

doi: 10.7690/bgzdh.2024.12.014

# 基于改进的 MDLS 算法的城市作战资源调度技术

吕家发<sup>1</sup>, 黄炎焱<sup>1</sup>, 谢文<sup>2</sup>

(1. 南京理工大学自动化学院, 南京 210094; 2. 陆军炮兵防空兵学院南京校区, 南京 211132)

**摘要:** 为避免在城市作战中一个任务投入过多超过需求的剩余资源, 影响其他任务的资源分配而造成作战时间延长, 将多维动态列表调度算法(multidimensional dynamic list scheduling, MDLS)应用于城市作战资源调度。通过在算法的平台选择阶段做出改进, 加入节约模块和支援模块, 建立多目标规划模型。仿真实验对比结果表明, 该技术研究可为城市作战资源调度提供理论参考。

**关键词:** MDLS; 多目标优化; 城市作战; 资源分配调度

**中图分类号:** E911 **文献标志码:** A

## Urban Combat Resource Scheduling Technology Based on Improved MDLS Algorithm

LYU Jiafa<sup>1</sup>, Huang Yanyan<sup>1</sup>, Xie Wen<sup>2</sup>

(1. School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China;

2. Nanjing Campus, Army Academy of Artillery and Air Defense Force, Nanjing 211132, China)

**Abstract:** In order to avoid investing too much surplus resources in one task in urban combat, which will affect the resource allocation of other tasks and prolong the combat time, the multidimensional dynamic list scheduling algorithm (multidimensional dynamic list scheduling MDLS) is proposed, is applied to urban combat resource scheduling. By improving the algorithm in the stage of platform selection, saving module and support module are added, and a multi-objective programming model is established. The simulation results show that the research of this technology can provide a theoretical reference for urban combat resource scheduling.

**Keywords:** MDLS; multi-objective optimization; urban operations; resource allocation and scheduling

## 0 引言

近年来, 随着世界上一系列局部战争的产生, 城市作战越来越成为世界关注的焦点。城市作战涉及多种作战资源, 而且作战场景和作战任务也是相对多样, 作战资源规划水准的高低会对作战结果产生巨大的影响。如何将作战资源规划得精确、合理、高效且不浪费, 是现代信息化战场作战研究的一个重要方向。

作战资源的分配和调度, 是将有限的资源合理分配到作战任务中去, 以求高效完成最终使命。由于任务层次和作战资源的多样性以及随着任务的进行作战资源发生的改变, 各阶段资源规划的不同会使得任务、资源、效益呈现不同的表现。城市战场作战资源主要是陆军方面的战斗人员、作战装备等, 作战效益通常是对于时间、作战价值、战斗风险等。作战效益是资源调度的追求目标, 作战效益的多样性也反映了本研究是一种多目标优化的决策过程。

合理的任务规划是任务处理决策的关键, 这就

需要考虑资源的调度和任务之间的并行处理, 这种问题在很多场景下都会出现, 比如联合作战的兵力规划调度、多处理器系统的动态调度、车辆运输调度等<sup>[1-3]</sup>。对这一问题的解决已经有许多调度算法的研究, 如遗传算法、模拟退火算法、多维动态列表调度算法(MDLS)等启发式算法, 其中 MDLS 算法在复杂的作战调度问题上比较常用<sup>[4-5]</sup>。

MDLS 算法由 Levchuk 等<sup>[4]</sup>提出, 是通过分析优先级的规则来获取资源调度策略。现有研究分析了该算法的效果, 并证实了其可行性。但是算法对于优先级的排序模式过于单一, 容易陷入局部最优, 只能保证在较短时间内获取可行解。这也赋予了 MDLS 算法在解决复杂调度问题中的速度优势。

结合城市作战场景, 区别于大规模联合作战, 小规模巷战场景较多, 支援协同较为困难, 且城区掩体较多, 很多大杀伤力武器难以发挥效能, 针对性的重型武器往往不在作战部队编制内。容易出现任务的个别资源需求很高, 虽然可以应付, 但是作战资源的消耗量也会很高, 从而造成其他任务难以

收稿日期: 2024-06-21; 修回日期: 2024-07-22

基金项目: 信息系统需求重点实验室开放基金课题(LHZZ2021-M05); 地面靶标与毁伤评估系统(LJYX2019-05-001)

第一作者: 吕家发(1995—), 男, 安徽人, 硕士。

分配资源而不得不延迟推进，造成作战时间延长，且降低了部队应付突发状况的能力。使用传统 MDLS 算法对城市作战场景进行处理，部分情况下会使得算法无法跳出这种特殊情况，从而产生不合理结果。

过去对于城市作战的资源规划研究较少，笔者将 MDLS 算法与多目标优化相结合，任务选择阶段依然使用 MDLS 算法中的任务优先权函数来获取，平台选择阶段加入应对平台优先级相同时的节约模块和应对个别高资源需求任务的支援模块，建立多目标规划模型，再使用相关算法求解获取解集，择优作为平台分配结果。

## 1 城市作战资源调度模型

### 1.1 作战资源相关概念

作战资源调度问题是在给定的作战任务和现有的可供调度的作战资源情况，对作战资源按照任务的需求进行调度，追求所有任务尽快的完成。对于作战资源的量化处理是资源分配调度的前提。

1) 战斗力指数  $w$ ：是对实际战斗效能的定量描述，由战斗力对应的火力指数  $f$  和战斗力系数  $\alpha$  共同确定，如部队的对空战斗力可由其对空火力和战斗力系数计算获取。计算公式为： $w=f \cdot \alpha$ 。战斗力系数又受作战单元的作战素质、指挥能力、战术运用、作战士气、战斗环境等影响，由战前对这些客观因素和敌我态势的分析评估获取，则公式可表示为<sup>[6]</sup>：

$$w = f \cdot \prod_{i=1}^n \alpha_i \quad (1)$$

式中： $\alpha_i$  为第  $i$  个战斗力修正系数，为便于计算，取  $\alpha_i$  均为 1； $f$  为火力指数； $w$  为战斗力指数。

2) 作战资源。对于进攻方，体现的是部队各方面的作战能力，比如防空部队在对空方面的战斗力远高于其他部队，但是其他方面的战斗力较低；对于防守方，体现的是区域部署的各方面的防御能力，对于作战资源的需求即是对于进攻方突破防御的作战能力的需求<sup>[7]</sup>。笔者使用战斗力指数来定量描述作战资源。考虑到城市作战场景，作战兵种多为陆军，防御方多以城市中独有的楼宇地形和错综复杂的道路为依托展开隐蔽又灵活的防御部署，进攻方则通过前期侦察数据为依托，以更多的兵力和合适的兵力调配，将防御阵地各个击破。防御装备的类型包括装甲类武器、轻型武器、无人机、火炮等。

掩体多为房屋、地面、雷区、障碍等。可将需要的战斗力指数归纳为：对人战斗力指数  $w_1$ 、对空战斗力指数  $w_2$ 、对装甲战斗力指数  $w_3$ 、对建筑战斗力指数  $w_4$  和扫雷破障战斗力指数  $w_5$ ，用向量  $w = \{w_1, w_2, w_3, w_4, w_5\}$  表示。战斗力指数对应的火力指数可表示为  $F = \{f_1, f_2, f_3, f_4, f_5\}$ 。

3) 剩余资源。指的是在对任务进行资源分配调度后，其被分配的总资源超过该任务资源需求的那一部分。可知，剩余资源越多，资源的浪费越多。

4) 节约模块。MDLS 算法改进的部分，用于在给任务分配平台时，对出现相同优先级平台的处理，追求符合资源需求的情况下尽量减少剩余资源。

5) 支援模块。MDLS 算法改进部分，用于在进行平台分配时，单个任务出现剩余资源过多，远高于对应任务的资源需求时，判断是否使用支援。

### 1.2 城市作战问题描述

城市作战作为现代战场常见的战争形态，作战方案的优劣直接影响到城市作战的难易与成败，而作战资源的分配调度是作战方案中关键的组成部分。笔者以一次营级规模的城市作战为例，首先假定进攻方已经实现通过各种侦察手段了解了敌方的大致部署，作战战场环境、敌方部署位置如图 1 所示。

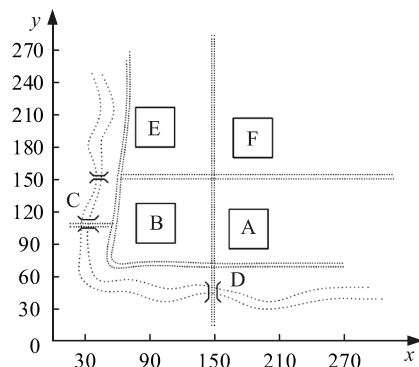


图 1 城市作战战例想定环境

在 A、B、E、F 区包括中间的道路边上都是敌方防御部署区域，并且在 C、D 桥梁靠近城区的区域也有兵力部署。B、E 正西方有我进攻方搭建的浮桥。笔者研究对 B 区的进攻任务，并且由在 B 区对岸的进攻方营级作战单位来负责。对 B 区进攻的阶段性作战如图 2 所示。

根据守军的防御部署，可将其分为 5 个区域：

1) 非守军火力覆盖下的安全区，作为我军的临时指挥阵地，执行一些战术制定、资源分配和调度等工作；2) 机动防御区，部署有守军的高机动单位，比

如坦克部队、装甲机动部队等，铺设反坦克地雷和路障，主要发生机动遭遇战；3)防御前沿，是保护城区的固定阵地，有雷区、碉堡、炮兵阵地等，主要发生阵地争夺战；4) 城区，这里环境较为复杂，楼间、楼里、地上、地下、天空都可能有防御单位部署，且城区内重型武器很难发挥效能，主要由轻型部队协同支援力量承担巷战任务；5) 后方区，守军撤退区域。进攻方会经过这几个区域从而发生阶段性作战，每个阶段的任务有所不同，可作为任务模型的建立依据<sup>[8]</sup>。

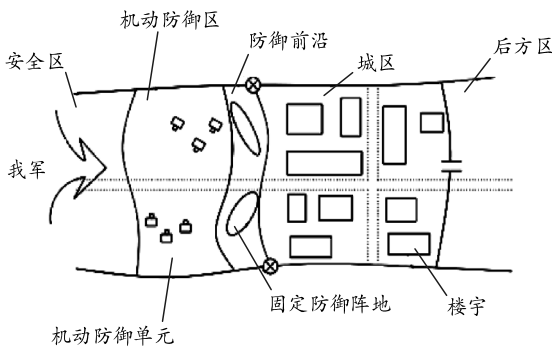


图 2 城市进攻阶段性作战

### 1.3 任务模型

城市作战中的作战任务是双方在城市交战时采取的一系列军事行动的主要依据，在作战过程中关于各种资源的调度都需要根据当前制定的作战任务来安排<sup>[9]</sup>。在执行具体的作战行动之前，会由上级将最终作战使命下达。作战使命是整场战斗或者战役的意图，一般较为笼统，所以需要先分解为更加具体的子任务之后交由下级作战单位做具体执行。层层分解的结果是获取到能够由一个或多个基本作战单元直接执行的子任务集合。将基本作战单元分配调度执行具体作战任务的过程就是作战资源调度的过程。

任务模型是 MDLS 算法中对于作战任务的抽象化描述，其中作战任务的需求被量化为各种作战资源，使用集合  $w_r$  表示，满足作战任务的需求是作战资源调度的最基本要求。如 1.1 节所交代，使用战斗力指数表示作战资源，则作战任务的资源需求可以矢量化为  $w_r = \{w_{r1}, w_{r2}, w_{r3}, w_{r4}, w_{r5}\}$ ， $w_m$  就是任务  $r$  所需要的第  $n$  种资源；此外任务模型中还包括任务的位置信息  $l_r(x, y)$  和任务需要的处理时间  $t_r$ ，是后续资源分配过程中优先级确定的依据；将整个战例的使命进行分解之后获得到的由基本作战单元可以直接执行的子任务集合  $R = \{R_1, R_2, R_3, \dots, R_i\}$ ， $i$  是子任务的数量。

针对本文中想定的城市作战战例，将 B 区的全面占领作为作战使命。综合考虑作战环境和守军部署以及城市进攻阶段性作战顺序，可以将使命分解为子任务集合  $R = \{R_1, R_2, R_3, \dots, R_9\}$ ，包括对于固定阵地的清除，与防御机动单位的遭遇战，城区房屋、地面、地下空中防御力量的清除，以及步步推进后的防御部署等；再根据具体子任务集和任务之间的依赖关系，获取任务执行序列  $O_R$  如图 3 所示，圆中为任务具体行动。

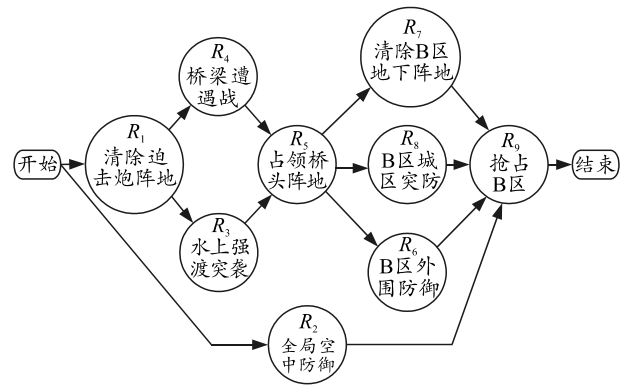


图 3 进攻任务序列

子任务集合  $R$  和  $O_R$  是 MDLS 算法的输入部分，所以这里对作战任务的具体分解过程和执行序列图的生成不过多赘述。参考前期的侦察结果和战前分析，制定任务参数如表 1 所示。考虑到城市作战场景，房屋林立，道路复杂，空间狭小，目标多为隐蔽的作战单位，结合 1.1 节中的分析将需求的火力指数  $F_s$  归为  $f_1$  对人火力指数， $f_2$  对空火力指数， $f_3$  对装甲火力指数， $f_4$  对建筑火力指数， $f_5$  扫雷破障火力指数，数值由有经验的军事人员给出并经过换算处理。任务位置为图 1 坐标系中的坐标给出。

表 1 任务参数

任务	任务区域需要的火力指数					处理时间/h	位置(x, y)
	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$		
$R_1$	5	0	20	0	0	0.5	(70,90)
$R_2$	0	10	0	0	0	5.0	(30,150)
$R_3$	40	2	30	0	5	0.5	(40,150)
$R_4$	40	2	20	0	5	1.0	(50,110)
$R_5$	40	3	10	0	0	1.0	(60,100)
$R_6$	10	5	10	0	0	2.0	(70,110)
$R_7$	5	0	5	40	0	1.5	(100,130)
$R_8$	40	5	30	20	20	1.5	(100,110)
$R_9$	15	0	10	15	0	1.0	(110,110)

### 1.4 平台模型

城市作战平台是进攻部队中提供作战资源的基本作战单元。平台是任务的执行对象，是单元化的作战部队。根据作战想定，由一支营级单位来完成全面占领 B 区的作战任务。考虑到营级指挥部的指挥压力，一般对于连排一级拥有直接调动能力，所

以这里将排级作为基本作战单元,即排级作战平台。

平台模型是 MDLS 算法中对于作战平台的抽象化描述,包括当前可用的系统中的所有平台的集合,矢量表示为  $A_s=\{a_1, a_2, a_3, \dots, a_m\}$ ,  $m$  是平台数量;平台提供的资源种类与前文关于任务所需资源相同,用战斗力指数表示为  $w_s=\{w_{s1}, w_{s2}, w_{s3}, \dots, w_{sm}\}$ ,  $w_{sm}$  就是平台  $A_s$  所能提供的第  $m$  种资源;平台自身属性包括平台的行进速度  $v_s$  和平台所在位置  $l_A(x, y)$ 。

所有平台的参数如表 2 所示,为应对城市作战中重型武器难以发挥效能的特点,轻装合成营相较于重装合成营更容易发挥,所以这里使用前者。参考目前我军装甲合成营的基本配置,本次战例的作战平台分别为:平台 1 为装甲步兵排,平台 2 为防空排,平台 3 为炮兵排,平台 4 为坦克排,平台 5 为支援排,发动进攻之前各平台分别位于河流对岸指定位置。每个作战平台能提供的火力指数与任务需求的火力指数对应,  $f_1$  到  $f_5$  分别为对人、对空、对装甲、对建筑和扫雷破障火力指数,部队参数数值都由有经验的军事人员计算给出并进行换算处理。

表 2 平台参数

平台	能提供的火力指数					行进速度	数量	位置(x, y)
	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$			
1	5	3	5	3	2	0.5	9	(0, 110)
2	1	8	0	0	0	5.0	2	(0, 120)
3	15	0	15	10	5	0.5	2	(0, 110)
4	10	0	10	5	8	1.0	4	(0, 150)
5	0	0	0	0	35	1.0	3	(10, 110)

## 2 使用多目标规划改进的 MDLS 算法

笔者关于资源调度的假定与传统研究方法相同,即在总的任务流程中,任务间可以存在直接或间接的依赖关系。假定每一项任务的处理时间和资源需求一直不变,且任务进行过程中无法中断;每种资源唯一,在同一个时间点至多分配给一项任务。

MDLS 算法是通过分析优先级关系直接获取分配调度结果的算法,主要分为任务选择和平台选择 2 部分。根据已选任务的资源需求,以最小化任务完成时间为追求,将已选择的平台集合分配到任务中。

### 2.1 MDLS 任务选择部分

任务选择的过程,是根据任务间的顺序关系,将所有前导已经完成的任务加入到任务集 Ready 中,该集合里的任务都需要被资源分配。分配的顺序根据任务的优先级来决定。任务优先级的具体确

定算法包括关键路径法、层次分配法等,本研究使用关键路径法。

根据任务流程图和任务预计执行时间绘制关键路径网络  $Q_p$ ,如图 4 所示。

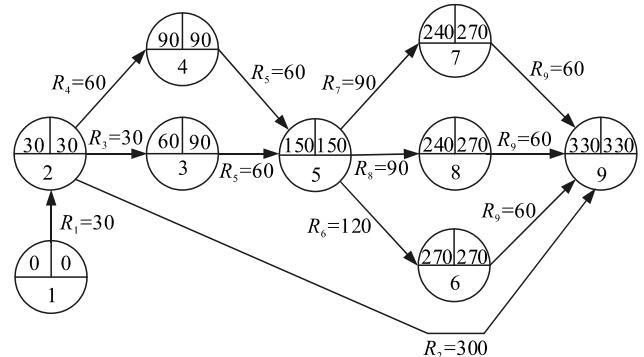


图 4 任务网络

图中,圆的左上方数字是任务最早开始时间,右上方是最迟完成时刻,连接线边是任务执行所需时间。优先权函数定义为<sup>[10]</sup>:

$$P(i)=CP(i). \tag{2}$$

式中:  $P(i)$  为优先权值;  $CP(i)$  为从当前任务开始到全部结束的最短路径时间,时间越小则权值越小,否则权值越大。以此作为 Ready 集任务的优先级排序规则,选择优先级高的任务优先分配平台。

### 2.2 平台分配模型

平台选择的前提是获取可选择任务的集合,要根据当前平台的状态,选择空闲的平台放入平台集 Free,集合中所有平台都是可处理任务的平台。之后会按照平台的优先级进行针对已选任务的平台选择分配。

平台优先级的确定函数为:

$$V(m) = s_{r(m)} + t_{r(m)} + \frac{d_{r(m),i}}{v_m} + \frac{B(m,i)}{BR(m) - B(m,i)}. \tag{3}$$

式中:  $r(m)$  为当前选择的平台  $m$  在当前时间最后处理的任任务;  $s_{r(m)}$  为处理该任务的开始时间;  $t_{r(m)}$  为平台  $m$  处理该任务的时间;  $d_{r(m),i}$  为平台从该任务到待处理任务的行进距离;  $v_m$  为平台  $m$  的行进速度;  $B(m, i)$  为处理当前任务  $i$  时平台  $m$  所能提供的满足任务的各类资源的总和;  $BR(m)$  为平台  $m$  对于当前就绪任务集 Ready 所有任务的满足程度,可知:

$$\left. \begin{aligned} B(m, i) &= \sum_{l=1}^L \min(S_{ml}, q_{il}) \\ BR(m) &= \sum_{i \in \text{Ready}} B(m, i) \end{aligned} \right\}. \tag{4}$$

式中:  $S_{ml}$  为平台  $m$  能提供给任务  $i$  资源的  $l$  数量;  $q_{il}$  为任务  $i$  所需资源的  $l$  数量。

2.2.1 传统 MDLS 算法应用于城市作战的问题

传统的 MDLS 算法会使用平台优先权等级顺序,按照从高到低叠加平台资源来填补所选择的任务资源需求,然后对选择的平台组进行修剪,保证达到任务资源需求并且资源开销最低<sup>[11]</sup>。这样做的优势很明显,能够根据步骤快速并且直接获取每个任务的平台分配情况,但是在面对城市作战场景特殊情况下可能存在问题。

首先是算法本身,当有多个平台的优先级差别不大时,按照优先级的高低进行平台的选择并没有考虑资源浪费情况,导致最后的结果选择的平台虽然能够满足任务需求,但是有部分资源远超需求,平台有限,这就可能使后续的任务得不到理想分配。为了追求总体更理想的分配结果,需要在符合任务需求的同时尽量节约资源,才能在后续可能出现的特殊任务需求中,拥有更多的资源以供分配。本研究也会将资源浪费的情况也考虑在内。

然后是场景方面,可以设想,在小规模城市作战的场景下,以合成营承担总体进攻作战的主体。由现实情况来看,合成营不可能拥有全部的作战平台规模,只能依靠作战之前的侦察评估,做出一些战斗力单位的分配和补充。这种情况下,如果遇到某些任务需要的个别资源需求量非常高,需要大量的平台来填补,以至于分配了该任务之后,Ready集中有其他任务无法分配而导致 MDLS 算法无法进行或者需要长期等待其他任务的完成而浪费时间。结合现实,这种情况下一般需要导弹部队、轰炸机部队等做支援处理。但这些部队的支援也不是随意使用的,不能作为平台放入营级作战指挥的平台分配模型中,否则既不符合营级编制,也可能会使得支援部队的优先级很高,以至于每个任务都会参与,违背了事实情况。所以这里需要对 MDLS 算法的平台选择环节进行改造。

2.2.2 MDLS 算法的改进

这里直接对平台选择环节建立多目标模型,并考虑支援模块的存在,为了达到更佳的资源分配效果,需要追求的目标主要有:1) 平台选择的综合优先级最高;2) 总的资源消耗最小。每个平台的优先级需要提前排序,然后在进行综合优先级计算。计算方法如式(5):

$$V_s = \sum_{m=1}^c x_m x_{\max}^{c-m} \quad (5)$$

式中:  $V_s$  为针对当前任务进行平台选择的综合优先权,用于评价平台选择的合理性;  $c$  为被选择的平台种类数量;  $x_m$  为平台按照优先级排序后的第  $m$  个平台的分配的数量;  $x_{\max}$  为所有平台中的最大数量。

在约束函数中不仅需要约束自变量的范围,还要约束各种资源量要超过任务需求,以及需要考虑是否使用支援。以此可以建立式(6):

$$\left. \begin{aligned} x_{i1}\alpha_{i1}f(A_1) + \dots + x_{ik}\alpha_{ki}f(A_k) + y\alpha_y f(A_y) &\geq F(R_i) \\ x_{i1}\alpha_{i1}f_1(A_1) + \dots + x_{ik}\alpha_{ki}f_1(A_k) + y\alpha_y f_1(A_y) &\geq F_1(R_i) \\ x_{i1}\alpha_{i1}f_2(A_1) + \dots + x_{ik}\alpha_{ki}f_2(A_k) + y\alpha_y f_2(A_y) &\geq F_2(R_i) \\ x_{i1}\alpha_{i1}f_3(A_1) + \dots + x_{ik}\alpha_{ki}f_3(A_k) + y\alpha_y f_3(A_y) &\geq F_3(R_i) \\ x_{i1}\alpha_{i1}f_4(A_1) + \dots + x_{ik}\alpha_{ki}f_4(A_k) + y\alpha_y f_4(A_y) &\geq F_4(R_i) \\ x_{i1}\alpha_{i1}f_5(A_1) + \dots + x_{ik}\alpha_{ki}f_5(A_k) + y\alpha_y f_5(A_y) &\geq F_5(R_i) \\ x_{ij} \in n_{ij}, y \in [0, 1] \end{aligned} \right\} \quad (6)$$

式中:  $F_n(R_i)$  为对于任务  $R_i$  的第  $n$  种资源的需求量;  $F(R_i)$  为综合资源需求量;  $f_n(A_i)$  为平台  $A_i$  能提供的第  $n$  种资源量,使用战斗力指数表示;  $x_{ij}$  为对于任务  $R_i$  选择平台  $A_j$  的数量;  $\alpha_{ij}$  为平台  $A_i$  的第  $j$  种火力指数对应的战斗力系数,用于计算  $A_i$  的综合战斗力  $f(A_i)$ :

$$f(A_i) = \alpha_{i1}f_1(A_i) + \alpha_{i2}f_2(A_i) + \dots + \alpha_{i5}f_5(A_i) \quad (7)$$

式中  $\alpha_{ij}$  和  $f_n(A_i)$  与式(6)含义相同。为体现支援力量的作用,加入支援模块,使用  $f(A_y)$  表示支援力量  $A_y$  的综合战斗力指数,这里为便于计算只考虑一种支援部队。同时限定  $x_{ij}$  的数量都得符合 Free 集中的数量,  $y=0$  是不使用支援,  $y=1$  是使用支援。

同时需要考虑战斗力资源的节约,加入节约模块。综合以上,目标函数为:

$$\left. \begin{aligned} \min(F) &= x_{i1}\alpha_{i1}f(A_1) + x_{i2}\alpha_{i2}f(A_2) + \dots + \\ &x_{ik}\alpha_{ki}f(A_k) + y\alpha_y f(A_y) \\ \max(V_s) \end{aligned} \right\} \quad (8)$$

式中:  $x_{ij}$ 、 $\alpha_{ij}$  和  $f(A_i)$  与式(6)含义相同;  $\min(F)$  为综合资源量提供量的最小值,即综合战斗力指数的最小值;  $\max(V_s)$  为针对当前任务进行平台选择的最大值综合优先权。

由此,平台分配就转化为一个多目标整数规划问题。之后将获取的  $y=1$  的最优解和  $y=0$  的最优解进行分析,结合实际,加入节约条件,即如果  $y=1$  的最优解综合战斗力资源的提供量大于需求量的  $n$  倍,则结果取另一个;否则结果取该解。 $n$  的取值需要结合战场实际,这里取  $n=3$  来计算,即解的综合资源提供超过了 3 倍综合资源需求就可以舍弃。

### 2.3 整体算法步骤

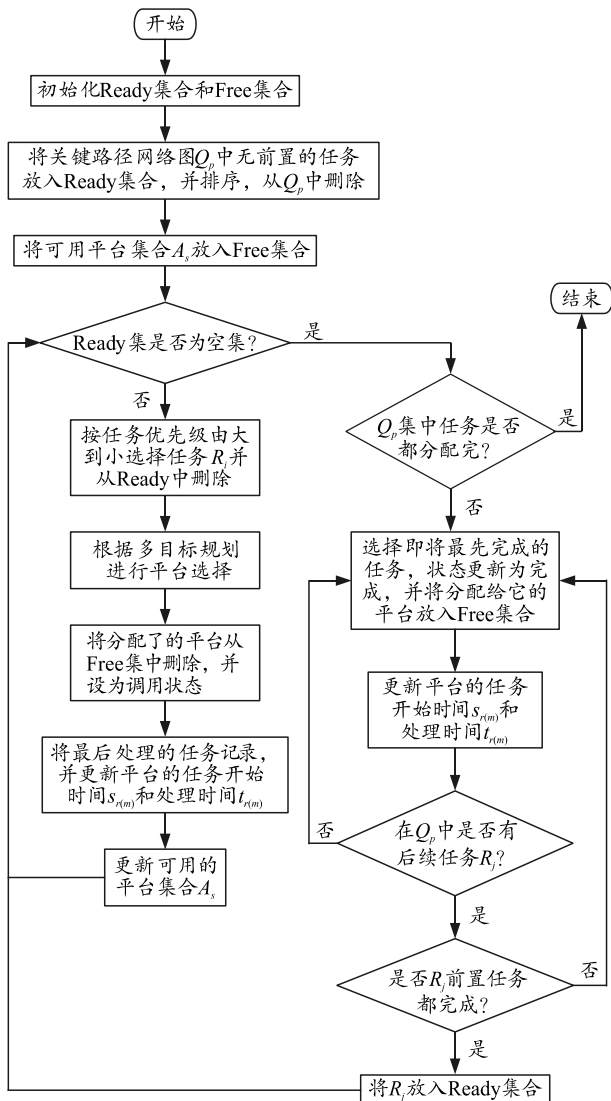


图 5 资源分配调度算法步骤

算法步骤如下：

初始化：把所有没有前置任务的放入 Ready 集中，将所有平台放入 Free 集中。

第 1 步：初始阶段跳过该步骤，直接进入第 3 步；若是全部的任务都分配完成，则结束；若不是，从当前正在处理的里选择马上完成的任务进行更新，将它的状态设置为完成，将分配给它的所有资源放入 Free 集中，并且更新这些平台的最后任务处理完成时刻。再看该任务是否有后续任务，如果有，再看该后续任务的前置任务是否都完成，如果是则放入 Ready 集，任一条件不符合则继续第 1 步。

第 2 步：如果 Ready 中的每个任务的资源需求都不能通过调用 Free 集中的平台来填补，那么返回第 1 步；否则进入第 3 步。

第 3 步：如果 Ready 集为空集合，则直接返回

第 1 步；否则在能满足资源的任务中选出最大优先级的任务作为待分配平台的任务  $R_i$  并在 Ready 中删除它，转到第 4 步。

第 4 步：按照上一节建模过程建立模型，然后计算获取平台分配结果。

第 5 步：将分配了的平台从 Free 集中删除，并设为调用状态，且将他们的最后处理任务设为  $R_i$ ，将各平台的任务开始时间  $s_{r(m)}$  和处理时间  $t_{r(m)}$  更新，将可提供平台更新为剩余的平台，然后返回第 3 步。

## 3 作战想定与仿真分析

### 3.1 作战想定描述

在我军的一次现代化的城市作战军事演习中，分为蓝军和红军双方，红军使用的是我军标准编制和指挥系统，蓝军则为模仿外军的编制架构和防御策略。红军一路进攻推进到由蓝军防守的某个城区。城区基本的防御区域如图 1 所示，有 A、B、E、F 4 个区域。通过前期侦察可知，城区南北长约 3 km，东西约 4 km，西部和南部有一条河流包围，共由约一个合成营 L 作防御部署。

现由红军一轻装合成营 H 承担进攻 B 区的任务，且在城区的正西方的河流对岸集结完毕。图 1 中已经标出了蓝军部队防御要地，城区 B 区和城区西面的桥梁区域 C 区都在此次红军进攻的阶段性目标区域。为了配合其他部队的进攻，达成完全占领城区的目标，H 营接收到上级下达的使命任务为清除 B 区守军并占领。

蓝军模仿外军对于 B 区进行防御展开，基本如图 2 所示，主要有：B 区西面防御前沿的固定防御阵地，约有一个炮兵排作部署；C 区和前沿之间的机动防御阵地，约有一个机步排和部分坦克部署；以及 B 区城区的立体式防御阵地，约有 2 个机步排和部分坦克作地下防御和楼宇间防御部署；部分道路和房屋间设有反坦克雷区和障碍。对于蓝军防御位置对应的具体任务集合构成如图 3 所示，对于任务区域的位置和资源需求如表 1 所示。

红军的装甲合成营 H 的平台编制如表 2 所示，表示战斗开始之前 H 营的全部作战资源。具体的平台参数在 1.4 节中已经交代。此外，作为合成营编制，H 营拥有向上级申请使用其他兵种平台支援的能力，本次作战想定其可申请导弹部队使用精确制导导弹支援，且只可使用一轮支援，支援平台所能给予的作战资源量在后文的仿真分析中给出。H 营需要在符合各任务需求的条件下进行资源分配调

度，以求更快速的完成使命任务。

### 3.2 仿真分析

根据作战想定，初始总共包含 9 个任务和 5 种不同平台。分别使用传统 MDLS 算法和优化过的 MDLS 方法对问题仿真，假定支援平台可提供的火力指数为  $F_s=\{5, 0, 5, 20, 0\}$ ，战斗力系数都取 1，获得 2 种算法的结果如表 3 和 4 所示。

表 3 使用传统 MDLS 算法生成的调度方案

任务	平台分配情况	作战时间段/min
$R_1$	$m_{3a}+m_{3b}$	0~30
$R_2$	$m_{2a}+m_{2b}$	30~330
$R_3$	$m_{1i}+m_{4a}+m_{3a}+m_{3b}$	30~60
$R_4$	$m_{1a}+m_{1b}+m_{1c}+m_{1d}+m_{1e}+m_{1f}+m_{1g}+m_{1h}$	30~90
$R_5$	$m_{3a}+m_{3b}+m_{1a}+m_{1b}$	90~150
$R_6$	$m_{1a}+m_{1b}+m_{1c}$	150~270
$R_7$	$m_{3a}+m_{3b}+m_{4a}+m_{4b}+m_{4c}+m_{4d}$	150~240
$R_8$	$m_{5a}+m_{1c}+m_{1d}+m_{3a}+m_{3b}$	240~330
$R_9$	$m_{4a}+m_{4b}+m_{4c}$	330~390

表 4 使用优化的 MDLS 算法生成的调度方案

任务	平台分配情况	作战时间段/min
$R_1$	$m_{3a}+m_{3b}$	0~30
$R_2$	$m_{2a}+m_{2b}$	30~330
$R_3$	$m_{4a}+m_{4b}+m_{4c}+m_{1c}+m_{1d}$	30~60
$R_4$	$m_{3a}+m_{3b}+m_{1a}+m_{1b}$	30~90
$R_5$	$m_{3a}+m_{3b}+m_{1c}+m_{1f}$	90~150
$R_6$	$m_{1g}+m_{1h}+m_{1i}$	150~270
$R_7$	$m_{3a}+m_{3b}+k$	150~240
$R_8$	$m_{4a}+m_{4b}+m_{1a}+m_{1b}+m_{1c}+m_{1d}$	150~240
$R_9$	$m_{4a}+m_{4b}+m_{4c}$	270~330

表中  $m_{ij}$  指的是第  $i$  种平台中的第  $j$  个平台个体， $k$  指的是支援平台个体， $R_i$  指的是第  $i$  个任务，作战时间段由任务时间、任务流程图和平台分配方式共同决定，由于平台的行动时间相较于任务时间可以忽略，所以这里没有考虑。

由表 3 和 4 中的资源分配结果可以看出，使用笔者优化过的 MDLS 算法在调度时间上优于传统算法。而且对资源的使用方面，表 4 中的资源的浪费更低，节约了资源之后，使部队应付突发状况的能力更强。

通过分析 2 个表，可以发现产生这种情况的原因。

首先在任务  $R_3$  和任务  $R_4$  处，由于平台  $m_1$ 、 $m_3$ 、 $m_4$  的优先级相同，所以传统算法会默认从前到后排序，一直使用  $m_1$  来填补任务  $R_4$  的需求，且  $R_3$  只能使用一个  $m_1$  平台，就只能使用消耗高资源平台填补任务需求，导致这个时间段 2 个任务的分配的资源浪费过大，相较于改进算法结果多使用了近 30 的综合战斗力，这就会使部队应付突发状况的能力降低。

另一点在于任务  $R_7$ ，在改进的 MDLS 算法下，分别考虑使用和不使用支援模块的仿真结果如图 4 和 5 所示。

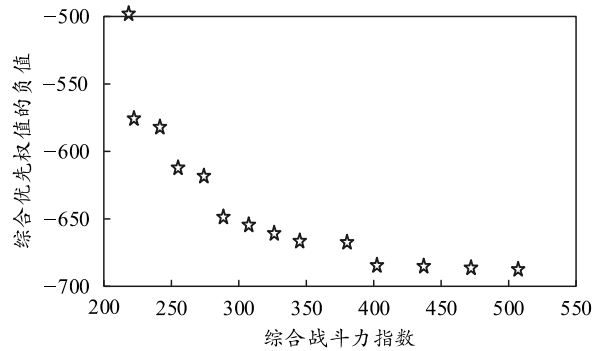


图 6 不使用支援的帕累托最优前沿

在不使用支援模块时，多目标优化的帕累托最优解集中，即使是最低的综合战斗力都有 218，高于任务所需求的 3 倍以上，而传统的 MDLS 算法结果比 218 还要高一些，所以资源浪费更多。这里就可以考虑使用支援模块。

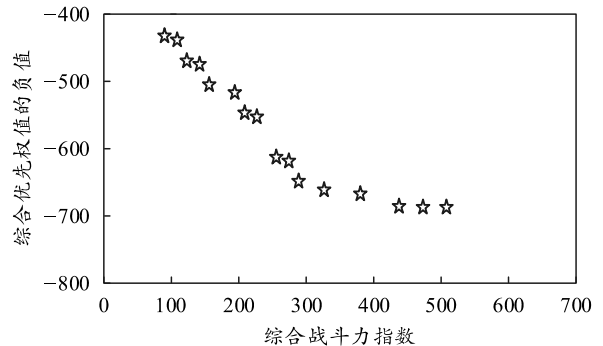


图 7 加入支援模块的帕累托最优前沿

在加入支援模块后，帕累托最优解集中就存在有符合节约条件的解了。根据解的拥挤度最小原则，从解集中获取符合条件的解，再根据节约条件，可得在  $F=90$  的解最符合要求，即使用 2 个  $m_3$  和 1 次支援。如果加入支援后依然无法达到节约条件，则使用拥挤度最小的解。

而传统算法不考虑资源节约会使用过多的  $m_4$  平台，从而导致  $R_8$  得资源无法通过此时 Free 集中得平台来满足，只能等待前面的任务完成开放其他平台来补充，最后使任务完成时间向后推移。改进的算法分配结果能够有效的考虑到战场支援对于资源分配结果的影响，以及提高对请求支援时机的判断能力。

## 4 结论

笔者提出一种改进的作战资源分配调度算法，在传统的 MDLS 算法的平台选择阶段加入节约模块

和支援模块，建立多目标规划模型，并将该算法应用于城市作战资源调度。通过仿真，对比分析了在假定的城市作战场景下使用传统算法和改进算法的结果。从仿真结果来看，获得如下结论：

1) 在应对平台优先级相同时的分配过程，改进的算法能够获得在完成需求的同时尽量减少剩余资源的结果。在同等的初始作战资源量下，使用改进的算法分配作战资源能够在作战中期拥有更多的可分配资源，一定程度上提高了整个部队应付突发状况的能力。

2) 在城市作战场景下，由于防御方依托城市地形使得进攻方部分火力资源难以发挥效能，使用传统 MDLS 算法会使部分任务分配资源过多，综合资源远超需求，而改进的算法能够准确判断使用支援的时机，过滤掉剩余资源过多的分配情况，间接使得各个作战阶段的可分配资源提升，也使得最终分配结果相较于传统算法使用的时间更少。

3) 证明了改进的 MDLS 算法在城市作战场景下进行资源分配的可行性以及相较于传统 MDLS 算法的优越性。为指挥决策人员分配调度作战资源和判断申请支援力量的时机以及后续的研究工作提供参考，具有一定的应用价值。

该算法还有一些不足，如承担支援的平台较为单一且没有考虑作战的资源消耗状况，后续研究应该将其加入考虑，使其实际应用价值得到提升。

### 参考文献：

- [1] 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.
- [2] RAMAMRITHAM K J, STANKOVIC A P F. Efficient Scheduling Algorithms for Real-time Multiprocessor Systems[J]. IEEE Transactions on Parallel and Distributed Systems, 1990, 1(2): 184-194.
- [3] MAHESWARAN M, SIEGEL H J. A Dynamic Matching and Scheduling Algorithm for Heterogeneous Computing Systems[C]// Proceedings Seventh Heterogeneous Computing Workshop. Orland, USA: 1998: 57-69.
- [4] LEVCHUK G M, LEVCHUK Y N, JIE L, et al. Normative design of organizations-Part I: Mission planning[J]. IEEE Transactions on Systems Man and Cybernetics-Part A Systems and Humans, 2002, 32(3): 346-359.
- [5] LEVCHUK G M, LEVCHUK Y N, JIE L, et al. Normative Design of Organizations-Part I: Mission Planning[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part a: Systems and Humans, 2002(3): 346-359.
- [6] 贺毅辉. 作战模拟基础[M]. 北京: 国防工业出版社, 2012: 131-145.
- [7] 王少然, 王瑞林, 王宇建, 等. 步兵分队城市作战效能分析与模糊综合评价[J]. 兵工自动化, 2006, 25(8): 3-4, 12.
- [8] United States Government US Army. Army Techniques Publication ATP 3-06 Urban Operations[M]. North Charleston, SC, USA: Create Space Independent Publishing Platform, 2017: 59-93.
- [9] 陶伟琪, 魏宇. 战场资源统一管理和调度技术综述[J]. 自动化与仪器仪表, 2021, (7): 55-57, 63.
- [10] 李刚洲. 面向服务的作战计划生成方法研究[D]. 长沙: 国防科学技术大学, 2009.
- [11] 包卫东, 王江峰, 张茂军. 一种改进的基于 MDLS 与 GA 的作战资源分配算法[J]. 火力与指挥控制, 2008, 33(9): 18-21.