

doi: 10.3969/j.issn.1006-1576.2011.11.009

基于蚁群算法的舰船维修资源优化调度

张鑫, 赵金超, 张勇明
(海军工程大学管理工程系, 武汉 430033)

摘要: 针对资源有限的舰船维修问题的实际特殊性, 在分析舰船维修资源的基础上, 建立舰船维修资源优化调度模型, 采取蚁群算法进行优化实现, 并以实例进行结果验证。验证结果表明: 该方法是有用的, 利用信息素的更新, 能增强蚂蚁对较优路径的搜索能力, 较好地解决维修过程中出现的维修资源冲突问题, 缩短舰船维修工期。

关键词: 舰船维修; 资源优化; 蚁群算法

中图分类号: TJ83 **文献标志码:** A

Optimized Dispatching of Warship Maintenance Resources Based on Ant Colony Algorithm

Zhang Xin, Zhao Jinchao, Zhang Yongming
(Dept. of Management Engineering, Naval University of Engineering, Wuhan 430033, China)

Abstract: Aiming at the practical situation of resource finite at warship maintenance, Based on the analysis of warship maintenance resources, put forwards anew model of optimized dispatching of warship maintenance resources. Using ant colony algorithm progress optimize realize, and taking example process validate. Through uses pheromone update prove the method is effective, can enhance search capability to better path, reduction of maintenance time, ensures problems better solved.

Keywords: warship maintenance; optimized resources; ant colony algorithm

0 引言

在舰船维修任务实施过程中, 资源始终是制约维修任务调配的重要因素。由于舰船维修通常具有任务数量大、任务并行度高、预定完成任务工期短、资源需求强度大等特点, 而通过对维修资源优化调度能帮助解决维修过程中出现的维修保障资源分配不均、计划不周、预测不准等问题^[1], 从而充分利用现有维修资源, 缩短待修舰船维修周期, 提高维修资源利用率, 在短时间内恢复舰船的战斗力, 最大限度地保持和恢复装备的完好性, 保证舰艇部队遂行正常的战备、训练任务。

在资源有限的项目调度问题上, 国内外学者都做了许多研究。1996年, Boctor首次尝试用模拟退火算法解决经典 RCPSP, 该方法基于任务链表编码, 串行进度生成机制解码, 领域函数采用插入式^[2]; 2003年, Artigues等人将他们提出的插入技术应用于禁忌算法, 把第一个被抛弃的解用基于网络流的算法计算后重新插入^[3]。杨利宏通过遗传算法采用基于活动优先权的十进制编码方式, 结合活动的存储邻接矩阵, 有效地解决活动调度违例现象; 运用优先抢占模式的资源分配方式安排活动资源, 避免资源分配中的冲突问题^[4]。郑超针对资源受限

的项目调度问题的具体特点, 提出了蚂蚁巡游网络图的动态生成方式、信息素的表示及更新方式以及启发信息的计算方法, 并且针对 PSPLIB 中的测试集对算法中的主要参数进行了优化^[5]。因此, 笔者根据资源有限的舰船维修问题的实际特殊性, 采取蚁群算法进行优化实现, 基于复杂的参数设置, 得到较好的实验结果。

1 舰船维修资源分析

舰船维修资源是指与海军舰船维修有直接关联的维修经费、维修物资器材、维修设施设备、维修人才、维修信息以及维修管理等方面有形和无形的资源。而在实际工作中, 只考虑了以实物形式存在的舰船维修资源, 即: 维修物资器材、维修设施设备和维修人才。

根据实际情况, 舰船维修工程可分成坞内修理工程和坞外修理工程。一般现实中的修理工厂只配备一个船坞平台, 并且舰船在船坞内进行修理时不能中止。而大多数舰船都会存在一部分维修项目必须要在船坞内进行, 那么当有多艘舰船同时进行维修, 就会产生资源冲突。一旦出现资源冲突, 每个维修项目的工序进度都会受到影响, 就容易造成整

收稿日期: 2011-07-16; 修回日期: 2011-08-26

作者简介: 张鑫(1986—), 男, 辽宁人, 硕士研究生, 从事军事运筹学方向研究。

体维修任务工期拖后的现象。要合理解决资源冲突,有效利用关键资源,有序分散并行进行的维修项目,保证多项目、多工序交叉进行和推进,就势必要对维修资源进行优化调度。

多艘舰船维修任务要有组织地被安排到修理厂,考虑到维修任务中各个维修项目所需的资源配置,再根据各个修理厂的分工职能,对各个修理厂所属的维修车间的维修资源进行调配。

2 舰船维修资源优化调度模型的建立

由于舰船维修任务比较特殊,与一般的元器件制造任务相比,不仅具有高度的并发性,还有关键资源的制约。假设有 X 艘舰船同时进行修理,每艘舰船都具有维修项目数为 Y ,其中每个维修项目包含 N 个工序阶段,并且每个项目中的各项工序都存在着紧前关系。则给定一个工序集合 $A = \{ a_{xy_0}, a_{xy_1}, \dots, a_{xy_N} \}$ 和一个资源种类的集合 $R = \{ r_0, \dots, r_k \}$,其中完成工序 a_{xi} 所执行的时间为 d_{xi} ,而工序开始时间为 s_{xi} ,添加 2 个虚工序表示维修项目的开始和结束,即开始工序 a_{xy_0} 和结束工序 a_{xy_N} ,其消耗资源量都为 0; G_t 为 t 时刻正在进行的工序集合; P_{xy_1} 为工序 a_{xy_1} 的紧前工序集合;若完成工序 a_{xy_1} 需要对资源 r_k 的消耗量为 R_{xy_1k} ,资源 r_k 总量为常量 $R_k (k=1,2,\dots,K)$;则该维修任务资源优化调度问题可建立如下数学模型:

$$\text{s.t. } s_{xyi} - s_{xyj} \geq d_{xyj}, \forall xyj \in P_{yi} \quad (2)$$

$$\sum_{a_{xyi} \in G_t} R_{xyik} \leq R_k, \forall k, t \quad (3)$$

其中式 (1) 为目标函数,即代表加权后的维修任务工期最短;式 (2) 为工序紧前约束关系,即工序在其紧前工序完成后开始;式 (3) 为资源约束,即在 t 时刻工序对资源 r_k 消耗量不得超过这种资源的总量。

若以舰船的小修为例,修理工序基本分为 6 大阶段:进厂准备、拆卸出舱、内场修理、回装调试、单项验收和试航扫尾。把进厂准备阶段作为所有维修项目的紧前工序,即最早开始的工序(0 工序);把试航扫尾阶段作为所有维修项目的紧后工序,即最晚完成的工序(N 工序)。这 2 项工序均为不消耗资源的虚活动,但要将其所执行的时间计算在内,并且这 2 项工序分别代表整个维修任务的开始和结束。而中间 4 个阶段由于已经进厂修理,可以按照各维修项目的划分,进行工序安排。

在简易的网络图 1 中,将整个维修任务的结构用一张有向网络图表示,图中节点代表各个维修项目的工序阶段,弧线代表各项工序间的紧前关系,箭头上方表示具体工序。

图 1 突显出舰船维修任务的特殊性,在拆卸出舱阶段后,分成坞内修理工程和坞外修理工程 2 部分进行维修任务。在所有维修工程结束后,统一进入试航扫尾阶段。此时,坞内工程和坞外工程这 2 个节点都是虚活动,不占用任何维修资源,也不需要执行时间,其作用就是清晰易懂地表达网络图的结构。

$$\min \gamma = \sum_{x=0}^X s_{xN} \quad (1)$$

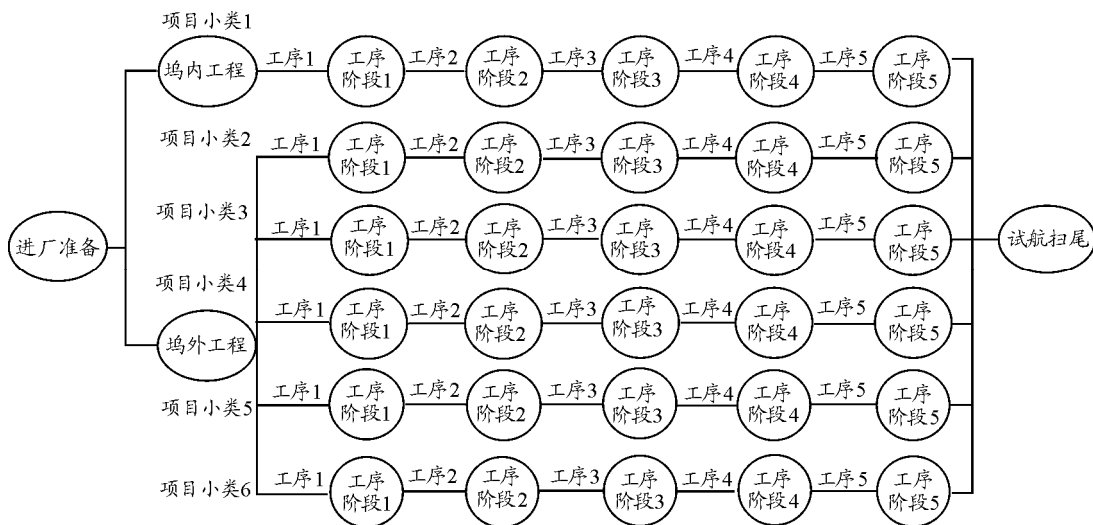


图 1 X 舰艇简易的维修网络计划示意图

3 蚁群算法的设计

蚁群优化 (ant colony optimization, ACO) 是意大利 Marco Dorigo 等学者在真实蚂蚁觅食行为的启发下提出的一种高度创新性的元启发式算法^[6-7]。蚁群算法的灵感来源于蚂蚁群体寻找食物的行为。某只蚂蚁从蚁穴出发寻找食物, 它会在蚁穴和食物源之间的路径上自身释放一种化学物质信息, 这种信息素会影响蚂蚁群体的路径选择。若在蚁穴和食物源之间有多条路径, 蚂蚁群体会根据路径上信息素的浓度来选择较优的路径。那么, 蚁群算法设计的人工虚拟蚂蚁, 在解决实际问题中, 探索不同的路径, 根据信息素浓度选择较优的路径。

3.1 主要思想

笔者主要采用精英蚂蚁系统 (elitist strategy for ant system, EAS) 来解决舰船维修资源优化调度问题。

把图 1 作为此问题的构建图, 而该构建图由所有代表需要调度的工序的结点 (工序阶段), 以及 2 个表示开始和结束的哑元结点组成 (进厂准备和试航扫尾)。其中, 每个工序都要被调度, 还要必须满足所有工序中的优先约束和资源约束。这里的信息素 τ_{ij} 指的是将工序 j 调度成序列中第 i 个工序的期望度。而启发式信息 η_{ij} 代表将工序 j 调度成序列中第 i 个工序的启发式期望度。由于本问题的优化调度目标是完成工期最短, 所以采用最迟可能开始时间优先规则 EST, 即该问题的启发式信息为:

$$\eta_{ij} = \max_{l \in N} LS_l - LS_j + 1$$

式中的 LS_j 是工序 j 的最迟可能开始时间, 而 N 是经过串行调度生成策略 (serial schedule generating scheme) 定义的一个部分调度后符合条件的工序集合。

EAS 算法还有 2 个主要步骤, 即蚂蚁构建问题的解和信息素的更新。这 2 个步骤是 EAS 算法解决舰船维修资源优化调度问题的关键所在。

3.2 解的构建

在每一次迭代过程中, 假设存在着 m 只蚂蚁, 每只蚂蚁都使用调度生成方法来构建一个调度序列, 而使用这个调度生成方法时要考虑本问题的资源需求。调度生成方法的作用就是使每只蚂蚁只建立包含工序集合中所有操作一次且仅一次的序列。

开始构建解时, 蚂蚁首先分配工序 0 到位置零处。当构建到步骤 i 时, 在已生成的调度序列中, 第 k 只蚂蚁用启发式信息 η_{ij} 和信息素 τ_{ij} 来计算 j 工序分配到序列中第 i 个位置上概率:

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \text{ 如果 } j \in N_i^k$$

这里的 α 和 β 是 2 个参数, 它们分别决定了信息素和启发式信息的相对影响力。 N_i^k 包含了所有到目前为止还没有被调度的工序。在这个概率规则下, 选择某一条边 (i,j) 的概率由该边所对应的信息素 τ_{ij} 以及启发式信息的值 η_{ij} 决定。

3.3 信息素的更新

当所有蚂蚁都构建好路径后, 各边上的信息素将会被更新。首先所有边上的信息素都会减少一个常量因子的大小, 然后在蚂蚁经过的边上增加信息素。信息素的蒸发:

$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij}, \forall (i,j) \in L$$

其中 ρ 是信息素的蒸发率, 有 $0 < \rho \leq 1$ 。参数 ρ 的作用就是避免信息素无限积累, 使算法忘记较差路径。而 L 是蚂蚁构建路径边的集合。

在信息素蒸发步骤之后, 通过信息素的释放得到更新后的信息素:

$$\tau'_{ij} = \tau_{ij} + \sum_{k=1}^m \Delta \tau_{ij}^k + e \Delta \tau_{ij}^{bs}$$

其中 $\Delta \tau_{ij}^k$ 是第 k 只蚂蚁向它经过的路径释放的信息素量, 如果边 (i,j) 在路径 T^k 上则 $\Delta \tau_{ij}^k = 1 / C^k$, 否则为 0; C^k 代表的是第 k 只蚂蚁建立路径 T^k 所代表的工期; 而 $e \Delta \tau_{ij}^{bs}$ 是指因算法从开始到目前为止找到的最优路径实施的一种强化手段而增加的信息素。 E 是一个参数, 定义路径 T^{bs} 的权值大小。

在 EAS 中有较多的参数变量, 经过许多科研学者多次的测试研究, 得到较好的参数设置, 如表 1。

表 1 AS 算法的参数设置

AS 算法 参数	参数 值	AS 算法 参数	参数 值	AS 算法 参数	参数 值
α	1	ρ	0.5	m	n
β	2~5	e	n		

通过以上方式, 获得较短工期的蚂蚁所经过的路径上将会遗留更多的信息素, 使得在以后的循环中, 蚂蚁会更容易地获得最优解。

4 算例

初始化数据：令舰船的数量 $a=4$ ，每艘舰船的维修项目数 $b=6$ ，每个维修项目所包含的工序数量 $c=5$ 。下面分别是 4 艘船舶每种维修项目中每项工序的执行时间，即 d_{ij} 指的是第 i 种项目第 j 项工序的执行时间。而这里的项目 1 专指坞内维修项目，其余维修项目都属坞外工程。

$$A_1 = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} & d_{15} \\ d_{21} & d_{22} & d_{23} & d_{24} & d_{25} \\ d_{31} & d_{32} & d_{33} & d_{34} & d_{35} \\ d_{41} & d_{42} & d_{43} & d_{44} & d_{45} \\ d_{51} & d_{52} & d_{53} & d_{54} & d_{55} \\ d_{61} & d_{62} & d_{63} & d_{64} & d_{65} \end{bmatrix} = \begin{bmatrix} 1 & 15 & 30 & 20 & 5 \\ 1 & 10 & 40 & 15 & 5 \\ 2 & 10 & 40 & 12 & 5 \\ 3 & 7 & 20 & 12 & 5 \\ 3 & 7 & 30 & 15 & 5 \\ 3 & 9 & 20 & 20 & 5 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} & d_{15} \\ d_{21} & d_{22} & d_{23} & d_{24} & d_{25} \\ d_{31} & d_{32} & d_{33} & d_{34} & d_{35} \\ d_{41} & d_{42} & d_{43} & d_{44} & d_{45} \\ d_{51} & d_{52} & d_{53} & d_{54} & d_{55} \\ d_{61} & d_{62} & d_{63} & d_{64} & d_{65} \end{bmatrix} = \begin{bmatrix} 1 & 13 & 28 & 18 & 7 \\ 2 & 11 & 45 & 10 & 3 \\ 2 & 8 & 36 & 13 & 5 \\ 3 & 9 & 19 & 17 & 3 \\ 1 & 7 & 40 & 10 & 5 \\ 3 & 6 & 25 & 21 & 7 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} & d_{15} \\ d_{21} & d_{22} & d_{23} & d_{24} & d_{25} \\ d_{31} & d_{32} & d_{33} & d_{34} & d_{35} \\ d_{41} & d_{42} & d_{43} & d_{44} & d_{45} \\ d_{51} & d_{52} & d_{53} & d_{54} & d_{55} \\ d_{61} & d_{62} & d_{63} & d_{64} & d_{65} \end{bmatrix} = \begin{bmatrix} 3 & 10 & 32 & 20 & 4 \\ 1 & 9 & 30 & 17 & 4 \\ 1 & 15 & 37 & 11 & 7 \\ 1 & 10 & 18 & 10 & 3 \\ 3 & 5 & 27 & 20 & 5 \\ 1 & 10 & 16 & 15 & 6 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} d_{11} & d_{12} & d_{13} & d_{14} & d_{15} \\ d_{21} & d_{22} & d_{23} & d_{24} & d_{25} \\ d_{31} & d_{32} & d_{33} & d_{34} & d_{35} \\ d_{41} & d_{42} & d_{43} & d_{44} & d_{45} \\ d_{51} & d_{52} & d_{53} & d_{54} & d_{55} \\ d_{61} & d_{62} & d_{63} & d_{64} & d_{65} \end{bmatrix} = \begin{bmatrix} 1 & 18 & 18 & 15 & 3 \\ 2 & 12 & 50 & 17 & 5 \\ 1 & 8 & 41 & 16 & 5 \\ 1 & 7 & 23 & 11 & 7 \\ 3 & 6 & 35 & 15 & 7 \\ 1 & 10 & 13 & 20 & 2 \end{bmatrix}$$

以下是表现每种维修项目各项工序的关系矩阵，即优先约束 C_{ij} 指的是第 i 项工序完成之后进行第 j 项工序的发生概率。

$$B = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} \\ c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \\ c_{41} & c_{42} & c_{43} & c_{44} & c_{45} \\ c_{51} & c_{52} & c_{53} & c_{54} & c_{55} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

R 是各种维修项目所对应的资源总量； r 是每艘船舶每种维修项目所对应的资源需求量。

$$R = [1 \ 2 \ 1 \ 2 \ 2 \ 1]$$

$$r = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} & r_{15} & r_{16} \\ r_{21} & r_{22} & r_{23} & r_{24} & r_{25} & r_{26} \\ r_{31} & r_{32} & r_{33} & r_{34} & r_{35} & r_{36} \\ r_{41} & r_{42} & r_{43} & r_{44} & r_{45} & r_{46} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

通过蚁群算法实现后，得到的优化结果如图 2、图 3。图 3 中，当蚂蚁数量分别取值 5、6、7、8、9、10 和循环次数分别为 1~10 的时候得到最短工期的收敛情况。实验结果说明：当蚂蚁数目超过 10 或者迭代次数超过 7 的时候，得到的解已经趋于定值，表明早已收敛。最后的优化结果如表 2。

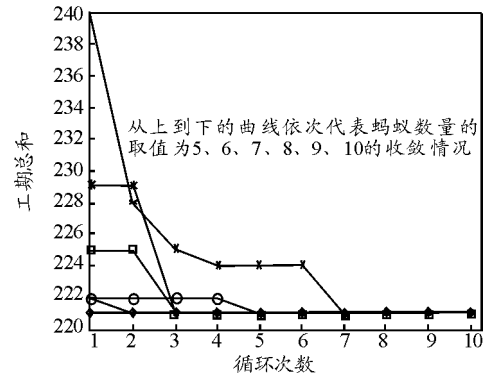


图 2 算法实现后的二维坐标图

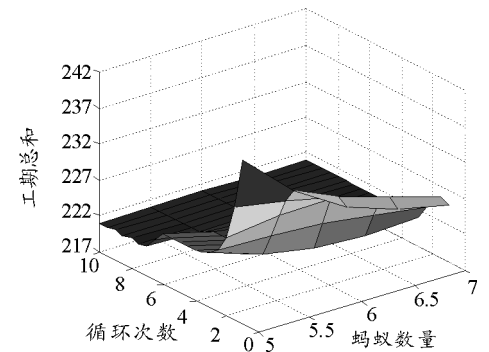
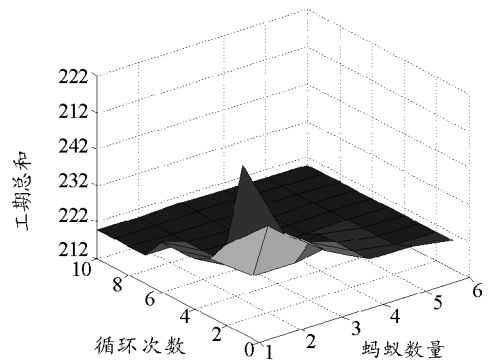


图 3 算法实现后的三维坐标图

表 2 循环 10 次的优化结果

蚂蚁数量	最短工期 /d	蚂蚁数量	最短工期 /d	蚂蚁数量	最短工期 /d
5	222	7	221	9	222
6	225	8	223	10	221

5 总结

通过用蚁群算法求解后, 达到了舰船维修资源优化调度的目的, 得到了较优的调度方案。用虚拟的人工蚂蚁对较短工期的路径进行搜索, 通过信息素的更新, 提高了算法的搜索能力, 从而增强了解的精确度。

建立舰船维修资源优化调度方案, 能够较好地解决维修过程中出现的维修资源冲突问题, 缩短了舰船维修的工期, 是一种科学的决策方法。

参考文献:

[1] 杨晶, 曾斌. 基于遗传算法的维修任务调度优化及仿真[J]. 计算机工程, 2009, 35(18): 243.

[2] Boctor, F.F. Resource-constrained project scheduling by simulated annealing[J]. International Journal of Production Research, 1996, 34: 2335-2351.

[3] Artigues, C., Michelon, P., Reusser, S. Insertion techniques for static and dynamic resource-constrained project scheduling[J]. European Journal of Operational Research, 2003, 149: 249-267.

[4] 杨利宏, 杨东. 基于遗传算法的资源约束型项目调度优化[J]. 管理工程, 2008, 21(4): 61-67.

[5] 郑超, 高连生. 蚁群算法在资源受限项目调度问题中的应用[J]. 计算机工程与应用, 2005, 27: 205-208.

[6] Marco Dorigo, Thomas Stutzle. Ant Colony Optimization [M]. 北京: 清华大学出版社, 2007(1): 161-174.

[7] 陈城辉, 徐永能, 杨爱梅, 等. 遗传算法在多车型军备物资配送路径优化中的应用[J]. 四川兵工学报, 2010, 31(2): 13.

(上接第 11 页)

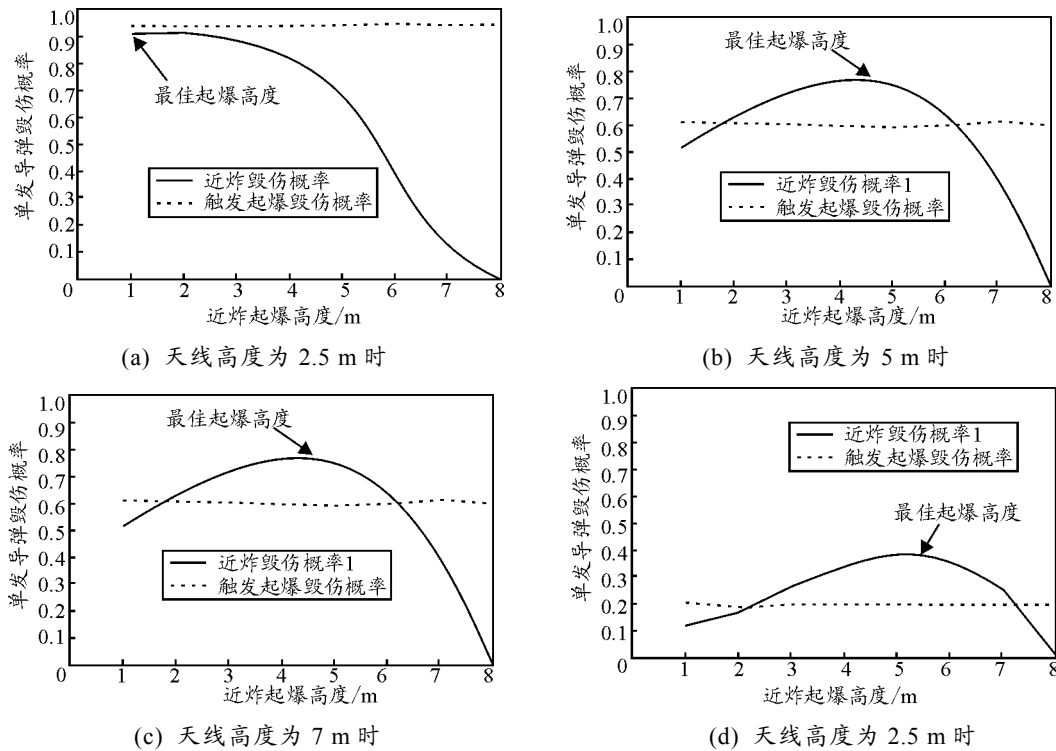


图 4 单发导弹毁伤概率

6 结束语

通过 Monte-Carlo 仿真方法, 求出了不同天线高度所对应的最佳起爆高度。仿真结果表明: 针对不同类型目标, 应采用不同引战配合方式, 才能获得较好毁伤效果。

参考文献:

[1] 曹柏楨. 飞航导弹战斗部与引信[M]. 北京: 宇航出版社, 1995: 237-238.

[2] 谷良贤, 温炳恒. 导弹总体设计原理[M]. 西安: 西北工业大学出版社, 2004: 217-224.

[3] 谢邦荣, 尹健, 宋劲松. ARM 导弹对雷达目标的毁伤概率仿真建模[J]. 系统仿真学报, 2004, 16(9): 2044-2051.

[4] 隋树元, 王树山. 终点效应学[M]. 北京: 国防工业出版社, 2000: 293-303.

[5] 张国伟. 终点效应及靶场试验[M]. 北京: 北京理工大学出版社, 2009: 174-181.

[6] 丁永丽, 李应岐, 田野. 导弹武器系统效能评估的模糊 AHP 评判模型[J]. 四川兵工学报, 2010, 31(6): 31.