



Discrete Optimization

Reconfiguration of satellite orbit for cooperative observation using variable-size multi-objective differential evolution

Yingguo Chen^a, Vladimir Mahalec^b, Yingwu Chen^{a,*}, Xiaolu Liu^a, Renjie He^a, Kai Sun^a^a College of Information System and Management, National University of Defense Technology, Changsha 410073, PR China^b School of Computational Engineering and Science, McMaster University, 1280 Main Street West, Hamilton, Ontario L8S 4L7, Canada

ARTICLE INFO

Article history:

Received 13 August 2013

Accepted 21 September 2014

Available online 10 November 2014

Keywords:

Satellite orbit reconfiguration

Variable-size optimization

Multi-objective differential evolution

Evolutionary computations

Estimation of Distribution Algorithm

ABSTRACT

A novel self-adaptive variable-size multi-objective differential evolution algorithm is presented to find the best reconfiguration of existing on-orbit satellites for some particular targets on the ground when an emergent requirement arises in a short period. The main contribution of this study is that three coverage metrics are designed to assess the performance of the reconfiguration. Proposed algorithm utilizes the idea of fixed-length chromosome encoding scheme combined with expression vector and the modified initialization, mutation, crossover and selection operators to search for optimal reconfiguration structure. Multi-subpopulation diversity initialization is adopted first, then the mutation based on estimation of distribution algorithm and adaptive crossover operators are defined to manipulate variable-length chromosomes, and finally a new selection mechanism is employed to generate well-distributed individuals for the next generation. The proposed algorithm is applied to three characteristically different case studies, with the objective to improve the performance with respect to specified targets by minimizing fuel consumption and maneuver time. The results show that the algorithm can effectively find the approximate Pareto solutions under different topological structures. A comparative analysis demonstrates that the proposed algorithm outperforms two other related multi-objective evolutionary optimization algorithms in terms of quality, convergence and diversity metrics.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

A distributed satellite system (DSS) may be needed to perform cooperative observation to satisfy the mission requirements in the case of emergency, because the characteristics of temporary reconnaissance missions, such as earthquake relief, flood monitoring and terrorism combat, launch of a new satellite may be uneconomic. Fortunately, there are hundreds of satellites carrying out various tasks in low Earth orbit. If necessary, the existing on-orbit satellites can be optimally reconfigured as a generalized DSS which performs cooperative observation of some particular targets during a specified time period. It is desirable in most cases to expand the performance of this generalized DSS by orbital maneuvers under affordable cost.

Observing some given targets within an assigned time frame is an orbit design problem. The initial orbit design can be solved from ground track points (Abdelkhalik, 2010; Abdelkhalik & Gad, 2011; Vtipil & Newman, 2010; Wall & Conway, 2009). An optimization problem in satellite constellation was constructed to pursue better revisit time (RT) or resolution according to the user requirements (Chen,

Mahalec, Chen, He, & Liu, 2013; Williams, Crossley, & Lang, 2001). The number of satellites in the above orbit designs is predefined, but it may be a design variable in the satellite reconfiguration case. Therefore, it is necessary to optimize the number of satellites and specify which satellites from an alternative set to perform orbital maneuver. This is a variable-size optimization problem because the number of satellites defines the state vector length during optimization. Moreover, reconfiguration is a typical multi-objective optimization problem that requires a trade-off between the multi-mission metrics. In this study, the reconfiguration of on-orbit satellites aims to improve the coverage metrics during a short period. To avoid drastically shortening the lifetime of the satellites, the reconfiguration only relates to changing the relative locations of satellites on the initial orbit.

The contribution of this study is twofold. First, three types of coverage metric are designed to assess the performance of reconfiguration: the average revisit time (ART) for single target, the total coverage time (TCT) for multi-targets and the coverage statistics based on task scheduling (CSTS) which measures performance with respect to user requirements. Then satellite reconfiguration is formulated as a multi-objective optimization problem that considers coverage metrics, fuel consumption, and maneuver time. Second, the fixed-length encoding scheme combined with expression vector in chromo-

* Corresponding author. Tel.: +86 13739072834.

E-mail address: ygchen@nudt.edu.cn (Y. Chen).

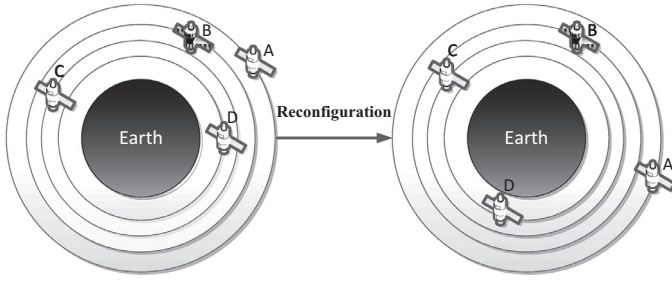


Fig. 1. Concept of on-orbit satellite orbit reconfiguration.

some building operator, multi-subpopulation diversity initialization, mutation based on estimation of distribution algorithm (EDA) (Ferringer & Spencer, 2006), adaptive crossover and modified selection operators are integrated into the multi-objective differential evolution to manipulate variable-length chromosomes and finally optimize the topological structure (satellite combination) and component values (maneuver variables of each satellite involved in reconfiguration) simultaneously.

The effectiveness of the proposed algorithm is verified by three case studies. The comparison with two other related multi-objective optimization algorithms demonstrates that the proposed algorithm performs better in terms of quality, convergence and diversity metrics defined on Pareto front.

2. On-orbit satellite reconfiguration

2.1. Background

Reconfiguration which is traditionally used in satellite constellation refers to implementing a series of essential maneuvers to maintain or improve the performance if some satellites fail (Larrañaga & Lozano, 2002). Various reconfiguration issues have been solved, such as optimal reconfiguration and formation-keeping for formation flying satellites (de Weck, Scialom, & Siddiqi, 2008); optimal reconfiguration of satellite formation to continue operation after loss of one satellite (Park, Park, & Choi, 2011); reconfiguration of GPS constellations after loss of capacity (Ahn & Spencer, 2002). A general framework was proposed to transform an initial constellation A into constellation B and was employed in communication satellites (Larrañaga & Lozano, 2002). In these examples, a reconfiguration is accomplished via orbital maneuvers of on-orbit satellites from a constellation with a known number of maneuvered satellites.

In this study, it is not required that the satellites involved in reconfiguration are from the same constellation. These satellites, which form a more generalized DSS for a temporary mission in a short period, may have been launched for performing specific tasks at different phases. They lie in different orbits with no active control to maintain relative positions. Although one or more classical orbital elements of a satellite can be changed to satisfy the mission requirements, the satellite in most cases is kept close to its initial orbit during the whole lifetime after it arrives at its mission orbit. Orbital maneuvers, especially non-coplanar maneuvers, are seldom carried out; otherwise the lifetime of the satellite will be drastically shortened.

An economic approach of reconfiguration is to rearrange the relative positions through phase changing in order to obtain a desired performance in a specified period. In the case of a modest transfer time, phase changing may require an acceptable fuel cost (Ferringer, Spencer, & Reed, 2009). Fig. 1 visualizes the concept of reconfiguration. It is assumed that there are four alternative satellites in different orbits providing the desired performance for some specific targets. The number of satellite combinations for orbital maneuver are 2^4 . If satellites A, C and D are chosen to rearrange their relative positions

from satellite B which is kept on its original orbit, the four satellites form a generalized DSS to perform cooperative observation.

In this study, satellite reconfiguration is defined as “adjusting the relative locations of multiple on-orbit satellites through changing the mean anomaly”. Not only the best combination but also the optimal maneuver values for each satellite should be computed with the design goal of maximizing the performance with minimum total fuel consumption and minimum maneuver time.

2.2. Computation of objective functions

The following notations are defined prior to the presentation of the objective functions.

Parameters

a_{tgt} semimajor axis of the target orbit

a_{phase} semimajor axis of the phase orbit

H orbit height

K the number of satellites in reconfiguration

K_{int} revolutions of satellite on phasing orbit

K_{tgt} revolutions of satellite on target orbit

$Lat_T, Long_T$ the target's latitude and longitude

$Lat_{SSP}, Long_{SSP}$ the sub-satellite point's latitude and longitude

m^i the number of gap durations of satellite i

MT_{total} total maneuver time

P the performance of the satellite system

R_e equatorial radius of the Earth

t_0 initial time of the scene

w_i priority weight for target i

w_p penalty weight

w_{tgt} angular velocity of the target orbit

w_s^{ij}, w_e^{ij} j th start and end time of coverage window for satellite i

t_s^{ij}, t_e^{ij} j th start and end time of gap duration for satellite i

α_i the target i on the schedule ($=1$) or not ($=0$)

λ Earth central angle

η^{\max} half maximum angle of field of view

η half current angle of field of view

ρ angular radius of the Earth

μ gravitational parameter

θ_0 angle of the target is ahead or behind of the interceptor

ν true anomaly

Δt step length of satellite propagation

ΔV_{total} total velocity changes

2.2.1. Coverage metrics

Coverage metrics assess the satellite-provided performance for a set of targets. Three types of coverage metric are used in this study: average revisit time (ART), total coverage time (TCT) and coverage statistics based on task scheduling (CSTS). Visibility between satellites and targets need to be determined so as to compute these coverage metrics.

Visibility depends on many factors, such as the location of the target, the number of satellites in the DSS, and each satellite's

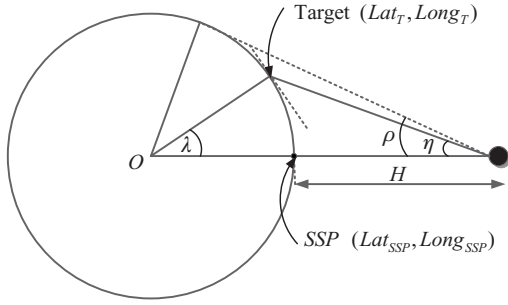


Fig. 2. Geometry of FOV.

orbit elements and field of view (FOV). ART and TCT are statistically distributed parameters that cannot be easily characterized (Ferrer et al., 2009), especially when the computation involves several satellites. Therefore, a numerical simulation considering the J2 perturbation (James, David, & Jeffery, 2011, chap. 10) is carried out to compute visibility. The target is visible to a satellite when it lies in the FOV (Fig. 2).

The Earth's angular radius ρ is defined as

$$\sin \rho = \frac{R_e}{R_e + H} \quad (1)$$

where R_e is the Earth's radius, and H is the orbit height. The half angle of FOV, η^{\max} , is measured at the satellite from sub-satellite point (SSP) to the acceptable fastest target restrained by the elevation angle. We define the geocentric coordinates of the target and SSP as $(Lat_T, Long_T)$ and $(Lat_{SSP}, Long_{SSP})$ respectively, then λ can be given as (Ferrer et al., 2009)

$$\lambda = \arccos [\sin Lat_T \sin Lat_{SSP} + \cos Lat_T \cos Lat_{SSP} \cos \Delta L] \quad (2)$$

where $\Delta L = Long_T - Long_{SSP}$. If $\lambda > 90$ degrees, the target is invisible, and if $\lambda \leq 90$ degrees, the angle η which is measured at the satellite from the SSP to the current target is computed as follows (Ferrer et al., 2009)

$$\eta = \arctan \left[\frac{\sin \rho \sin \lambda}{(1 - \sin \rho \cos \lambda)} \right] \quad (3)$$

(1) Average revisit time (ART)

The revisit time measures the gap during which the coverage is not provided, and ART is the average duration of all gaps. For satellite i at time t , the SSP from the ground track is combined to check if $\eta_t^i > \eta_i^{\max}$. If yes, the target is invisible current, and thus time t is considered as the start time of gap duration t_s^i of satellite i . At time $t = t_0 + \Delta t$, this process is repeated until the target lies in the satellite's FOV at time t_e^i . This time is considered as the end time of gap duration. If there are K satellites, ART is calculated by accumulation of the m^i gap durations $[t_s^{ij}, t_e^{ij}] (j = 1, \dots, m^i)$ for satellite i :

$$ART = \frac{\sum_{i=1}^K \sum_{j=1}^{m^i} (t_e^{ij} - t_s^{ij})}{\sum_{i=1}^K m^i} \quad (4)$$

(2) Total coverage time (TCT)

TCT measures the amount of time during which the targets are covered. For target i at time t , if $\eta_t^i \leq \eta_i^{\max}$, then t is marked as the start time of coverage window, w_s . At $t = t + \Delta t$, the position of a satellite is updated from the ground track and the computation is repeated until the target is out of FOV, and the corresponding time is marked as the end time of coverage window, w_e . Accumulating p^i coverage windows yields the TCT for K satellites:

$$TCT = \sum_{i=1}^K \sum_{j=1}^{p^i} (w_e^{ij} - w_s^{ij}) \quad (5)$$

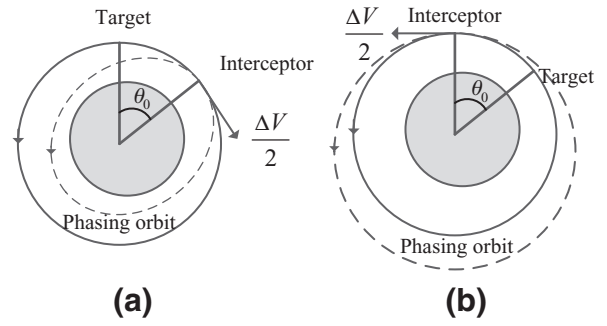


Fig. 3. Illustration of (a) interceptor trailing target and (b) interceptor leading target.

(3) Coverage statistics based on task scheduling (CSTS)

In some cases, it is desirable that the reconfiguration satisfies more than one mission requirement, such as requirement that the higher priority targets should be observed for longer period, that observation time must be at least equal to the threshold observation time, or that each target can be observed at least once. The computation of this coverage metric needs to employ task scheduling (Vallado, 2007). Task scheduling is defined as “creating a schedule which selects exact start and stop observation time for each target based on $[w_s^{ij}, w_e^{ij}]$ ”. In this study, priority dispatch (Vallado, 2007) is applied to create the schedule and then the objective function is defined as:

$$CSTS = \sum_{i=1}^n \left[w_i \alpha_i \left(\sum_{j=1}^{n_i} Duration_{bonus}^j \right) - w_p (1 - \alpha_i) \right] \quad (6)$$

where n is the number of original targets, and n_i is the number of the recurring observations of target i ; $w_i = [(largestPriority - smallestPriority + 1)/target_iPriority]$ is the priority weight for target i ; $Duration_{bonus}^j = Duration_{actual}^j / Duration_{desired}^j$ rewards the cases where the actual duration is longer than the threshold observation duration. A penalty term, $w_p (1 - \alpha_i)$, aims to guarantee each target can be observed at least once; the penalty weight is w_p creates a high cost for the targets which are not on the schedule ($\alpha_i = 0, \alpha_i \in \{0, 1\}$).

2.2.2. Co-orbital rendezvous

Orbital maneuver in this study focuses on how to move a satellite back and forth in the same plane. This is a co-orbital rendezvous problem in which two satellites are located in the same orbit and one satellite maneuvers its orbit through two-impulse Hohmann transfer (James et al., 2011, chap. 10) to catch up with the other one. There are two cases of rendezvous: interceptor trailing target and interceptor leading target (James et al., 2011, chap. 10) (Fig. 3).

Co-orbital rendezvous only changes the mean anomaly of the satellite in a circular orbit. If there are several satellites, the performance can be effectively improved by reasonable arrangement of the satellites' relative positions. In execution of an orbital maneuver, two important factors to be considered are the amount of energy consumption (or velocity changes, ΔV) and the total maneuver time. In the case of interceptor trailing target, the target is assumed to be ahead of the interceptor by an amount θ_0 , so the target must travel an angle of θ in K_{igt} revolutions to arrive the interceptor's current position. θ is calculated as

$$\theta = K_{igt}(2\pi) - \theta_0 \quad (7)$$

If the interceptor is ahead of the target by an amount θ_0 , then

$$\theta = K_{igt}(2\pi) + \theta_0 \quad (8)$$

In either case, the target orbit's angular velocity w_{tgt} is expressed as:

$$w_{tgt} = \sqrt{\frac{\mu}{a_{tgt}^3}} \quad (9)$$

where μ is the Earth's gravitational parameter. The total maneuver time MT_{total} of the K satellites involved in reconfiguration is given by

$$MT_{total} = \sum_{i=1}^K MT_i = \sum_{i=1}^K \frac{\theta^i}{w_{tgt}^i} \quad (10)$$

For satellite i , the maneuver time is equal to the K_{int}^i periods of the phase orbit, and thus the semimajor axis of the phase orbit can be defined as

$$a_{phase}^i = \left(\mu \left(\frac{MT_i}{2\pi K_{int}^i} \right)^2 \right)^{1/3} \quad (11)$$

The phase orbit is elliptical. The necessary ΔV s for the rendezvous can be computed when the semimajor axis of the orbit is known. The first ΔV changes the orbit from the initial orbit to the phase orbit, and the second ΔV makes the satellite reenter the initial orbit. Regardless of θ_i being ahead or behind the target point, the ΔV s of the two burns are of the same magnitude. Therefore, the velocity changes of orbital maneuver ΔV_{total} for all satellites can be given by (James et al., 2011, chap. 10)

$$\Delta V_{total} = \sum_{i=1}^K \Delta V_i = \sum_{i=1}^K 2 \left| \sqrt{\frac{2\mu}{a_{tgt}^i}} - \sqrt{\frac{\mu}{a_{phase}^i}} - \sqrt{\frac{\mu}{a_{tgt}^i}} \right| \quad (12)$$

Eqs. (10) and (12) reveal that if the integer value of K_{tgt} or K_{int} increases, then MT_{total} increases and ΔV_{total} decreases. Usually a large difference between K_{tgt} and K_{int} will lead to a long maneuver time. It is reasonable to assume that K_{tgt} is equal to K_{int} . Noticeably, the perigee radius of the phase orbit should not be smaller than the Earth's radius when adjusting K_{tgt} (K_{int}).

2.3. General formulation of multi-objective optimization

Given some specified targets on the ground, the on-orbit satellites can be reconfigured by changing the mean anomaly (ν) and K_{tgt} such that the DSS obtains the optimal performance with minimum ΔV_{total} and MT_{total} . Therefore, the multi-objective problem is formulated as follows:

$$\begin{aligned} &\text{Maximize: } P(\nu) \\ &\text{Minimize: } \{\Delta V_{total}(\nu, K_{tgt}, K_{int}), MT_{total}(\nu, K_{tgt}, K_{int})\} \\ &\text{S.T.: } K_{tgt} = K_{int} \end{aligned} \quad (13)$$

where ν and K_{tgt} are decision vectors; ν is a continuous variable and K_{tgt} is a discrete variable; $P(\nu)$ is the performance of the reconfiguration, where maximizing $P(\nu)$ means minimizing $ART(\nu)$, or maximizing $TCT(\nu)$ and $CSTS(\nu)$. The reconfiguration optimization aims to investigate the effects of satellite combinations on cooperative observation. The three objective functions often conflict with one another, and thus a Pareto front should be identified, in which no candidate solution is dominated by other solutions.

3. Variable-size multi-objective differential evolution (VSMODE)

Evolutionary algorithms have been shown to be effective in conventional satellite orbit design (Wolfe & Sorensen, 2000) or satellite reconfiguration (Ahn & Spencer, 2002; de Weck et al., 2008). DE is a population-based optimization algorithm introduced by Storn and Price (Kim, Bang, & Jung, 2008). The classical DE method consists of four basic steps:

(i) Randomly generate NP individuals (each individual is presented by a chromosome) in the design space. For example, the initial population ($g = 0$) of dimension j in individual i is generated as:

$$X_{ij}(0) = X_{ij}^L + \text{rand}_j(0, 1)(X_{ij}^U - X_{ij}^L) \quad i = 1, \dots, NP \quad (14)$$

(ii) Produce individual $U_{ij}(g)$ by the mutation (Eq. (15)) and crossover (Eq. (16)) operations according to the scaling factor F and the crossover rate CR :

$$V_i(g) = X_{i0}(g) + F(X_{i1}(g) - X_{i2}(g)), \quad (i0 \neq i1 \neq i2) \quad (15)$$

$$U_{ij}(g) = \begin{cases} V_{ij}(g) & \text{if } (\text{rand} \leq CR \text{ or } j = C_{rand}) \\ X_{ij}(g) & \text{otherwise} \end{cases} \quad (16)$$

where $V_i(g)$ is created by adding the scaled difference F between two randomly chosen vectors $X_{i1}(g)$ and $X_{i2}(g)$ to a randomly chosen base vector $X_{i0}(g)$; parameter C_{rand} in the Eq. (16) is a random index which avoids the trial individual is the duplicate of X_i .

(iii) Employ a selection operator (17) to replace the current individual by a new individual that has a higher fitness function (for maximum problem):

$$X_i(g+1) = \begin{cases} U_i(g) & \text{if } (f(U_{i,g}) \geq f(X_{i,g})) \\ X_i(g) & \text{otherwise} \end{cases} \quad (17)$$

(iv) Repeat this process until a stopping criterion is satisfied.

Experimental comparisons of DE with other evolutionary algorithms (genetic algorithms, and particle swarm optimization) have shown better performance both in robustness and convergence speed (Price, Storn, & Lampinen, 2005; Storn & Price, 1997). For reconfiguration optimization problem, multi-objective differential evolution (MODE) algorithms (Gong & Cai, 2009; Vesterstrom & Thomsen, 2004; Wang, Wu, & Yuan, 2010) are suitable search engines to obtain the Pareto front. An obvious manner of applying MODE for reconfiguration is to compare the difference between the optimal and the original values. If the difference is less than a threshold value, this satellite is not involved in the reconfiguration. However, this manner of application may raise problems, such as the limited exploring ability under different structures, or the difficulty in setting an appropriate threshold value in practice.

In the reconfiguration, the optimal number of satellites and the satellite to undergo orbital maneuver may be unknown. The problem hence involves a varying number of design variables and requires a search algorithm suitable for optimization on variable-length chromosomes. For a large satellite combination, it is inefficient and impractical to run the algorithm many times while changing the prescribed number of satellites until the optimal number is found.

A better solution is to synchronize the optimization of reconfiguration topology and component values. One way to deal with this issue is to adjust the length of chromosome by comparing the current function value with the current incumbent value (i.e. the best function value found so far) (Ali, Siarry, & Pant, 2012). For multi-objective optimization, however, it is difficult to decide whether to add or delete a component (a satellite need to maneuver in reconfiguration), because each combination may exist on the current Pareto front. A VSMODE algorithm containing problem-oriented operators is proposed to address this problem.

3.1. Modification of DE operators

3.1.1. Chromosome building operator

Since all components contain the same design variables, the fixed-length mechanism (Su et al., 2012) is employed to express variable-length chromosomes. Fixed-length does not mean the size of a problem is fixed, but that the population length is fixed to the maximum possible length of the problem. Then a K -dimensional expression vector $f = (f_1, f_2, \dots, f_K)$ with values of 0 or 1 is established. f determines

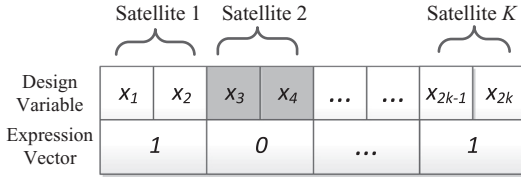


Fig. 4. The structure of chromosome building in reconfiguration.

the state of being expressed (=1) or unexpressed (=0) of the design variable, which finally determines the effective length of the solution. The general structure of chromosome building is shown in Fig. 4.

Only the expressed (or effective) component is evaluated in the optimization. For example, if satellite 2 is an unexpressed genome (Fig. 4), its initial values are kept during function evaluation. Noticeably, a position unexpressed in the current generation may be expressed in the next generation. Fixed-length mechanism combined with the expression vector makes the individual have the same length in all populations.

3.1.2. Multi-subpopulation diversity initialization

The concept of multi-subpopulation diversity initialization is to initialize a population that consists of multiple subpopulations with different effective lengths. All of the individuals in each subpopulation have the same effective length. In the absence of a priori information, each subpopulation has the same size. The successive steps of initialization are listed as follows:

- 1) Divide the population (NP) into $Nsub$ subpopulations ($Nsub$ is equal to the number of satellites K). Create all subpopulations to be equal size of the individual: $Nsubpop = NP / Nsub$.
- 2) For $i = 1: NP$, randomly generate design variables according to Eq. (14).
- 3) For $j = 1: Nsub$, generate expression vector (0 or 1) in subpopulation j which enables the sum of the expression vector to be equal to the effective length j .

The above processes actually initialize two sets of variables: the design variables aim to optimize the values of each satellite, while the expression vector aims to optimize the structure of the reconfiguration. The subpopulations will change size during later stage to produce more fit members.

3.1.3. Adaptive mutation operator

The mutation operator only deals with design variables. Classical mutation operator (Eq. (15)) is performed at the same position among the randomly selected vectors ($X_{i1}(g)$, $X_{i2}(g)$ and $X_{i0}(g)$) and it is unable to extract or use the global information of the design space. In satellite reconfiguration, a novel and efficient adaptive mutation inspired by Estimation of Distribution Algorithm (EDA) for producing promising individual is employed. EDA directly extracts the global statistical knowledge from the current generation and builds a probability model to generate a new population. In this study, the K components or $2K$ design variables are independent. Then the optimal NP individuals from the current population are selected and the joint density function of the g th generation based on Gaussian mixture distribution is

$$p(\mathbf{x}(g)) = \prod_{i=1}^{2K} p(x_i(g)) \quad (18)$$

where $p(x_i(g))$ is the margin density function of the i th dimension:

$$p(x_i(g)) = \frac{1}{NP} \sum_{j=1}^{NP} N(\mu_{ij}^g, \sigma_{ij}^g) \quad (19)$$

where $N(\mu_{ij}^g, \sigma_{ij}^g)$ is normal distribution. The mean, μ_{ij}^g , and the standard deviation, σ_{ij}^g can be estimated as follows:

$$\mu_{ij}^g = \frac{1}{NP} \sum_{i=1}^{NP} x_{ij}(g), \quad \sigma_{ij}^g = \sqrt{\frac{1}{NP} \sum_{i=1}^{NP} (x_{ij}(g) - \mu_{ij}^g)^2} \quad (20)$$

After the probability model is constructed, the adaptive mutation can be described as follows: if (r and $(0, 1) < \delta$), the EDA scheme is employed to produce new vector, otherwise the classical scheme described in Eq. (15) is employed. The initial value of δ ($0 \leq \delta \leq 1$) is 0.5, and then δ is equal to the proportion how many individuals on Pareto front are from the EDA scheme. The EDA-based adaptive mutation can combine differential information and global knowledge, thus avoiding the local convergence of one component and improving the exploration capability.

3.1.4. Adaptive crossover operator

The crossover operator performs not only on design variables, but also on expression vector. The crossover on design variables determines whether the element of a trial vector is from the mutant vector or the target vector (Eq. (16)). The crossover on expression vector will change the effective length. A two-part adaptive strategy is adopted. First, the effective length set ($L_{effective}$) on the Pareto front is obtained. If the difference set $K - L_{effective}$ ($K = \{1, 2, \dots, K\}$) is not null, one expression vector with effective length in the difference set is randomly generated to replace the original expression vector. Second, experiments show that it is difficult to find the optimal structure when the effective length is close to the maximum, because a subpopulation with longer effective length needs more individuals for better convergence. Hence, the adaptive crossover on expression vector is

$$\begin{cases} E(f_{U_i}(g)) = \text{random } k & \text{if } (K - L_{effective} \neq \emptyset \text{ and } k \in K - L_{effective} \text{ and } rand_j \leq CR_i(g)) \\ f_{U_{ij}}(g) = \begin{cases} 1 & \text{elseif } rand_j \leq CR_i(g) \\ f_{X_{ij}}(g) & \text{otherwise} \end{cases} \end{cases} \quad (21)$$

where $E(f_{U_i}(g))$ is the effective length of expression vector for individual $U_i(g)$, and $f_{U_{ij}}(g)$ if the expression vector of individual $U_i(g)$ on dimension j ($j = 1, 2, \dots, K$). The adaptive crossover rate is set as follows:

$$CR_i(g) = \begin{cases} 2P_A CR \left(\frac{CurGen}{MaxGen} \right) - CR, & \text{if } CurGen > \frac{MaxGen}{P_A} \\ CR, & \text{if } CurGen < \frac{MaxGen}{P_A} \end{cases} \quad (22)$$

where CR is the predefined crossover rate, $CurGen$ is the current generation, $MaxGen$ is the maximum generation number and a large P_A means the adaptive strategy is adopted in the early stage of optimization. The adaptive crossover which enables the algorithm to pay more attention to the missing effective length on the Pareto front or the individuals with longer effective length achieves the probability increasing as the optimization proceeds. The crossover operation on design variable enables the algorithm to explore the locality independently, while the expression vector concentrates more on the global optimal reconfiguration structure.

3.1.5. Selection operator

VSMODE enters the selection phase after the crossover. The idea of selection in VSMODE originates from the fast non-dominated sorting, ranking, and elitism techniques utilized in non-dominated sorting genetic algorithm-II (NSGA-II) (Ryerkerk, Averill, Deb, & Goodman, 2012) but with some differences. In VSMODE, the trial individual is compared to the target individual, and if target individual is dominated by trial individual, trial individual replaces target individual immediately in the current population. This comparison process creates two populations: current population and dominated population.

Table 1
Orbital element of alternative satellites.

No.	<i>a</i> (kilometers)	<i>i</i> (degrees)	<i>e</i>	Ω (degrees)	ω (degrees)	ν (degrees)
1	7264.2	99.1	0.0017	263.1	25.9	334.3
2	7231.4	98.6	0.0015	279.1	124.3	236.0
3	7240.2	98.7	0.0013	309.4	276.1	83.9
4	6985.2	97.7	0.0012	358.7	109.6	250.7
5	7118.8	98.5	0.0014	350.5	32.4	327.8

The two populations are combined and called combined population. The newly created solution entering the population instantly takes part in the creation of new solutions. This is helpful to maintain the population diversity. Moreover, VSMODE also incorporates an effective elite-preserving and an explicit diversity-preserving strategy. The selected individuals are classified by the effective length (associated sub-population). The elite count, a small portion of the fittest individuals, is selected to be 10 percent in each sub-population, and guarantees that the fittest individuals of each effective length can survive to the next generation. Not all effective length can exist on the true Pareto front, so the elite-preserving strategy is only employed in the first half of generation.

3.2. Outline of VSMODE

The VSMODE is a heuristic algorithm, so the solutions obtained by the proposed algorithm are approximate Pareto solutions. The pseudo-codes of the proposed algorithm are listed as follows:

- 1 Initialize the values of parameters: NP , F , CR , $\delta = 0.5$, elite count, $\max Gen$ and the number of satellites K .
- 2 Generate NP individuals $\{X_1(0), X_2(0), \dots, X_{NP}(0)\}$ in search space using multi-subpopulation diversity initialization method described in Section B.
- 3 Evaluate objective functions at the NP individuals, find $f(X_i(0))$, $i = 1, 2, \dots, NP$, and build probability model $p(X(g))$.
- 4 Store NP solutions in the current population pop_1 .
- 5 $currentGen \leftarrow 1$
- 6 While ($currentGen < \max Gen$)//Main loop of VSMODE
- 7 For($i = 1$ to NP)//Iteration loop starts here
- 8 Two generation schemes in an adaptive mutation are employed only on design variables.
- If ($rand(0, 1) > \delta$)
- $V_i(g)$ is produced using Eq. (15).
- Else
- $V_i(g)$ is sampled from the probability model $p(X(g))$
- EndIf
- 9 Perform adaptive crossover both on design variable and expression vector using the operation described in Section D.
- 10 Perform selection on trial individual $U_{ij}(g)$ with target individual $X_i(g)$. If ($U_{ij}(g)$ dominates $X_i(g)$), replace $X_i(g)$ by $U_{ij}(g)$ immediately.
- 11 EndFor
- 12 Select NP fittest solutions using fast non-dominated sorting, ranking, and elitism techniques from the combined population.
- 13 Update parameters δ , CR and probability model $p(X(g))$.
- 14 EndWhile

If the complexity is defined as the total number of function value comparisons, the overall computational complexity of latest versions of Multi-Objective Evolutionary Algorithms (MOEAs) or other popular multi-objective algorithms (e.g. NSGA-II) is $O(N_{obj} \times 2(NP)^2)$ (Wang et al., 2010) in which N_{obj} is the number of objective functions. The increased complexity of one iteration in VSMODE caused

Table 2
Rang of design variables.

Variable	Lower bound	Upper bound	Type
ν	−180 degrees	180 degrees	Continuous
K_{tgt}	4	12	Discrete

by the problem-oriented operators (especially selection operation) is $O(N_{obj} \times NP)$ which is smaller than $O(N_{obj} \times 2(NP)^2)$ when NP is large, so the overall complexity of VSMODE can be considered to be $O(N_{obj} \times 2(NP)^2)$. Although VSMODE needs some extra computational cost in performing the problem-specific operators, such as sorting the population based on EDA or generating individuals using mutation and crossover operators, most computational costs come from function evaluation.

4. Case studies

This section presents a series of case studies of reconfiguration optimization that investigate how coverage metrics can be improved by employing the proposed algorithm.

A summary of five alternative satellites categorizing the initial orbital elements in terms of semimajor axis a , inclination i , eccentricity e , argument of perigee ω , right ascension of the ascending node Ω , and mean anomaly are listed in Table 1 more details of satellite database can be found from webpage <http://www.agi.com/products>. η_{\max} is assumed to be 15 degrees for all satellites.

The design variables and ranges for each satellite in the following reconfiguration scenario are summarized in Table 2. The bound of K_{tgt} is based on the consideration that the fuel consumption is very large if K_{tgt} is less than 4 revolutions, while the maneuver time is excessively long if K_{tgt} is more than 12 revolutions.

To demonstrate the effectiveness of the optimization mechanism (fixed-length mechanism combined with expression vector) and modified operators, we compare the proposed algorithm with two related multi-objective variable-size algorithms: 1) MODE (Wang et al., 2010) with the pattern to compare the optimal and original ν to determine the topological structure (denoted as MODE-C). The satellite is not involved in reconfiguration if the difference is less than δ times of original value; 2) replace the initialization, mutation, crossover and selection operators in VSMODE with that in MODE (denoted as VSMODE-R). Three performance measures are used to quantitatively assess the performances of these algorithms.

The first performance measure is the number of solutions contained in set A that are not dominated by any solution contained in set B, denoted as NNS.

$$NNS = |\{a \in A \mid \forall b \in B, b \text{ does not dominate } a\}| \quad (23)$$

NNS measures the quality of Pareto front. Obviously, the better algorithm produces more non-dominated solutions.

The second performance measure, denoted as C_p , is the convergence of solutions solved by each algorithm. If P_s is a set of uniformly distributed points along the ideal Pareto front in the objective space,

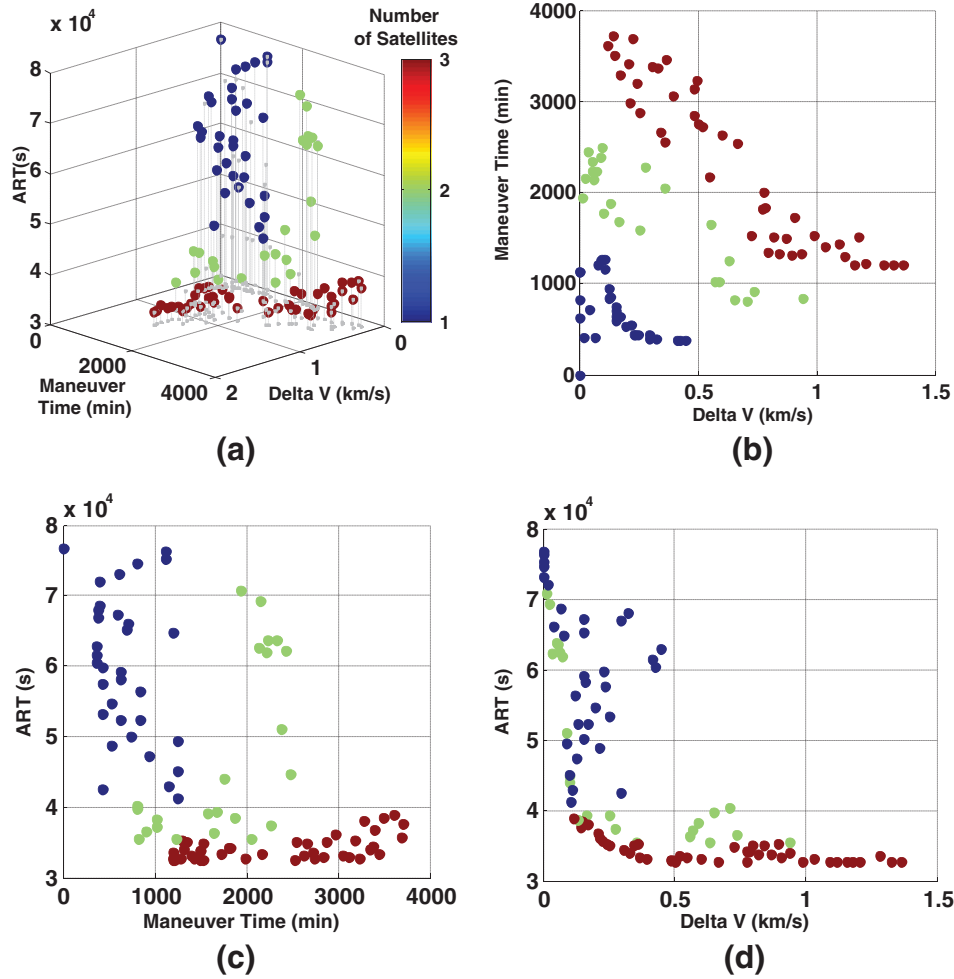


Fig. 5. Case study 1: Pareto front in 3D space (a) and three 2D planes (b, c, d).

the mathematical definition of C_P is (<http://www.agi.com/products/>):

$$C_P(P_a, P_*) = \frac{\sum_{\Delta \in P_*} d(\Delta, P_a)}{|P_*|} \quad (24)$$

where $d(\Delta, P_a)$ is the minimum Euclidean distance between Δ and the approximate Pareto front P_a . The ideal Pareto front P_* can be created by merging all non-dominated solutions produced by the compared algorithms.

The last performance measure, denoted as D_E , is the diversity of solutions on the effective length of expression vector. If $E = \{e_i | e_i \in [1, 2, \dots, K], i = 1, 2, \dots, NP\}$ is the effective length of the approximate Pareto front, then the elements of E with the same effective length are counted, denoted as set $E_j^s (j = 1, 2, \dots, K)$. D_E can be defined as the variance of E_j^s . A smaller D_E means that the optimal solutions spread out over different numbers of satellites involved in reconfiguration.

The following three case studies are selected to represent different coverage metrics. The proposed algorithm and other two related algorithms are implemented and compared under equal condition in each case study. The algorithm settings are: $NP = 100$ (case study 1), $NP = 150$ (case study 2 and 3), $MaxGen = 300$ (maximum number of generation is considered as stopping criterion), $CR = 0.4$, $F = 0.6$, $P_A = 2$ and $\delta = 0.5$. The elite count of each sub-population is 10 percent in the first half of generation. The Matlab implementation of VSMODE and the supplementary material of the case studies can be downloaded from the webpage <http://www.mongofiles.com/file/70108/VSMODE.zip#>.

4.1. Reconfiguration for reducing ART

In this case study, three alternative satellites (No. 1 to No. 3) are reconfigured to reduce the ART of a specified target (N31 degree/E103 degree) in a short period (2014/05/01 to 2014/05/15, 15 days). This example is designed to investigate the effect of satellite reconfiguration on the revisit time of a single target.

The original revisit time of this target using all the alternative satellites without any orbital maneuvers is 76,670 s. The goal of reconfiguration is to minimize the ART with the minimum total fuel consumption and maneuver time.

When the algorithm terminates after $MaxGen$ generations ($MaxGen = 300$), the solutions from the final generation approximate the problem's Pareto front. The Pareto front, as well as the corresponding number of satellites involved in orbital maneuver, is shown in Fig. 5. Apparently, different satellite combinations can be found by the proposed algorithm to constitute the non-dominated designs.

Three clusters are created if the approximate Pareto front is classified by the number of maneuvered satellites. Useful information can be obtained if we explore different satellite combinations: (1) the boundary is clear between different clusters; (2) each satellite has opportunity to join in reconfiguration in each cluster; (3) the solution with the minimum revisit time can usually be achieved by involving more satellites in reconfiguration, but at the cost of larger ΔV or maneuver time.

The problem has also been solved by MODE-C and VSMODE-R, and the comparative results on three performance metrics are shown in Table 3.

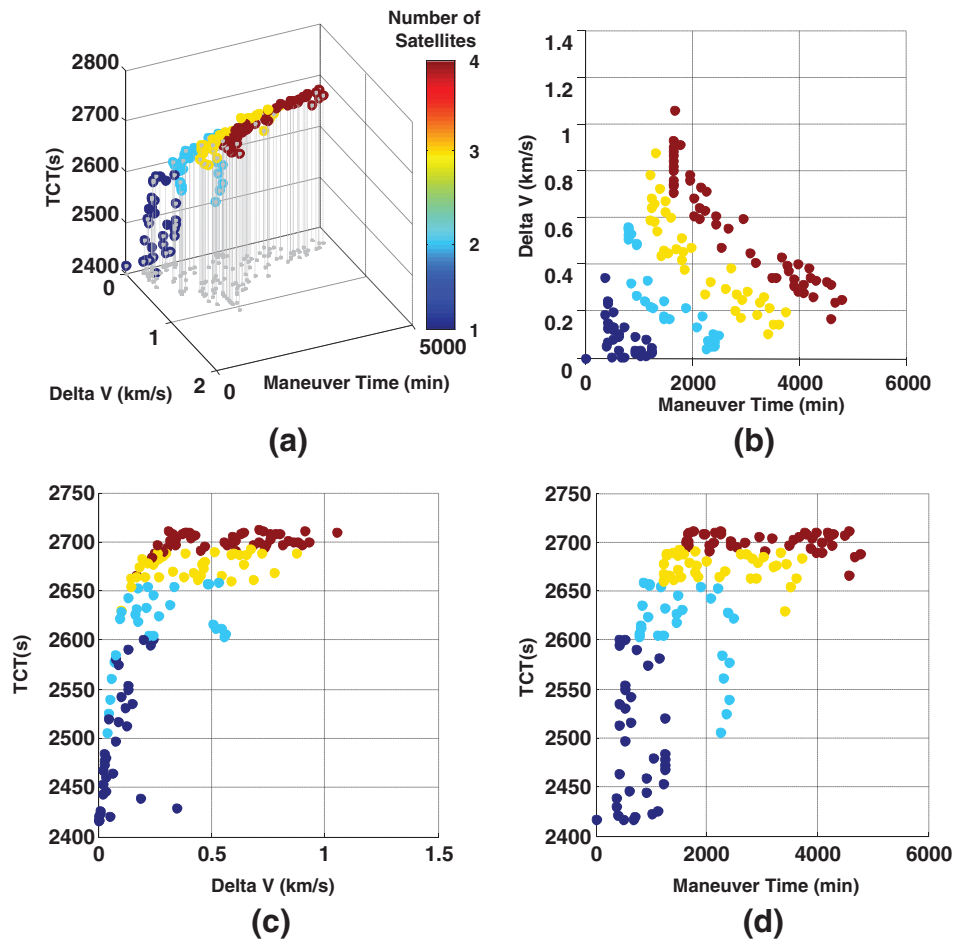


Fig. 6. Case study 2: Pareto front in 3D space (a) and three 2D planes (b, c, d).

Table 3

Case study 1: comparative results of the three algorithms.

	MODE-C	VSMODE-R	VSMODE
NNS	20	73	97
C_P	0.093	0.041	0.029
D_E	45.545	14.942	10.073

Table 5

Case study 2: comparative results of the three algorithms.

	MODE-C	VSMODE-R	VSMODE
NNS	5	118	132
C_P	0.221	0.065	0.030
D_E	53.792	9.535	8.654

4.2. Reconfiguration for improving TCT

Four alternative satellites (No. 1 to No. 4) are reconfigured to maximize the TCT of several point targets (listed in Table 4) in a period (2014/05/01 to 2014/05/7, 7 days).

The TCT of these targets is 2416 s without reconfiguration. The goal of reconfiguration is to maximize the TCT with the minimum total fuel consumption and maneuver time. The solutions from the final generation of the optimization, as well as the corresponding number of satellites involved in orbital maneuver, are shown in Fig. 6.

The results also indicate that the solutions providing the optimal TCT are achieved by involving more satellites in reconfiguration, but at the cost of larger ΔV or maneuver time. The solutions on the Pareto front are characterized by: (1) there are four clusters and the boundary is clear; (2) satellite 1 seldom performs maneuver in cluster 1 and 2; (3) satellite 2 always performs maneuver in cluster 3.

The problem has also been solved by MODE-C and VSMODE-R, and the comparative results with respect to three performance metrics are shown in Table 5.

Table 4

Case study 2: details of the observation targets.

Target id	Lat (degrees)	Long (degrees)	Target id	Lat (degrees)	Long (degrees)
1	39.91	116.39	6	15.55	32.53
2	52.52	13.33	7	49.27	−122.96
3	−15.79	−47.90	8	−25.73	28.22
4	−35.35	149.04	9	19.43	−99.13
5	33.72	73.06	10	−33.48	−70.65

Table 6
Case study 3: details of the observation targets.

Target id	Lat (degrees)	Long (degrees)	Priority	Minimum duration (seconds)
1	39.91	116.39	7	90
2	37.77	118.52	3	100
3	37.07	121.95	8	90
4	33.83	119.85	10	80
5	30.84	121.12	6	90
6	28.05	120.74	10	80
7	25.38	119.02	1	100
8	23.36	116.06	2	100
9	21.95	112.39	8	90
10	19.43	−99.13	3	100
11	52.52	13.33	9	80
12	−35.35	149.04	3	100
13	15.55	32.53	8	90
14	−25.73	28.22	7	90

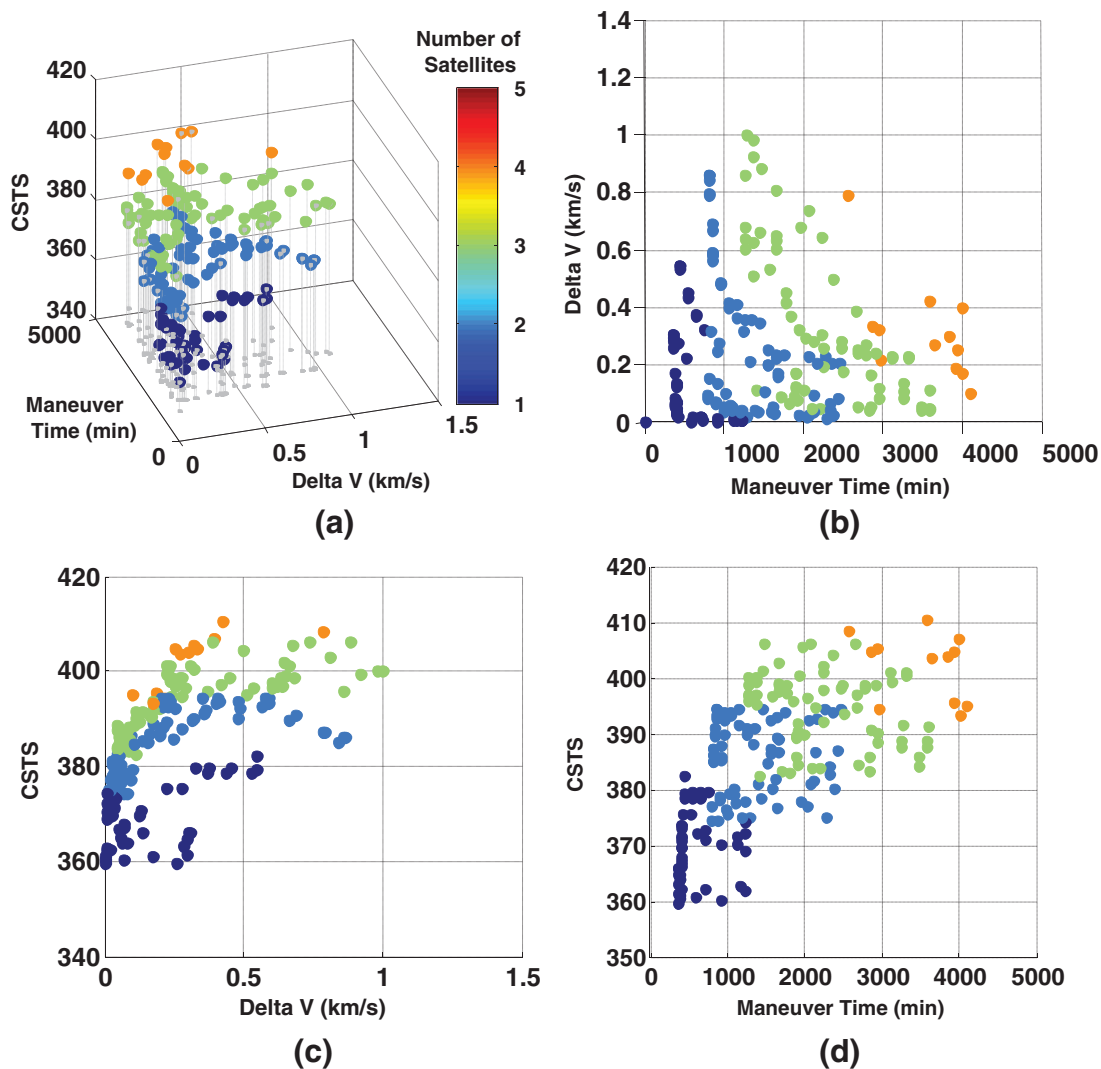


Fig. 7. Case study 3: Pareto front in 3D space (a) and three 2D planes (b, c, d).

4.3. Reconfiguration for improving CSTS

This case study reconfigures five alternative satellites (No. 1 to No. 5) to maximize the CSTS of multi-targets defined by different latitude, longitude, priority and minimal duration observation time (Table 6) in a short period (2014/05/01 to 2014/05/15, 15 days). This reconfiguration instance considers the task completion rate, the preference for higher priority targets and the user-specified duration.

The CSTS of these targets is 359.43 without reconfiguration. The solutions from the final generation of the optimization, as well as the corresponding number of satellites involved in orbital maneuver, are shown in Fig. 7.

The CSTS increases with the increase in the number of maneuvered satellites up to four satellites involved in reconfiguration while there are five alternative satellites. Moreover, the characteristics of these solutions are: (1) satellite 3 performs little to no maneuver;

Table 7
Case study 3: comparative results of the three algorithms.

	MODE-C	VSMODE-R	VSMODE
NNS	9	104	128
C_p	0.152	0.078	0.048
D_E	62.091	34.428	26.504

(2) although there are four clusters, most solutions that balance the three conflicting objectives are achieved by maneuvering two or three satellites; (3) there are total 2^5 combination patterns, and thus it is easy to understand the overlap between clusters.

The comparative results on three performance metrics are shown in Table 7.

4.4. Discussion and analysis of results

Three characteristically different scenarios are used to assess the proposed algorithm. Noticeably, the evaluation of coverage metrics is time-consuming and expensive. Hence, the goal of the proposed algorithm is to seek near-optimal solutions for a given problem. The results from above optimization indicate that different satellite combinations are found to balance the performance, propellant consumption and maneuver time. The improvement of performance is limited after a threshold value and even taking more propellant or maneuver time. Each cluster contains a set of solutions with the same effective length. The boundary between clusters is clear if the alternative satellites are limited. However, a large alternative set will result in overlap between clusters because the number of combinations increases exponentially.

The comparative results show that the proposed algorithm is generally more effective than other two algorithms. The key observations are:

- (1) VSMODE produces more non-dominated solutions and provides a higher degree of convergence and diversification of solutions compared to MODE-C and VSMODE-R.
- (2) VSMODE-R also works much better in terms of the three performance measures compared to MODE-C.
- (3) MODE-C is almost ineffective for variable-size optimization. It cannot find the solutions spreading out on different effective lengths even we adjust the threshold value δ .

The significant difference between the proposed VSMODE and the other two algorithms originates from the optimization mechanism and modified operators. The high performance of VSMODE can be attributed to three reasons. First, the optimization mechanism provides a localized effect in exploring different effective lengths of the search space around the potential candidates. Second, the multi-subpopulation diversity initialization makes sure that each subpopulation has the same opportunity to take part in reconfiguration. Its random nature in each subpopulation also helps to preserve the diversity. Third, as the search procedure progresses, VSMODE can avoid the loss of diversity in the population owing to its modified mutation and selection operators. Finally, the adaptive crossover can potentially increase the explore opportunity at longer effective length.

An average ΔV below 2 kilometers per second can be considered as a cheap reconfiguration. The above results show that the average ΔV in each solution is much less than 2 kilometers per second, indicating satellite reconfiguration by adjusting mean anomaly is feasible and effective way to improve coverage metrics. From the above results, some typical combination patterns on the Pareto front can be found and used as knowledge to guide search in the future. If the maximum generation and population size are equal for different algorithms, the computation costs are almost the same, because the

computation cost mainly comes from the fitness evaluation, rather than the modified operators. We have not focused on finding the best parameter setting for the algorithm and leave this task for a future study.

5. Conclusions

This work addresses the problem of reconfiguring on-orbit satellites for a temporary mission. Three objectives including coverage metrics, total fuel consumption and total maneuver time are considered. Since the reconfiguration topology and component values should be optimized simultaneously, a novel multi-objective optimization approach based on a fixed-length mechanism combined expression vector for implementing variable-size optimization has been constructed. This approach is based on multi-objective differential evolution (MODE) with three significant attributes:

- Initially, multi-subpopulation diversity initialization was employed to produce multiple subpopulations, and each subpopulation owns the same effective length.
- Then, adaptive mutation operator based on EDA was adopted to improve the exploration capability.
- Finally, the adaptive crossover was adopted with an increasing probability as the optimization proceeds, which performs not only on design variables, but also on expression vector to find the optimal component values under different topological structures.

Three case studies considering different coverage metrics were designed to evaluate the effectiveness of the proposed algorithm. The implementation demonstrated that VSMODE can effectively compute different satellite combinations to perform reconfiguration. The comparative results demonstrate that the proposed algorithm is better than other two variable-size algorithms. Moreover, the proposed algorithm not only provides a useful way to solve the reconfiguration problem, but also gives a framework for solving other variable-size problems.

Acknowledgements

This research has been supported by the [National Natural Science Foundation of China](#) (Grant No. 61473301 and 61203180) and Youth Foundation of China High Resolution Earth Observation (Grant No. GFZX04060103-5-18) and by McMaster University. The authors kindly thank the anonymous reviewers for the useful suggestions to improve the paper.

References

- Abdelkhalik, O. (2010). Initial orbit design from ground track points. *Journal of Spacecraft and Rockets*, 47(1), 202–205.
- Abdelkhalik, O., & Gad, A. (2011). Optimization of space orbits design for earth orbiting missions. *Acta Astronautica*, 68(7–8), 1307–1317.
- Y. T. Ahn, D. B. Spencer, Optimal reconfiguration of a formation-flying satellite constellation, in: 53rd international astronautical congress of the international astronautical federation, October 2002.
- Ali, M., Siarry, P., & Pant, M. (2012). An efficient Differential Evolution based algorithm for solving multi-objective optimization problems. *European Journal of Operational Research*, 217, 404–416.
- Chen, Y. G., Mahalec, V., Chen, Y. W., He, R. J., & Liu, X. L. (2013). Optimal satellite orbit design for prioritized multi-targets with threshold observation time using self-adaptive differential evolution. *Journal of Aerospace Engineering*. doi:10.1016/(ASCE)AS.1943-5525.0000393.
- Ferringer, M. P., & Spencer, D. B. (2006). Satellite constellation design tradeoffs using multiple-objective evolutionary computation. *Journal of Spacecraft and Rockets*, 43(6), 1404–1411.
- Ferringer, M. P., Spencer, D. B., & Reed, P. (2009). Many-objective reconfiguration of operational satellite constellations with the large-cluster epsilon non-dominated sorting genetic algorithm-II. *IEEE evolutionary computation congress*, 340–349.
- Gong, W., & Cai, Z. (2009). An improved multi-objective differential evolution based on Pareto-adaptive-dominance and orthogonal design. *European Journal of Operational Research*, 198, 576–601.

- James, R. W., David, F. E., & Jeffery, J. P. (2011). *Space mission engineering: The New SMAD*. Hawthorne, CA: Microcosm Press.
- Kim, H. D., Bang, H., & Jung, O. C. (2008). A heuristic approach to the design of an orbit for a temporary reconnaissance mission using a few LEO satellites. *Proceedings of the AAS/AIAA space flight mechanics meeting*, Galveston, USA.
- Larrañaga, P., & Lozano, J. A. (2002). Estimation of distribution algorithms. *A new tool for evolutionary computation*: 2 Springer.
- Park, H. E., Park, S. Y., & Choi, K. H. (2011). Satellite formation reconfiguration and station-keeping using state-dependent Riccati equation technique. *Aerospace Science and Technology*, 15(6), 440–452.
- Price, K., Storn, R., & Lampinen, J. (2005). *Differential evolution: A practical approach to global optimization*. New York: Springer-Verlag.
- Ryerkerk, M. L., Averill, R. C., Deb, K., & Goodman, E. D. (2012). Optimization for variable-size problems using genetic algorithms. *12th AIAA aviation technology, integration, and operations (ATIO) conference and 14th AIAA/ISSM*, Indianapolis, Indiana.
- Storn, R., & Price, K. (1997). Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4), 341–359.
- Su, Z. G., Wang, P. H., Shen, J., Li, Y. G., Zhang, Y. F., & Hu, E. J. (2012). Automatic fuzzy partitioning approach using Variable string length Artificial Bee Colony (VABC) algorithm. *Applied Soft Computing*, 12, 3421–3441.
- Vallado, D. A. *Fundamental of astrodynamics and applications*, third ed., Microcosm Press and Springer, 2007.
- Vesterstrom, J., & Thomsen, R. (2004). A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. *IEEE evolutionary computation congress*: 2 1980–1987.
- Vtipil, S. D., & Newman, B. (2010). Designing a constrained optimal orbit for earth observation satellites based on user requirements. *AIAA/AAS astrodynamics specialist conference*, Toronto, CA.
- Wall, B. J., & Conway, B. A. (2009). Genetic algorithms applied to the solution of hybrid optimal control problems in astrodynamics. *Journal of Global Optimization*, 44(4), 493–508.
- Wang, Y. N., Wu, L. H., & Yuan, X. F. (2010). Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure. *Soft Computing*, 14, 193–209.
- de Weck, O. L., Scialom, U., & Siddiqi, A. (2008). Optimal reconfiguration of satellite constellations with the auction algorithm. *Acta Astronautica*, 62(2), 112–130.
- Williams, E. A., Crossley, W. A., & Lang, T. J. (2001). Average and maximum revisit time trade studies for satellite constellations using a multiobjective genetic algorithm. *The Journal of the Astronautical Sciences*, 49(3), 385–400.
- Wolfe, W. J., & Sorensen, S. E. (2000). Three scheduling algorithms applied to the earth observing systems domain, 46(1), 148–168.