

用离散粒子群优化算法求解 WTA 问题

曲在滨^{1,2}, 刘彦君², 徐晓飞¹

(1. 哈尔滨工业大学 计算机科学与技术学院, 150001 哈尔滨, zbqu@hotmail.com;

2. 哈尔滨理工大学 计算机科学与技术学院, 150080 哈尔滨)

摘 要: 为了提高武器-目标分配(WTA)问题的求解效率和性能,提出了一种用离散粒子群(DPSO)算法求解此问题的新方法.对粒子群算法中的速度和位置进行了重新定义,使其可求解 WTA 这类离散组合优化问题,并采用贪心的启发式策略对迭代产生的方案进行调整,以利于快速找到最优或次优的分配方案.算法测试表明新算法执行速度快,结果令人满意,提出的算法能快速给出 WTA 问题的最优或近优分配方案.

关键词: 武器-目标分配;约束优化;粒子群优化算法;遗传算法

中图分类号: TP301.6

文献标志码: A

文章编号: 0367-6234(2011)03-0067-03

Discrete particle swarm optimization for solving WTA problem

QU Zai-bin^{1,2}, LIU Yan-jun², XU Xiao-fei¹

(1. School of Computer Science and Technology, Harbin Institute of Technology, 150001 Harbin, China, zbqu@hotmail.com;

2. School of Computer Science and Technology, Harbin University of Science and Technology, 150080 Harbin, China)

Abstract: To improve the solving efficiency and performance, a new Discrete Particle Swarm Optimization (DPSO) algorithm for the problem of weapon-target assignment (WTA) was presented, in which the velocity and position in the original PSO formulas were redefined to solve a discrete combination optimization such as WTA. The greedy heuristic policies were exploited to adjust the solutions generated during the execution of the algorithm to find optimal or suboptimal solution rapidly. The experiments show that the new algorithm can get the optimal or suboptimal solution quickly, and the result is satisfying.

Key words: Weapon-target Assignment (WTA); constrained optimization; Particle Swarm Optimization (PSO) algorithm; Genetic Algorithm (GA)

武器-目标分配(Weapon-target Assignment, WTA)问题是现代战争中一个十分重要的问题,其解空间随武器种类数和目标总数的增加而呈指数级的增加,使其成为一个多参数、多约束 NP 完全问题^[1-2].对于 WTA 问题,国内外主流的求解方法多采用诸如神经网络、遗传算法、模拟退火算法和蚁群优化等求次优解方法,但它们的求解效率都还不是十分令人满意,所以探索有效的求解方法解决 WTA 问题仍然十分重要.

目前,关于单一兵器火力分配问题的文献已比较多,然而由于在现代作战中,经常是多种类型的兵器联合作战,故需对多种不同类型的兵器的

武器-目标分配问题进行探讨.文献[3-6]虽然解决的是多种武器类型的火力分配问题,但所建立模型还是一种简化模型,在实际使用中一般需添加额外约束条件,如对各目标分配的武器数目限制等.本文针对多类型兵器的火力分配问题给出更实用的数学模型,并将粒子群算法求解问题的思路应用于此问题,用实例验证了方法的可行性及有效性.

1 问题的描述

具体问题可描述为:设有 n 个目标; m 种类型武器; V_j 为目标 j 的威胁度; W_i 为可分配给目标的 i 类型武器数量; p_{ij} 为 i 类型的一个武器对目标 j 的杀伤概率; x_{ij} 为给目标 j 分配 i 类型武器的数量,目标 j 最多可分配武器数量为 N_j . WTA 问题是要

收稿日期: 2010-01-21.

作者简介: 曲在滨(1962—),男,博士研究生,副教授;

徐晓飞(1962—),男,教授,博士生导师.

确定分配给各目标的各类武器数量以使所有目标的总期望生存值最小, 这个问题可形式化为

$$\begin{aligned} & \text{minimize } \sum_{j=1}^n V_j \left(\prod_{i=1}^m q_{ij}^{x_{ij}} \right); \\ & \text{s. t. } \sum_{j=1}^n x_{ij} \leq W_i, \quad i = 1, 2, \dots, m; \\ & \quad \sum_{i=1}^m x_{ij} \leq N_j, \quad j = 1, 2, \dots, n; \\ & x_{ij} \geq 0 \text{ 且为整数, } i = 1, 2, \dots, m, j = 1, 2, \dots, n. \end{aligned}$$

式中 $q_{ij} = 1 - p_{ij}$, 即目标 j 受到一发 i 类型武器打击的生存概率。

2 粒子群优化算法

粒子群优化 Particle Swarm Optimization (PSO) 算法是由 Kennedy 等^[7] 在 1995 年提出的, 此算法受鸟群觅食行为的启发, 并用于解决优化问题。同遗传算法相比, PSO 算法不但具有遗传算法的全局寻优能力, 还具有较强的局部寻优能力。

算法将每个个体看作是在 n 维搜索空间中的一个没有重量和体积的粒子, 并在搜索空间中以一定的速度飞行。每个粒子的飞行速度根据其本身的飞行经验和群体的飞行经验不断调整。每个粒子代表解空间的一个候选解, 解的优劣程度由适应函数决定, 而适应函数根据优化目标定义。PSO 随机初始化一群粒子, 其中第 i 个粒子在 n 维解空间的位置为 $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ 。粒子通过在每一次迭代中动态跟踪 2 个极值来更新其速度和位置为:

1) 粒子本身从初始到当前迭代搜索产生的最优解: 个体极值 $p_i = (p_{i1}, p_{i2}, \dots, p_{in})$;

2) 整个粒子种群目前的最优解: 全局极值 $p_g = (p_{g1}, p_{g2}, \dots, p_{gn})$ 。

如果第 i 个粒子的速度表示为 $v_i = (v_{i1}, v_{i2}, \dots, v_{in})$, 粒子根据式(1), 式(2) 来动态调节自己的“飞行”, 得到搜索问题的最优解为

$$V_{id}(t+1) = \omega V_{id}(t) + c_1 r_1 (p_{id}(t) - X_{id}(t)) + c_2 r_2 (p_{gd}(t) - X_{id}(t)). \quad (1)$$

$$X_{id}(t+1) = X_{id}(t) + V_{id}(t+1), \quad i = 1, 2, \dots, n. \quad (2)$$

式中: t 为迭代次数; c_1, c_2 分别为正常数, 称为学习因子, 一般有 $c_1 = c_2 = 2$; r_1, r_2 分别为 $(0, 1)$ 之间的随机数; ω 为惯性因子。

3 离散 PSO 算法求解 WTA 问题

3.1 约束处理方法

WTA 问题的解可表示为一个矩阵 $X =$

$(x_{ij})_{m \times n}$, 其中 x_{ij} 为第 i 类武器分配给第 j 个目标的数目。在这里, 每个粒子的位置便对应着武器的一种分配方案。对粒子群迭代中产生的不可行解采用修正结果值方案的方法进行修正, 使其在可行解空间范围内搜索。具体实现时利用一个可行性函数检查新产生的解是否满足所有约束。当不满足时, 采用启发式策略进行调节。调节策略为:

1) 如果分配方案中某一目标的武器分配总数超过约束限制, 则取消对此目标杀伤概率最小的超出数目的武器分配; 如果此种武器分配数少于超出数目, 则需取消此种武器的全部分配。重复步骤 1), 直到各目标满足各自的武器分配总数约束。

2) 如果分配方案中某种武器的分配不满足约束, 则在已分配这种武器的目标中选择“威胁度与此种武器对此目标杀伤概率之和”最小的目标, 取消超出单位数的武器分配; 如果超出单位数大于分给此目标的武器数, 则余下部分再从占用此种武器且满足选择准则的目标中取消分配。重复步骤 2), 直到各种武器分配满足约束为止。

本启发式策略运用了贪心思想, 尽可能使所做调节有利于提高武器对目标打击的有效性, 从而加速找到最优或次优的分配方案。

3.2 离散 PSO 算法实现

本文对粒子群算法中的速度和位置进行了重新定义。对速度进行的重新定义为

$$V_i(t+1) = r_1 [P_{lbest}(t) - X_i(t)] \oplus r_2 [P_{gbest}(t) - X_i(t)]. \quad (3)$$

式中: $r_1 [P_{lbest}(t) - X_i(t)]$ 为第 i 个粒子的位置 X_i 的任一列以概率 r_1 与其个体极值 $P_{lbest}(t)$ 的对应列做减法运算, 未做减法运算的列一律清零; $r_2 [P_{gbest}(t) - X_i(t)]$ 为第 i 个粒子的位置 X_i 的任一列以概率 r_2 与全局极值 $P_{gbest}(t)$ 的对应列做减法运算, 未做减法运算的列一律清零; $\oplus$ 为对作为操作数的 2 个矩阵产生 1 个新的矩阵, 当 2 个操作数矩阵的对应列所有元素均为 0, 则结果矩阵的相应列也为全 0; 如果只有一列不全为 0, 则此列为结果矩阵的对应列; 否则任选其中一列为结果矩阵的对应列。

上述定义解决了速度难于表示问题, 而进行位置的更新为

$$X_i(t+1) = X_i'(t) + V_i(t+1). \quad (4)$$

式中: $X_i'(t)$ 为将 $X_i(t)$ 随机选择一行根据相应类型武器数目对各目标产生随机分配而其余各行保持不变的结果, 而“+”运算则为 2 个矩阵对应元素相加。

求解 WTA 问题的离散 PSO 算法 (DPSO - WTA)描述为:

- 1) 初始化粒子群,即随机设定各粒子的初始位置 X 和速度 V ;
- 2) 计算每个粒子的个体适应度值(即目标函数值);
- 3) 将每个粒子的适应度值与个体极值 P_{lbest} 比较,若更好,则更新 P_{lbest} ;
- 4) 将每个粒子的适应度值与全局极值 P_{gbest} 比较,若更好,则更新 P_{gbest} ;
- 5) 根据式(3)和式(4)更新粒子的速度和位置,并使用启发式策略调整新位置;
- 6) 如未达到最大的迭代次数,则返回步骤 2);否则结束.

算法在对每一粒子初始化时根据各类武器的数目随机产生分配方案,且使其满足各约束条件.

4 算法测试

假设有 6 个目标, $T_1, T_2, \cdots, T_6$,各目标的威胁度如表 1 所示. 5 种武器: W_1, W_2, W_3, W_4, W_5 的数目为 $\{4,3,3,2,2\}$,对各目标最多可使用武器数目为 $\{1,1,2,1,1,2\}$,各类武器对各目标的单发杀伤概率如表 2 所示.

表 1 各目标的威胁性系数

目标	1	2	3	4	5	6
威胁性系数	1.0	0.9	0.7	1.0	0.5	0.8

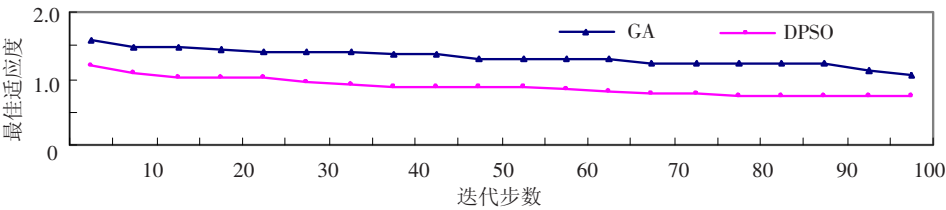


图 1 适应度函数值(平均值)变化情况

需平均时间为 217.3 ms,所得最佳方案的适应度平均值为 1.061,与最优值相差 0.427,可明显看出离散 PSO 算法大大优于遗传算法,求解精度也明显高于文献[8]提出的粒子群优化算法.

经过对大量实例的计算发现,随着武器和目标数的增加,所提出离散粒子群算法对遗传算法的优势也越发明显.图 2 给出当目标数为 15,武器种类数为 5,且每类武器数为(4,6,5,5,4)时 2 种算法执行时的最佳适应度值随时间变化情况,此例分配方案的适应度最优值为 1.404.总而言之,使用本文提出的离散粒子群优化算法能

表 2 武器对目标的杀伤概率表

武器	目标					
	1	2	3	4	5	6
1	0.5	0.7	0.2	0.5	0.8	0.4
2	0.4	0.1	0.8	0.7	0.6	0.9
3	0.1	0.7	0.3	0.7	0.6	0.8
4	0.8	0.3	0.2	0.8	0.2	0.1
5	0.4	0.9	0.5	0.7	0.6	0.2

算法具体实现时,为避免陷入局部最优,若全局极值在进化一定代数后保持不变,则只保留最优粒子,其他粒子全部重新生成.本例中的粒子数取为 30,最大迭代数为 100.在 P4 1.7 G/512 M 微机上重复执行算法 50 次得出的最优方案的适应度值情况如表 3 所示,最优分配方案的适应度函数值为 0.634,与枚举算法所得分配方案的适应度函数值相同.

表 3 最优方案的适应值情况

最佳值	最差值	平均值
0.634	0.976	0.732

进化过程中的适应度函数值(平均值)变化情况如图 1 所示,图 1 中也同时给出与遗传算法(GA)的对比.对于此例,采用枚举算法求解大约需 1 142 s,而提出的离散粒子群算法迭代 100 代所需平均时间为 156.9 ms,所得最佳方案的适应度平均值为 0.732,与最优值相差 0.098;对遗传算法进行同样次数独立实验,每次迭代 100 代所

快速给出武器 - 目标分配问题的最优或近优解,验证了算法的有效性.

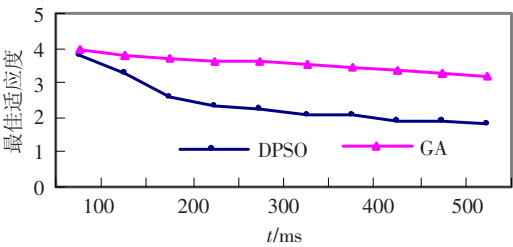


图 2 适应度函数值(平均值)随时间变化情况