单巷道固定货架路径规划问题的研究

王 罡[1],李梅娟[1,2],陈雪波[3]
WANG Gang[1],LI Mei-juan[1,2],CHEN Xue-bo[3]

1.鞍山师范学院 计算机系,辽宁 鞍山 114005
2.大连理工大学 信息与控制研究中心,辽宁 大连 116024
3.辽宁科技大学 电子与信息工程学院,辽宁 鞍山 114004
1.Department of Computer,Anshan Normal University,Anshan,Liaoning 114005,China
2.Research Center of Information and Control,Dalian University of Technology,Dalian,Liaoning 116024,China
3.School of Electronic and Information Engineering,Liaoning University of Science and Technology,Anshan,Liaoning 114004,China
E-mail:asmeijuanli@126.com

WANG Gang,LI Mei-juan,CHEN Xue-bo.Research on path planning problem for single aisle fixed storage rack.Computer Engineering and Applications,2008,44(16):205-207.

Abstract: Fixed storage rack system is an important part of the automated warehouse.The efficiency of warehouse depends on the control strategies of order picking.According to the order picking procedure in the single aisle rack,a mathematic model is constructed with multiple constraints and the optimization aim is to minimum the storage and retrieval time.It adopts dynamic change on algorithm parameters and awaiting nodes set based on the basic ant colony algorithm,and designs an improved ant colony algorithm for the order picking path planning problem.Simulation results demonstrate the improved algorithm can solve medium or large scale problem,it also has better overall search ability and quickly astringency.
Key words: single aisle storage rack;path planning problem;improved Ant Colony Algorithm

摘 要: 固定货架系统是自动化立体仓库的重要组成部分,仓库运行效率主要取决于对货物进行拣选的控制策略。针对单巷道固定货架系统拣选作业过程,以存取时间最小为目标,构建了含多个约束条件的拣选作业路径规划问题的数学模型。在基本蚁群算法基础上,采取自适应调整算法参数、候选节点集合等改进措施,设计了一种改进的蚁群算法对问题进行求解。仿真实验表明该算法能够很好地解决中大规模的拣选作业问题,全局寻优能力强,收敛速度快。
关键词: 单巷道固定货架;路径规划问题;改进蚁群算法
DOI: 10.3778/j.issn.1002-8331.2008.16.063 **文章编号:** 1002-8331(2008)16-0205-03 **文献标识码:** A **中图分类号:** TP18

1 引言

固定货架系统的拣选作业效率是影响自动化立体仓库(automated warehouse)吞吐率的重要因素之一[1,2]。单巷道固定货架拣选作业时,若改变货单中各货位的访问顺序将对应不同的存取路径,寻找具有最小存取时间的路径规划问题,可归结为TSP问题,是组合最优化问题。由于该问题的变量多,求解过程复杂,其复杂程度远高于TSP,用传统方法很难求得问题的近优解。研究表明基于群集智能的蚁群优化算法ACO(Ant Colony Optimization)在求解组合优化问题方面具有很强的发现较好解的能力,并在求解TSP中取得了成果[3-5];文献[6,7]又证明了改进的ACO算法在求解TSP问题时能够收敛到全局近优解。本文在基本ACO的基础上设计了改进的蚁群优化算法求解单巷道固定货架拣选作业路径规划问题,取得了较好的优化结果。

2 问题描述及数学模型

对问题做以下定义:

定义1 对于固定货架系统,相邻两排货架之间有一条巷道,每个巷道口设置一入/出库台I/O(Input/Output)台。如果每条巷道内都配备一台巷道堆垛机,负责该巷道内相应的两排货架的拣选作业,则所研究的问题称为单巷道固定货架拣选作业问题SAOPP(Single-Aisle Order Picking Problem)。

拣选作业过程为:计算机发送货单后,堆垛机携带空货箱从巷道口出发,对第一条目所对应的货位进行存取操作,然后

基金项目: 国家自然科学基金(the National Natural Science Foundation of China under Grant No.60574010);辽宁省高等学校优秀人才支持计划项目和创新团队支持计划项目(the Liaoning College Talented People Supporting and Innovation Team Program Foundation of China under Grant No.2006R31,No.2007T082)。
作者简介: 李梅娟(1967-),女,博士生,教授,主要研究领域为智能控制,先进控制理论及方法;陈雪波(1960-),男,博士生导师,教授,主要研究领域为智能控制,复杂系统的结构与分散控制,先进控制理论及方法。
收稿日期: 2007-11-28 **修回日期:** 2008-02-25

访问下一货位，直到执行完货单中所有条目，堆垛机携带货箱返回 I/O 台，将货箱送至输送系统，完成一次作业。图 1 表示拣选作业流程，(a)表示入库作业流程，(b)表示出库作业流程。

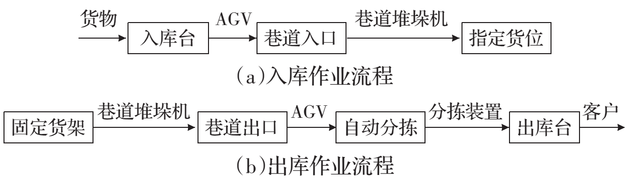

图 1　拣选作业流程

单巷道固定货架结构如图 2，图中 x 表示列号，y 表示层号，黑点为堆垛机需要存取的货位点，以坐标(X,Y)标志，将(0,0)视为 I/O 台作为拣选作业的附加货位点。单个货格长度为 a，宽度为 b，高度为 h，且 a、b、h 均为常数。对于 n 个编号为{1,2,3,…,n}的待拣选货位，所求目标是寻找一条最佳拣选顺序，使完成一个货单的拣选作业花费时间最短。基本设定如下：

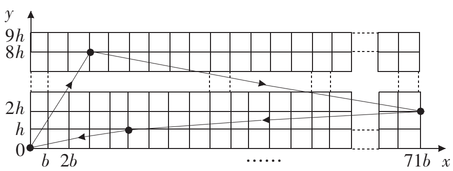

图 2　单巷道固定货架结构示意图

设定 1　堆垛机恒速运行，忽略堆垛机启动和制动及货物存取时间。

设定 2　堆垛机水平运行速度为 V_x，垂直运行速度为 V_y；能同时沿 x 轴(水平方向)和 y 轴(垂直方向)运行。

设定 3　货单条目中每次需拣选的货位点只被堆垛机服务一次，具有单向性。

根据以上假设，堆垛机拣选时从 I/O 台出发，存/取完货位 i 再对货位 j 操作，花费时间是：

$$T_{ij}=\max\{(|X_i-X_j|)/V_x,(|Y_i-Y_j|)/V_y\} \tag{1}$$

其中 X_i,Y_i——货位点 i 的坐标；X_j,Y_j——货位点 j 的坐标。

设 x_{ij} 是决策变量，$x_{ij}\in\{0,1\}$　$i,j=1,2,\cdots,n,i\neq j$。$x_{ij}=1$ 表示 S/R 机行走的路线是访问完货位 i 接着访问货位 j；$x_{ij}=0$ 表示 S/R 机行走没选择这条路线。构造数学模型如下：

$$\min\sum_{i=1}^{n}\sum_{j=1}^{n}T_{ij}x_{ij} \tag{2}$$

$$\text{s.t.}\sum_{j=1}^{n}x_{ij}=1,i=1,2,\cdots,n \tag{3}$$

$$\sum_{i=1}^{n}x_{ij}=1,j=1,2,\cdots,n \tag{4}$$

$$\sum_{i,j\in s}^{n}x_{ij}\leqslant|s|-1,2\leqslant|s|\leqslant n-2,s\subset\{1,2,\cdots,n\} \tag{5}$$

其中：T_{ij} 为堆垛机从货位 i 运行到货位 j 花费的时间代价；n 为待拣选货位总数；目标(2)是确定堆垛机对货位进行拣选的时间最短；约束(3)和(4)要求每个货位只被堆垛机访问一次；约束(5)表示堆垛机运行路线是单向的，不能在货位点之间形成回路，访问完所有的货位点后返回 I/O 台，式中$|s|$表示集合 s 中元素个数。

3　改进的蚁群算法设计

3.1　基本蚁群算法分析

对 Dorigo M 等[8]提出的 ACO 算法分析如下：

(1)设 m 是蚂蚁的数量，$d_{ij}(i,j=1,2,\cdots,n)$表示城市 i 到城市 j 的距离，第 t 个搜索周期内，蚂蚁 $k(k=1,2,\cdots,m)$根据各条路径的信息量从节点 i 向 j 转移的概率为

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}[1/d_{ij}]^{\beta}}{\sum\limits_{k\notin tabu_k}[\tau_{ik}(t)]^{\alpha}[1/d_{ik}]^{\beta}},j\notin tabu_k\\0,\qquad\text{否则}\end{cases} \tag{6}$$

其中，$\tau_{ij}(t)$表示在第 t 个周期内路段 i,j 上的信息素，$tabu_k$ 为禁忌表记录蚂蚁 k 当前走过的城市。

(2)每一搜索周期结束后，路段 i,j 上的信息素按式 $\tau_{ij}(t)=\rho\tau_{ij}(t-1)+(1-\rho)\Delta\tau_{ij}(t-1)$更新，其中 ρ 为信息素挥发速度系数，$1-\rho$ 表示信息消逝程度，$\Delta\tau_{ij}(t-1)$是第 $t-1$ 次循环中路段 i,j 上信息素的增量

$$\Delta\tau_{ij}(t-1)=\sum_{k=1}^{M}\Delta\tau_{ij}^{k}(t-1)$$

$$\Delta\tau_{ij}^{k}(t-1)=\begin{cases}Q/L_k,\text{第 }k\text{ 只蚂蚁在 }t-1\text{ 循环内经过路段 }ij\\0,\text{否则}\end{cases} \tag{7}$$

其中，$\Delta\tau_{ij}^{k}(t-1)$表示第 k 只蚂蚁在第 $t-1$ 次循环中留在路段 i,j 上的信息素的增量；Q 为常数；L_k 表示第 k 只蚂蚁走过的路径长度。

3.2　改进的蚁群算法原理

ACO 是一种通用型随机优化算法[9]，存在求解速度和全局寻优能力之间的矛盾，针对本文研究的问题，对基本 ACO 算法做了以下改进：

3.2.1　自适应调整信息素挥发速度系数 ρ

标准的蚁群算法中 ρ 为常数，信息素挥发速度系数 ρ 的取值大小对算法的收敛速度，以及是否收敛到全局最优解非常重要。对于组合优化问题，在求解的初始阶段，需要 ρ 的取值稍大一些增强蚁群系统探索新解的能力，在后期 ρ 越小，搜索就越集中。基于这种思想，利用参数 $\rho_{nc}=(NC_{\max}-nc)/NC_{\max}\times\rho_0$ 作为信息素挥发速度系数更新规则，其中 nc 是当前迭代步数，$NC_{\max}$ 是最大迭代步数，ρ_0 取一个较大的常数，运算过程中自适应动态调整系数 ρ。运算时，先设定一个阈值 NC_1，当迭代步数 nc 大于 NC_1 时，按照上式修正 ρ。这样，前 NC_1 步迭代，先尽量扩大全局搜索范围，再加快收敛速度，向局部最优解收敛。自适应动态调整 ρ 的策略在问题规模较大的时候具有很明显的优势，不仅可以加快收敛速度，而且搜索质量也有提高。

3.2.2　候选节点集合策略

对于较大规模的实际问题，蚁群算法需要很长的计算搜索时间，为了提高算法的搜索效率，设计了候选节点集合策略。对编号为{1,2,3,…,n}的货单，在堆垛机访问所有货位点的最短路径中，货位点 i 连接到货位点 j，j 不可能是离货位点 i 最远的那些节点。根据这一点，设计了候选节点集合策略，就是将下一个要选择的待拣选货位点的数量限制在一个合适的子集或候选表内，候选节点集合中的待拣选货位点被局限于离当前货位点 i 较近的部分节点，仅对这部分节点的转移概率进行计算。

3.2.3　最优解选择操作

设一个全局变量 L^*，代表到目前为止找到的目标函数值最好的蚂蚁轨迹。每一次迭代完成时，将 L^* 与当前最优方案的蚂

蚁 $L_k^*=L_{k\min}$（$L_{k\min}=\min L_k, k=1,2,\cdots,m$）的轨迹相比较，如果 L_k^* 优于 L^*，则 $L^*=L_k^*$，这样能使算法快速收敛到某一个次优解上。

3.3 算法流程

改进的蚁群算法步骤如下：

步骤 1 初始化算法参数。设定 $m,\rho_0,NC_1,NC_{\max},t=0,nc=1$，$\tau_{ij}(0)=\tau_0$ 为一个很小的常数，$\Delta\tau_{ij}=0$。

步骤 2 将 n 个待拣选货位点按（X，Y）坐标排序，找出距附加货位点 I/O 台（0，0）较近的 N 个节点，以（0，0）为起点生成排序表。$N=n/r$，$r\in[1,10]$，n 越大，r 取值可越大，若 n 较小时，r 取较小的值。将 m（$m=n$）只蚂蚁放置在（0，0）点，并设置到禁忌表中，开始循环搜索。

步骤 3 对于每一只蚂蚁，以当前节点 i 为中心，按较近节点原则选取离 i 最近的 n/w 个节点作为下一个节点 j 的选择范围（根据节点 i 的距离排序表选择前 n/w 个节点），$n/w\in[5,20]$。从 n/w 个节点中找出 z 个未走过的节点$\{j_1,j_2,\cdots,j_p,\cdots,j_z\}$，即 $j_p\notin tabu_k$。

步骤 4 在 z 个节点中，按式（6）选择节点 j；将 j 加入到禁忌表 $tabu_k$ 中。

步骤 5 计算信息素挥发速度系数 ρ，按式（8）更新信息素 $\tau_{ij}(t+1)$。

步骤 6 当 m 只蚂蚁选择完节点 j 后，令 $i_{new}=j$；$j_{new}=j_{old}+1$；返回步骤 3，选择下一个节点，直到所有蚂蚁完成一次周游为止。

步骤 7 最优解选择操作。m 只蚂蚁完成一次周游以后，计算 L^* 和 L_k^*，并使 $L^*=L_k^*=\min L_k(k=1,2,\cdots,m)$，$L_k$ 是蚂蚁 k 完成一次周游的路径长度，保留 L^* 和本次最优路径表 $tabu_L$。

步骤 8 判断 $NC_{\max}-1$，不等于 0，清空并初始化禁忌表，重复上述过程，否则终止程序。

4 实验分析

为了验证算法求解拣选作业问题的有效性，采用以下测试数据，自动化立体仓库货架系统由 10 排，72 列组成，每排 10 层共 720 个货位，货架系统及堆垛机各参数如下：$V_x=3$ m/s，$V_y=1$ m/s；$a=1$ m，$b=1$ m，$h=1$ m；$r=5$，$n/w=5$；最大迭代步数为 $NC_{\max}=1\,000$。

应用与文[10]相同的 80 个货位点坐标。首先，对这 80 个货位点，选择参数 $r=5$，$n/w=5$，将附加货位点和 N（$N=n/r$）个节点进行排序生成排序表，将其中 n/w 个货位节点生成候选节点集合。针对不同的算法参数进行实验，平均运行 30 次作为结果，见表 1。

表 1 数据表明，当 ρ 不变时，Q 取值由大变小，运算结果趋于全局最优，但对算法收敛速度影响不明确；α、β 由 1.0 改变到 2.0 时，使算法收敛速度稍有提高，但全局搜索能力下降，对算法影响不大；若参数 Q、α、β 保持不变，改变 ρ 对结果影响非常显著，ρ 取值越大，算法全局搜索能力强，运算结果趋于全局最优，但是收敛速度很慢；相反 ρ 取值越小，其全局搜索能力变弱，但是收敛速度明显提高。

表 1　算法不同参数的实验结果

ρ	Q	α	β	nc	T/s
0.9	1 000	1.0	1.0	92	92.4
0.9	1 000	2.0	2.0	87	95.6
0.9	500	2.0	2.0	98	94.8
0.9	500	1.0	1.0	102	97.4
0.9	100	2.0	2.0	95	94.3
0.9	100	1.0	1.0	94	93.1
0.5	1 000	1.0	1.0	43	94.8
0.5	1 000	2.0	2.0	40	98.5
0.5	500	2.0	2.0	40	94.2
0.5	500	1.0	1.0	37	93.9
0.5	100	2.0	2.0	40	94.2
0.5	100	1.0	1.0	51	95.3
0.1	1 000	1.0	1.0	21	98.6
0.1	1 000	2.0	2.0	17	103.4
0.1	500	2.0	2.0	21	96.8
0.1	500	1.0	1.0	21	98.3
0.1	100	2.0	2.0	23	94.8
0.1	100	1.0	1.0	26	97.5

实验中 $\rho_0=0.9$，当迭代次数 nc 约为 50 时，减小 ρ，将搜索空间完全控制在局部区域内进行，尽快收敛到最优解。应用改进算法对 80 个拣选点的作业进行 30 次实验，求解结果见表 2，可以看出货位规模为 80 时，最优拣选代价为 92.38s，拣选代价标准差是 0.901，最优解出现概率是 76.67%。与文[10]的混合遗传算法相比，最优拣选代价减少 11.62 s，拣选代价标准差下降 0.406。本文算法在迭代到 102 代时就能找到全局最优解，收敛速度比混合遗传算法提高约 6 倍。再用本文算法对待拣选货位规模为 20、40、60 的作业分别进行实验，对每一组数据实验结果及与文[10]的结果进行比较见表 2。实验表明本文算法比混合遗传算法有了极其显著的改善，随着货位规模增大，解的质量有所提高，算法的收敛速度明显加快。

5 结束语

分析了自动化立体仓库单巷道固定货架系统拣选作业路径规划问题的特点，并构建了数学模型。设计了一种改进的蚁群算法求解该问题，实验验证了本文算法在求解大规模货物拣选作业优化问题时收敛速度显著提高，具有很好的可行性和明显的优势，并可以应用于求解其他组合优化问题中。

参考文献：

[1] Jeroen P, van den Berg.Analytic expressions for the optimal Dwell point in an automated storage/retrieval system[J].Int Production Economics, 2002, 76(1): 13-25.

表 2　不同算法性能比较表

货位规模	混合遗传算法					改进蚁群算法				
	最优拣选代价（T/s）	最佳个体出现频率	平均拣选代价（T/s）	拣选代价标准差	收敛代数	最优拣选代价（T/s）	最佳个体出现频率	平均拣选代价（T/s）	拣选代价标准差	收敛代数
20	49.33	30/30	49.33	0	注	49.33	30/30	49.33	0	8
40	69.67	10/30	70.67	0.922	注	68.23	30/30	68.23	0	19
60	88.67	3/30	90.46	1.085	注	83.19	25/30	83.46	0.785	38
80	104.00	*	105.76	1.307	561	92.38	23/30	93.25	0.901	102

注：* 表示文[10]中未注明项。

（下转 239 页）