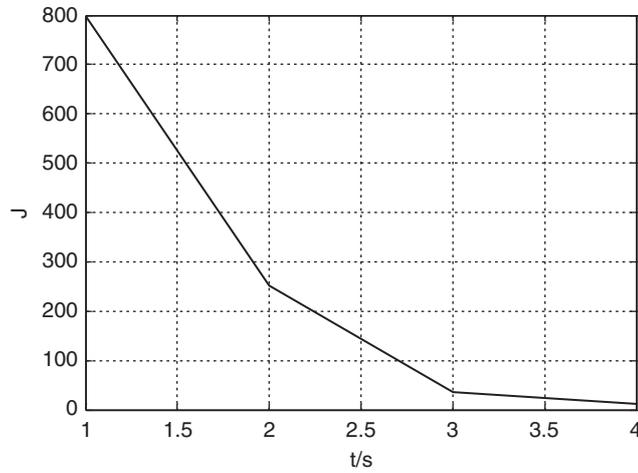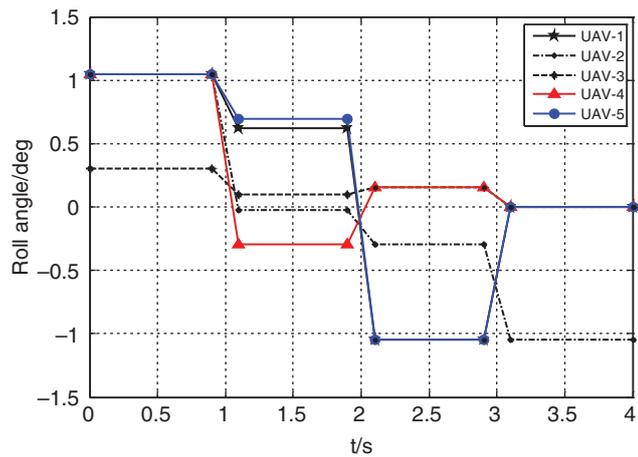
Figure 10 Fitness function value

Figure 11 Roll angle at each time horizon

UAVs, which is less than the communication distance and greater than safe anti-collision distance.

Another simulation is as follows. The initial states of each UAV are set as $[2, \pi, 0, 20]$, $[2, \pi, 0, 10]$, $[2, \pi, 0, 0]$, $[2, \pi, 0, -10]$ and $[2, \pi, 0, -20]$. Continuing to choose UAV-3 as the leader, the initial formation is as a '- - - - -' shape, and the desired formation is $\mathbf{X}_{ref} = [0, 0, -20, 20; 0, 0, -10, 10; 0, 0, 0, 0; 0, 0, -10, -10; 0, 0, -20, -20]$, which is also a V-shape formation. Figures 14–18 show the reconfiguration results after five time horizons. Similarly to the first experiment, multi-UAVs flied to their relative configuration successfully, and the distance constraints are also satisfied.
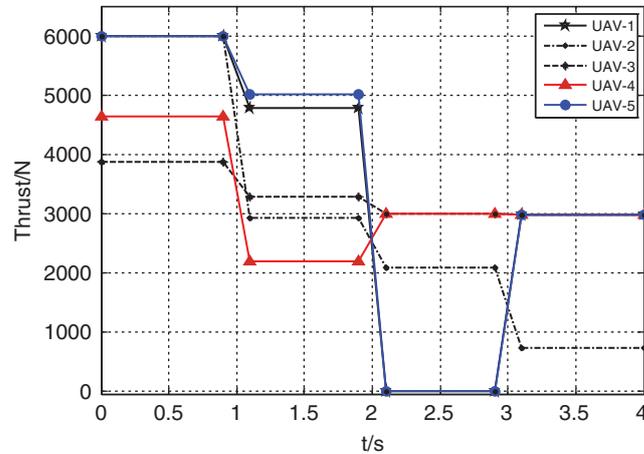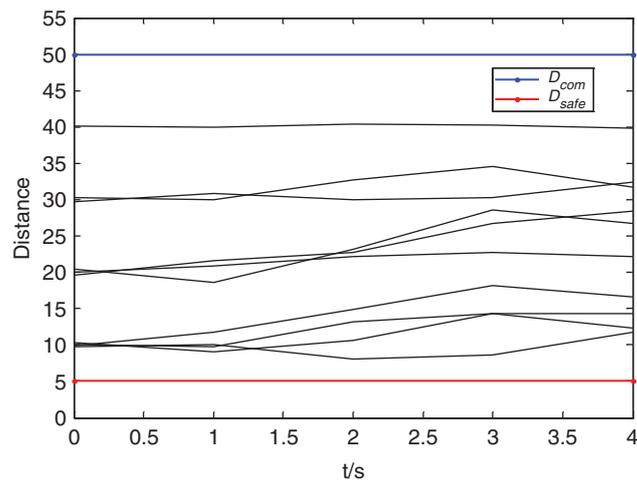
Figure 12 Thrust at every time horizon

Figure 13 Distance between each two UAVs

From these experiment results, it is obvious that the DE-based RHC approach can always find sequences of optimal solutions to meet the fitness function requirements and various constraints to achieve desired configuration. Because the parameters are much fewer within each optimization computing process using our proposed DE-based RHC approach, it can greatly decrease the computational complexity. However, compared with the optimal results using the CPTD method in Bai *et al*. (2009), Duan *et al*. (2008) and Xiong *et al*. (2008), as well as the polynomial optimal method in the Zelinski *et al*. (2003), our proposed RHC approach can also solve the formation reconfiguration problem efficiently.
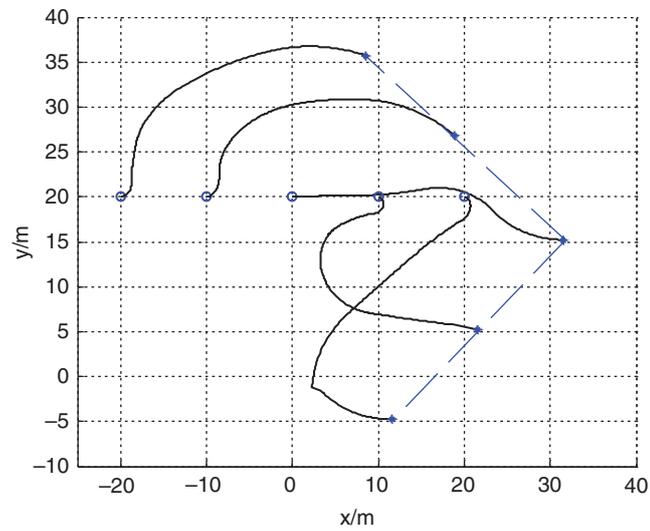
Figure 14 Reconfiguration trajectory of multi-UAVs; 'o' is the initial position, and '•' is the terminal position
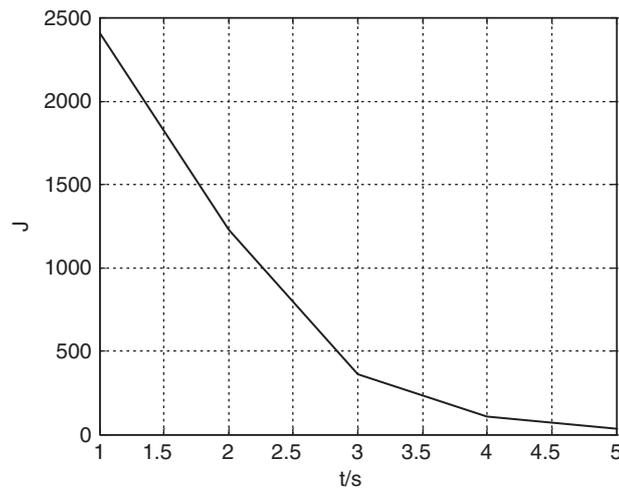

Figure 15 Optimal fitness function value at each time horizon

6.   Conclusions

The issue of multi-UAVs formation reconfiguration is a complicated constrained non-linear problem, and it is difficult for the traditional controller to divert each UAV to its desired relative position. In this work, a DE-based RHC method is proposed.
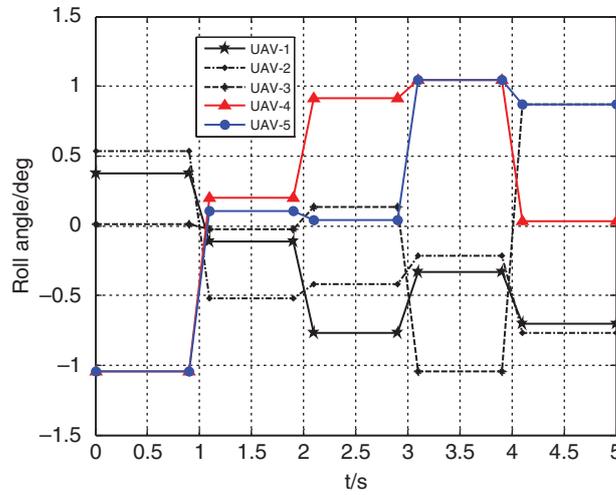
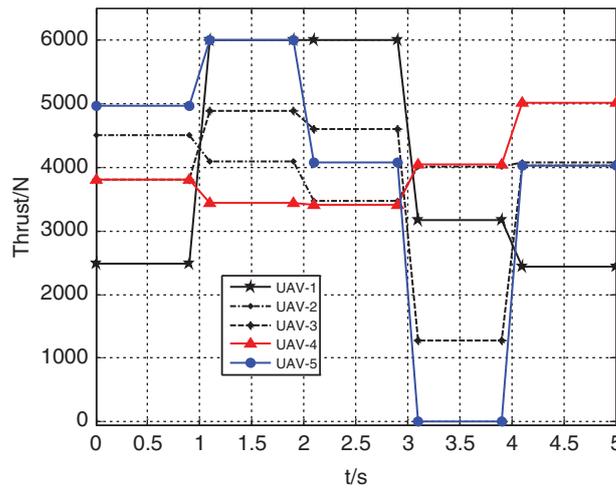**Figure 16** Roll angle at each time horizon

Figure 17 Thrust at each time horizon

The RHC method regards the global control problem as a series of local optimal problems, and the DE algorithm is utilized to work out the predicted control actions directly. Compared with the CPTD method, our proposed method has much fewer parameters to be optimized in one optimization process. Experimental results show that the proposed control approach can solve the optimal control problem of the multi-UAVs formation reconfiguration effectively. Our future work will focus on
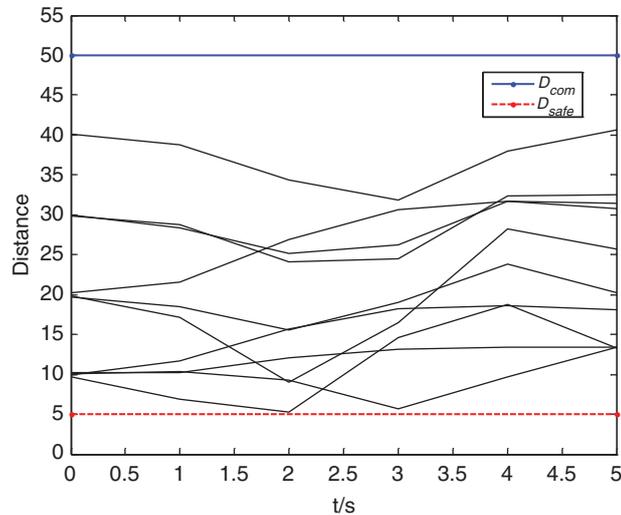
Figure 18 Distance between each two UAVs

developing a more exact formation reconfiguration model, with real, complicated environments taken into account. Moreover, considering the real-time requirement of UAV, the time complexity of our proposed method should also be analysed in the following work.

Acknowledgement

References

Bai, C., Duan, H.B., Li, C. and **Zhang, Y.** 2009: Dynamic multi-UAVs formation reconfiguration based on hybrid diversity-PSO and time optimal control. *Proceedings of 2009 IEEE Intelligent Vehicles Symposium*, Xi'an, China, 3–5 June, 775–79.

Bhattacharya, R., Balas, G.J., Kaya, A. and **Packard, A**. 2001: Nonlinear receding horizon control of F-16 aircraft. *Proceedings of the American Control Conference*, Arlington, VA, 25–27 June, 518–22.

Chichka D.F., Speyer, J.L. and **Park, C.G.** 1999: Peak-seeking control with application to formation flight. *Proceedings of the 38th IEEE Conference on Decision and Control*, Arlington, VA, 7–10 December, 3: 2463–70.

Duan, H.B., **Ma, G.J.** and **Luo, D.L.** 2008: Optimal formation reconfiguration control of multiple UCAVs using improved particle swarm optimization. *Journal of Bionic Engineering* 5, 340–47.

Furukawa, T. 2002: Time-subminimal trajectory planning for discrete non-linear systems. *Engineering Optimization* 34, 219–43.

Furuknwa, T., Durrant-Whyte, H.F., Bourgault, F. and **Dissanayake, G.** 2003: Time-optimal coordinated control of the relative formation of multiple vehicles.

Proceedings of 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation, Kobe, 16–20 July, 256–64.

Giuletti, F., **Pollini, L.** and **Innocenti, M.** 2000: Autonomous formation flight. *IEEE Control Systems Magazine* 20, 34–44.

Hendrickx, J.M., **Fidan, B.**, **Yu, C.**, **Anderson, B.D.O.** and **Blondel, V.D.** 2008: Formation reorgnization by primitive operations on directed graphs. *IEEE Transactions on Automatic Control* 53, 968–79.

Hu, X.B., **Chen, W.H.** and **Paolo, E.D.** 2007: Multiairport capacity management: genetic algorithm with receding horizon. *IEEE Transactions on Intelligent Systems* 8, 254–63.

Keviczky, T., Fregene, K., Borrelli, F., Balas, G.J. and **Godbole, D.** 2006: Coordinated autonomous vehicle formations: decentralization, control synthesis and optimization. *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, 14–16 June, 2022–27.

Mayne, D.Q., **Rawlings, J.B.**, **Rao, C.V.** and **Scokaert, P.O.M.** 2000: Constrained model predictive control: stability and optimality. *Automatica* 36, 789–814.

Paul, T., **Krogstad, T.R.** and **Gravdahl, J.T.** 2008: Modelling of UAV formation flight using 3D potential field. *Simulation Modelling Practice and Theory* 16, 1453–62.

Raffard, R.L., Tomlin, C.J. and **Boyd, S.P.** 2004: Distributed optimization for cooperative agents: application to formation flight. *Proceedings of 43rd IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, 14–17 December, 2453–59.

Saber, R.F., Dunbar, W.B. and **Murray, R.M.** 2003: Cooperative control of multi-vehicle systems using cost graphs and optimization. *Proceedings of the 2003 American Control Conference*, Denver, CO, 2217–22.

Shannon, Z., John, K.T. and **Shankar, S.** 2003: Hybrid system design for formations of autonomous vehicles. *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, HI, 1–6.

Simeon, T., **Leroy, S.** and **Lauumond, J.P.** 2002: Path coordination for multiple mobile robots: a resolution-complete algorithm. *IEEE Transactions on Robotics and Automation* 18, 42–49.

Storn, R. and **Price, K.** 1995: Differential evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report, International Computer Science Institute, Berkeley, CA.

Storn, R. and **Price, K.** 1997: Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* 11, 341–59.

Ueno, S. and **Kwon, S.J.** 2007: Optimal reconfiguration of UAVs in formation flight. *Proceedings of SICE Annual Conference 2007*, Kagawa University, Japan, 17–20 September, 2611–14.

Vachtsevanos, G., **Tang, L.**, **Drozeski, G.** and **Gutierrez, L.** 2005: From mission planning to flight control of unmanned aerial vehicles: strategies and implementation tools. *Annual Reviews in Control* 29, 101–15.

Xiong, W., **Chen, Z.J.** and **Zhou, R.** 2008: Optimization of multiple flight vehicle formation reconfiguration using hybrid genetic algorithm. *Acta Aeronautica et Astronautica Sinica* 29, s209–14.

Zielinski, K. and **Laur, R.** 2008: Stopping criteria for differential evolution in constrained single-objective optimization. *Advances in Differential Evolution* 143, 111–38.

Zielinski, K., Weitkemper, P., Laur, R. and **Kammeyer, K.-D.** 2006: Parameter study for differential evolution using a power allocation problem including interference cancellation. *Proceedings of the IEEE Congress on Evolutionary Computation*, Vancouver, 6748–55.

Zelinski, S., Koo, T.J. and **Sastry, S.** 2003: Optimization-based formation reconfiguration planning for autonomous vehicles. *Proceedings of the 2003 IEEE International Conference on Robotics & Automation*, Taipei, 14–19 September, 3758–63.