

# DLS 和 NGA 结合的平台资源调度方法

胡诗骏, 姚佩阳, 孙 昱, 李 锴

(空军工程大学信息与导航学院,西安,710077)

**摘要** 研究战场环境平台资源调度问题,针对传统使用的动态列表调度算法易陷入局部最优,提出一种基于动态列表调度和小生境遗传算法相结合的求解方法。考虑任务激烈度来更准确地描述平台资源能力的动态损耗,以使命完成时间为指标,在保证平台资源利用率的前提下,采用具有全局性的小生境遗传算法,得到最优的平台资源调度方案。通过一个联合作战案例,得出所提方法作战使命完成时间更短,为战场资源调度提供依据。

**关键词** 平台资源调度;任务激烈度;损耗系数;动态列表调度;小生境遗传算法

**DOI** 10.3969/j.issn.1009-3516.2015.03.017

**中图分类号** TP391.9 **文献标志码** A **文章编号** 1009-3516(2015)03-0082-05

## An Approach to Platform Resource Scheduling Based on DLS and NGA

HU Shijun, YAO Peiyang, SUN Yu, LI Kai

(Information and Navigation College, Air Force Engineering University, Xi'an 710077, China)

**Abstract:** In this paper, the platform resource scheduling problem has been carefully studied. Aimed at the DLS that is liable to plunge into local optima, this paper presents an approach to solving the problem based on the dynamic list scheduling (DLS) and niche genetic algorithm (NGA). Task intense is introduced to describe the dynamic loss of the platform capacity in the process of combat. By taking the finish-time of mission as an index and on the premise of guaranteeing the platform resource utilization ratio, the NGA with overall importance is adopted to get the best approach to platform resource scheduling. Finally, through a case of joint campaign, a conclusion is obtained that the use of proposed approach can make the mission finished in short time, which provides a basis for battlefield resource scheduling.

**Key words:** platform resource scheduling; task intense; loss coefficient; dynamic list scheduling; niche genetic algorithm

动态列表调度方法<sup>[1]</sup>是解决平台资源调度问题的常用方法。文献[1~2]提出了多维动态列表调度(multi-dimensional dynamic list scheduling, MDLS)方法;文献[3]针对 MDLS 中优先级计算方式的不合理性进行了改进,优化效果更佳;文献[4~5]针对 MDLS 中优先级设置的不合理性,提出了多优先级列表动态调度方法;文献[6]通过引入 PWE 方法对 MDLS 进行了改进,提出了循环 MDLS 的

方法。MDLS 存在局部搜索和优先权函数的合理性问题;MDLS 及其改进算法都是基于贪心机制的,无法保证所得到的方案是最优的。

## 1 问题的数学描述

### 1.1 基本概念

1) 任务(Task):部队为完成预定的作战目的所

收稿日期:2015-01-06

基金项目:国家自然科学基金资助项目(61273048)

作者简介:胡诗骏(1990-),男,浙江杭州人,硕士生,主要从事组织结构模型设计研究.E-mail:shijunhu0504@163.com

**引用格式:**胡诗骏,姚佩阳,孙昱,等. DLS 和 NGA 结合的平台资源调度方法[J]. 空军工程大学学报:自然科学版,2015,16(2):82-86. HU Shijun, YAO Peiyang, SUN Yu, et al. An Approach to Platform Resource Scheduling Based on DLS and NGA[J]. Journal of Air Force Engineering University: Natural Science Edition, 2015, 16(3): 82-86.

执行的任务,由作战使命分解形成,记为任务集为  $T = \{T_1, T_2, \dots, T_N\}$ ,  $N$  为使命分解得到的任务数量。任务激烈度<sup>[7]</sup>  $Sd_k$ ,即任务对抗的激烈程度,通常  $Sd_k \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ 。

2) 平台(Platform, P):平台即指处理任务所需要的物力资源实体,记平台集为  $P = \{P_1, P_2, \dots, P_Q\}$ ,  $Q$  为平台数量。

### 1.2 变量定义

1) 平台-任务的分配变量  $T_k P_m$ 。若平台  $P_m$  分配给任务  $T_k$ ,则  $T_k P_m = 1$ ;否则  $T_k P_m = 0$ 。

2) 平台在任务间的转移变量  $TP_{ijm}$ 。若平台  $P_m$  在执行完  $T_i$  后分配给  $T_j$ ,则  $TP_{ijm} = 1$ ;否则  $TP_{ijm} = 0$ 。特别的,若  $i = j$ ,则  $TP_{ijm} = 0$ 。

3) 任务执行顺序变量  $T_{ij}$ 。若  $T_i$  必须在  $T_j$  开始执行前完成,则  $T_{ij} = 1$ ;否则  $T_{ij} = 0$ 。

4) 使命的完成时间  $T_f$ 。即最后一个任务的结束时间。

5) 任务的持续时间  $Dt_i$ 。 $Dt_i = Ft_i - St_i$ ,  $St_i$ ,  $Ft_i$  分别是任务  $T_i$  的开始和结束时间。

6) 资源满足度  $Sr_i$ 。即完成某任务实际能使用的平台资源与所需求平台资源之间的比值。

### 1.3 平台资源能力更新

以平台自身的损耗系数与作战任务激烈程度的乘积作为总的资源能力损耗系数,则资源能力更新公式:

$$r_{ml}^{\text{renewed}} = r_{ml} (1 - Sd_k W_l \frac{r_{ml}^{\text{real}}}{r_{ml}})$$

$$(l = 1, 2, \dots, L; k = 1, 2, \dots, N) \quad (1)$$

式中:  $r_{ml}^{\text{renewed}}$  为  $P_m$  在执行完该任务后所拥有的第  $l$  种资源能力的大小;  $r_{ml}^{\text{real}}$  为  $P_m$  在执行该任务中第  $l$  种资源能力的实际损耗值。

### 1.4 数学模型的建立

#### 1.4.1 任务平台的匹配分析

若平台  $P_m$  在执行完某一个任务  $T_a$  之后被分配执行  $T_i$ ,并且在执行完任务  $T_i$  后又会分配去执行下一个任务  $T_j$ ,即:

$$\sum_{a=0}^N TP_{aim} - T_i P_m = 0 (a, i = 1, 2, \dots, N; m = 1, 2, \dots, Q) \quad (2)$$

$$\sum_{j=0}^N TP_{ijm} - T_i P_m = 0 (i, j = 1, 2, \dots, N; m = 1, 2, \dots, Q) \quad (3)$$

#### 1.4.2 资源能力需求分析

任务的资源能力需求与资源满足度的乘积要不大于实际所提供的资源能力,即:

$$\sum_{m=1}^Q r_{ml} \cdot T_i P_m \geq Sr_i Pre_i (i = 1, 2, \dots, N; l = 1, 2, \dots, L) \quad (4)$$

#### 1.4.3 处理任务的时间分析

任务开始时间约束如下:

$$St_j \geq St_i + Dt_i + TP_{ijm} (\frac{D_{ij}}{V_m} + T_{ij} T_f') - T_{ij} T_f' \quad (5)$$

( $i, j = 1, 2, \dots, N; m = 1, 2, \dots, Q$ )

$T_f'$  不小于  $T_f$ ,是一个较大的值; $D_{ij}$ 是任务  $T_i$  与任务  $T_j$ 之间的距离。

$$D_{ij} = \sqrt{(Lx_j - Lx_i)^2 + (Ly_j - Ly_i)^2} \quad (6)$$

同时,使命完成时间必须满足:

$$T_f \geq St_i + Dt_i (i = 1, 2, \dots, N) \quad (7)$$

综合以上分析,调度问题的数学模型如下:

$$\begin{cases} \min T_f \\ \sum_{a=0}^N TP_{aim} - T_i P_m = 0; \sum_{j=0}^N TP_{ijm} - T_i P_m = 0 \\ \sum_{m=1}^Q r_{ml} \cdot T_i P_m \geq Sr_i \cdot Pre_i (l = 1, \dots, L) \\ St_j \geq St_i + Dt_i + TP_{ijm} (\frac{D_{ij}}{V_m} + T_{ij} T_f') - T_{ij} T_f' \\ D_{ij} = \sqrt{(Lx_j - Lx_i)^2 + (Ly_j - Ly_i)^2} \\ T_f \geq St_i + Dt_i \\ \sum_{i=0}^N TP_{oim} = \sum_{j=0}^N TP_{jom} = 1 \\ 0 \leq T_f \leq T_f'; St_i \geq 0; T_i P_m, TP_{ijm} \in \{0, 1\} \\ a, i, j = 1, 2, \dots, N; m = 1, 2, \dots, Q \end{cases} \quad (8)$$

## 2 问题求解

### 2.1 基于 DLS 的任务选择流程

基本流程见图 1。

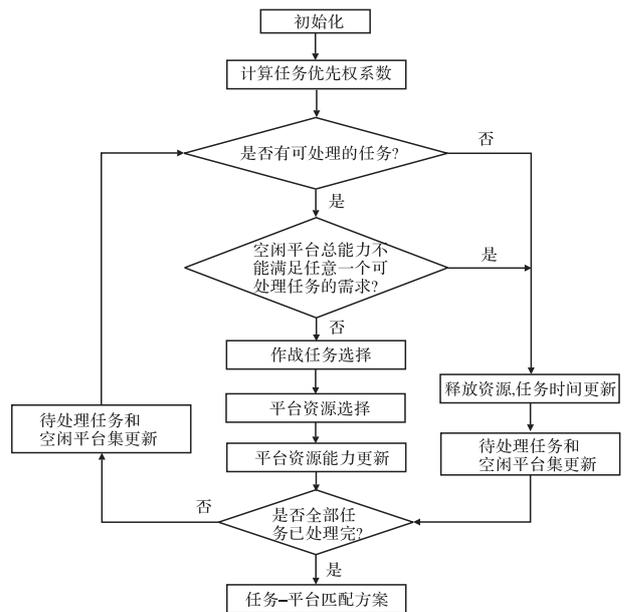


图 1 基于 DLS 和 NGA 调度方法流程

Fig.1 Scheduling method based on DLS and NGA

任务选择首先要求出各个任务的优先级函数,任务优先级的计算取决于任务序列图。本文采用加权长度算法 WL<sup>[8]</sup>进行任务优先权计算 TP。

## 2.2 基于 NGA 的平台匹配

平台资源调度的关键问题是任务分配最佳的平台组,在此通过采用一种改进的小生境遗传算法<sup>[9]</sup>来为选择的任务匹配最佳的平台组。

### 2.2.1 编码与解码

从当前的空闲平台集  $P_{free}$  中为所选定的任务  $T_K (K \in \{1, 2, \dots, N\})$ , 分配最佳的平台组, 这是一个多维的 0-1 背包问题。

#### 2.2.1.1 编码

为了提高 NGA 的效率, 采用双串结构<sup>[10]</sup>对染色体进行编码。

#### 2.2.1.2 解码

解码主要是通过特定的方式进行可行解的构造, 按以下步骤进行解码:

**Step1** 设染色体解码后的平台资源真实状态为  $(x_1, \dots, x_k, \dots, x_{N_{free}})$ , 资源能力总和为  $Pal = (rs_1, rs_2, \dots, rs_L)$ , 任务  $T_K$  的资源能力需求为  $Pre_K$ ;

**Step2** 令  $i = 1; Mid = 0$ ;

**Step3** 假如  $B_{A(i)} = 1$ , 令  $x_{A(i)} = 1, Mid = Mid + PO_{A(i)}$ , 转 **Step4**; 否则, 转 **Step4**;

**Step4** 令  $i = i + 1$ , 若  $i > N_{free}$ , 结束循环; 否则, 转 **Step2**;

**Step5** 令  $Tmp = Sr_k * Pre_K - Mid$ ;

**Step6** 若  $Tmp < 0$  (每个元素都小于 0), 结束解码; 否则, 转 **Step7**;

**Step7** 令  $j = 1, Bet = 10\ 000$ ;

**Step8** 若  $B_{A(j)} = 0$ , 求出  $Tmp$  和  $PO_{A(i)}$  之间的欧氏距离  $d$ ; 否则, 转 **Step10**;

**Step9** 若  $Bet < d$ , 转 **Step10**; 否则,  $Bet = d, m = A(j)$ ;

**Step10** 令  $j = j + 1$ , 若  $j > N_{free}$ , 则  $x_m = 1, Tmp = Tmp - PO_m$ ; 否则, 转 **Step8**。

### 2.2.2 适应度函数

考虑到本文小生境遗传算法的特殊应用环境, 本文主要通过考虑缩短使命的完成时间和提高资源的利用率两方面来综合设定 NGA 的适应度函数。

#### 2.2.2.1 时间优先权系数 $P_{time}$ 的计算:

当每个平台都到达任务执行地点时, 任务才开始执行, 即:

$$ST(AP_{free}^{(i)}) = \max_{m \in AP_{free}} (St_{T(m)} + Dt_{T(m)} + \frac{D_{T(m), T_K}}{V_m}) \quad (9)$$

$T(m)$  为  $P_m$  最后处理的任务。

$$ST_{min} = \min_{i \in 1, 2, \dots, Popsize} (ST(AP_{free}^{(i)})) \quad (10)$$

$$P_{time}(AP_{free}^{(i)}) = \frac{ST(AP_{free}^{(i)}) - ST_{min}}{\sum_{k=1}^{Popsize} (ST(AP_{free}^{(k)}) - ST_{min})} \quad (11)$$

#### 2.2.2.2 资源能力优先权系数 $P_{resource}$ 的计算:

记平台组  $AP_{free}^{(i)}$  分配给  $T_K$  的资源能力冗余程度为  $RER(AP_{free}^{(i)})$ , 即:

$$RER(AP_{free}^{(i)}) = \sum_{l=1}^L (\sum_{m \in AP_{free}^{(i)}} PO_{ml} - R_{Kl}) \quad (12)$$

记平台组  $AP_{free}^{(i)}$  中各个平台相对于  $T_{ready}$  中其他任务的平均资源满足程度  $SAR(AP_{free}^{(i)})$ , 即

$$SAR(AP_{free}^{(i)}) = \frac{1}{Num^{(i)}} \sum_{m \in AP_{free}^{(i)}} \sum_{n \in T_{ready} \setminus \{K\}} \sum_{l=1}^L \min(PO_{ml}, R_{nl}) \quad (13)$$

综合式(15)和式(16), 得出  $AP_{free}^{(i)}$  的资源能力影响因子, 即:

$$SRR(AP_{free}^{(i)}) = RER(AP_{free}^{(i)}) + SAR(AP_{free}^{(i)}) \quad (14)$$

记该种群中最优个体所对应的资源能力影响因子最小, 即:

$$SRR_{min} = \min_{i \in 1, 2, \dots, Popsize} SRR(AP_{free}^{(i)}) \quad (15)$$

$$P_{resource}(AP_{free}^{(i)}) = \frac{SRR(AP_{free}^{(i)}) - SRR_{min}}{\sum_{k=1}^{Popsize} (SRR(AP_{free}^{(k)}) - SRR_{min})} \quad (16)$$

综合式(14)和式(19), 得到总的优先权系数, 将其作为 NGA 的适应度函数, 即:

$$Fitness = \alpha P_{time}(AP_{free}^{(i)}) + \beta P_{resource}(AP_{free}^{(i)}) \quad (17)$$

式中:  $\alpha, \beta \in (0, 1)$ , 且  $\alpha + \beta = 1$ 。Fitness 越小, 越接近优化的目标。

### 2.2.3 遗传算子

#### 2.2.3.1 自适应交叉、变异算子

针对固定的交叉、变异概率, 有可能导致局部收敛, 本文将引入自适应策略得到动态的交叉、变异概率。自适应交叉概率  $P_C$  和变异概率  $P_M$  具体求解如下:

$$P_C = \begin{cases} P_{Cbegin} (f_C \geq f_{avg}) \\ P_{Cbegin} \frac{f_C - f_{min}}{f_{avg} - f_{min}} (f_C < f_{avg}) \end{cases} \quad (18)$$

$$P_M = \begin{cases} P_{Mbegin} (f_M \geq f_{avg}) \\ P_{Mbegin} \frac{f_M - f_{min}}{f_{avg} - f_{min}} (f_M < f_{avg}) \end{cases} \quad (19)$$

#### 2.2.4 小生境淘汰操作

针对传统的遗传算法容易导致局部收敛, 难以搜索到全局最优解的缺点, 采用具有全局性的小生境遗传算法。由种群  $D_{pbegin}, D_{pc}$  和  $D_{pm}$  合并为一个种群  $D_{psum}$ , 将种群  $D_{psum}$  按个体适应度值升序排列, 将其前  $M$  个优秀个体的欧氏距离作为小生境

距离参数,即

$$\Gamma = \min \|AP_m - AP_n\| = \min \sqrt{\sum_{k=1}^{\text{Bitlength}} (AP_{mk} - AP_{nk})^2} \quad (20)$$

$m = 1, 2, \dots, M - 1, n = m + 1, m + 2, \dots, M - 1$

$AP_m$  和  $AP_n$  是种群中的 2 个个体。每代的适应度值升序排列不同,前  $M$  个个体的最小欧式距离也是不一样的,也就得到了动态的小生境距离参数。对种群  $D_{p_{\text{sum}}}$  中的任意 2 个个体,求出其欧式距离  $d$ ,若  $d < \Gamma$ ,则比较两者的适应度值大小,对其中适应度值较大的个体施加一个强罚函数,增大其淘汰概率。施加的罚函数如下:

$$F_{\text{penalty}} = \frac{f_{\text{max}} - f_{\text{min}}}{|f_A - f_B|} \quad (21)$$

式中:  $f_A$  和  $f_B$  是两个个体的适应度值,  $f_{\text{max}}$  和  $f_{\text{min}}$  分别为种群的最大和最小适应度值。

### 2.2.5 基于 NGA 的算法流程

NGA 算法的具体步骤描述如下:

**Step1** 参数初始化。根据空闲平台集  $P_{\text{free}}$ , 设置染色体基因位数  $\text{Bitlength}$  和合适的种群规模  $\text{Popsiz}$ ,设置最大迭代次数  $\text{Generationmax}$ ,初始交叉概率  $P_{C_{\text{begin}}}$  和初始变异概率  $P_{M_{\text{begin}}}$ 。

**Step2** 生成初始种群。随机产生  $2\text{Popsiz}$  个染色体,经小生境淘汰控制操作产生  $\text{Popsiz}$  个优秀个体作为初始种群  $D_{p_{\text{begin}}}$ ,令迭代计数器  $k = 1$ 。

**Step3** 适应度计算。计算种群中的所有个体的适应度值,并求出平均适应度值  $f_{\text{avg}}$  和最小适应度值  $f_{\text{min}}$ 。

**Step4** 自适应交叉、变异操作。

**Step5** 进行小生境淘汰控制操作。对种群中任意 2 个个体,求出个体间的距离  $d, d < T$ ,则添加罚函数,并将适应度值进行升序排列,取出  $P_{\text{opsize}}$  个个体替代  $D_{p_{\text{begin}}}$ 。

**Step6** 判断是否满足迭代终止条件。若不满足,则  $k = k + 1$ ,并转至 **Step3**;否则小生境遗传算法结束,输出结果。

## 3 算例分析

为了验证方法的优越性和适用性,以文献[6]中的联合登岛作战为例,进行仿真实验。

各个任务对抗的激烈程度  $Sd_k = 1(k = 1, 2, \dots, 18)$ ,损耗系数为  $W = [0.1, 0.1, 0.1, 0.05, 0.05, 0.05, 0.1, 0]$ ,分别用  $\text{MDLS}, \text{MDLS} + \text{PWE}$  以及本文设计的方法进行仿真,仿真结果见图 2,本文设计的方法所设置的 NGA 参数为:群体规模 50,迭代次数 500,  $M = 10, P_{C_{\text{begin}}} = 0.8, P_{M_{\text{begin}}} = 0.05, \alpha = \beta = 0.5$ 。

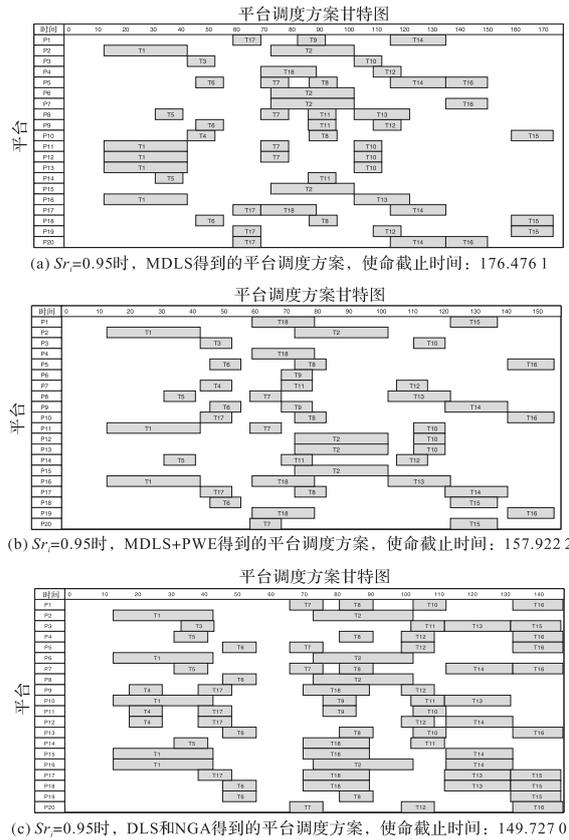


图 2 不同方法得到的平台调度方案

Fig.2 Different methods for platform resource scheduling

由仿真的结果可得,当资源满足度阈值为 0.95 时,MDLS 得到的平台调度方案的使命完成时间为 176.476 1,MDLS+PWE 得到的平台调度方案的使命完成时间为 157.922 2,而 DLS+NGA 得到的平台调度方案的使命完成时间为 149.727 0,表明了本文设计的方法优于 MDLS 和 MDLS+PWE,得到的平台资源调度方案具有更小的使命完成时间,同时还提高了平台资源利用率。

## 4 结语

本文结合 DLS 和 NGA,提出了一种更具全局性的平台资源调度方法。通过将任务激烈度融入资源能力损耗系数,更准确地描述了平台资源能力在作战过程中的动态损耗;同时采用小生境技术,提出了使用 NGA 为选定任务分配平台资源,并设计了 NGA 的解码方法,NGA 的自适应交叉、变异算子,以及小生境淘汰操作;并对其进行了仿真验证,证明了所提方法的优越性。

### 参考文献(References):

[1] Levchuk G M, Levchuk Y. N, Luo J, et al . A Library of Optimization Algorithms for Organizational Design[C]// Proceedings of the 2000 Command and

- Control Research and Technology Symposium, Monterey, CA: NPS, 2000.
- [2] Levchuk, G M, Levchuk Y N, Luo J, et al. Normative Design of Organizations-Part I: Mission Planning [J]. IEEE Transactions on Systems, Man, and Cybernetics, 2002, 32(3): 346-359.
- [3] 鲁音隆, 阳东升, 刘忠, 等. 联合作战规划中资源调度算法研究[J]. 火力与指挥控制, 2006, 31(2): 12-16. LU Yinlong, YANG Dongsheng, LIU Zhong, et al. Research on Algorithm of Resource Allocating in Joint Operation [J]. Fire Control and Command Control, 2006, 31(2): 12-16. (in Chinese)
- [4] 阳东升, 彭小宏, 张维明, 等. C2 组织结构设计: 平台-任务关系设计[J]. 火力与指挥控制, 2006, 31(3): 9-13. YANG Dongsheng, PENG Xiaohong, ZHANG Weiming, et al. Design of C2 Organizational Structure: Designing Relations Among Platforms and Tasks [J]. Fire Control And Command Control, 2006, 31(3): 9-13. (in Chinese)
- [5] 陈洪辉, 赵亮, 芮红, 等. 作战任务和资源间的匹配模型及求解算法研究[J]. 系统工程与电子技术, 2008, 30(9): 1712-1716. CHEN Honghui, ZHAO Liang, RUI Hong, et al. Research on the Match Model and Solving Method Between Operational Tasks and Resources [J]. Systems Engineering and Electronics, 2008, 30(9): 1712-1716. (in Chinese)
- [6] 张杰勇, 姚佩阳, 李凡. 完成时间限制下的任务-平台关系设计模型及算法[J]. 系统工程与电子技术, 2012, 34(8): 1621-1629. ZHANG Jieyong, YAO Peiyang, LI Fan. Task-platform Relation Design Model and its Algorithm under Completion Time Constraint [J]. Systems Engineering and Electronics, 2012, 34(8): 1621-1629. (in Chinese)
- [7] 牟亮, 张维明, 陈涛, 等. 不确定条件下 C2 组织结构的“任务-平台”关系设计模型及算法[J]. 系统工程与电子技术, 2010, 32(12): 2576-2583. MU Liang, ZHANG Weiming, CHEN Tao, et al. Task-platform Relation Design Model and Algorithm for C2 Organization Structure under Uncertainty [J]. Systems Engineering and Electronics, 2010, 32(12): 2576-2583. (in Chinese)
- [8] Shirazi B, Wang M F. Analysis and Evaluation of Heuristic Methods for Static Task Scheduling [J]. Journal of Parallel and Distributed Computing, 1990, 10: 222-232.
- [9] 陆青, 梁昌勇, 杨善林, 等. 面向多模态函数优化的自适应小生境遗传算法[J]. 模式识别与人工智能, 2009, 22(1): 91-100. LU Qing, LIANG Changyong, YANG Shanlin, et al. An Adaptive Niche Genetic Algorithm for Multimodal Function Optimization [J]. Pattern Recognition and Artificial Intelligence, 2009, 22(1): 91-100. (in Chinese)
- [10] 张杰勇, 姚佩阳, 周翔翔, 等. 基于 DLS 和 GA 的作战任务-平台资源匹配方法[J]. 系统工程与电子技术, 2012, 34(5): 947-954. ZHANG Jieyong, YAO Peiyang, ZHOU Xiangxiang, et al. Approach to Operation Task and Platform Resource Matching Based on DLS and GA [J]. Systems Engineering and Electronics, 2012, 34(5): 947-954. (in Chinese)

(编辑: 姚树峰)