sensors

MDPI

Article

Coordinated Target Tracking via a Hybrid Optimization Approach

Yin Wang [1,2,*] and Yan Cao [2]

[1] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China
[2] College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; caoyan626@126.com
* Correspondence: yinwangee@nuaa.edu.cn; Tel.: +86-138-1399-5163

Academic Editors: Felipe Gonzalez Toro and Antonios Tsourdos
Received: 30 December 2016; Accepted: 23 February 2017; Published: 27 February 2017

Abstract: Recent advances in computer science and electronics have greatly expanded the capabilities of unmanned aerial vehicles (UAV) in both defense and civil applications, such as moving ground object tracking. Due to the uncertainties of the application environments and objects' motion, it is difficult to maintain the tracked object always within the sensor coverage area by using a single UAV. Hence, it is necessary to deploy a group of UAVs to improve the robustness of the tracking. This paper investigates the problem of tracking ground moving objects with a group of UAVs using gimbaled sensors under flight dynamic and collision-free constraints. The optimal cooperative tracking path planning problem is solved using an evolutionary optimization technique based on the framework of chemical reaction optimization (CRO). The efficiency of the proposed method was demonstrated through a series of comparative simulations. The results show that the cooperative tracking paths determined by the newly developed method allows for longer sensor coverage time under flight dynamic restrictions and safety conditions.

Keywords: unmanned aerial vehicles; UAV cooperation; persistent tracking; evolutionary algorithm

1. Introduction

Recently, unmanned aerial vehicles (UAV) have been widely used in both defense and civilian applications. In many of these applications, the UAV is required to continuously track a moving target, such as in the surveillance and tracking of ground objects. This task becomes challenging when the target is moving in a complex and dynamic environment, where the target may be occluded by ground features, such as buildings and mountains. In the case where the tracking cannot be achieved by using a single UAV, multiple UAVs may be deployed to improve the tracking robustness. The methods to coordinate the movements of multiple UAVs can be generally grouped into two categories, namely centralized and distributed algorithms, respectively [1]. In centralized methods, the mission of each UAV is dispatched from a central station or agent based on global information collected from the all of the aircraft. These methods generally tend to produce better solutions since the global information is available. In contrast, the distributed approaches determine the motion of each UAV by using the local information obtained from the UAV itself and its neighboring UAVs, which enjoys the advantage of computational efficiency and robustness when compared with their centralized counterparts. In a moving target tracking problem, besides generating the tracking path for each UAV, the motion of the target should be estimated since such information cannot be obtained from prior knowledge. As the purpose of this paper is to determine the optimal path of each UAV for cooperative tracking tasks, the motion pattern of the target is assumed to be available when it can be observed by the tracking UAVs.

The aim of cooperative tracking is to keep the target under the coverage of the UAV sensors at all time. To address this problem, Yang et al. adopted the Lyapunov Vector Field Control (LVC) approach to make the unmanned aerial vehicles converge to a circular area on top of the target [2]. However, this method does not take environmental factors into account, such as non-fly zones and the occlusion of sensors' field of view. Besides, in their method prior knowledge about the motion of the target is required, which is not available in most cases. In order to cope with this limitation, Belkhouche et al. [3] proposed to apply active avoidance strategies for obstacles in the environment, and determine feasible fly-zones by calculating the speed and position of the UAV relative to the obstacle. Cheng et al. presented a time-related optimal path planning algorithm to determine the cooperative tracking trajectories for each UAV [4]. Quintero and colleagues adopted a dynamic programming technique to minimize the distance error covariance for solving the cooperative tracking task, however the relative angle between UAVs and the target was ignored [5]. Chen et al., presented a hierarchical method, combining the Lyapunov Guidance Vector Field (LGVF) and Tangent Guidance Vector Field (TGVF) approaches, to solve the path planning problem for coordinated tracking [6]. In this method, the TGVF algorithm is firstly utilized to determine an optimal path from the initial point to a limiting circle, and then UAV would converge to the desired limit circle by optimizing the heading command through the use of the LGVF approach.

As discussed previously, due to the complexity and uncertainty of the ground environment, the target may be out of the sight of the UAV sensors, thus the motion of the target should be estimated. Shaferman et al. considered the sensor field-of-view (FOV) convergence in the urban environment, and determined an optimal path by maximizing the sensor convergence time to the target [7]. In order to improve the tracking performance of unmanned aerial vehicles (UAVs) under unknown conditions. Yu and his colleagues proposed an online path planning algorithm by estimating the location of the target based on its past states using a Bayesian filter based method [8]. Zhang et al., applied the Model Predictive Control (MPC) concept to solve the cooperative path planning problem with unknown target motions [9]. Beard and his co-workers present a decentralized cooperative aerial surveillance method by solving the associated combinational optimization problem, where both flight dynamics and environmental constraints are taken into account [10]. He and Xu proposed a model predictive control-based solution for the cooperative tracking path planning problem by considering urban area occlusions, however, the secure distances between UAVs were not taken into account [11].

Evolutionary algorithms, such as genetic algorithm approaches, also applied to solve the cooperative path planning problem [12–14], achieve good results in many offline path planning applications. Wise et al., compared commonly used methods for cooperative tracking path planning problems and in their work [15], readers can find a comprehensive review of the relevant literature.

This work present a novel cooperative tracking path planning approach for the visual tracking tasks of multiple UAVs, by coupling the receding horizon control principle and chemical reaction optimization framework [16]. The proposed method does not require any prior information about the motion and trajectory of the targets, and is capable of dealing with the occlusion of the sensors' field of view. Followed by this Introduction, the kinematics model of UAV and sensor visible regions model are described in Section 2. Section 3 presents the proposed method in great detail, and we demonstrate the performance of the proposed method through a series of comparative simulations in the following section. Finally, we conclude this work and discuss possible future work directions in Section 5.

2. Problem Description

2.1. Kinematics Model of UAV

Assuming the UAVs are flying at a constant altitude with a fixed speed, the motion of the UAV can be reduced to a 2-dimensional Dubins model. Figure 1 shows the 2D planar view of the Dubins coordinate system, where X_I, Y_I are the Cartesian inertial reference frame. v_a represents the velocity of the aircraft and ψ denotes the heading of the UAV.

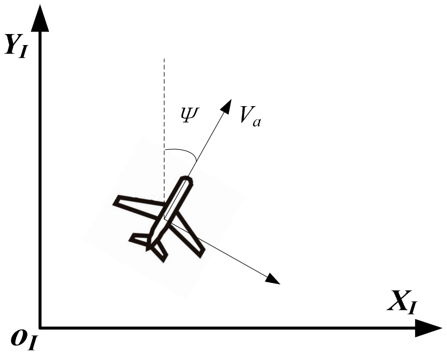


Figure 1. Schematic diagram illustrates the coordinate of the Dubins model.

In this paper, the angle of attack and the angle of side slip are neglected, then the kinematics equations of each aircraft defined in the inertial frame can be represented as:

$$\begin{aligned} \dot{x}_i(t) &= v_a \cos(\psi_i(t)) \\ \dot{y}_i(t) &= v_a \sin(\psi_i(t)) \\ \dot{\psi}_i(t) &= \tfrac{g \tan \phi_i(t)}{v_g}, \quad |\phi_i(t)| \leq \phi_{\max} \end{aligned} \tag{1}$$

where $x_i(t)$ and $y_i(t)$ represent the position of the i-th UAV at time instant t. $\phi_i(t)$ denotes the roll angle of the UAV, which is bounded to the range within $-\phi_{\max}$ and $\phi_{\max}$. Assuming that each UAV is equipped with an autopilot so that it is able to follow the given roll angle, let T_s denote the sampling rate and we apply zero-order hold to the input signal, then the discrete kinematics of the UAV can be found as:

$$\begin{aligned} \dot{\psi}_i(k) &= \tfrac{g \tan \phi_i(k)}{v_g} \\ x_i(k+1) &= \tfrac{v_g}{\dot{\psi}_i(k)} \left[\sin(\dot{\psi}_i(k) T_s + \psi_i(k)) - \sin(\psi_i(k)) \right] + x_i(k) \\ y_i(k+1) &= \tfrac{v_g}{\dot{\psi}_i(k)} \left[\cos(\psi_i(k)) - \cos(\dot{\psi}_i(k) T_s + \psi_i(k)) \right] + y_i(k) \\ \psi_i(k+1) &= \dot{\psi}_i(k) T_s + \psi_i(k) \end{aligned} \tag{2}$$

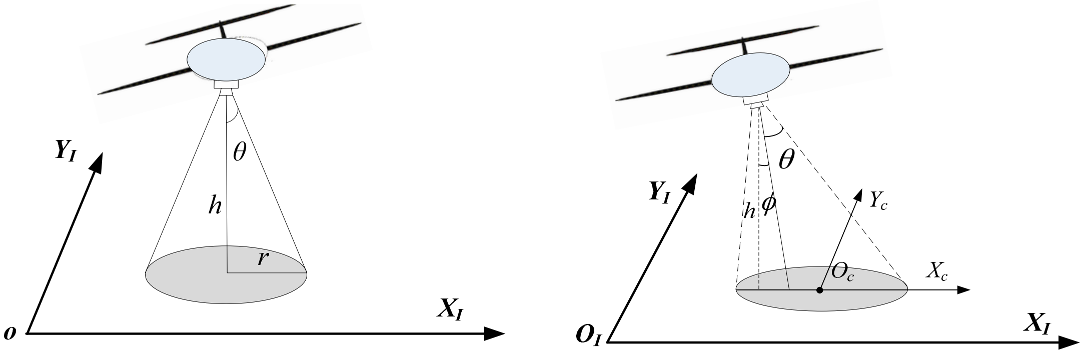
If the input roll angle is zero, Equation (2) can be rewritten as:

$$\begin{aligned} x_i(k+1) &= vT_s \cos\left(\dot{\psi}_i(k)\right) + x_i(k) \\ y_i(k+1) &= vT_s \sin\left(\psi_i(k)\right) + y_i(k) \\ \psi_i(k+1) &= \psi_i(k) \end{aligned} \tag{3}$$

2.2. Sensor Visible Regions in Complex Environment

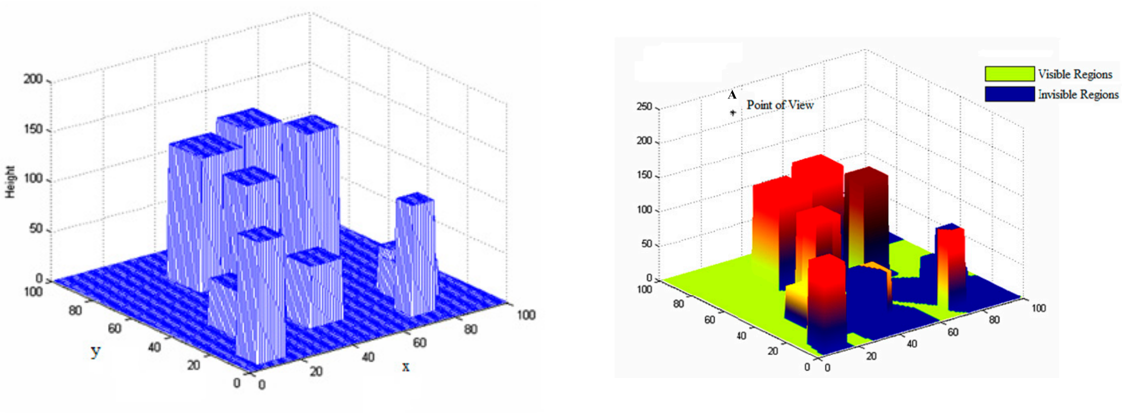
The sensors carried by the UAV can be categorized into to two groups, in terms of the installation conditions, namely gimbaled sensors and body fixed sensors. As shown in Figure 2a, gimbaled installed sensors are capable of pointing to a fixed direction regardless of the attitude of the UAV. The sensor converge area is a cone-shaped region under the aircraft. The pointing orientation of body fixed sensors, on the other hand, are usually affected by the state of the UAV, such as the roll and pitch angles, as shown in Figure 2b. To simplify the problem, in this work UAVs are assumed to be equipped with gimbaled sensors. Hence the sensor convergence on the ground would not be affected by the changes of UAV's attitude.

In many applications, the target to be tracked is moving in a complex environment, which may be occluded by terrain features, such as buildings in an urban environment. Figure 3 illustrates a simulated urban environment, where the 3D rectangular structures represent buildings. Due to the occlusion of buildings, not all of the convergence area of the sensor is visible. The actual visible area depends on the height and the field of view of the sensor. As shown in Figure 3, assuming the sensor is located at (sx, sy, sz), the visible regions on the ground can be determined calculated by using the line of slight (LoS) methods. Rather than using conventional geometrical analysis approaches, such as LoS algorithms, to define the visible areas of the sensors, we employ a fast yet robust spatial analysis method developed in [17], which greatly improves the efficiency of this procedure.



(**a**) (**b**)

Figure 2. Illustration of the sensor convergence areas with different types of sensors. (**a**) illustrates the visibility regions of gimbaled sensors when the aircraft is turning; (**b**) depicts the visibility regions of body-fixed sensors when the aircraft is turning.



(**a**) (**b**)

Figure 3. Illustrates of the sensor visible areas in complex environments. (**a**) The simulated urban environment with buildings (rectangular structures); (**b**) shows the visibility regions of a sensor located at the point **A**.

Figure 3b depicts the visible areas looked from the point **A**, the blue regions are invisible areas due to the occlusion of buildings or terrains. Hence, the detectable regions of the sensor can be defined as:

$$R_{vis}(p) = R_{con}(p) \cap R_{in}(p) \tag{4}$$

where R_{con} is the convergence region of the sensor positioned at p, and R_{in} denotes all of the possible visible areas visible from p. Given the field of view angle of the gimbaled sensor θ, and the height of the sensor h, the convergence region on the ground can be calculated as:

$$R_{con} = \pi r^2 \quad , \quad r = h \tan\theta \tag{5}$$

2.3. The Motion of the Ground Target

The ground object to be tracked is assumed to move within the 2-dimensional plane, and the associated motion can be described by a state vector $\boldsymbol{x}_t = [x_t,\ y_t,\ \dot{x}_t,\ \dot{y}_t,\ \ddot{x}_t,\ \ddot{y}_t]$. Let ***F***() represent the state-transition function of the target dynamic, the motion of the target can be described as follows:

$$\boldsymbol{x}_t(k+1) = \boldsymbol{F}[\boldsymbol{x}_t(k)] + \boldsymbol{W}(k) \tag{6}$$

where ***W***() denotes the state noise and is usually assumed to be subject to a zero-mean Gaussian distribution, i.e., p(**W**)~N(0, **Q**), **Q** is the covariance matrix. The possible motion of the target can be predicted by filtering approaches [13].

3. The Proposed Method

In this paper, we present a solution to the problem of tracking by multiple coordinated UAVs based on the concept of the Receding Horizon Control (RHC) method, which minimizes the uncertainties arising from the application environment and the motion of the target. The associated optimal control sequence estimation problem is then solved by using a novel bio-inspired optimization approach, based on the framework of chemical reaction optimization.

3.1. The Receding Horizon Control in Coordinated Tracking

Since the motion of the target is not available as prior knowledge, it requires predicting the trajectory of the target based on previous observation. Receding horizon control (RHC) approach is an advanced model predictive controller allows for determining the optimal solution at current time instant while taking the future states of the target into account. Let $u[k{:}k + L]$ denote the control sequence starts from the time *k*, and *L* is a constant indicating the window size of the RHC sequence. $\boldsymbol{X}[k]$ represents the state of the dynamic system, including the positions of the UAV and the target, respectively. The main steps of RHC method for solving the target tracking problem are as follows:

Step 1: Determine the optimal control sequence based on the state $\boldsymbol{X}\ [k]$:

$$u^*[k:k+L] = \text{argmin} J(\boldsymbol{X}[k], u[k:k+L]) \tag{7}$$

where u^* denotes the optimal control sequence minimizing the cost function *J*. Since the dynamic model of the UAV is obtained from prior knowledge, the trajectory of the UAV can be updated in an iterative fashion based on the input control signal and its current state. Because the motion of the target is unknown, its future positions can only be estimated based on previous observation. The cost function *J* returns the goodness of the tracking based on the relative position between the target and the UAV, which would be discussed in detail in Section 3.2.

Step 2: Apply the optimal input sequence u^* to the system, and update the states of the system.

Step 3: Repeat Steps (1) and (2) for the next time instant.

3.2. The Cost of the Tracking Path

3.2.1. Dynamic Constraint of the UAV

In practice, the roll angle of the UAV should be within a reasonable scale when changing its orientation, due to safety considerations. Hence, the input roll angle should satisfy the following constraint:

$$\phi \in (-\phi_{\max}, \phi_{\max})$$

where $\phi_{\max}$ is a constant representing the maximal roll angle of the UAV during turning process.

3.2.2. The Target Observation Time

The objective of the coordinate target tracking is to maintain the target located within the detectable region of at least one UAV at any time instant. In addition, it favors the targets located next to the center of the sensor visible area. Let (rx, ry) denote the relative position of the target with respect to the center of the visible area of the sensor:

$$J_s = \sum_{i=1}^{N} \sum_{k=1}^{k+L} sen(\boldsymbol{X}[k]) \tag{8}$$

$$Sen = \begin{cases} -Ce^{-\frac{rx^2+ry^2}{2\sigma^2}}, & \text{the target is within the visible region of at least one sensor} \\ P, & \text{the target is out of the visible region for all sensors} \end{cases}$$

where the function *Sen*() quantifies whether the target can be observed by the sensors of UAVs based on the relative position between the UAVs and the target, This term reaches its minimum if the target is located within the center of the visible area of all of the sensors, and approaches zero if the target is moving to the boundaries of the sensor visible regions. P is a positive constant used to penalize the case where the target is out of the visible region of all of the sensors.

3.2.3. The Anti-Collision Constraint

In this paper, we assume the UAVs applied to track the target are distributed on the same flight level, thus it is necessary to prevent the UAV from colliding with each other during tracking:

$$J_{AC} = \sum_{i=1}^{N} \sum_{j\in\Omega(i)} erfc[k \cdot (\|\boldsymbol{x}_i - \boldsymbol{x}_j\| - d)] \tag{9}$$

where k is a constant, d indicates the desired distance between UAVs, $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ represent the positions of i-th and j-th UAVs in the inertia reference frame, respectively. *erfc* denotes the Gaussian error function which is defined as:

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-x^2} dx \tag{10}$$

This term would reach its minimum when the positions of the UAVs are evenly distributed around the target.

3.2.4. The Regulation Term

In order to ensure the planned path is smooth, the change of the input signal is used as the regulator to penalize a jagged path:

$$J_{reg} = \sum_{i=1}^{N} \sum_{j=k+1}^{k+L} (u_i(j) - u_i(j-1)) \tag{11}$$

This term approaches zero when a straight path is generated, and it would be large in magnitude if the resulting path is highly curved. The total cost can be obtained by summing the aforementioned terms together, leading to the following function:

$$J = w_1 J_s + w_2 J_{AC} + w_3 J_{reg} \tag{12}$$

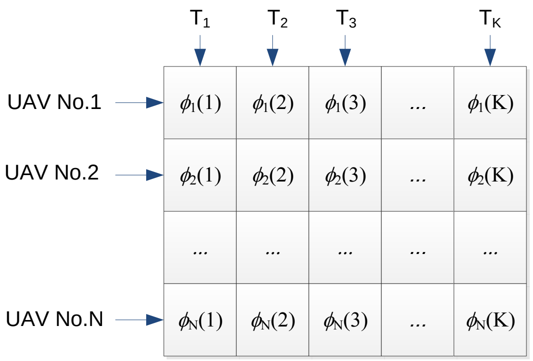
where w_1, w_2 and w_3 are constants controlling the influence of each term on the total energy.

3.3. *Chemical Reaction Optimization-Based Coordinated Tracking*

The Chemical Reaction Optimization (CRO) algorithm, first reported by Lam and Li [16], is a nature-inspired optimization algorithm based on the principle of chemical reactions. It mimics the process of chemical reactions occuring in a closed container. CRO is a type of population-based algorithm where the basic individuals are molecules. Each molecule is described by some attributes, including the molecular position, i.e., a possible solution of the optimization problem, the potential energy (PE) and the kinetic energy (KE). The molecule structure represents a possible solution of the optimization problem, the potential energy (PE) is associated with the cost of the resulting solution based on the objective function. In general, a small value of the potential energy may indicate a better solution of the optimization problem. Kinetic energy (KE) defines the tolerance of the system for accepting a worse solution than the existing one. In addition to molecules, the container (buffer), where the chemical reaction takes place, is another key component of the CRO algorithm. It is able to supply or absorb the energy resulting during the process of the molecular reactions. Different from many existing population-based optimization algorithms, the total number of molecules may not be the same in each iteration of the optimization process, but the entire amount of energy (the energies of molecules and the energy of the buffer) of the system stays constant throughout the reaction process.

3.3.1. Coding Scheme

As shown in Figure 4, the molecular structure is defined as a sequence of bounded input signals, i.e., the desired roll angle to the autopilot in the time from t$_k$ to t_{k+1} for each UAV. Thus, the dimensions of each molecule would be $N \times L$, where N represents the total number of UAVs employed for coordinated tracking, and L is the number of steps in the predicted ahead current state.



	T$_1$	T$_2$	T$_3$		T$_K$
UAV No.1	$\phi_1(1)$	$\phi_1(2)$	$\phi_1(3)$	...	$\phi_1(K)$
UAV No.2	$\phi_2(1)$	$\phi_2(2)$	$\phi_2(3)$	...	$\phi_2(K)$
	...	...	...	...	...
UAV No.N	$\phi_N(1)$	$\phi_N(2)$	$\phi_N(3)$	...	$\phi_N(K)$

Figure 4. The coding scheme of the CRO algorithm.

3.3.2. On-Wall Ineffective Collision

In this elementary process, the new path is determined by slightly deforming the current path, which can be considered as a local search around the current solution. Let KE_ω and PE_ω represent the kinetic and potential energy of a molecule before it hitting the wall of the container, respectively. The potential energy can be calculated based on Equation (12), and the kinetic energy for each molecule is initialized randomly. Due to the collisions, a certain amount of the energy may transfer to the wall of the container (i.e., the buffer), and thus the potential $PE_{\omega'}$ and kinetic energy $KE_{\omega'}$ of such molecule at the new statue can be computed as:

$$KE_{\omega'} = (PE_\omega + KE_\omega - PE_{\omega'}) \times \alpha, \quad \text{if } (PE_\omega + KE_\omega - PE_{\omega'}) \geq 0 \tag{13}$$

Let *KElossRate* (*KElossRate* ϵ [0, 1]) be a constant defining the percentage of energy that lost before and after the molecule hits the wall, α is a random number evenly distributed from *KElossRate* to 1. Based on the law of conversion of energy, the energy absorbed by the buffer can be expressed as:

$$KE_{\omega'} = (PE_{\omega} + KE_{\omega} - PE_{\omega'}) \times (1 - \alpha) \tag{14}$$

If the kinetic energy of the molecule before wall collision is large enough, the after collision potential energy may larger than PE_{ω}, leading to a worse solution when compared with the existing one. It allows the local search algorithm the ability of 'hill-climbing', which increases the probability of avoiding local optimal solutions.

3.3.3. Decomposition

Unlike the process of the 'on-wall ineffective collision', the molecule breaks into parts after it hits the wall of the container ($\omega \rightarrow \omega_1{}' + \omega_2{}'$). For simplicity we only consider two parts. Because more solutions are created through the decomposition process, the sum of KE and PE energies of the original molecule may be less than the sum of potential energies of the resulting molecules:

$$PE_{\omega} + KE_{\omega} < PE_{\omega'_1} + PE_{\omega'_2} \tag{15}$$

In this case, such decomposition is not valid since it disobeys the law of energy conservation. In order to increase the possibility of a correct decomposition process, the buffer may supply a certain amount of the energy to the decomposition process so that the following condition can be satisfied:

$$PE_{\omega} + KE_{\omega} + \delta_1 \times \delta_2 \times buffer \geq PE_{\omega'_1} + PE_{\omega'_2} \tag{16}$$

where δ_1 and δ_2 are two independently random variables uniformly distributed in the range of [0, 1]. The remaining energy is translated as the kinetic energies for two new molecules as:

$$\begin{array}{c} PE_{\omega'_1} = (PE_{\omega} + KE_{\omega} + \delta_1 \times \delta_2 \times buffer - PE_{\omega'_1} - PE_{\omega'_2}) \times \delta_3 \\ PE_{\omega'_2} = (PE_{\omega} + KE_{\omega} + \delta_1 \times \delta_2 \times buffer - PE_{\omega'_1} - PE_{\omega'_2}) \times (1 - \delta_3) \end{array} \tag{17}$$

where δ_3 is another independent random variable range from 0 to 1. The decomposition process provides the molecule the ability of exploring more solution regions with respect to the initial position ω.

3.3.4. Inter-Molecular Ineffective Collision

The reaction process is very similar to its single molecule reaction counterpart, where the collision takes place between molecules. The number of the molecules is assumed to be unchanged before and after the collision, and the molecular energies of the associated reaction process should satisfy the condition:

$$PE_{\omega_1} + KE_{\omega_1} + PE_{\omega_2} + KE_{\omega_2} \geq PE_{\omega'_1} + PE_{\omega'_2} \tag{18}$$

The kinetic energies of the new molecules are:

$$\begin{array}{l} KE_{\omega'_1} = E_{\text{inter}} \times \delta_4 \\ KE_{\omega'_2} = E_{\text{inter}} \times (1 - \delta_4) \\ E_{\text{inter}} = PE_{\omega_1} + KE_{\omega_1} + PE_{\omega_2} + KE_{\omega_2} - PE_{\omega'_1} + PE_{\omega'_2} \end{array} \tag{19}$$

The inter-molecule collision allows a median range local search, thus increasing the probability of detecting a better solution in a local area.

3.3.5. Synthesis

Synthesis is the opposite reaction process of decomposition, where two (multiple) molecules merge into a single molecule. As the new molecule is produced from multiple molecules, it is more likely to meet the following requirement for such a process:

$$PE_{\omega_1} + KE_{\omega_1} + PE_{\omega_2} + KE_{\omega_2} \geq PE_{\omega'} \tag{20}$$

The remaining energy of the new molecule is:

$$KE_{\omega'} = PE_{\omega_1} + KE_{\omega_1} + PE_{\omega_2} + KE_{\omega_2} - PE_{\omega'} \tag{21}$$

In this reaction process, the new molecule is a combination of multiple old molecules, thus the newly produced molecule may exhibit a large deviation from the original ones in the solution space.

The workflow of the CRO based optimization framework is shown in Figure 5.

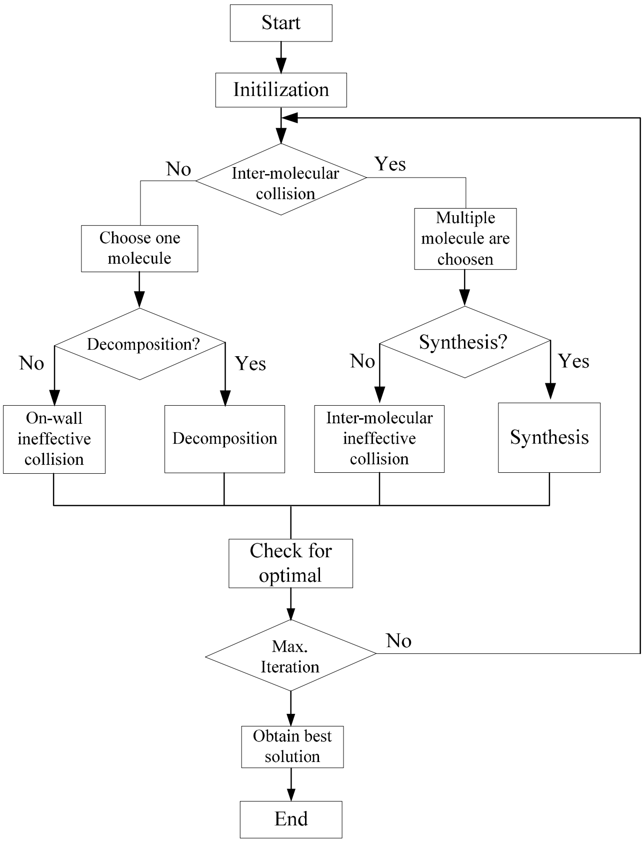


Figure 5. Schematic diagram that illustrates the CRO optimization framework.

4. Simulation Results and Analysis

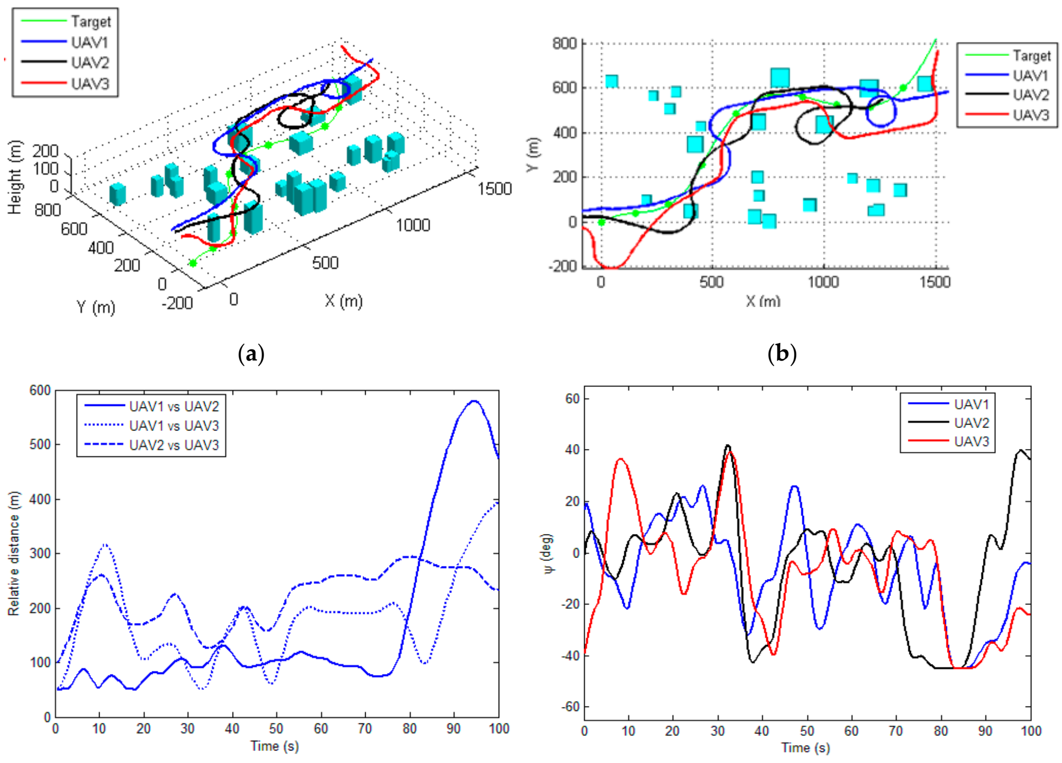
This section presents comparative simulation results to demonstrate the efficiency and accuracy of the proposed technique in the coordinated UAV tracking problem. The simulations were conducted using 3D synthetic environmental data, which allows for generating a variety of virtual environments with different resolutions and urban features. The simulations were carried out in MATLAB (R2010b, MathWorks, Natick, MA, USA) on a standard specification PC (Dell Precision T3500, DELL, Round Rock, TX, USA, Inter(R) Xeon(R) CPU operating at 2.67 GHz).

The dimensions of the virtual environment are defined within the range of 1.5 km × 1.5 km, and 30 buildings with different dimensions are randomly generated. Three UAVs are employed to track one moving target, and the motion of the UAVs and target are described using the dynamic model discussed in Section 2. The flight speed v_g for each UAV is set to be 30 m/s, the flight level is 200 m, the maximum roll angle $\phi_{\max}$ is set to 45° and the minimal distance between each UAV is 50 m. The trajectory of the target is randomly generated in the flexible region on the ground, with a constant speed at 15 m/s. The RHC window size L is set as 5 s, and the total simulation time is 100 s, and the sampling time is 1 s. The parameter settings of the CRO algorithm are shown in Table 1.

Table 1. Parameter settings of CRO algorithm.

KELossRate	0.2	InitialKE	200
MoleColl	0.3	PopSize	100
buffer	0	α	500
B	10	c_1	0.15
c_2	2.5		

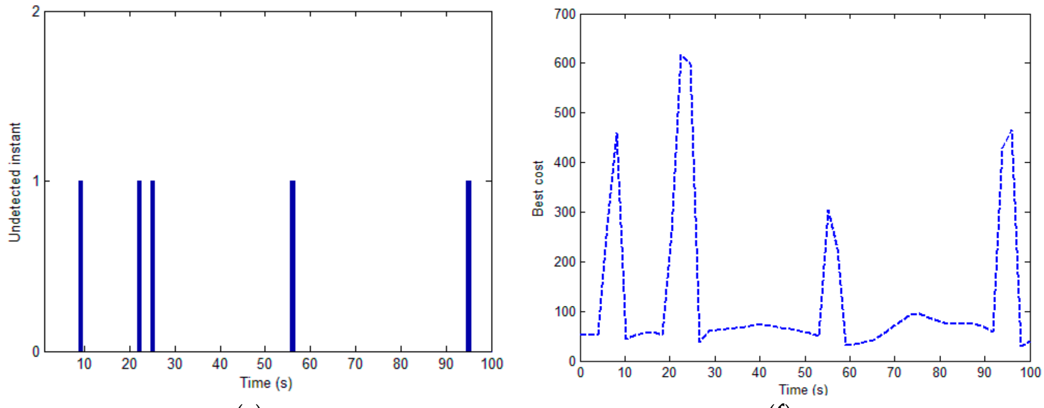
Figures 6 and 7 illustrate the performance of the proposed method for tracking the moving target in two scenarios. The total visibility time for these two scenarios are 95 s and 94 s (as shown in Figures 6e and 7e, respectively), which approach the total flight time (100 s). This indicates that the proposed method is able to generate an optimal tracking path for each UAV. The costs of the proposed method at each sampling time are depicted in Figures 6f and 7f, respectively. The cost grows rapidly when the target is out of the visibility region for all of the sensors, and for the rest the time the costs are almost constant, implying that the proposed algorithm can determine the optimal path within the sampling time (1 s). As shown in Figures 6d and 7d, the control input for each UAVs, i.e., the roll angle, was changing frequently during the tracking process, since the UAVs are required to maneuver to track the moving target while avoid the occlusions of the buildings. The control inputs determined by the proposed method are generally smooth and changing slowly, which can be tracked by the autopilot of the UAV.



(**a**) (**b**)

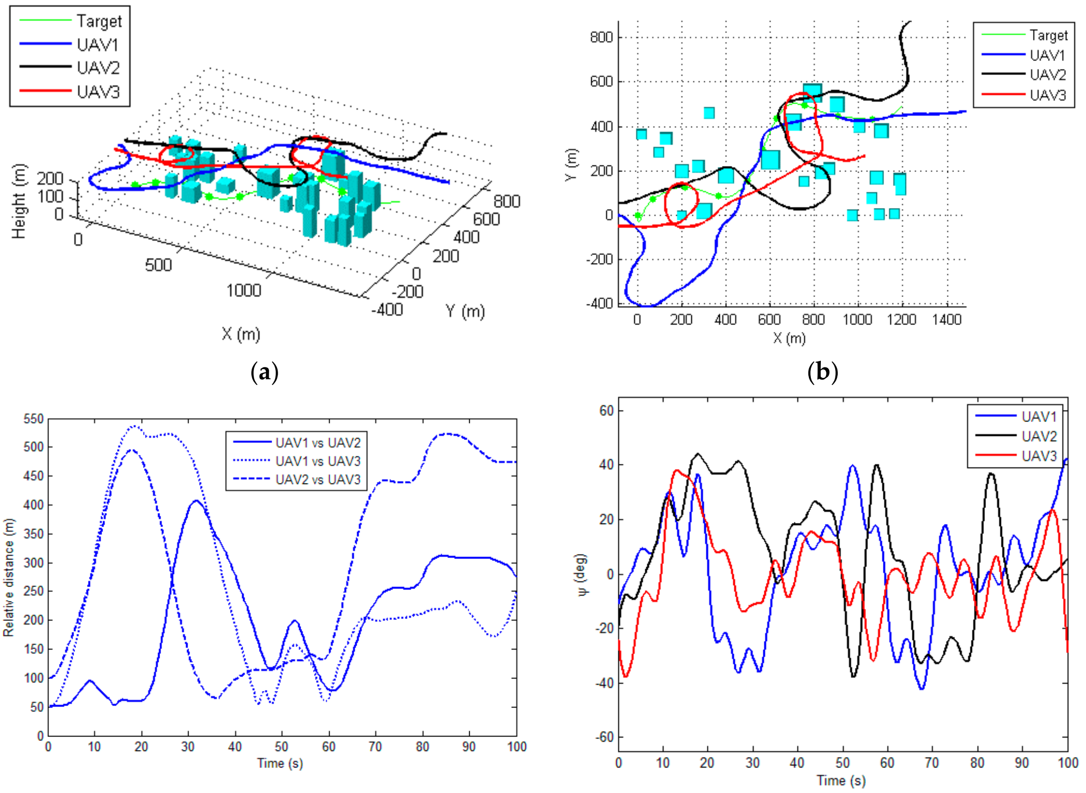
(**c**) (**d**)

Figure 6. *Cont.*



(**e**) (**f**)

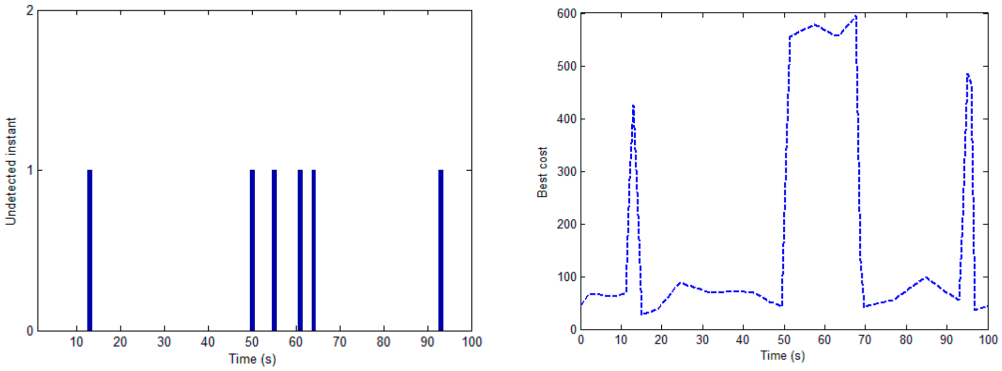
Figure 6. The performance of the proposed method for scenario 1. (**a**) 3D view of the planned path for each UAV; (**b**) Top view of the resulting paths; (**c**) The relative distance among each UAV during tracking; (**d**) The roll-command for each UAV (**e**) The undetected time-instant; (**f**) The total cost of the tracking process.



(**a**) (**b**)

(**c**) (**d**)

Figure 7. *Cont.*



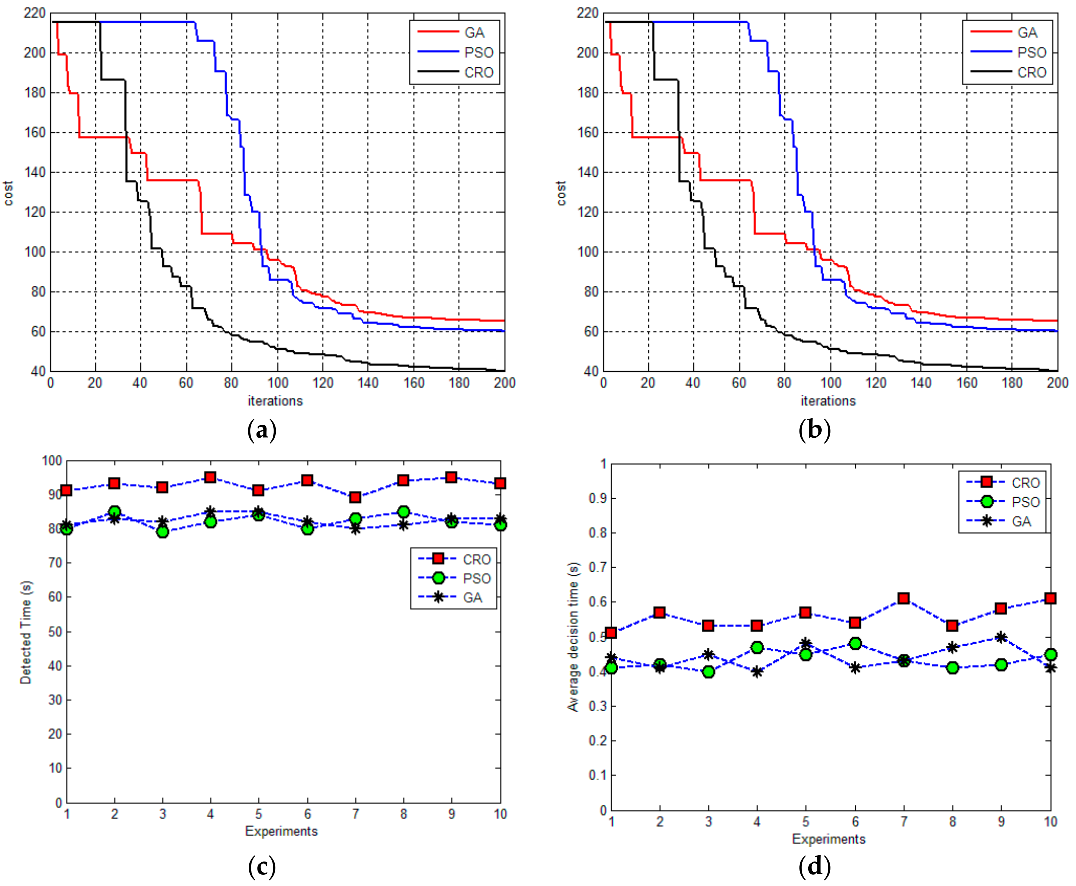
(**e**) (**f**)

Figure 7. The performance of the proposed method for scenario 2. (**a**) 3D view of the planned path for each UAV; (**b**) Top view of the resulting paths; (**c**) The relative distance among each UAV during tracking; (**d**) The roll-command for each UAV (**e**) The undetected time-instant; (**f**) The total cost of the tracking process.

Figures 6c and 7c illustrate the relative distances between each UAV during the tracking process for two scenarios, respectively. It can be seen that the generated path by the proposed method always satisfies the safety constraint (i.e., the minimal distance between each UAV). In order to demonstrate the efficiency of the proposed CRO based cooperative tracking path planning algorithm, this section firstly compares the proposed approach with commonly used evolutionary algorithms based optimizer, i.e., the genetic algorithm (GA) [13] and particle swarm optimization (PSO) [14], for determination of the tracking paths.

Figures 8 and 9 illustrate the performances of the proposed method, GA-based method and PSO- based algorithm for the tracking task, respectively. Figure 8a,b and Figure 9a,b illustrate the convergence of the proposed method and the other evolutionary-based optimizers from randomly selected sampling times for two scenarios. It can been seen that the proposed approach outperforms to GA- and PSO-based optimizer in terms of convergence speed and optimality (in terms of the cost value). Figures 8c and 9c depict the visibility time of the ground target of the aforementioned algorithms obtained from 10 repeat simulations. The overall performance of the proposed method is better than GA- and PSO-based planners in terms of the visibility time. This is because the CRO is a variable population-based metaheuristic approach, which allows for generating more solutions near the optimality and thus increasing the convergence speed. Figures 8d and 9d are the average execution times for each decision point (i.e., the sampling point), where the proposed method requires more time than the GA- and PSO-based planners. However, this execution time (around 0.55 s for the proposed method) is acceptable when compared the sampling time (1 s).

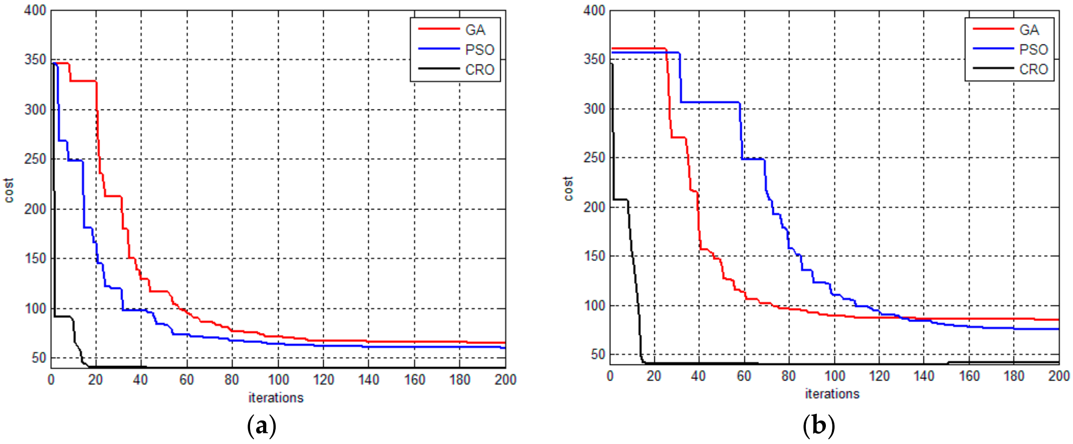
Next, the proposed method is compared with a recently developed Lyapunov guidance vector field (LGVF)-based approach for planning the cooperative tracking path [18]. Figures 10 and 11 illustrate the results of using the LGVF-based approach for two scenarios. It can be observed that the proposed method achieves better solution to the cooperative path planning problem in terms of the visibility time. The minimal distances between UAVs are not always satisfactory (i.e., less than the safety distance, i.e., 50 m), as shown in Figures 10c and 11c. The statistics of the proposed method and the LGVF-based approach obtained from 10 repeat simulation tests are shown in Tables 2 and 3, respectively. The mean, maximum and minimum visibility times of the proposed method are better than those of the LGVF-based planner. This is because it is difficult for the LGVF-based optimizer to incorporate nonlinear and non-convex constraints, such as the visibility of sensors and anti-collision constraints in the tracking optimization problem, and thus cannot always reach the optimal solution.



(**a**) (**b**)

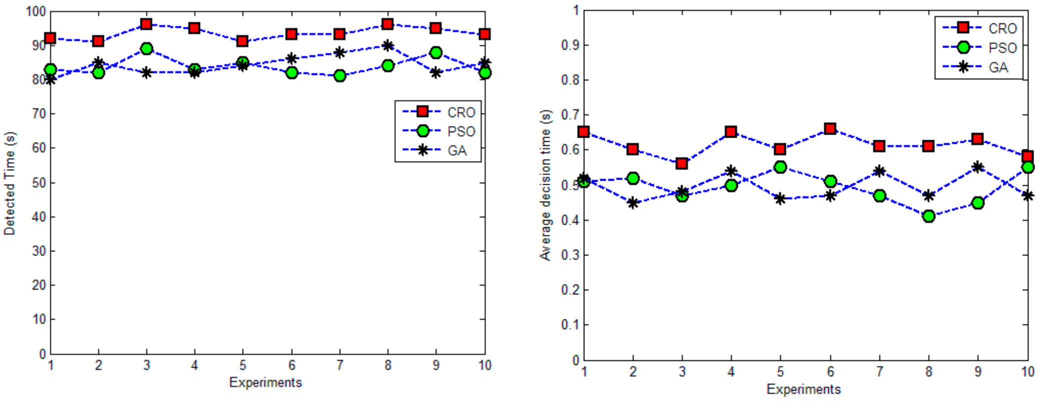
(**c**) (**d**)

Figure 8. The performances of using different evolutionary optimizers for determination of the tracking path for scenario 1. (**a**) and (**b**) illustrate the convergence of different methods in selected decision time, respectively; (**c**) shows the visibility time of the ground time for the proposed method, PSO and GA based algorithms, respectively; (**d**) is the average execution time of the aforementioned methods.



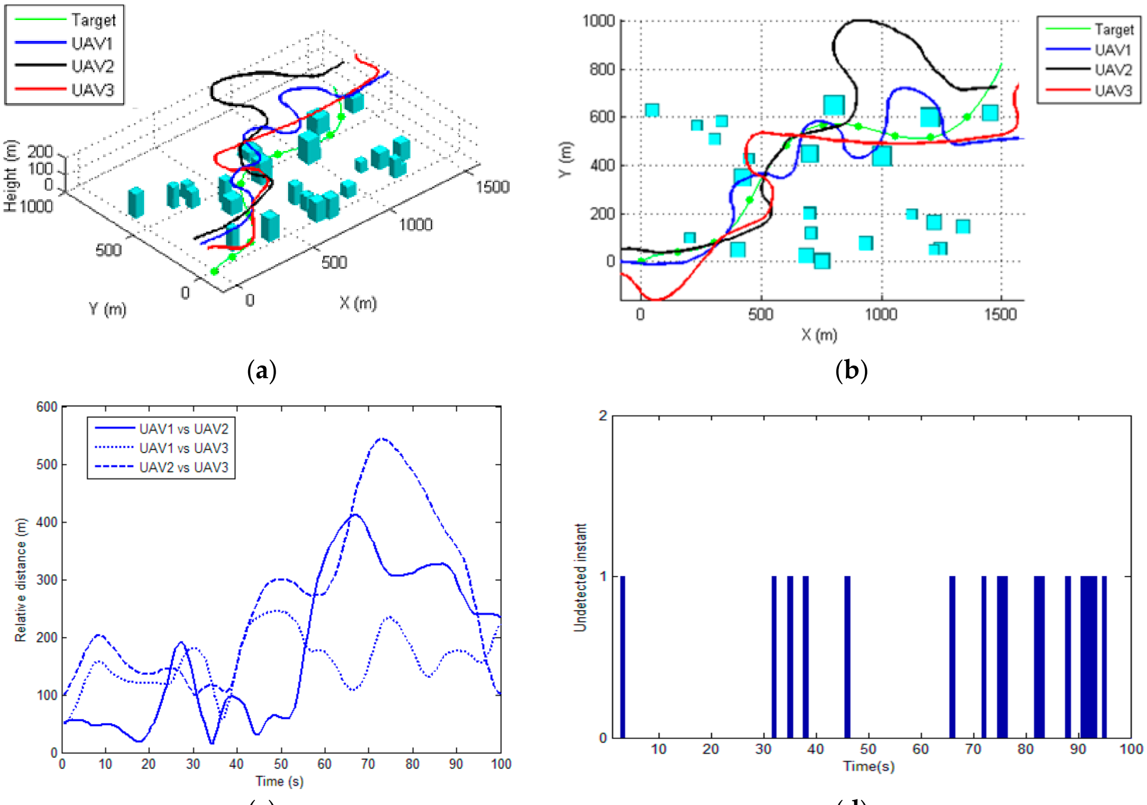
(**a**) (**b**)

Figure 9. *Cont.*



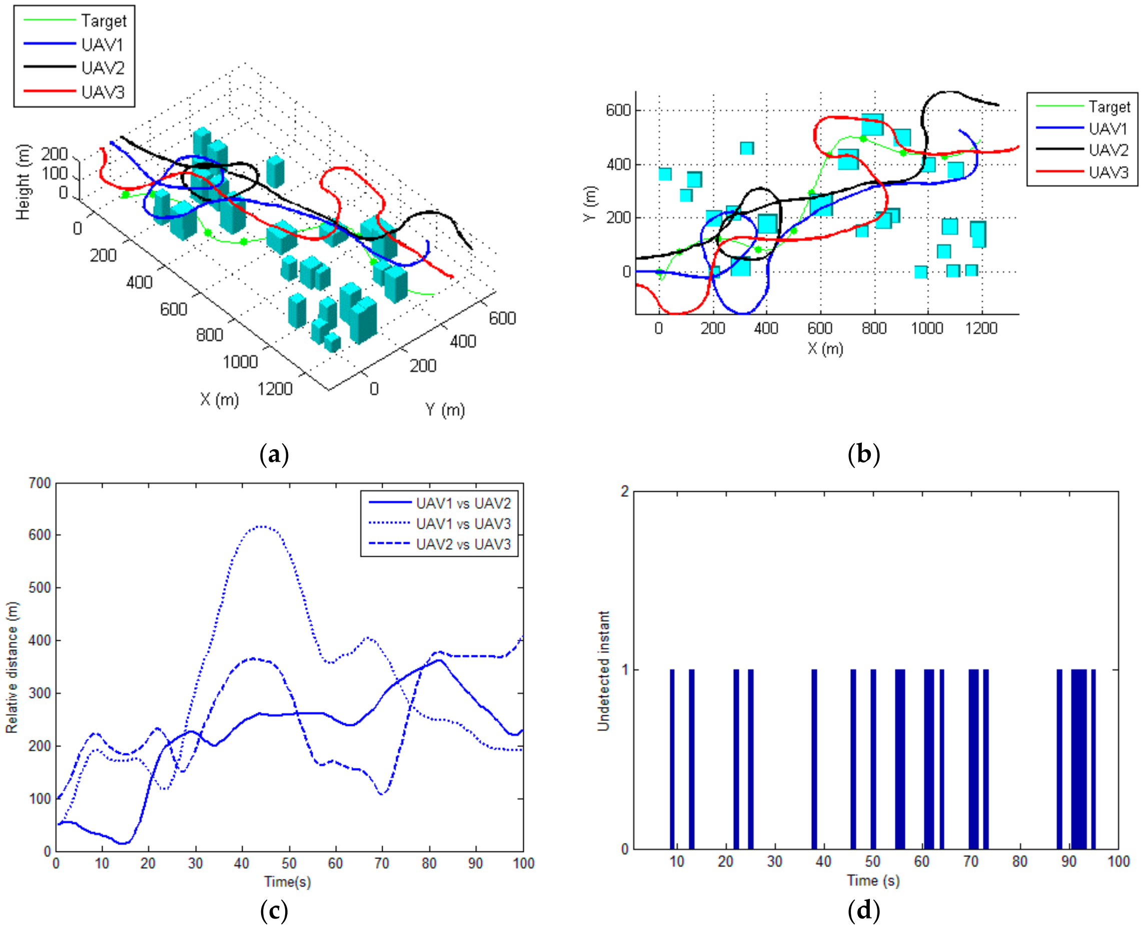
(**c**) (**d**)

Figure 9. The performances of using different evolutionary optimizers for determination of the tracking path for scenario 2. (**a**) and (**b**) illustrate the convergence of different methods in selected decision time, respectively; (**c**) shows the visibility time of the ground time for the proposed method, PSO and GA based algorithms, respectively; (**d**) is the average execution time of the aforementioned methods.



(**c**) (**d**)

Figure 10. The results obtained by using LGVF based approach for scenario 1. (**a**) 3D view of the planned path for each UAV; (**b**) Top view of the resulting paths; (**c**) The relative distance between each UAV during tracking; (**d**) The undetected time-instant.



(**a**) (**b**)

(**c**) (**d**)

Figure 11. The results obtained by using LGVF based approach for scenario 2. (**a**) 3D view of the planned path for each UAV; (**b**) Top view of the resulting paths; (**c**) The relative distance between each UAV during tracking; (**d**) The undetected time-instant.

Table 2. Statistics results of the tracking performances for scenario 1.

Performance	Proposed Method	LGVF-Based Algorithm
Mean visibility time (s)	95	82
Maximum visibility time (s)	96	79
Minimum visibility time (s)	91	84
Variation of visibility time (s)	4	3
Range of relative distances (m)	[51, 588]	[11, 548]

Table 3. Statistics results of the tracking performances for scenario 2.

Performance	Proposed Method	LGVF-Based Algorithm
Mean visibility time (s)	94	83
Maximum visibility time (s)	97	81
Minimum visibility time (s)	91	89
Variation of visibility time (s)	4	7
Range of relative distances (m)	[54, 608]	[15, 611]

5. Conclusions

Tracking moving targets using UAVs is a challenging task due to the uncertainties arising from the motion of the target and the environmental features. This paper presents a novel method, coupling the concept of model predictive control into the framework of chemical reaction optimization, to solve the coordinated tracking path planning problem in complex urban environments. The sensor visible regions under urban environment, flight dynamics and anti-collision constraints are considered. Particularly, the LoS occlusion caused by dense buildings is discussed, and the FOV for a gimbaled sensor is also analyzed. A series of comparative simulations have shown that the proposed method outperforms other metaheuristic and mathematical methods in terms of tracking ability and safety, while achieving similar computational efficiency. The simulation results imply that the proposed method is capable of improving the performance of target tracking in urban environments, with more visible time, higher security and stronger robustness compared to other methods.

Acknowledgments: This work was funded partially by the Opening Foundation of State Key Laboratory of Virtual Reality Technology and Systems of China under Grant No. BUAA-VR-15KF-11 and National Natural Science Foundation (NSFC) of China under Grant No. 61503185.

Author Contributions: For research articles with several authors, Yin Wang conceived and designed the experiments and wrote this paper. Yin Wang and Yan Cao performed the simulation and analyzed the results.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Horri, N.M.; Dahia, K. Model Predictive Lateral Control of a UAV Formation Using a Virtual Structure Approach. *Int. J. Unmanned Syst. Eng.* **2014**, *2*, 1–11. [CrossRef]
2. Yang, K.; Sukkarieh, S. 3D smooth path planning for a UAV in cluttered natural environments. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 794–800.
3. Belkhouche, F. Reactive Path Planning in a Dynamic Environment. *IEEE Trans. Robot.* **2009**, *25*, 902–911. [CrossRef]
4. Cheng, P. Time-optimal UAV trajectory planning for 3D urban structure coverage. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 22–26 September 2008; pp. 2750–2757.
5. Quintero, P.; Papi, F.; Klein, D.J.; Chisci, L.; Hespanha, J.P. Optimal UAV coordination for target tracking using dynamic programming. In Proceedings of the 49th IEEE Conference on Decision and Control, Atlanta, GA, USA, 15–17 December 2010; pp. 4541–4546.
6. Chen, H.D.; Chang, K.C.; Agate, C.S. UAV path planning with Tangent-plus-Lyapunov vector field guidance and obstacle avoidance. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 840–856. [CrossRef]
7. Shaferman, V.; Shima, T. Unmanned aerial vehicles cooperative tracking of moving ground target in urban environments. *J. Guid. Control Dyn.* **2008**, *5*, 1360–1371. [CrossRef]
8. Yu, H.; Beard, R.W.; Argyle, M.; Chamberlain, C. Probabilistic path planning for cooperative target tracking using aerial and ground vehicles. In Proceedings of the 2011 American Control Conference, San Francisco, CA, USA, 29 June–1 July 2011; pp. 4673–4678.
9. Zhang, C.G.; Xi, Y.G. Robot path planning in globally unknown environments based on rolling windows. *Sci. China* **2001**, *2*, 131–139.
10. Beard, R.W.; McLain, W.; Nelson, B. Decentralized Cooperative Aerial Surveillance Using Fixed-Wing Miniature UAVs. *Proc. IEEE* **2006**, *7*, 1306–1324. [CrossRef]
11. He, Z.; Xu, J.X. Moving target tracking by UAVs in an urban area. In Proceedings of the IEEE International Conference on Control and Automation, Hangzhou, China, 12–14 June 2013; pp. 1933–1938.
12. Besada-Portas, E.; de la Torre, L.; de la Cruz, J.M.; de Andrés-Toro, B. Evolutionary Trajectory Planner for Multiple UAVs in Realistic Scenarios. *IEEE Trans. Robot.* **2010**, *26*, 619–634. [CrossRef]
13. Wang, L.; Su, F.; Zhu, H.; Shen, L. Active sensing based cooperative target tracking using UAVs in an urban area. In Proceedings of the International conference on Advanced Computer Control, Shenyang, China, 27–29 March 2010; Volume 2, pp. 486–491.
14. Vincent, R.; Mohammed, T.; Gilles, L. Comparison of Parallel Genetic Algorithm and Particle Swarm Optimization for Real-Time UAV Path Planning. *IEEE Trans. Ind. Inform.* **2013**, *9*, 132–141.

15. Wise, R.A.; Rysdyk, R.T. UAV coordination for Autonomous Target Tracking. In Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Keystone, CO, USA, 21–14 August 2006; pp. 1–22.
16. Lam, A.S.; Li, V.K. Chemical Reaction Optimization: A tutorial. *Mim. Comput.* **2012**, *4*, 3–17. [CrossRef]
17. Gal, O.; Doytsher, Y. Fast and Accurate Visibility Computation in a 3D Urban Environment. In Proceedings of the Fourth International Conference on Advanced Geographic Information Systems, Applications, and Services, Valencia, Spain, 30 January–4 February 2012; pp. 105–110.
18. Yao, P.; Wang, H.; Su, Z. Real-time path planning of unmanned aerial vehicle for target tracking and obstacle avoidance in complex dynamic environment. *Aerosp. Sci. Technol.* **2015**, *47*, 269–279. [CrossRef]