

学号 2008301500227

密级

武汉大学本科毕业论文

多策略集装箱船舶配载计划

院（系）名 称：计算机学院

专 业 名 称 ： 计算机科学与技术

学 生 姓 名 ： 石 磊

指 导 教 师 ： 胡文斌 副教授

二〇一二年五月

BACHELOR'S DEGREE THESIS
OF WUHAN UNIVERSITY

Container Ship Stowage Plan Based on
Multi-Strategy

College : School of Computer

Subject : Computer Science and Technology

Name : Lei Shi

Directed by : Wen-Bin Hu Associate Professor

May 2012

郑 重 声 明

本人呈交的学位论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本学位论文的知识产权归属于培养单位。

本人签名：_____

日期：_____

摘要

集装箱码头船舶配载调度是集装箱海运运输中的一个重要环节。船舶配载调度需要在满足船舶安全性和运输经济性的基础之上，确定待装船集装箱在船上的堆放方案，这是一个典型的组合优化问题。随着集装箱船舶的大型化，待装船集装箱的数量急剧增多，船舶配载调度的问题规模越来越大，如何在最短的时间内求解出一套有效的、满足实际要求的配载调度方案成为港口面临的新挑战。

传统的集装箱船舶配载计划以确定每一个集装箱在船上的具体堆放位置作为决策内容，这种思路存在两个问题：（1）集装箱船舶的大型化导致采用此种方案很难在较短的时间内求解出一套有效、满足实际要求的解决方案；（2）确定集装箱的具体堆放位置，势必给后面的集装箱装船过程带来很大的局限性，不利于满足堆场业务的实时性要求。基于此，本文提出了集装箱的分类策略和堆放单元的概念。我们基于集装箱的重量，尺寸等指标把集装箱进行归类，利用一个类中的一些数量的集装箱拼接堆放单元。把集装箱的堆放问题转化为堆放单元的堆放问题，把一个三维堆放问题转化为一个二维堆放问题，问题的解空间大大减小。本文以确定堆放单元的相对堆放位置和船舶每一个箱位堆放的集装箱类型作为决策内容。以最小化船上翻箱率，最小化船舶重心高度，保持适当吃水差和重箱中间堆放作为决策目标。在此基础上建立了船舶配载调度模型，通过提出一种基于群体编码方式的遗传算法和基于贪心算法的倍内堆放策略对模型进行求解。

本文较好地解决了船舶大型化情形下的集装箱船舶配载调度问题。采用本文提出的模型和算法进行求解，可以在保证船舶稳性、重箱中间堆放等约束的基础上，降低船上的翻箱率，满足了船舶的安全性和运输的经济性要求。同时，本文的决策结果可以保证装船过程具有很大的灵活性，满足了堆场业务的实时性要求。

关键词：船舶配载；遗传算法；策略；集装箱码头；群体编码

ABSTRACT

Container terminal ship stowage scheduling is an important step in the transportation of container shipping. Ship stowage schedule needs to determine stacking plan on the basis of meeting the safety of ship and transport economy. And this is a combinatorial optimization problem. With the large-scale container ships and the increase of container numbers, the problem becomes more and more complex. How to get an effective solution in the shortest time is a new challenge for the port.

Traditional container ship stowage plan determines the exact stacking position for each container. This method has two shortcomings. Firstly, it's difficult for this method to get an effective solution in a short time period with the large-scale container ship. Secondly, determining the exact stacking position will bring many limitations for loading container, which is not conducive to meet the real-time requirements of the container yard services. Based on this, this paper presents a container classification strategy and the concept of stacking unit. Similar container will be classified into classes based on a strategy which considering the weight, size and other factors of containers and containers from the same class will make up stacking units. The problem of containers' stacking will be transformed into stacking units' stacking and the problem of a three-dimensional stacking will be transformed into a two-dimensional problem, the problem space is greatly reduced. The strategic decisions of this paper are that which bay each stacking unit is assigned and the stacking location of each stacking unit in bay. The goals are minimizing container rehandle rate on board, minimizing the center of gravity of the ship, holding appropriate trim and ensuring that the heavy containers stacking in the middle of the ship. The classification strategy and stacking strategy in bay are proposed. Based on this, a scheduling model of ship stowage is established. A genetic algorithm which bases on group coding method and the stacking strategy in bay which bases on the greedy algorithm are proposed as the resolution of the model.

This paper has solved the stowage schedule problem in the case of large ships. The

proposed model and algorithm can ensure the stability of the ship, heavy contains stacking in the middle and reduce rehandle rate. It meets the safety of ships and transport economic. What's more, it can guarantee that the shipment process has great flexibility to meet the real-time requirements of the container yard services.

Key words: Ship stowage; Genetic algorithms; Strategy; Container terminal; Group coding

目 录

第 1 章 引言	1
1.1 研究背景与意义	1
1.2 国内外研究现状	2
1.3 本文研究内容	3
1.4 论文组织结构	4
第 2 章 出口集装箱业务流程介绍	5
2.1 出口集装箱业务流程	5
2.2 集装箱码头堆存分配	5
2.3 集装箱配载调度	6
2.4 集装箱装船	7
2.5 本章小结	7
第 3 章 配载调度建模	8
3.1 船结构抽象和分析	8
3.2 分类策略和堆放单元	9
3.3 倍内堆放策略	10
3.4 配载调度模型描述	12
3.5 模型的数学描述	12
3.6 本章小结	14
第 4 章 基于 GA 的船舶配载调度算法	16
4.1 遗传算法简介	16
4.2 本文采用的编码方式	16
4.3 遗传算子描述	17
4.4 算法步骤	18
4.5 本章小结	18
第 5 章 算例及结果分析	20
5.1 实验假设	20
5.2 模型和算法的有效性验证	20
5.3 不同参数配置下的优化情况	21
5.4 本章小结	23
第 6 章 配载调度软件模块设计与开发	25
6.1 配载调度软件模块流程图	25
6.2 配载调度软件模块数据表设计	25
6.3 配载调度软件模块的子模块实现	27
6.4 配载调度软件模块软件功能演示	28
6.5 本章小结	30

第 7 章 总结与展望	31
7.1 论文总结	31
7.2 论文展望	32
参考文献	33
致谢	35

第1章 引言

1.1 研究背景与意义

集装箱运输是目前最重要的一种运输方式。出口集装箱在其生命周期中一般要经历以下两个阶段：第一个阶段是集装箱到达港口之后，需要根据堆场的实际情况为其在堆场分配一个堆放位置；第二个阶段是装载此集装箱的运输船舶到达港口之后，需要为集装箱在船上分配一个堆放位置。第二个阶段就是集装箱船舶配载计划。集装箱船舶配载计划是集装箱船舶运输中的重要一环，主要解决船舶在装载集装箱时，在遵循配载基本原则和保证船舶稳定性和强度的基础上，确定集装箱在船上的堆放位置，以此形成集装箱装载顺序^[1]。配载计划主要关注船舶的安全性和运输的经济性。传统的配载方案通过人工经验手工完成，但是随着集装箱船舶的大型化，配载问题变得原来越复杂，各种研究机构和船舶航运公司开始研究借助于计算机来解决配载问题，希望通过制定科学合理的配载计划，提高码头运作和船舶运输能力。

集装箱船舶配载计划是一个复杂的综合性问题，需要满足船舶的运输要求，即船舶的航行安全、集装箱及其货物的运输质量和船舶营运的经济效益，还要考虑集装箱码头装卸工艺和操作方式，使码头能合理、有序、有效地组织生产，提高整个系统作业效率。配载计划主要需要考虑以下因素：

减少卸货时的翻箱率。集装箱船舶在营运过程中通常需要挂靠多个港口，且一般在各个港口需要装卸部分集装箱。因此，集装箱船舶配载时应当尽量减少先卸的集装箱被后卸的集装箱压住或堵住其卸箱通道的现象发生，以免发生船上翻箱的现象。在船上产生翻箱，会降低装卸速度，造成损失。

保证船舶良好的稳性。所谓稳性，是指集装箱船舶受外力作用而发生倾斜，当外力消失后自动恢复到原来平衡位置的能力。稳性是衡量航行安全的最重要指标，特别是由于集装箱船甲板上堆放着大量集装箱，会导致整个船体的重心上升，造成船舶稳性下降，配载的时候必须充分考虑这一因素，满足集装箱船的稳性要求。

保持船舶首尾适当的吃水差。首尾吃水差是指船舶首尾吃水的差值，适当的吃水差可以使船舶具有良好的航行性能，节省燃油，充分发挥主机功率。不同船舶的吃水差要求往往不同。

防止船体中拱。在集装箱配载过程中，应该保证中部的集装箱总重量分别大于首部和尾部的集装箱总重量。这样可以防止中拱现象的发生。

在满足以上约束的基础上得到一套满足实际要求的配载调度方案是目前港口和船舶运输公司的迫切要求。

1.2 国内外研究现状

早在 60 年代，国外航运公司和研究机构就开始研究集装箱船舶配载调度问题，目的是要寻求一种在满足既定约束条件下翻箱最少，作业高效的配载调度方案。从目前收集到的资料看，对集装箱配载问题的研究，从方法上讲，可以归纳为5类：随机模拟方法、启发式方法、决策支持系统、数学规划模型、专家系统。

文献[2]提出了多类型集装箱装船模型，并采用COIN—OR对建立的模型进行行求解。针对多个港口，一定数量的集装箱的情况进行实验，优化后的翻箱次数明显减少。文献[3]在考虑安全性和经济性的前提下，提出了基于pareto遗传算法的最优解决方案。该方案能够在减少翻箱次数的同时增大船舶的净载量。文献[4]提出了集装箱船的单元块配载，这种分块思想在集装箱船挂靠多个港口的情形下能很好地解决多港箱的港序安排问题。文献[5]提出了单倍多港配载模型，以最小化翻箱数作为优化目标，但没有考虑重量等因素。目前很多研究均以确定每个集装箱在船上的具体装载位置作为决策内容。文献[6] 通过仿真集装箱码头现场的运行情况,对集装箱物流系统设计的各环节进行有效地了解,分析系统瓶颈,预测系统结果,展现系统形象,实现系统优化。文献[7]采用遗传算法优化船舶实配过程，并实现了一个基于web的集装箱船舶箱管系统。文献[8]提出了集装箱船在一个航次中挂靠多个港口的混装Bay位排箱模型，采用隐式图启发式搜索技术对各个港口的集装箱进行安排，所提出的模型能快速产生翻箱次数最少的全航线Bay位排箱方案。文献[9]通过对生产实际中散装货船的多品种、大批量货物的配载问题的分析,总结了散装船舶配载问题的特点和钢铁产品的货物堆装规则，构造了一种混合遗传算法来解决此类多目标、多约束的组合优化问题。文献[10]提出利用预测模型预测后续港

口的箱量信息，为船舶全航线配载提供数据支持，在此基础上建立全航线预配和Bay位排箱优化模型，使用配载评价模型对配载方案进行综合评价及优选。文献[11]探讨了多种集装箱船舶智能配载方法，并实现了一个系统原型。文献[12]提出了基于稳定约束的集装箱配载计划启发式算法，对重心高度和翻箱数这两个目标进行权衡和优化。文献[13]采用Petri网对空箱调运进行仿真和建模，取得了较好的优化效果。文献[14]将配载问题分解为Bay位选择和Bay中集装箱排序两个子问题，将Bay位选择看成装箱问题，按照某些规则对Bay位中的集装箱进行排序，取得了较好的效果。文献[15]将稳性、减少翻箱等重要因素分解为评估策略，按优先级将各种评估策略划分等级，尤其将稳性问题具体到船的每行及每个箱位，以寻找最优解。文献[16]将配载问题分解为倍位选择和倍位中集装箱箱位指定两个子问题。主要研究了倍位中装载同一目的港集装箱时，不同重量集装箱在倍位中的排箱方法，为集装箱船全航线配载优化提供了一个实用的模型。随着集装箱船舶的大型化，集装箱数量增多，问题规模急剧增大[17]，采用这种方式很难在短时间内求解出一套满足实际要求的配载方案。

1.3 本文研究内容

本文的研究重点在于如何在最短的时间内，针对具体的船舶求解出一套高效的船舶配载方案。本文要解决的问题如下：

（1）集装箱分类和操作单元形成。由于集装箱船舶的大型化，配载调度问题的规模急剧增加，传统的方法以确定每一个集装箱的具体堆放位置作为决策内容，很难在短时间内求解出一套有效的解决方案，同时，不利于满足堆场业务的实时性要求，极大地限制了后期装船计划的灵活性。基于此，本文提出了集装箱分类策略和操作单元的概念。我们把满足一定相似度的集装箱归为一类，并在各个类中划分操作单元，从而把一个高维的组合优化问题降低为一个低维的组合优化问题。同时，以确定堆放单元的相对堆放层次作为决策目标。决策输出只会指定某一个箱位堆放的集装箱的类型，而不指定具体的集装箱。这样为后续的装船计划提供了很大的灵活性。满足了堆场业务的实时性要求。

（2）船结构抽象和分析。众所周知，不同的船舶，其船结构是不同的。本文为了建立一套通用的集装箱船舶配载调度解决方案，对船结构进行了抽象。结合

实际中的船结构模型，我们把集装箱船舶的堆放空间抽象为一个长方体空间。在针对某一条船舶进行具体的配载调度时，我们可以通过禁用某些箱位来反映实际的船结构。

(3) 堆放单元的倍内堆放策略。影响堆放单元倍内堆放的因素主要有两个。一个是堆放单元的重量，另外一个为堆放单元的卸货港。如果堆放的不合理，可能会导致船舶重心高度偏高，不利于航行的安全性。也可能导致卸货时的翻箱率过高，增大经济开销。倍内堆放策略同时考虑了这两种因素，并且设置了可调参数，以满足实际工程的需要。

(4) 给出了船舶配载调度的模型和数学描述，并采用基于群体编码方式的遗传算法求解模型。本文还给出了一些算例，证明本文所提出模型和算法的有效性。同时，讨论了在倍内堆放策略采用不同参数配置的情形下，对堆放效果的影响。

(5) 本文采用C#语言实现了集装箱船舶配载调度的演示模块。

1.4 论文组织结构

本文共分为7章。

第一章为引言，简要地介绍了配载计划的背景，存在的挑战，国内外的研究现状和本文的研究内容。

第二章为出口集装箱业务流程介绍，介绍了出口集装箱的业务流程。

第三章为配载调度建模。在这一章里，介绍了船舶结构和本文对船舶结构的抽象。同时，还给出了集装箱分类策略和堆放单元拼接的方案。提出了一种集装箱倍内堆放的贪心策略。最后，给出了集装箱船舶配载调度模型的数学描述。

第四章为基于遗传算法的配载调度模型求解。首先，简要地介绍了遗传算法。然后，给出了本文的编码方式和遗传算子设计。最后，给出了算法流程。

第五章为算例和结果分析。

第六章为配载调度软件模块设计与开发。简要介绍了本文模型的一个实例。

第七章为对现在工作的总结和对未来工作的展望。

第2章 出口集装箱业务流程介绍

2.1 出口集装箱业务流程

由于冷藏箱、特种箱等的堆放有其它因素的限制，本文只针对普通出口箱进行建模。本文研究的主要对象是港口的出口集装箱。典型的出口集装箱业务流程如下图2.1所示。

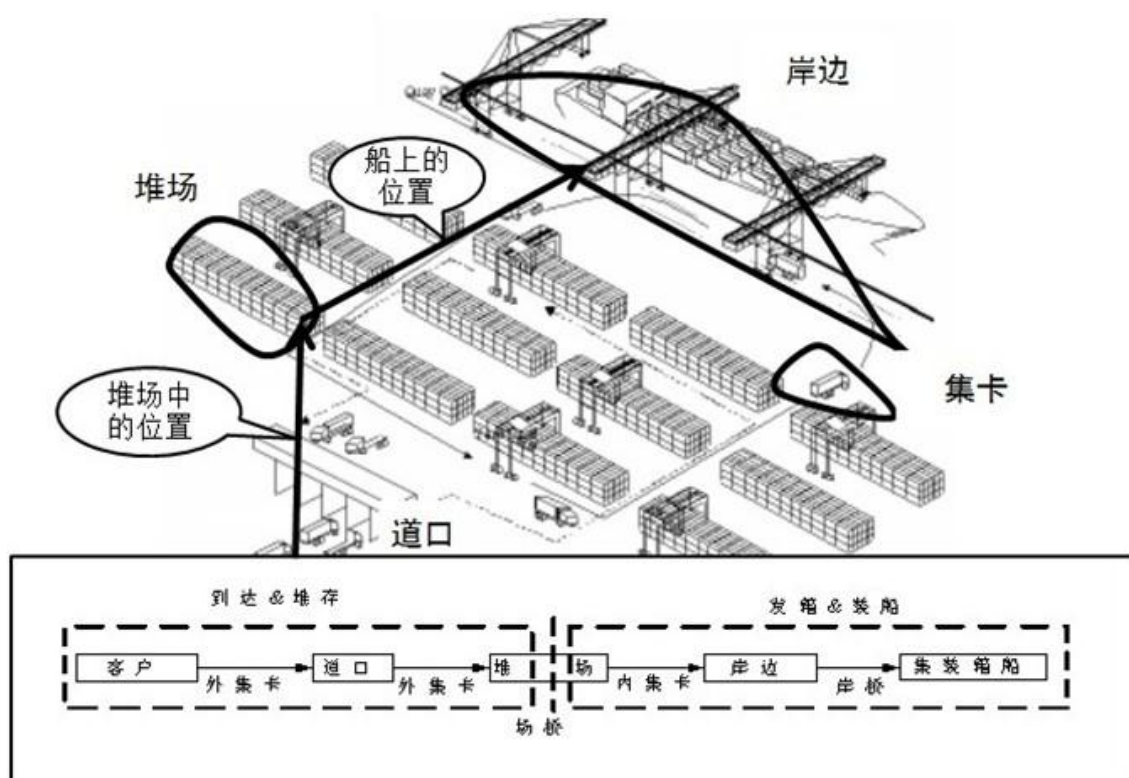


图 2.1 出口集装箱业务流程图

一般而言，出口集装箱在其生命周期中，会经历两个阶段。第一个阶段是指集装箱到达码头的道口之后，为这个集装箱确定其在堆场的堆放位置，这个阶段通常被成为出口集装箱码头堆场堆存分配；第二个阶段是指装载集装箱的船舶到达码头之后，为集装箱在船上分配一个堆放位置，这个过程通常被称为出口集装箱船舶配载调度。

2.2 集装箱码头堆存分配

集装箱码头堆场是船舶装卸进出口集装箱的场地，通常有专门的装卸、运输设备，集运、贮放集装箱的宽阔货场，拆卸集装箱和货物分类用的货棚等。它是

集装箱运输关系方的重要组成，在集装箱运输中起到重要作用。由于港口极大的吞吐量和频繁的费用高昂的集装箱调度，如何优化堆场空间的利用率，减少各设备的使用次数，是研究港口问题的重中之重。而堆场中的出口箱堆存分配是集装箱码头堆存空间管理中的核心环节。

针对集装箱码头堆存空间分配，文献[18]提出了采用区位计划、倍位计划和集装箱进场选位多阶段决策模型。在区位计划模型中，在保证堆场利用率以及装船效率的同时，以最小化各船舶与堆存区段的总体距离和尽量避免区段内多条船舶同时装船的情况为目标，进行双目标优化，确定各个集装箱所属的箱区。倍位计划将保证其前面的区位计划顺利执行，保证装船发箱的连续性和减少轮胎调大车的移动，同时为后面的集装箱进场选位的优化打下基础。采取预留式倍位分配策略，确定集装箱所属的倍位。集装箱进场选位中提出多阶段决策分配箱位法，基于混堆排模式的混堆方式，以此来解决集装箱进场选位需求的实时性以及降低压箱数、提高堆场利用率等问题。

2.3 集装箱配载调度

集装箱船舶配载调度解决的主要问题是确定集装箱在船上的堆放位置。这也是本文的研究重点。配载调度过程中，主要关心安全性、时效性和经济性。

安全性主要是指船舶在运输中的安全性，包括船舶的稳性，吃水差，左右重量差等因素。一般而言，要求船舶的重心尽可能的低，这样才能保证船舶具有良好的稳性。同时，堆放的集装箱应该在重量上较为均匀，这样可以保证船舶左右的平衡。

时效性主要是指，在船舶到达港口之后，堆场资源调度和装船过程的时效性。制定出的配载计划必须利于满足堆场业务的实时性要求，具有更大的灵活性。港口处于极其繁忙的状态，在集装箱装船过程中需要调集设备运送集装箱，然后堆放到船上，所以制定的配载计划一定要具有时效性、灵活性。

经济性主要涉及装卸集装箱过程中的翻箱率等因素。有数据显示，上海港在其堆场中的一次翻箱操作的代价是50元人民币，而在船上的翻箱成本则更高。所以在配载过程中，还需要考虑如何减少在船上的翻箱率，以节省港口的经济开销。

2.4 集装箱装船

集装箱装船是指在船舶到达港口之后，在配载计划的基础上，根据堆场实时的资源配置，调用堆场设备将集装箱堆放到船上的过程。这个过程中，主要涉及场桥，集装箱运输卡车和岸桥的调度和使用。

2.5 本章小结

本章介绍了出口集装箱的业务流程。出口集装箱的业务流程主要包括两个方面：1.出口集装箱到达道口之后，为其在堆场分配一个堆放位置，即出口集装箱堆存分配；2.装载集装箱的船舶到达码头之后，为集装箱分配一个在船舶上的堆放位置，即出口集装箱船舶配载调度。

第3章 配载调度建模

3.1 船结构抽象和分析

典型的集装箱船舶的结构如图3.1所示。从侧面看去，船舶堆放空间由若干个倍构成。每一个倍中又有若干个箱位。一个箱位用于堆放一个集装箱。不同船舶的结构略有差异。

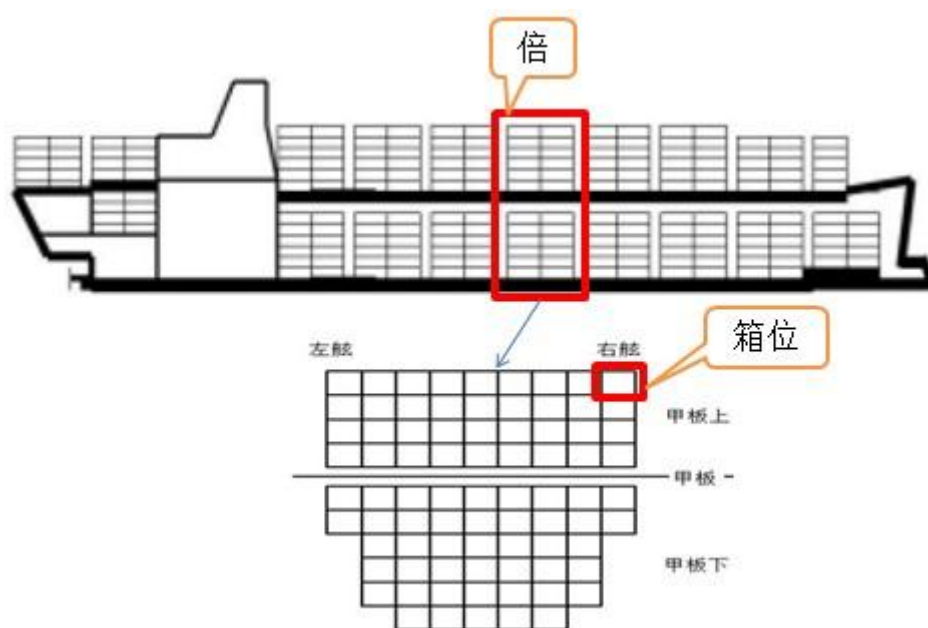


图 3.1 船舶结构图（侧视图）

由于不同船舶集装箱堆放空间的结构是不同的，为了提高本文所构建的调度模型的通用性，可以抽象所有船舶的集装箱堆放空间为一个长方体空间，在实际的操作过程中通过分层方式扩展为其他任何不规则的堆放空间。每个集装箱会被堆放在这个长方体空间的某一位置。

如图3.2所示为船舶倍图。左图为实际船舶的倍图，右图为本文抽象后的船舶倍图。对于不同的船舶，甲板上的堆放空间一般是相同的，都为长方体空间，但甲板下的堆放空间和具体的船结构有关。本文建模时采用了右图所示的长方体空间。这种空间的通用性主要表现在三个方面。（1）可以增加一些限制条件，例如某些箱位禁止堆放，从而将右图的堆放空间转化为左图的堆放空间。（2）无需对右图堆放空间进行转化，而是在堆放集装箱时，保持集装箱的相对堆放层次。（3）将整条船舶的集装箱堆放空间划分为几个子堆放空间，划分后的子堆放空间均为

长方体空间。采用本文模型和算法对各个子堆放空间进行求解。

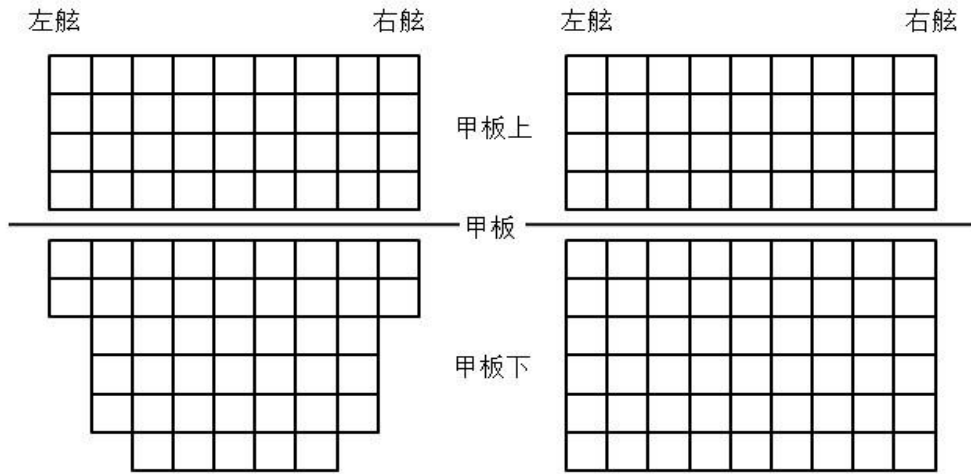


图 3.2 船舶倍图

3.2 分类策略和堆放单元

定义1（箱组）同一箱型（集装箱的类型，包括冷藏箱、干货箱等），同一尺寸，同一船，同一目的港，同一重量级的集装箱的集合称为同一个箱组^[18]。

多数集装箱码头船舶配载调度的文献都把指定每一个集装箱在船上的具体堆放位置作为决策内容。这种方式在集装箱数量较少的情况下是有效的，但是随着集装箱船舶的大型化，配载计划所涉及的集装箱数量增多，这种方式的劣势越发凸显。总体来说，这种方式有如下两个比较突出的弱点：（1）集装箱船舶的大型化导致每一次配载计划需要考虑的集装箱数量增多，这使得算法寻优空间很大，很难在短时间内求解出一套有效的船舶配载调度方案（2）计划完成后，每个集装箱在船上的装载位置完全确定，严格限制了集装箱装船过程中的发箱顺序，导致装船过程中的灵活性不够，不利于满足堆场业务的实时性要求。本文提出了集装箱分类的思想。在集装箱分类的基础上，再划分堆放单元。本文以确定每一个堆放单元的堆放位置作为决策内容，算法的寻优空间大大减少，可以保证在一个较短的时间内求解出一套有效的配载调度方案。同时，本文的决策结果将确定每一个箱位堆放的集装箱类别，而不是具体的集装箱。这样将为后续的集装箱装船过程带来很大的灵活性，有利于满足堆场业务的实时性要求。

定义2（分类策略）分类策略构建于箱组的基础之上。在同一个箱组内，根据集装箱之间重量的相似性来划分类别（1）如果一个箱组中集装箱的重量差别不大，

直接把这个箱组中的集装箱划为一类。(2) 如果一个箱组中的集装箱的重量差别比较大, 则按照集装箱重量进行聚类, 从而形成几个集装箱类。

定义3 (堆放单元) 堆放单元指由若干个属于某一个集装箱类的集装箱拼接而成的, 可以占据一个倍的一层箱位的集装箱的集合。

本文以确定堆放单元的堆放位置作为决策内容。这种方式可以大大地减小解空间, 以保证能够在短时间内求解出一套有效的船舶配载调度方案, 同时, 由于只是确定了堆放单元的堆放位置, 也就是说只确定了各个箱类的堆放位置, 而没有确定具体集装箱的堆放位置, 这给后续的集装箱装船带来了很大的灵活性, 满足了堆场业务的实时性要求。由于堆放单元占据一层的箱位, 而堆放单元中的箱子属于同一个箱类, 所以重量差别不大。这样就不需要对船舶的左右平衡进行特别的考虑了。在实际装船过程中, 采用一些简单的策略就可以保证船舶的左右平衡。

堆放单元的形成: 假设一个类中的集装箱数为 n , 集装箱船一个倍位中一层可以堆放的集装箱数为 N , 计算 n/N , 得到商为 x , 余数为 y 。当 $y < N/4$ 时, 这一类集装箱形成 x 个堆放单元。另外剩下的 y 个集装箱不划入堆放单元, 可以在配载计划完成之后, 找到合适的位置进行堆放。当 $y > N/4$ 时, 这一类集装箱将形成 $x+1$ 个堆放单元, 其中, 有一个堆放单元的集装箱数为 y 个。

按照上面的堆放单元形成策略, 有一些堆放单元中包含的集装箱数没有 N 个, 不足以占用一个倍位一层中所有的箱位, 但是算法中仍然认为这些堆放单元占了一个倍位一层的箱位。在实际装船过程中, 可以用没有划入堆放单元的集装箱来填充这些空余的箱位, 也可以将分配到上一层的堆放单元的集装箱堆放到这些空余的箱位, 保持集装箱的相对堆放次序。

3.3 倍内堆放策略

影响倍内堆放单元堆放顺序的主要因素有两个, 一个是堆放单元的卸货港, 另一个是堆放单元的重量。一般来说, 配载计划应该尽量将比较重的堆放单元堆放在比较轻的堆放单元的下面以维持比较低的重心高度。同时, 尽量将卸货港远的堆放单元堆放在卸货港近的堆放单元下面, 这样可以减少船上的翻箱率, 降低

运营成本。但是上面两条原则很难在实际中同时得到满足。如果只考虑重量这个因素，严格按照比较重的堆放单元在下，比较轻的堆放单元在上的原则进行配载，势必导致整条船舶的翻箱率过高，增加经济开销。如果只考虑卸货港，按照卸货港远的堆放单元堆放在卸货港近的堆放单元下面的原则进行配载，势必导致船舶的重心高度较大，影响船舶稳性。本文提出了一种贪心策略来指导倍内堆放单元的堆放。这种策略综合考虑了卸货港和重量这两个因素。定义堆放单元的卸货港为 P ，重量为 W 。 $P=j$ 表示离开当前港口后到达的第 j 个港口。定义变量 $wMULp = w^\alpha * p^\beta$ ，其中 α ， β 称为 W 和 P 的权值。

定义 4（倍内堆放策略）对一个倍内的堆放单元按照 $wMULp$ 的值进行堆放， $wMULp$ 值越大的堆放单元堆放在越下面的位置， $wMULp$ 值越小的堆放单元堆放在越上面的位置。

如图 3.3,图 3.4 是对一个倍位内的情况进行模拟的结果，可以看到 α ， β 的相对取值对于翻箱率和重心高度的影响。

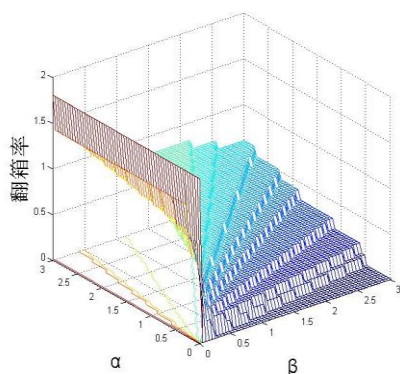


图 3.3 α ， β 对翻箱率的影响

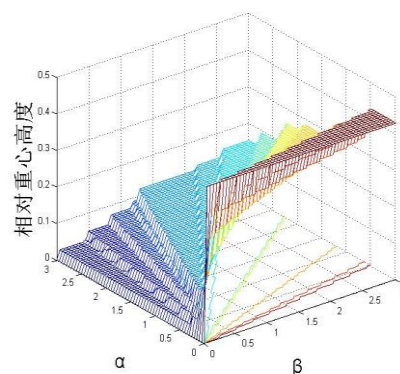


图 3.4 α ， β 对重心高度的影响

其中，重心高度为相对重心高度。计算方法为：相对重心高度=(绝对重心高度-最小重心高度)/最大最小重心高度之差。上面的两幅图成阶梯状，翻箱率和重心高度均和 α/β 有关系。可以观察到翻箱率和重心高度的变化趋势是相反的。随着 α/β 增大，翻箱率增加，重心高度减小。随着 α/β 减小，翻箱率减小，重心高度增加。从上面的两张图中，也可以看出上文中描述的两个基本事实：1.如果只考虑堆放单元的卸货港，那么会造成船舶中心高度很高，不利于满足船舶稳性，威胁航行安全；2.如果只考虑堆放单元的重量，那么会造成船上的翻箱率过高，造成很

大的经济开销。本文提出的这种策略是为了通过设置相关的权值，在翻箱率和重心高度之间找到一个折中，使其满足实际的船舶配载计划。

本文提出的倍内堆放策略可以有效地指导堆放单元的倍内堆放，可有效提高算法的效率。在实际配载过程中，可以根据港口和运输公司的实际需要调整 α/β 的值，以获得满足实际要求的优化结果。

3.4 配载调度模型描述

由于冷藏箱、特种箱等的堆放有其它因素的限制，本文只针对普通出口箱进行建模。本文的决策内容是每个堆放单元分配的倍位和其在倍位中的相对堆放位置。决策目标包括以下几点：

- (1) 最小化船上的翻箱率。船上的翻箱成本较高，并且容易导致一些不安全的因素，减少船上的翻箱次数可以取得很好的经济效益和保证船舶的安全性。
- (2) 最小化重心高度。较低的重心高度可以使船舶保持较好的稳性，保证航行中的安全。
- (3) 满足重箱的中间堆放。重箱的中间堆放可以使船舶受力更加合理，防止产生中拱。
- (4) 保持集装箱船首尾适当的吃水差。适当的吃水差可以使船舶具有较好的航行性能。由于不同船对于最佳首尾吃水差的要求不同，为了本文表述方便，这里约定首尾吃水差为 0 时为最佳吃水差。在实际应用中，根据具体船舶的要求，通过调整吃水差的评价函数来获得具体船舶的最佳吃水差。

3.5 模型的数学描述

- (1) 数学符号定义

BayNum : 船上的倍位数量。

TierNum : 每一个倍中一层可以堆放的集装箱数。

ContainerNum : 待装船的集装箱的数量。

TierNumBay_i : 第*i*个倍位的最大堆放层数。由于本文假设集装箱堆放空间是一个长

方体，所以 $TierNumBay_i = TierNumBay_j (i \neq j)$ 。

$BayW_i$: 第 i 个倍位堆放集装箱的总重量。

w_{ij} : 倍位 i 中第 j 层堆放的堆放单元的总重量。

h_j : 第 j 层的堆放单元距离堆放平面的距离。

l_i : 倍位 i 距离船中心的距离，且靠近船首的倍位距离船中心距离为正值，靠近船尾的倍位距离船中心距离为负值。

$port_{ij}$: 第 i 倍第 j 层的堆放单元的卸货港。卸货港以数字的形式进行定义，例如对于有 5 个卸货港的船舶， $port_{ij} \in \{1, 2, 3, 4, 5\}$ 。其中 $port_{ij} = 1$ 表示离当前港口最近的港口， $port_{ij} = 5$ 表示离当前港口最远的港口。

$TierUsedInBay_i$: 第 i 个倍位已经堆放的层数。

$ReHandleNumInBay_i$: 第 i 个倍位中的翻箱次数。其值由式 (1) 给出。

$$ReHandleNumInBay_i = \sum_{j=1}^{TierUsedInBay_i-1} \sum_{k=j+1}^{TierUsedInBay_i} (a_{ijk} * TierNum) \quad (1)$$

$$a_{ijk} = \begin{cases} 0, & port_{ij} \geq port_{ik} \\ 1, & port_{ij} < port_{ik} \end{cases} \quad (2)$$

$WeightCenter_i$: 第 i 个倍位的重心高度。其值由式 (3) 给出。

$$WeightCenter_i = \frac{\sum_{j=1}^{TierUsedInBay_i} w_{ij} * h_j}{\sum_{j=1}^{TierUsedInBay_i} w_{ij}} \quad (3)$$

$Center$: 重箱堆放船舶中间的评价值。其值由式 (4) 给出。

$$Center = \sum_{i=1}^{BayNum} c_i \quad (4)$$

c_i 的定义为 $c_i = \begin{cases} 1, & \text{if } (BayW_i \leq BayW_{centerp}) \\ 0, & \text{if } (BayW_i > BayW_{centerp}) \end{cases}$ ，其中， $centerp \in CENTER$ ， $CENTER$ 是指船舶中间的倍位编号组成的集合。

$Trim$: 船舶首尾吃水差。其值由式 (5) 给出。

$$trim = \frac{|\sum_{i=1}^{BayNum} l_i * BayW_i|}{\sum_{i=1}^{BayNum} |l_i| * BayW_i} \quad (5)$$

(2) 模型目标的数学描述

①最小化船上翻箱率

$$\min(\frac{\sum_{i=1}^{BayNum} ReHandleNumInBay_i}{ContainerNum}) \quad (6)$$

②最小化平均重心高度

$$\min(\frac{\sum_{i=1}^{BayNum} WeightCenter_i}{BayNum}) \quad (7)$$

③保证重箱堆放船舶中间位置

$$\max(center) \quad (8)$$

④保持船体适当首尾吃水差

$$\min(trim) \quad (9)$$

(3) 模型约束的数学描述

①装箱层数不能超过每个倍位的最大堆放层数

$$TierUsedInBay_i \leq TierNumBay_i \quad (10)$$

3.6 本章小结

本章给出了船舶的实际结构和本文中所提出的通用结构。采用本文提出的通用结构可以保证本文模型和算法的通用性。随着集装箱船舶的大型化，传统的已确定每个集装箱具体堆放位置作为决策内容的配载调度方案已经不在适用，很难在短时间内求解出一套有效的解决方案。基于此，本文提出了集装箱分类策略和操作单元的形成方法。把确定操作单元的堆放位置作为决策目标，有效的减少了问题的解空间。同时，最后的决策结果确定了船上每一个箱位堆放集装箱的类型，为后期的集装箱装船带来了很大的灵活性，满足堆场业务的实时性要求。本章给出了一种同时考虑堆放单元港序和重量的贪心策略，它可以有效的指导倍内堆放

单元的堆放，有利于加快算法的收敛速度。

本章还给出了船舶配载调度的模型描述。本文以确定堆放单元的堆放位置作为决策内容，以最小化船上翻箱率，最小化船舶重心高度，保持适当吃水差和重箱中间堆放作为决策目标。同时，还给出了船舶配载调度模型的数学描述。

第 4 章 基于 GA 的船舶配载调度算法

4.1 遗传算法简介

遗传算法 (GA: Genetic Algorithm) 是一种以达尔文自然进化论和孟德尔遗传变异理论为基础的全局随机搜索优化计算技术, 它同时考察多个候选解, 模拟生物遗传进化过程, 淘汰劣质解, 鼓励发展优质解, 逐步提高解群体的质量, 直至收敛, 获得最优解。同时, 由于遗传算法仅需知道如何根据解求得其相应的适应度, 而不要求适应度函数满足连续可微等条件, 因而得到了广泛的应用。

4.2 本文采用的编码方式

遗传算法编码方式将直接影响算法的性能。多数采用遗传算法求解配载问题的文献中, 采用了基于集装箱的编码方式, 编码串长度为集装箱的个数。这种编码方式在集装箱数比较少的情况下是适用的。但是目前船舶配载时, 集装箱数量一般较多, 这种编码方式显然不再适用。过长的编码长度会严重影响算法的执行效率。文献[21]在求解装箱问题时采用了基于群体的编码方式, 克服了传统编码方法的不足。本文也采用了这种编码方式。同时, 采用倍内堆放策略指导每个倍内操作单元的堆放, 提高算法的收敛速度。

本文遗传表示所要表达的信息是一个倍位中分配了哪几个堆放单元。倍内堆放单元的相对堆放顺序由倍内堆放策略决定。染色体的长度 (即基因数) 为船舶倍位数。一个基因代表一个倍中的堆放情况。一个 10 倍位的染色体如图 4.1 所示, 每个基因位中存储了分配到对应倍位中的堆放单元和这些堆放单元的相对堆放顺序 (也就是堆放单元的编号)。

1	11	2	15	3	4	6	5	7	8
10	12	13	16	17	18	19	21	25	9
31	32	14	33	36	38	20	22	26	27
46	43	45	34	37	39	30	23	29	28
47	44	48	35	49	50	40	24	41	42

图 4.1 染色体示意图

4.3 遗传算子描述

(1) 杂交算子

本文采用了染色体级杂交和基因级杂交两级杂交算子。染色体级杂交算子迭代过程如图 4.2 所示，其迭代步骤如下：

①从种群中选择两条染色体，随机选择两个交叉位置。两个交叉位置中间的基因片段为待交换的基因片段。

②建立集合 S_f , S_s , F , S 。其中 S_f 为基因片段 1 中的堆放单元集合, S_s 为基因片段 2 中的堆放单元集合。 $F = \{a | a \in S_f \wedge a \notin S_s\}$, $S = \{a | a \in S_s \wedge a \notin S_f\}$ 。

③交换基因片段 1 和基因片段 2。

④启动合法性检查。对于基因片段 1，遍历每一个堆放单元 a 。如果 $a \in F$ ，则从 S 中选择一个堆放单元 b 替换 a 。每次从 S 中取出的堆放单元必须是不同的。如果不能再从 S 中取出堆放单元进行替换，则直接删除 a 。检查整条染色体，将没有出现的操作单元重新分配到某一基因中。对于基因片段 2，遍历每一个堆放单元 c 。如果 $c \in S$ ，则从 F 中选择一个堆放单元 d 替换 c 。每次从 F 中取出的堆放单元必须是不同的。如果不能再从 F 中取出堆放单元进行替换，则直接删除 c 。检查整条染色体，将没有出现的操作单元重新分配到某一基因中。

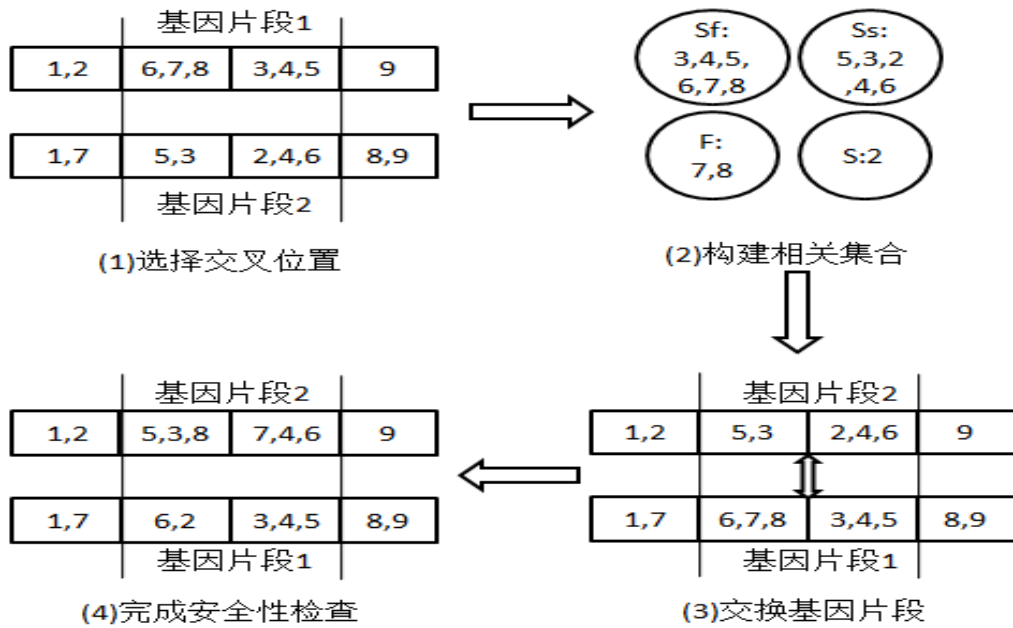


图 4.2 染色体级杂交算子示意图

基因级杂交算子：随机选择一条染色体的两个基因位，读取其中的堆放单元的信息。再随机地将这些堆放单元分配到这两个基因位中。

(2) 变异算子

在一条染色体中，随机选择一个基因位及与这个基因位相邻的基因位。交换这两个基因位的基因信息(即交换这两个基因位中的堆放单元信息)。

(3) 选择算子

采用轮盘赌的选择算子。

4.4 算法步骤

(1) 初始化种群。将堆放单元按照随机方式分配到倍位中。计算每个操作单元的 $wMULp$ ，使用本文提出的倍内堆放策略指导倍内堆放单元的堆放。转到 (2)。

(2) 根据实际配载过程中的相关业务规范对种群中的每一个个体进行评价，得到每一个个体的评价值，并确定其中的当前最优个体。转到 (3)。

(3) 判断算法是否满足结束条件。这里以迭代次数作为算法的结束条件。如果算法结束，则转 (8)，否则，转 (4)。

(4) 执行选择算子。这里采用了轮盘赌的选择策略。转 (5)。

(5) 执行杂交算子。首先执行染色体级杂交算子，启动安全检查，然后执行基因级杂交算子。转 (6)。

(6) 执行变异算子。转 (7)。

(7) 计算每个操作单元的 $wMULp$ ，采用倍内堆放策略指导操作单元的倍内堆放。运用实际配载过程中的相关业务规范对个体进行评价，确定其中的最优个体，并执行精英保活策略。转 (3)。

(8) 迭代结束，打印目前为止的最优解。运用业务规则对最优解进行解释。转 (9)。

(9) 算法结束。

4.5 本章小结

本章首先简要的介绍了遗传算法，遗传算法目前在工程优化领域有十分广泛的应用。然后给出了本文的编码方式。与传统的编码方式不同，本文采用了群体

编码方式，并采用倍内堆放策略指导倍内操作单元的堆放，算法会在更短的时间内收敛。最后，给出了本文采用的杂交算子，变异算子，选择算子和算法流程。

第 5 章 算例及结果分析

5.1 实验假设

本文中实验按照 50 个倍位、60 个倍位、70 个倍位、80 个倍位、90 个倍位、100 个倍位的情况开展。假设所涉及的集装箱均为 40 尺箱（实际中，两个 20 尺箱可拼成一个 40 尺箱）。船上每个倍位集装箱堆放层数为 10 层。船舶挂靠 5 个港口（不包括起始港）。这里考虑将 10 个堆放单元分配到 n （ n 取值 50, 60, 70, 80, 90, 100）个倍位中的情况。堆放单元的重量分布在 5-30 之间。这里给出的重量为堆放单元的相对重量，而非实际重量。

实验包含两个部分，第一个部分验证本文所提出的模型和算法的有效性，第二个部分给出倍内堆放策略的两个参数 α ， β 不同取值下的优化结果。

5.2 模型和算法的有效性验证

设置算法的杂交率为 0.9，变异率为 0.1，迭代次数为 1000。在随机堆放和使用本文提出的算法的情形下，可以得到如表 5.1 所示的结果。其中，倍内堆放策略的参数 α ， β 取值分别取为 3，10。

表 5.1 随机堆放和使用本文算法优化后的结果对比

倍位数	完全随机堆放			采用本文算法与策略		
	翻箱率	相对重心高度	吃水差	翻箱率	相对重心高度	吃水差
50	181.6%	0.4967	0.015044644	0%	0.1953	0
60	173%	0.5303	0.038355497	0%	0.1551	0
70	182%	0.5065	0.021358335	0%	0.2049	0.000004810
80	176.25%	0.4967	0.042130467	0%	0.2030	0.000060438
90	180.89%	0.5011	0.045425535	0%	0.1760	0.000026824
100	179.3%	0.4856	0.048068796	0%	0.2035	0.000011849

对表 5.1 的数据进行分析可以发现，使用本文的算法进行优化后，相关指标均取得了很好的优化效果。在随机堆放情况下，翻箱率在 170%以上，但是采用本文的算法进行优化后，翻箱率基本上保持在 0%。在随机堆放情况下，相对重心高度很高，大概在 0.5 左右。使用本文的算法进行优化之后，相对重心高度降低到 0.2 左右。在随机堆放情况下，前后吃水差离预定的较好吃水差相距很远。采用本文的优化算法之后，吃水差基本上趋于最优吃水差。图 5.1 显示了上述两种情况

下 50 个倍位时船上重量分布的情况。从图 5.1 可以看出，在使用本文提出的优化算法的情况下，中间倍位 (第 22-29 个倍位) 堆放的集装箱重量明显大于其它倍位堆放的集装箱重量，达到了重箱中间堆放的要求。随机堆放情况下，中间倍位堆放集装箱重量和其它倍位相差不大，没有达到重箱中间堆放的要求，容易产生中拱。60-100 倍位时的情况与 50 倍位类似，这里不再列出。

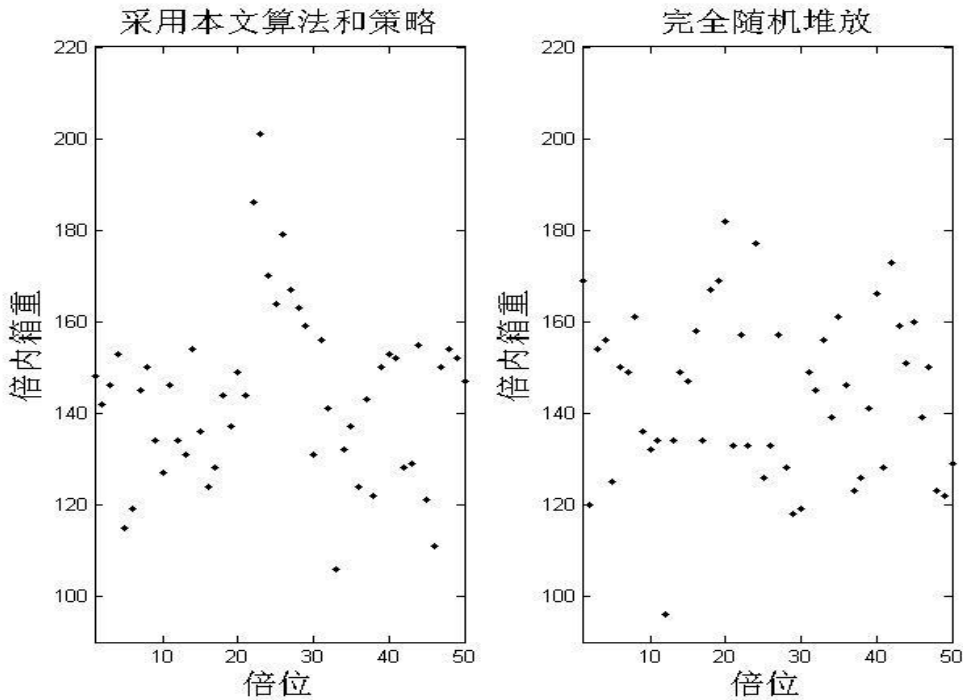


图 5.1 50 个倍位重量分布情况对比

5.3 不同参数配置下的优化情况

通过对本文所提出的倍内堆放策略进行分析不难发现，优化值与 α ， β 的绝对取值无关，只与它们的相对取值有关。这里分别探讨 α/β 在区间 0-1 变化和 β/α 在区间 0-1 变化的优化情况。以期为实际工程应用中的参数选择提供参考。

α/β 在区间 0-1 之间变化时，设置算法的杂交率为 0.9，变异率为 0.1，迭代次数为 1000。可以得到如图 5.2、图 5.3 的实验结果。

从图 5.2 和图 5.3 可以看到，翻箱率一直保持在一个很低的水平，最大的翻箱率不超过 6%。而相对重心高度也维持在 0.1-0.36 的范围内。可以发现， α/β 取值在 0.1-0.4 这个区间时，翻箱率和重心高度这两个指标均获得了比较好的优化值。

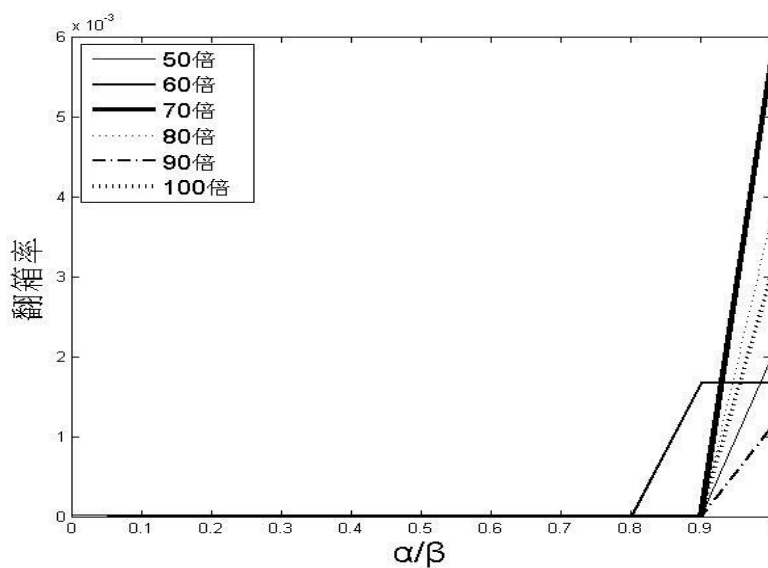


图 5.2 不同 α/β 取值下的翻箱率

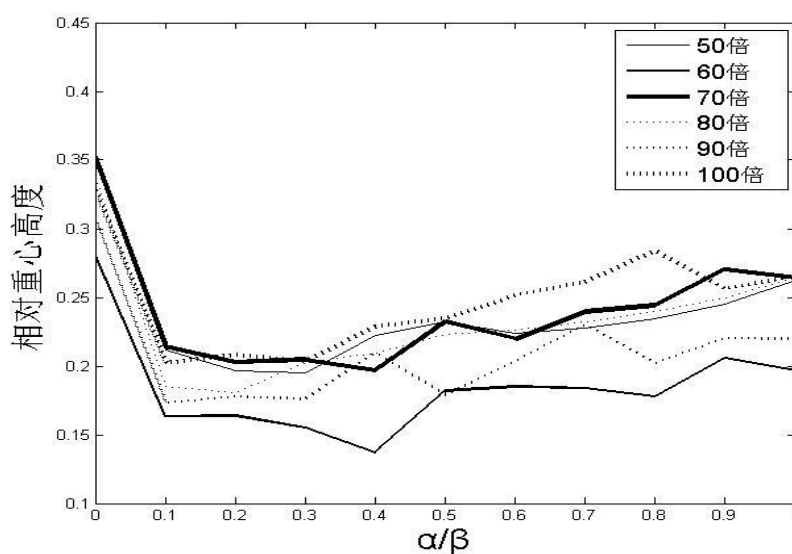


图 5.3 不同 α/β 取值下的相对重心高度

β/α 在区间 0-1 之间变化时，设置算法的杂交率为 0.9，变异率为 0.1，迭代次数为 1000。可以得到如图 5.4、图 5.5 的实验结果。

从图 5.4 和图 5.5 可以看到，随着 β/α 的增大，翻箱率呈现降低的趋势，而重心高度呈现上升趋势。 β/α 取值在 0.7-1 这个取值区间时，翻箱率和重心高度

这两个指标均获得了比较好的优化值。

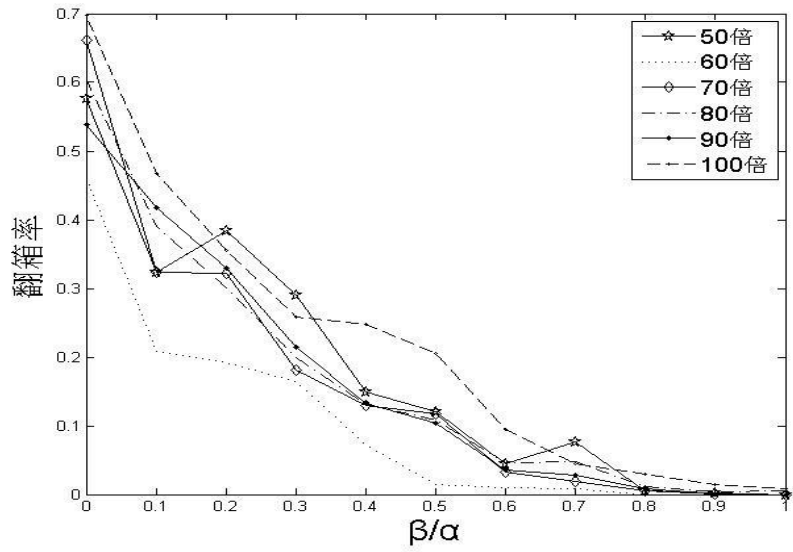


图 5.4 不同 β/α 取值下的翻箱率

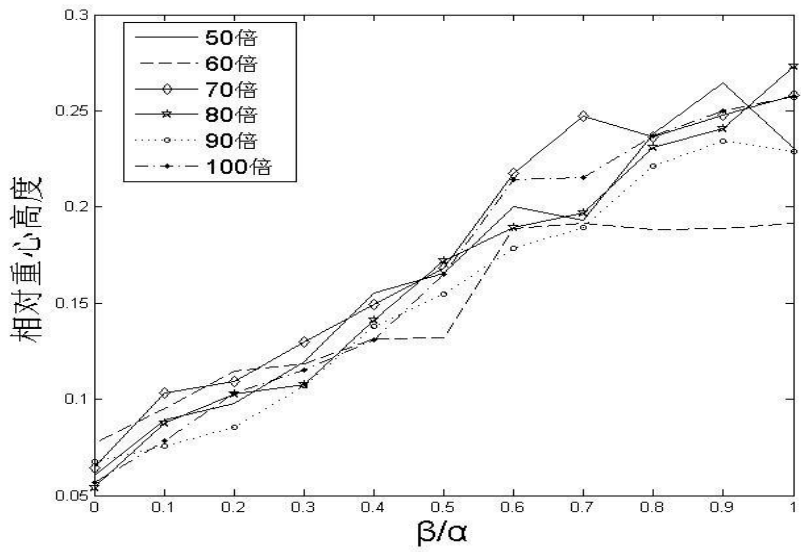


图 5.5 不同 β/α 取值下的相对重心高度

5.4 本章小结

本章分别在 50 个倍位、60 个倍位、70 个倍位、80 个倍位、90 个倍位、100 个倍位的情况下进行了实验。实验分为两个部分，第一部分验证本文提出的算法和策略的有效性；第二部分给出了在不同参数情况下的优化情况。

第一部分的实验表明，本文提出的算法和策略是有效的。能够在满足重箱中间堆放，维持较低重心和合理首尾吃水差的基础上，最小化船上翻箱率。在保证船舶运输安全性的基础上，给码头和船运公司带来了经济效益。

第二部分的实验论证了在几种不同的参数组合之下算法的有效性。通过对比，给出了一个参数区间，在这个区间里，可以获得较好的优化效果。

第 6 章 配载调度软件模块设计与开发

6.1 配载调度软件模块流程图

基于本文所提出的模型和算法，开发了一个配载调度的软件模块。如图 6.1 所示，是船舶配载调度软件模块的流程图。

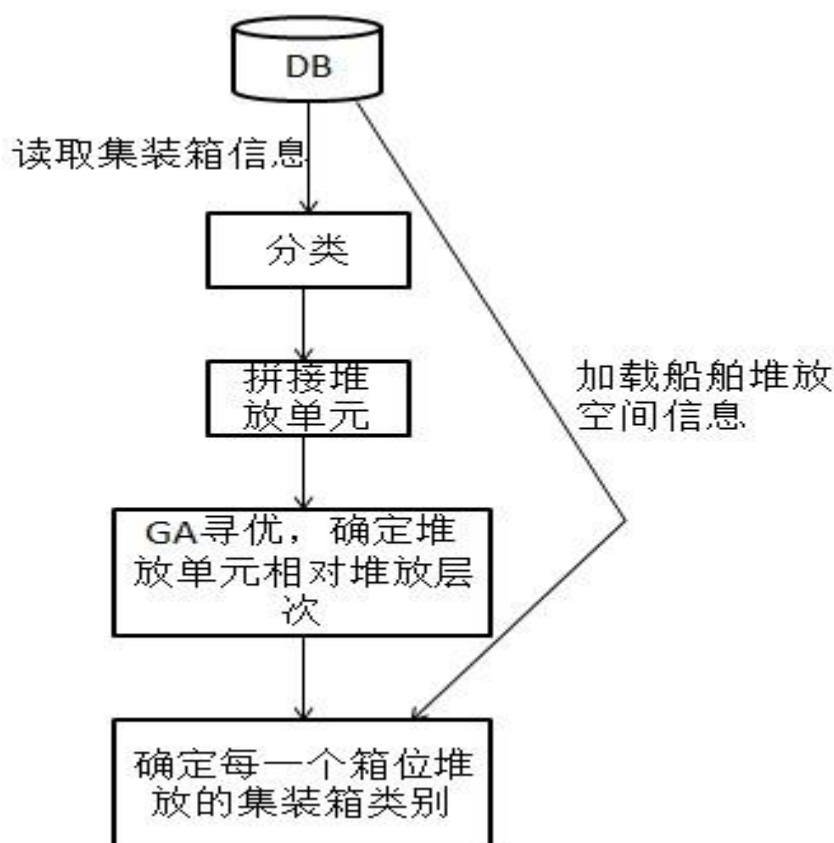


图 6.1 船舶配载调度软件模块流程图

该流程分为以下几步：

- 1) 从数据库中读取集装箱信息和船舶堆放空间信息；
- 2) 依据卸货港，尺寸，所属船舶和重量级将集装箱划分到不同的箱组，在每个集装箱箱组中，以集装箱重量作为指标进行集装箱分类，把重量比较接近的集装箱分为一类。
- 3) 在每一个集装箱类中，取出集装箱拼接堆放单元。
- 4) 采用遗传算法求解建立的配载调度模型，确定各个堆放单元的相对堆放层次。
- 5) 采用一定的策略确定船上每一个箱位堆放的集装箱类别。

6.2 配载调度软件模块数据表设计

(1) container 表

container 表中存储了堆场中的所有集装箱信息。包括集装箱的集装箱号、类型、尺寸、卸货港、船号、重量、核对号和箱类号等信息。其中，箱类号是指通过本文的集装箱分类策略所得到的集装箱的箱类。

container 表的表结构如表 6.1 所示：

表 6.1 container 表结构

表名		container				
主键		containerid				
序号	字段名称	数据类型(精度范围)	允许为空 Y/N	唯一 Y/N	默认值	约束条件/说明
1	containerid	char(12)	N	Y		集装箱号
2	type	varchar(25)	N	N		类型
3	size	int(11)	N	N		尺寸
4	port	varchar(25)	N	N		卸货港
5	shipid	varchar(25)	N	N		船号
6	weight	int(11)	N	N		重量
7	checkid	int(1)	N	N		核对号
8	classid	varchar(11)	Y	Y	NULL	箱类号

(2) stackspacedistribution 表

stackspacedistribution 表中存储了船舶堆放空间的结构信息和集装箱在船舶堆放空间的堆放信息。包括船号、倍号、排号、层号、箱位是否占用、箱类号和集装箱号。其中，倍号、排号和层号可以用来表示一个集装箱箱位。箱类号表示堆放在该箱位的集装箱的类型号，集装箱号表示堆放在该箱位的集装箱号。

stackspacedistribution 表的表结构如表 6.2 所示。

表 6.2 stackspacedistribution 表结构

表名		stackspacedistribution				
主键		ShipID, Bay_NO, Bay_X, Bay_Y				
序号	字段名称	数据类型(精度范围)	允许为空 Y/N	唯一 Y/N	默认值	约束条件/说明
1	ShipID	varchar(25)	N	Y		船号
2	Bay_NO	int(10)	N	N		倍号
3	Bay_X	int(10)	N	N		排号
4	Bay_Y	int(10)	N	N		层号
5	Flag	int(1)	N	N	0	是否已占用
6	ContainerClassID	varchar(11)	Y	N	NULL	箱类号
7	ContainerID	char(12)	Y	Y	NULL	集装箱号

6.3 配载调度软件模块的子模块实现

(1) 船结构配置子模块

本文构造了一个通用的集装箱船舶的堆放空间，该堆放空间为一个长方体空间。不同船舶的集装箱堆放空间一般是不同的，这里通过禁止相应的箱位来把堆放空间配置成相应的船舶的堆放空间。配置完之后，把信息写入数据库的 `stackspacedistribution` 表中。`Flag=0` 表示对应的箱位可以使用，`Flag=1` 表示对应的箱位被禁止。

(2) 分类子模块

分类子模块的功能实现主要分为两步，首先根据集装箱的装载船舶，卸货港，尺寸，类型来进行预划分，形成多个箱类。然后在每个箱类中，根据集装箱的重量进行聚类，从而形成最终的箱类划分。这里需要说明的是，在对集装箱进行分类时，往往是对指定船舶的集装箱进行分类，所以待分类集装箱的装载船舶都是相同的。

(3) 堆放单元拼接子模块

在完成集装箱分类操作之后，从每一个箱类中选择集装箱来拼接堆放单元。对于每一个集装箱类，每次从中抽取 N (N 表示一个倍位中一层可以堆放的集装箱数) 个集装箱来拼接成一个堆放单元。当一个箱类中的集装箱数不足 N 个时，可以根据具体情况进行处理。例如，可以把剩余的这些集装箱也拼接成一个集装箱类。本文在实现子模块功能时，就采用了这种方式。

(4) 基于群体编码方式的遗传算法子模块

在得到堆放单元之后，采用文中所描述的基于群体编码方式的遗传算法和倍内堆放单元堆放策略对模型进行求解。

(5) 确定箱位堆放箱类子模块

遗传算法求解完成之后，得到堆放单元的相对堆放层次。在这个的基础之上，需要确定每一个箱位所堆放的集装箱类型。这里所采用的策略是，按照堆放单元的相对堆放层次的顺序遍历堆放单元中的每一个元素，在结合船舶箱位的分布情

况来确定每一个箱位堆放的集装箱类别。

6.4 配载调度软件模块软件功能演示

(1) 配置集装箱船舶结构

如图 6.2，左侧的列表表示船舶信息表。分别显示船舶的编号和这条船舶目前所处的状态。中间的列表显示的是对应的船舶的倍位编号。选择其中一个倍位，右侧就会显示出该倍位的倍图。图中的绿色区域表示空闲的箱位。红色区域表示禁用的箱位。选择某一个箱位，右键，可以对该箱位进行禁止或启用操作，这样就可以对船舶的结构进行配置。

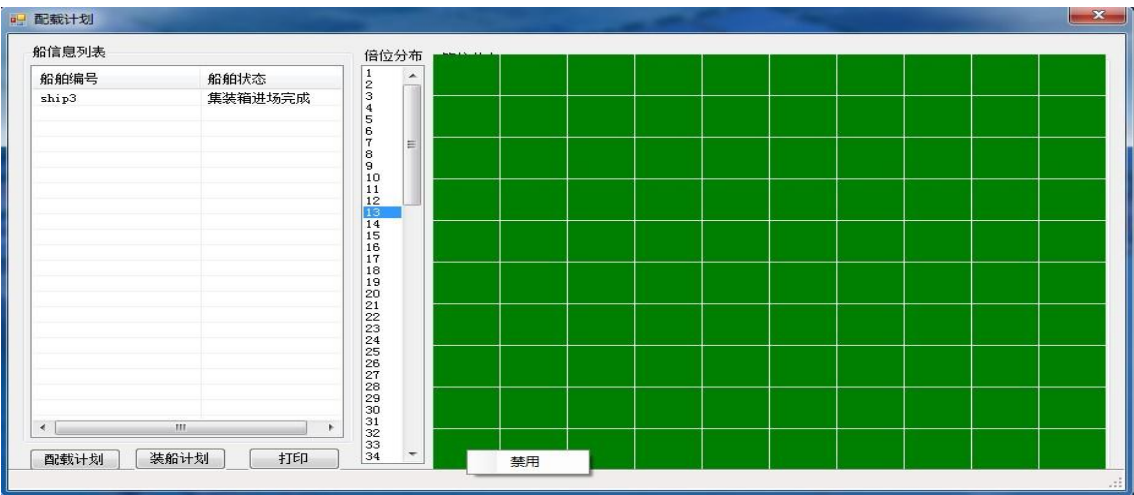


图 6.2 配载计划界面

如图 6.3 所示，单击右键点击“禁用”之后，相应箱位被禁用且被设置为红色。

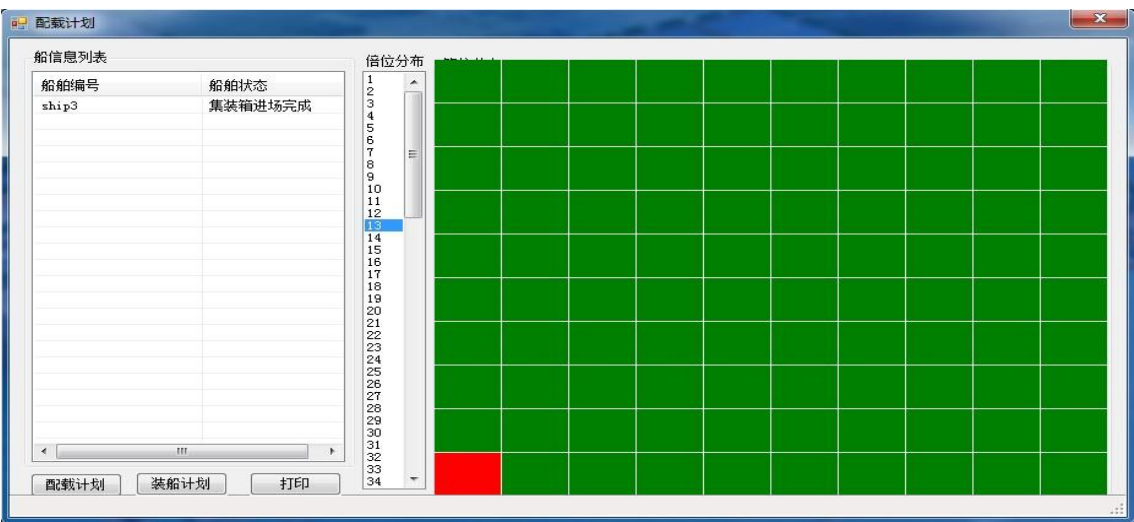


图 6.3 通过点击右键禁用一个箱位

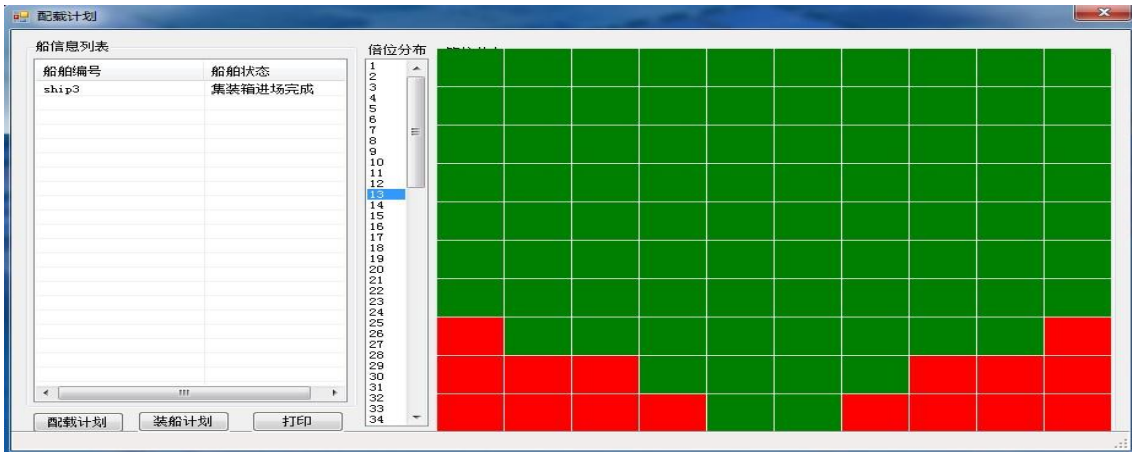


图 6.4 配置好的堆放空间

如图 6.4 表示一个配置好的船舶的一个倍。

(2) 选择船舶进行配载计划

如图 6.5, 在船信息列表中选择一条船, 点击配载计划按钮之后可以进行配载计划。如图 6.6, 表示配载计划正在进行。

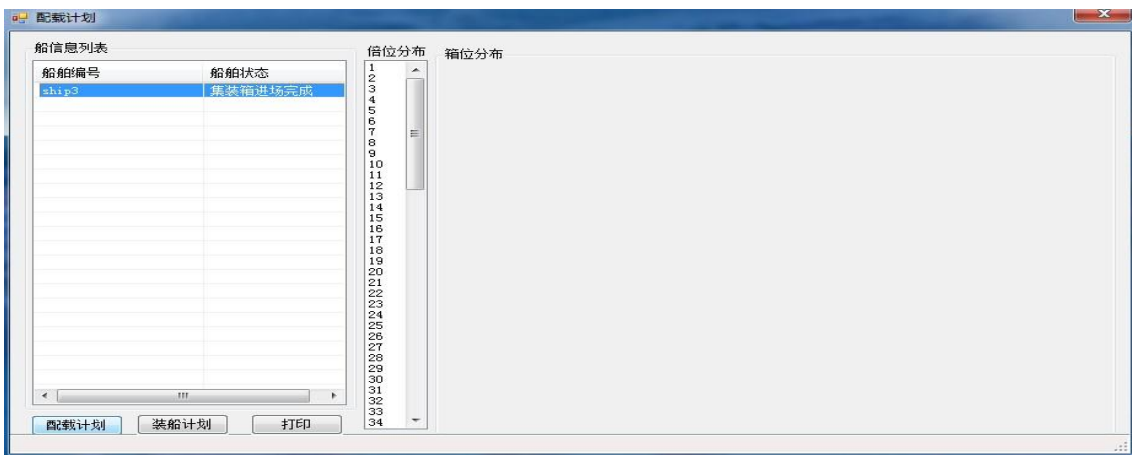


图 6.5 选择一条船舶进行配载计划

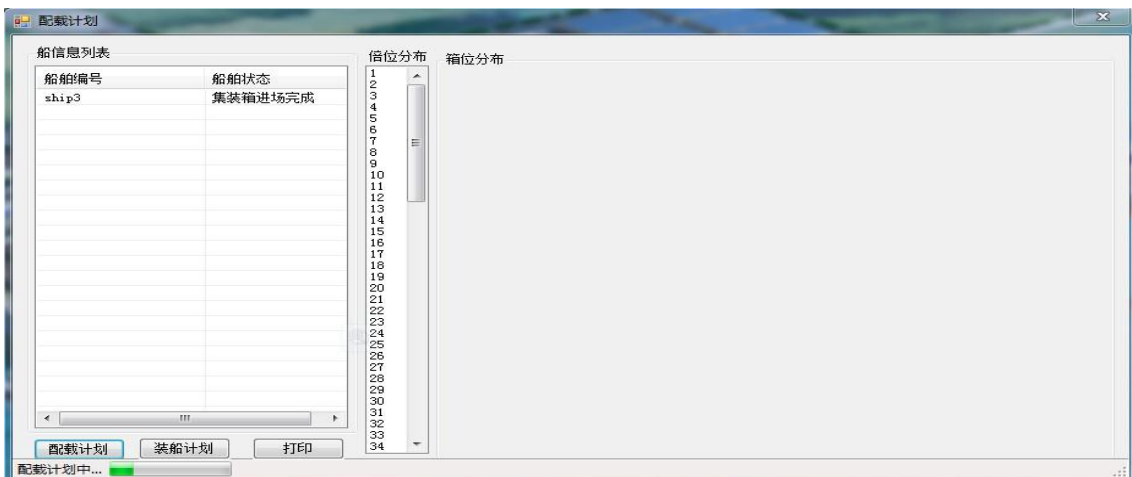


图 6.6 配载计划进行中

(3) 配载计划结果显示

如图 6.7，是指船舶 ship3 的第 44 个倍位的配载计划结果。图中红色区域表示禁止掉的箱位，也就是说在 ship3 的第 44 个倍位中，这些箱位是不存在的。粉红色区域表示已经有集装箱分配到了对应的箱位。上面给出的编号表示集装箱类别号。在装船过程中，对于某一个箱位，可以选择该类中的任意一个集装箱堆放在此处。

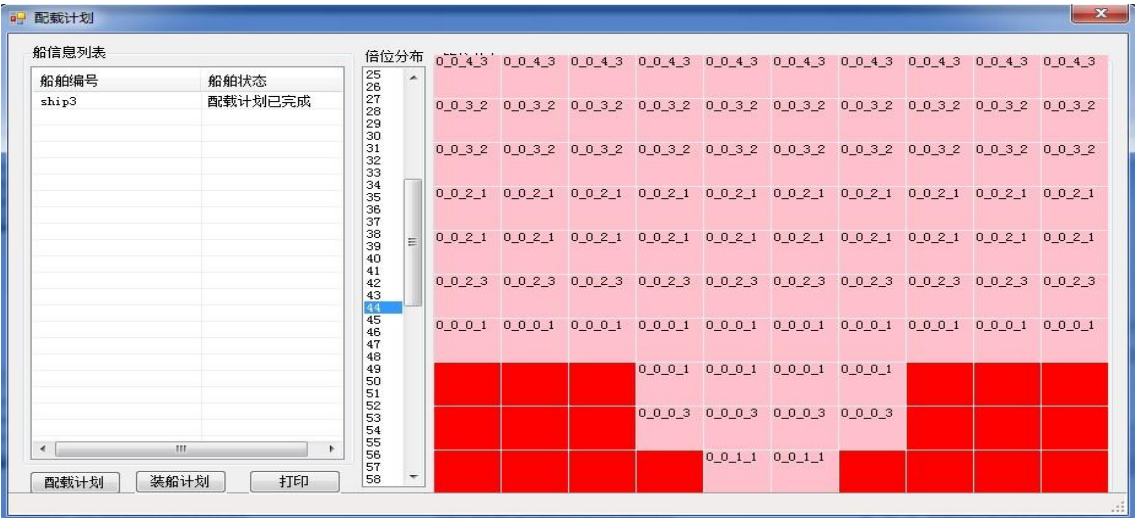


图 6.7 配载计划结果显示

6.5 本章小结

本章简要的介绍了配载调度软件模块的设计与开发，给出了模块流程图，数据表设计和子模块的实现，并给出了配载调度软件模块的功能演示。

第 7 章 总结与展望

7.1 论文总结

集装箱船舶配载调度问题是一类典型的组合优化问题。传统的集装箱船舶配载调度以确定每一个集装箱在船上的具体堆放位置作为决策内容，但是随着集装箱船舶的大型化，配载过程中的集装箱数量增加，问题规模急剧变大，增加了这类问题的求解难度，很难在一个较短的时间内求解出一套满足实际要求的、效果较好的解决方案。同时，采用这种方式进行求解，限制了装船过程的灵活性，不利于满足堆场业务的实时性要求。针对这些问题，本文从以下几个方面展开研究：

(1) 本文提出了集装箱分类策略、堆放单元拼接方法和倍内堆放单元的堆放策略。集装箱分类策略主要从集装箱的卸货港，尺寸，装载船舶，重量等几个指标出发，采用一定的策略对集装箱进行归类。在完成集装箱分类后，拼接堆放单元，本文中配载调度的操作实体是堆放单元。本文从倍内堆放单元的重量和卸货港这两个指标入手，提出了倍内堆放单元的堆放策略。此策略实际是对于重量和卸货港这两个重要指标的权衡，它可以有效地指导倍内堆放单元的堆放，帮助加快算法的收敛速度。

(2) 本文提出了船舶配载调度模型和求解算法。本文以堆放单元的相对堆放位置和集装箱船舶每一个箱位堆放的集装箱类型作为决策内容，以最小化船上翻箱率，最小化船舶重心高度，保持适当吃水差和重箱中间堆放作为决策目标。以堆放单元相对堆放位置作为决策目标，可以减小问题的解空间，在一定程度上解决了船舶大型化所带来的问题规模太大而无法求解的挑战。同时，最后的决策结果给出了每一个箱位堆放的集装箱类型，这样给后续的集装箱装船带来了很大的灵活性。本文采用了一种基于群体编码方式的遗传算法来求解模型。采用这种编码方式，染色体的编码长度大大减小，同时，采用堆放单元倍内堆放策略指导倍内堆放单元的堆放，算法收敛速度更快，求解效果更好。

(3) 实验仿真和船舶配载调度软件模块的实现。本文在 50 个倍位、60 个倍位、70 个倍位、80 个倍位、90 个倍位和 100 个倍位的条件下开展了实验。实验分为两个部分，第一个部分验证本文模型和算法的有效性；第二个部分给出了倍内

堆放策略不同配置下的优化情况，以期为实际工程中的应用提供参考。本文还给出了基于所提出的模型和算法开发的配载调度软件模块的功能演示。

7.2 论文展望

集装箱船舶配载调度问题是一个复杂的组合优化问题。配载中涉及的因素很多，本文在建模过程中，考虑了几个相对重要的因素，在以后的工作中，会把更多的因素考虑进来，使建立的模型能够更好的符合实际情况。同时，港口是一个对于实时性要求很高的场所，所以提高配载调度的效率也是未来需要解决的一个问题。

参考文献

- [1] 沈继平. 集装箱码头配载业务初探[J]. 集装箱化, 2007,18(1):9-12
- [2] 刘彦斌. 多类型集装箱装船模型及优化[J]. 小型微型计算机系统, 2007, 28(11): 1988-1991
- [3] 樊铁成. Pareto 遗传算法在集装箱配载优化中的应用[A].第二十四届中国控制会议论文集[C], 广州: 中国自动化学会, 2005 年: 1377-1381
- [4] 王启友. 集装箱船的单元块配载[J]. 集装箱化, 2007, 18(8): 26-29
- [5] 段成华. 基于整数规划的单贝多港 SBMP 配载模型研究[J]. 计算机辅助工程, 2004, 13(3): 9-13
- [6] 杨晓东. 仿真技术在集装箱船舶配载问题中的应用研究[J]. 应用技术, 2010,(26): 314-315
- [7] 宋毅. 基于 Web 的集装箱船舶箱管系统设计与实现[J]. 计算机工程, 2003, 29(10): 161-163
- [8] 张维英. 集装箱船全航线 Bay 位排箱优化模型[J]. 上海交通大学学报, 2007, 41(2): 199-204
- [9] 李虎. 一种求解船舶配载问题的混合遗传算法[J]. 工业工程与管理, 2006, 11(3): 27-31
- [10] 张维英. 集装箱船全航线配载智能优化研究[D]. 大连: 大连理工大学, 2006
- [11] 喻斌. 集装箱船舶智能配载方法研究[D]. 武汉: 武汉理工大学, 2007
- [12] 李波. 基于稳定约束的集装箱配载计划启发式算法研究[D]. 天津: 天津大学, 2008
- [13] 刘恒江. 基于 Petri 网的集装箱空箱调运仿真分析[J]. 交通运输工程学报, 2002, 2(3): 97-102
- [14] 张维英. 集装箱船全航线预配优化模型与算法研究[J]. 大连理工大学学报,

2008, 48(5): 673-678

- [15] 王莉莉. 集装箱装船顺序优化模型及遗传算法[J]. 计算机工程与应用, 2008, 44(15): 234-238
- [16] 张维英. 基于指派问题的 Bay 位排箱优化模型与算法[J]. 大连理工大学学报, 2011, 51(1): 61-67
- [17] 熊剑锋. 集装箱船的大型化趋势及其制约因素[J]. 船舶设计通讯, 2001(1): 8-10
- [18] LI Li. GA-based Container Terminal Stockpile Block Plan Optimization and Simulation[A]. 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing[C], Washington, DC: IEEE Computer Society, 2010: 438-442
- [19] SUN Ni. Reserving-based Strategy for Bay Allocation[A]. 2nd International Symposium on Computer Network and Multimedia Technology [C], Washington, DC: IEEE Computer Society, 2010: 291-295
- [20] MIN Zhengyu. Mixed-row-based Space Allocation Strategy for Outbound Containers[A]. 2nd International Conference on Information Engineering and Computer Science – Proceedings[C], Washington, DC: IEEE Computer Society, 2010: 231-236
- [21] 张大斌. 基于群体编码方式的遗传算法求解装箱问题[J]. 计算机工程与设计, 2008, 29 (12): 3154-3156

致谢

论文的顺利完成，首先要感谢我的导师胡文斌副教授，论文从选题、初稿到最终定稿的整个过程中，胡老师都提出了许多宝贵意见。导师学识渊博，充满激情，对待工作一丝不苟，对待学生认真负责。不仅使我在学业方面获益匪浅，也使我更加确信，只要认准目标，认真钻研，就一定会有所成。在此，特向他道声谢谢。

其次，要感谢大学期间一起做学生科研的伙伴，李立，罗鹏，鲁彬。李立师兄是我的算法入门导师，也是我最要好的朋友。相信在以后的日子里，我们会有更多的合作，一起进步。罗鹏具有较强的算法能力、鲁彬具有相当的钻研精神，他们对研究课题的热爱，促进我更加努力学习、钻研。感谢他们对我的支持和帮助，在此，向他们表示感谢，希望他们在未来的学习道路上展示自己的风采，绽放光芒！

一直以来，我的家人在背后默默的支持着我。当我因为学习压力较大时，是他们给了我鼓舞和信任，正是他们的信任，使我坚信自己可以做得更好，走得更远！

毕业设计，也许是我大学生涯最后的作业。想借此机会感谢四年以来给过我帮助的所有老师、同学，你们的友谊是我人生的财富，是我生命中不可或缺的一部分。感谢所有给过我帮助的师长、亲人、朋友！