文章编号:1671-4598(2015)05-1706-04 DOI:10.16526/j.cnki.11—4762/tp.2015.05.077 中图分类号:TP311.13 文献标识码:A

军械装备保障数据仓库中的ETL过程优化研究

谢 峰,孙江生,张西山
(军械技术研究所,石家庄 050000)

摘要:针对军械装备保障数据仓库中复杂的ETL过程,提出采用改进的粒子群算法进行ETL任务调度的优化策略;通过改进惯性向量,使其具备动态调整能力,形成更具寻优特性的动态w粒子群算法(DWPSO);在对数据仓库ETL调度过程进行数学化描述的基础上,将改进算法应用在以最小执行时间为目标函数任务调度中,并通过仿真实验证明了该算法的有效性。

关键词:数据仓库;ETL;粒子群

Research on Ordnance Equipment Support Data Warehouse ETL Process Optimizing

Xie Feng, Sun Jiangsheng, Zhang Xishan
(Ordnance Technical Research Institute, Shijiazhuang 050000, China)

Abstract: The structure of data warehouse ETL Process is studied in this paper. An improved method is put forward to optimize scheduling process for lifting efficiency of ETL process. By improving inertia vector, the performance of PSO gets promoted. Experiments proved that the improved PSO is much better than traditional algorithm. The use of DWPSO improves the performance of ETL process in ordnance equipment support data warehouse.

Keywords: data warehouse; ETL; PSO

0 引言

近年来,随着装备保障信息化的快速发展,围绕军械装备保障所产生的数据量急速增长,为实现数据的综合利用,军械装备保障数据仓库得以构建。军械装备保障数据仓库实现了对现有保障信息系统的数据集成,形成了覆盖全地域、全部门的数据集成应用体系。任务调度优化是数据抽取、转换、加载过程(ETL, extract transform load)高效执行的关键问题之一[1]。本文采用改进粒子群算法,对该过程实施优化,并通过仿真实验证明了算法的有效性。

1 ETL调度优化框架

任务调度、分配问题是典型的NP问题[2]。对于此类问题的求解,许多学者都提出了自己的研究思路。部分学者以遗传算法为技术手段进行此类问题的求解[3-4]。文献[5]给出了ETL执行流水线的优化方法,但该方法是以ETL各活动串行约束为前提,在通用性上存在一定欠缺。文献[6]采用贪婪算法,对ETL调度任务进行优化,这种调度方式没有细化到任务中的操作级,只有限应用与单个ETL任务。此外,还有部分研究着眼于ETL执行过程优化[7-8],其核心思想是通过减少或更改处理操作的方式,优化ETL执行流程,这种方式并未从ETL优化核心上进行改变,流程方法针对不同的数据仓库各有不同,通用性不强。基于以上研究,本文提出以粒子群算法解决ETL调度优化问题,该算法除在连续函数最优值求解上广泛应用外,在离散空间领域任务调度活动中也有成熟应用,通过建立连续空间与离散空间的映射模式,以连续空间的方法解决离散问题。

1.1 ETL多任务调度问题模型描述

ETL过程是相互独立的ETL任务集合,每个任务由多个具有时间序列的具体操作组成,形成一个完成的数据处理过程。其过程如图1所示。

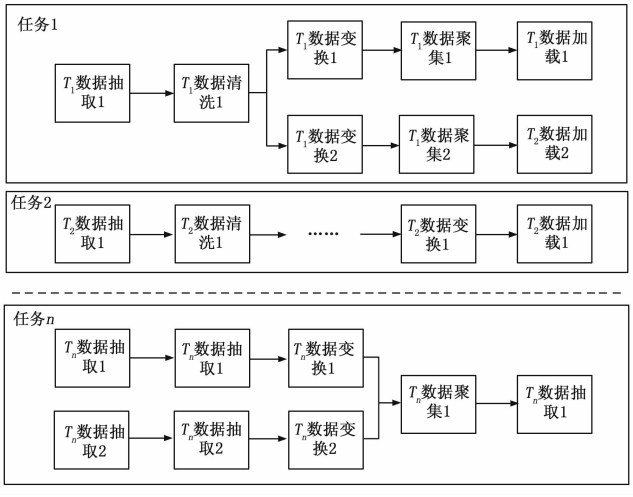

图1 数据仓库ETL任务组成

使用粒子群算法,解决ETL多任务分配的问题,需要对ETL过程抽象建模,该问题的数学化描述如下:

ETL集群系统配有处理机m台,准备对n个不同的处理

收稿日期:2014-09-03; 修回日期:2014-10-13。

作者简介:谢 峰(1986-),男,山东泰安人,硕士,主要从事装备保障信息化方向的研究。

任务 $Task_i$，每个处理任务包含处理操作 $Deal_{ij}$。其中，$Task_i$：处理任务 $Task$ 是数据处理的基本单元；$Deal_{ij}$：代表处理任务 $Task_i$ 的第 j 个处理操作；M：ETL 处理机，一个 ETL 处理机可以是一台 PC 或是一台服务器，代表处理任务的操作单元。为了便于通过数学方法解决 ETL 任务调度问题，做如下假设：

1）每个处理任务的操作具有有序性，即在完成上一操作后才能进行之后处理过程；

2）处理任务的一个操作不能同时在两个处理机上实施；

3）处理任务的单一操作开始以后就不会被其他操作所打断；

4）每个处理任务的一个处理操作只能实施一次，不可重复单一处理过程。

5）同一时刻，每台 ETL 处理机只能从事单一操作过程。

在已建立的约束条件下，根据数学模型，建立如下的调度函数：

$$\min T_{ts} = \max T_{ts}(M_i) \tag{1}$$

$$Q(Deal_{ijk}, Deal_{i'j'k}) = 1 \Leftrightarrow t_e(Deal_{i'j'k}) - t_s(Deal_{ijk}) \geqslant 0 \tag{2}$$

$$x_{ijk} \times x_{ijk'} = 0 \tag{3}$$

$$P(Deal_{ij}, Deal_{ik}) = 1 \Leftrightarrow t_e(Deal_{ijk'}) - t_s(Deal_{ijk'}) \geqslant 0 \tag{4}$$

1）处理操作序列序：

$$P(Deal_{ij}, Deal_{ik}) = \begin{cases} 1 Deal_{ij} \text{ 先于 } Deal_{ik} \text{ 执行} \\ 0 Deal_{ij} \text{ 后于 } Deal_{ik} \text{ 执行} \end{cases}$$

2）处理操作执行序列：

$$Q(Deal_{ijk}, Deal_{i'j'k}) = \begin{cases} 1 \ Deal_{ij} \text{ 先于 } Deal_{i'j'} \text{ 在处理机 } k \text{ 上执行} \\ 0 \ Deal_{ij} \text{ 后于 } Deal_{i'j'} \text{ 在处理机 } k \text{ 上执行} \end{cases}$$

3）　指示变量：

$$x_{ijk} = \begin{cases} 1 & \text{第 } i \text{ 个任务的 } j \text{ 操作在处理机 } k \\ 0 & \text{非上述情况} \end{cases}$$

4）ETL 完成时间：

$T(Deal_{ijk}) = t_e(Deal_{ijk}) - t_s(Deal_{ijk})$，该表达式表示第 i 个任务的第 j 个操作 $Deal_{ijk}$ 在第 k 个处理机上的处理操作时间。$t_s(Deal_{ijk})$ 代表了操作开始的时间，$t_e(Deal_{ijk})$ 代表了操作结束的时间，

5）处理机完成时间：

$T_{ts}(M_i)$，在调度方案 ts 下，处理机 M_i 完成所安排的处理任务的时间总和。

公式（1）为任务调度的目标函数，即最小化最大完成时间；公式（2），确保同一时刻，某个 ETL 处理机只能执行一个操作；公式（3），确保同一时刻，ETL 任务中的某一处理操作智能在一个处理机上执行；公式（4），决定了处理操作的有序性，所有的 ETL 处理操作，必须按照预先设定的执行顺序进行。

1.2　ETL 任务调度中的编码设计与更新

利用粒子群算法对调度任务进行求解时，必须以编码的方式，对离散的调度方式进行编码，适应 ETL 调度问题的特点，建立编码方式如下。

处理任务总处理操作数可以定义为 $Z = \sum_{j=1}^{n} n_j$，把每个粒子定义为一个 $2Z$ 维度的向量。为了直观展示粒子，对粒子的 $2Z$ 向量作以处理操作，分为 $X_{deal}[Z]$ 和 $X_{device}[Z]$ 两个分向量。$X_{deal}[Z]$ 由 Z 个非 0 自然数组成，其顺序决定了处理序列调度的顺序。假定某 ETL 活动包含 3 个处理任务，每个处理任务需完成两个处理操作，则编码为［1，2，2，3，3，1，1，2，3］，第一个“1”表示处理任务 1 的第一个处理操作。第二个“1”表示处理任务 1 的第二个处理操作，第三个“1”表示处理任务 1 的第三道工序。$X_{device}[Z]$ 表示各处理操作的处理机号。

对于粒子群中的粒子，速度向量 $V[Z]$ 可以表示为 $V_{\text{deal}}[Z]$ 和 $V_{\text{device}}[Z]$，按照粒子群更新方程对 $V_{\text{deal}}[Z]$、$V_{\text{device}}[Z]$ 和 $X_{\text{deal}}[Z]$、$X_{\text{device}}[Z]$ 进行更新。

2　改进粒子群算法在 ETL 调度中的应用

2.1　改进的粒子群优化算法

标准粒子群的速度及位置更新公式如下：

$$v_{id}(t+1) = wv_{id}(t) + c_1 r_1(p_{id}(t) - x_{id}(t)) + c_2 r_2(g_d(t) - x_{id}(t)) \tag{1}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), 1 \leqslant i \leqslant N, 1 \leqslant d \leqslant D \tag{2}$$

其中：w 为惯性权重，c_1，c_2 为加速因子，用于调节自身及全局的飞行步长。r_1，r_2 为［0，1］间的随机数。

实践证明，当数据源数量增大，迭代次数增多时，标准 PSO 会出现早熟收敛及迭代运行后期优化效率不高等问题，而惯性权重 w 的改变，可以对这种现象进行有效缓解。为了优化寻优过程，借鉴文献［8］的改进思想，本文建立了可动态调节惯性权重的方法，对 w 进行动态调节。在公式（1）、（2）基础上，更新如下方程：

$$v_{id}(t+1) = w(t)v_{id}(t) + c_1 r_1(p_{id}(t) - x_{id}(t)) + c_2 r_2(g_d(t) - x_{id}(t)) \tag{3}$$

$$w(t) = \left(\frac{1}{1 + e^{\frac{t+T_m}{h}}}\right) w_s \tag{4}$$

式中，w_s 为初始化权重值，根据以往实验，当 w 在取值区间在［0.4，0.9］时，更新方程具有更好的寻优特性，所以将 w_s 取值区间设定此区间。t 为当前迭代次数，T_m 为最大迭代次数，h 为优化常数，通常取值 0.5。其中 T_m 和 h 均由系统根据实验需求、相关实验数据及经验事前设定。

在区间［0，T_1］，$w(t)$ 取值较大，方程全局搜索能力更强，在区间［T_2，T_m］，$w(t)$ 取值较小，更利于方程的精细化寻优本文对 PSO 的改进思想便是对惯性权重 w 的动态调整，如图 2 所示，当初始运行时，种群处于最多样化状态，需要较大 w 值获得较强的全局搜索能力；随着迭代次数的增加，整体进入局部精细搜索状态，这时通过随迭代次数 t 动态调节 w 值，使得系统保持最佳优化状态，以便快搜索到最优值。

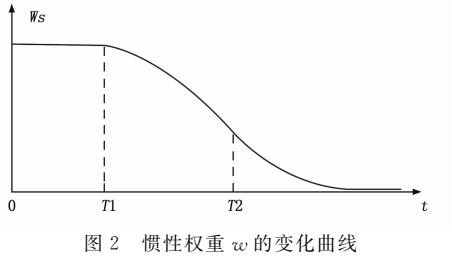

图 2　惯性权重 w 的变化曲线

2.2　算法的有效性验证

为验证本文所提出的 DWPSO（dynamic w particle swarm

optimization）算法的适用性能，选取与 ETL 任务调度寻优过程相似的典型 Schwefel 函数为测试对象，该函数为多峰多极点函数，且局部最优值和全局最优值较为相似，与 ETL 调度中的实际方案选择情况相吻合。运用 Matlab 进行仿真，将本文算法与其他相关算法：标准 PSO[9]、LDPSO[10] 及 CPSO[11] 进行对比验证。

Schwefel 函数数学表达式为：

$$f_1(x) = -\sum_{i=1}^{n}(x_i \sin(\sqrt{|x_i|})) - 500 \leqslant x_i \leqslant 500 \quad (5)$$

$$\min(f_1) = f(420.9687,..,420.9687) = -837.96. \quad (6)$$

其函数表示图如图 3 所示。

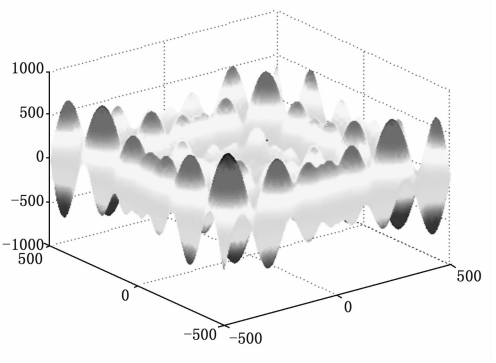

图 3 Schwefel 函数

仿真时，各算法的初始参数设置为：粒子群大小 N 设为 50，粒子以函数 $f(x)$ 的取值范围为搜索区间，并粒子的最大速度设定为其长度的 25%，经实验数据分析将本文所提的 DWPSO 算法和 LDPSO 算法中的惯性权重 w 的变动区间设为 [0.4 0.9]，本实验将测试函数的迭代次数定为 200，共运行 50 次，设速度因子 $C_1 = C_2 = 2$，*DWPSO* 算法的控制系数 $h = 0.5$。对于测试函数 $f(x)$，当误差小于 0.01 时，视为搜索出最优解。本实验通过对检验搜索测试函数的最优值平均运行时间以及收敛效率作为算法的适应度从而进行评估，运行结果如图 4 所示：将 4 种 PSO 算法分别对测试函数进行优化后的适应度曲线对比情况。由图中曲线变化分析可得：收敛速度最慢的是标准 PSO 算法，且其能够达到的精度也最低；另外两种算法的改进也都能实现快速搜索且全局性较好，而 DWPSO 算法在良好的搜索全局的性能基础上，相比于另外 3 种 PSO 算法能实现平均收敛时间最短，收敛速度明显加快，且迭代次数较小，寻优精度更高，在用时最短情况下能寻找到函数的最优解，同时，跳出局部最优点的能力较强，明显提高了算法的适用性能。

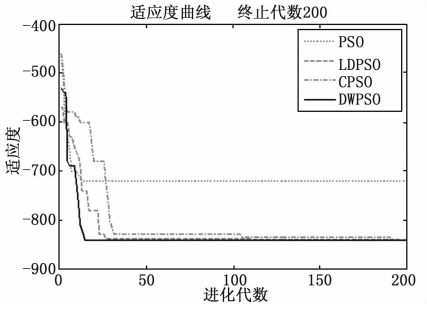

图 4 Schwefel 函数优化曲线比较

3 调度优化实验对比

文献 [12] 中采用遗传算法进行任务调度，取得良好的应用。为了检验本文所提改进粒子群算法的有效性，与该文献所采用的方法进行对比，通过实验的方式对比本文所提方法的优势。以 $n=6$，$m=6$ 条件为例，其初始化设定时间 $\boldsymbol{T}=$

$$\begin{bmatrix} 1 & 3 & 6 & 7 & 3 & 6 \\ 8 & 5 & 10 & 10 & 10 & 4 \\ 5 & 4 & 8 & 9 & 1 & 7 \\ 5 & 5 & 5 & 3 & 8 & 9 \\ 9 & 3 & 5 & 4 & 3 & 1 \\ 3 & 3 & 9 & 10 & 4 & 1 \end{bmatrix}$$

，以最小化最大完工时间 $\boldsymbol{T}$ 为优化目标来进行仿真实验。主要参数如下：粒子群规模为 40；$C_1 = C_2 = 2$，最大代数取 100；*GA* 的种群规模 $N=40$，变异概率 $p_m=0.01$，交叉概率 $p_c=0.9$，两种算法各运行 30 次，实验结果见表 1。

表 1 GA 与 DWPSO 实验结果对比

序号	GA		DWPSO	
	结果	用时	结果	用时
1	57	20.01	55	19.87
2	59	19.98	57	20.12
3	55	19.68	56	20.15
4	58	23.01	55	22.03
5	58	22.20	58	19.56
6	59	23.14	59	22.53
7	56	21.47	57	20.36
8	56	20.53	55	19.69
9	57	22.61	58	21.79
10	55	20.14	59	19.63
11	55	21.38	57	20.65
12	56	19.88	56	19.63
13	57	20.12	55	19.87
14	55	20.02	58	21.03
15	58	21.98	56	22.05
16	56	19.63	55	19.35
17	55	21.03	58	21.23
18	58	21.89	59	20.98
19	58	22.31	55	20.13
20	59	20.16	55	21.65

从实验的结果可以得出结论，使用 GA 对该 ETL 过程进行任务调度时，完成平均时间为 21.06，结果平均值为 56.85，达到最优值 55 的次数为 5。而使用笔者提出的 DWPSO 进行仿真实验时，完成平均时间为 20.61，结果平均值为 56.65，达到最优值次数为 7。说明该改进的粒子群算法无论是在收敛速度还是最优值寻解精度上，都比 GA 方法有优势。表 2 给出了一个可行的调度解。

4　结论

针对军械装备保障的复杂体系，军械装备保障数据仓库 ETL 需要面对大量的处理任务。本文针对 ETL 调度寻优问题进行研究，提出使用改进的粒子群算法优化 ETL 过程，通过将 ETL 任务进行数学描述，使用 DWPSO 成功解决了任务调度寻优问题。实践证明，该算法相比现阶段使用的标准粒子群算法更具寻优特性，并在改进军装装备保障数据仓库 ETL 过程的实践应用中起到良好效果。

表 2　最短调度时间任务分配表

处理机/操作	1	2	3	4	5	6
M_1	($Deal_1$, 1,4)	($Deal_4$, 13,18)	($Deal_3$, 18,27)	($Deal_6$, 28,38)	($Deal_2$, 38,48)	($Deal_5$, 48,51)
M_2	($Deal_2$, 0,8)	($Deal_4$, 8,13)	($Deal_6$, 13,16)	($Deal_1$, 16,18)	($Deal_5$, 22,25)	($Deal_3$, 27,28)
M_3	($Deal_1$, 0,1)	($Deal_3$, 1,6)	($Deal_2$, 8,13)	($Deal_5$, 13,22)	($Deal_4$, 22,27)	($Deal_6$, 42,43)
M_4	($Deal_3$, 6,10)	($Deal_6$, 16,19)	($Deal_4$, 27,30)	($Deal_1$, 30,37)	($Deal_2$, 48,52)	($Deal_2$, 52,53)
M_5	($Deal_2$, 13,23)	($Deal_5$, 25,30)	($Deal_4$, 30,38)	($Deal_6$, 38,42)	($Deal_3$, 42,49)	($Deal_1$, 49,55)
M_6	($Deal_3$, 10,18)	($Deal_6$, 19,28)	($Deal_2$, 28,38)	($Deal_1$, 38,41)	($Deal_2$, 41,45)	($Deal_4$, 45,54)

参考文献：

[1] Inmon W H. Building the Data Warehouse [M]. 4th ed. Indiana, USA: Wiley Publications, 2005.

[2] Hironori Kasahara, Seinosuke Narita. Practical multiprocessor scheduling algorithms for efficient parallel processing [J]. IEEE Trans on Computers, 1984, 33 (11): 1023－1029.

[3] Edwin S H. Hou, Nirwan Ansari. Genetic algorithm for multiprocessor scheduling [J]. IEEE Trans on Parallel and Distributed Systems, 1994, 5 (2): 113－120.

[4] 钟求喜，谢　涛，陈火旺. 基于遗传算法的任务分配与调度 [J]. 计算机研究与发展，2000，37 (10)：1197－1203.

[5] 韩京宇，徐立臻，董逸生. ETL 执行的流水线优化 [J]. 小型微型计算机系统，2005，26 (6)：134－138.

[6] 王　珊，陈　琨. ETL 中基于贪婪算法的任务调度方法研究 [J]. 微电子学与计算机，2009，26 (7)：130－133.

[7] 吴远红. ETL 执行过程的优化研究 [J]，计算机科学，2007，34 (1)：81－83.

[8] 姚全珠，赵双瑞. 基于状态空间搜索的 ETL 过程优化 [J]. 计算机工程与应用，2007，43 (26)：169－173.

[9] Karagiannis A, Vassiliadis P, Simitsis A. Scheduling strategies for efficient ETL execution [J]. Information Systems, 2012.

[10] Shi Y, Eberhart R C. Empirical study of particle swarm optimization [A]. In Proceedings of the 1999 Congress on Evolutionary Computation [C]. Piscataway, NJ, IEEE Service Center, 1999: 1945－1950.

[11] Krusienski D J, Jenkins W K. A modified particle swarm optimization algorithm for adaptive filtering [J]. IEEE Circuits and Systems, 2006, 21 (24): 136－140.

[12] 宋旭东，刘晓冰. 数据仓库 ETL 任务调度模型研究 [J]. 控制与决策，2011，2：271－275.

(上接第 1705 页)

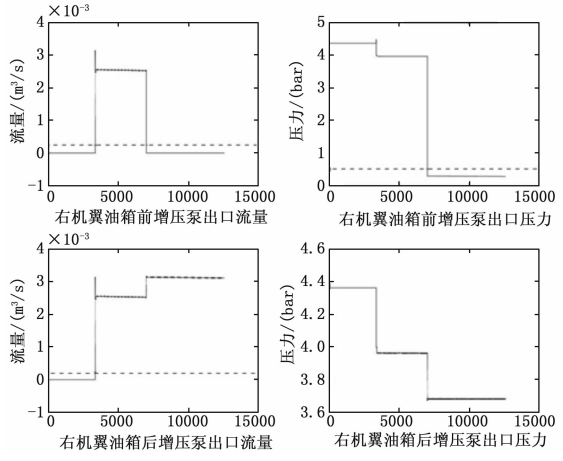

图 9　右机翼前增压泵出口阀无法打开仿真结果

且有相应的流量输出，满足供油要求。而由于机翼油箱前增压泵出口阀无法打开，供油压力一直居高不下，而供油流量为 0。

3　结论

本文针对飞机燃油供油系统，分析了其工作原理。利用流体仿真软件 Flowmaster 建立了相应的仿真模型，得到了飞机燃油系统在增压泵供油、交输供油以及重力供油条件下的结果，验证模型的正确性。飞机燃油系统部分元件、支路故障并不会致使系统无法工作。因此，仿真了供油系统典型的元件故障情况，并分析了所得结果。为之后的故障诊断打下了基础，具有一定的应用价值。

参考文献：

[1] 飞机设计手册 [Z].

[2] 高行山，刘永寿，岳珠峰. 某型飞机燃油输送系统供油稳定性研究 [J]. 机械科学与技术，2008，39 (12)：1541－1544.

[3] Papadopoulos Y. Safety－Directed System Monitoring Using Safety Cases [D], The University of York. 2000.

[4] Hybrid Modeling and Diagnosis in the Real World: A Case Study [A]. Sriram Narasimhan, Gautam Biswas, Tim Bowman, Mark Kay. Thirteenth International Workshop on Principles of Diagnosis [C]. 2002.

[5] Patton R J, Frank P M, and Clark R N. Issues of Fault Diagnosis for Dynamic Systems [M]. London, U. K. Springer－Verlag, 2000.

[6] 吕亚国，飞机燃油系统计算研究 [D]. 西安：西北工业大学，2006.

[7] 冯震宙，高行山，刘永寿，等. 某型飞机燃油/液压系统故障统计与分析 [J]. 飞机工程，2007 (1)：50－53

[8] 冯震宙，高行山，刘永寿，等. 某型飞机燃油系统数值建模方法与仿真分析 [J]. 飞机设计，2007 (5)：65.