# 基于船舶配载决策支持系统的研究

## 封面

## 承诺书

## 摘要

通过研究船舶配载决策支持系统的实现，基于B/S架构实现一套基本的船舶配载决策支持系统，系统支持船舶配载信息的监控，堆场箱位信息的实时监控， 船舶预配载信息的监控，船舶调度计划的实时更新，不同类型船舶结构信息的适配显示，堆场到船舶集装箱出场的手工配载，船舶贝位调整，船舶配载问题中新调度算法或决策模型的导入，基于遗传算法和模拟退火算法的初始模型库。通过人机友好界面设计以及模型库的引入辅助决策船舶配载问题。

**关键词：** B/S架构、模型库、辅助决策

## Abstract

By studying the implementation of the ship's stowage decision support system, we implement a basic ship's stowage decision support system based on Browser/Server architecture. The system supports the monitoring of ship stowage information, real-time monitoring of the containers' position in the yard, monitoring of ship pre-loading information, real-time updating of ship sheduling plan, adaptable display of different types of ship, manual loading of container from yard to ship, ship shell adjusting, the import of new scheduling algorithm or new decision model in ship scheduling problem, the initial model library based on genetic algorithms and simulated annealing algoritms. Through the Graphic User Interface design and the introduction of the model library to assist decision-making ship loading problem.

**Key words:** B/S architecture，model library, decision support

目录

[基于船舶配载决策支持系统的研究 1](#_Toc8128815)

[封面 1](#_Toc8128816)

[承诺书 2](#_Toc8128817)

[摘要 3](#_Toc8128818)

[Abstract 3](#_Toc8128819)

[第一章、绪论 5](#_Toc8128820)

[1.1 研究背景 5](#_Toc8128821)

[1.2 研究目的及意义 6](#_Toc8128822)

[1.3 国内外研究现状 7](#_Toc8128823)

[1.4 文章内容及结构 7](#_Toc8128824)

[第二章、 相关技术理论概述 8](#_Toc8128825)

[2.1 决策支持系统 8](#_Toc8128826)

[2.2 自动化集装箱码头 9](#_Toc8128827)

[2.3 集装箱码头的配载 9](#_Toc8128828)

[2.4 系统技术概念 10](#_Toc8128829)

[2.5 算法概念 13](#_Toc8128830)

[2.6 船舶配载规范 14](#_Toc8128831)

[第三章、 实体分析 16](#_Toc8128832)

[3.1 船舶实体分析 16](#_Toc8128833)

[3.2 岸桥实体分析 20](#_Toc8128834)

[第四章、 系统规划 24](#_Toc8128835)

[4.1 船舶结构 24](#_Toc8128836)

[4.2 船舶配载过程 24](#_Toc8128837)

[第五章、系统界限 26](#_Toc8128838)

[5.1 船舶相关的信息分析 26](#_Toc8128839)

[5.2 船舶配载业务功能调查 26](#_Toc8128840)

[5.3 船舶配载业务流程调查 26](#_Toc8128841)

[5.4 数据汇总 26](#_Toc8128842)

[5.5 数据流程图 26](#_Toc8128843)

[5.6 数据字典 26](#_Toc8128844)

[第六章、数学模型 26](#_Toc8128845)

[6.1 算法模型库 26](#_Toc8128846)

[第七章、 系统设计 26](#_Toc8128847)

[7.1 功能结构图设计 26](#_Toc8128848)

[7.2 系统流程图 27](#_Toc8128849)

[7.3 数据存储设计 27](#_Toc8128850)

[7.4 处理流程图 27](#_Toc8128851)

[7.5 程序设计 27](#_Toc8128852)

[第八章、模型评价 27](#_Toc8128853)

[8.1 参数指标 27](#_Toc8128854)

[第九章、 结论和展望 27](#_Toc8128855)

[参考文献 28](#_Toc8128856)

[致谢 29](#_Toc8128857)

[附录 30](#_Toc8128858)

[代码部分 31](#_Toc8128859)

## 第一章、绪论

### 1.1 研究背景

改革开发以后，各个行业不断发展。经济方面，我国的经济增长地趋势是不断的；科技方面，我国科技加强的趋势是不断的。这些说明，社会主义制度是足够的优越。与此同时，这个时代，它是足够开放的时代，它是足够包容的时代，它也是足够和平的时代。经济不断增长同时在各个国家不断发展，结果使得经济和文化的交流在国家与国家之间变得更加密切，相互的关系也变得更加紧密。日益紧密地经济文化交流，促使国际化商品交易更加频繁，对进出口贸易的需求，国际化运输的需求（包括公路，水路，铁路等）不断提高。而集装箱码头港口，作为可以充当进出口商品功能的门户，必然要承担相关需求的增长。又是因为港口面对需求的功能的特殊性，它不仅要处理好商品的正常的进出口，更要进一步提高商品进出口操作的效率效益。这里，就要求港口作业，要更高——更高的标准，更快——更快的响应时间，更强——更强大的系统支持。

然而，祖国港口的发展建设的过程中，机遇和挑战，这两种问题，它们一直存在着，一方面，2008年发生了国际性的金融危机，在这场经济危机发生之后，很多国家的经济秩序被冲击的程度已经相当严重，同时各国的实体经济遭受的影响也相当严重，于是，与实体经济密切相关的航运经济，也严重受损，曾经我国的航运业经济一度低迷；另一方面，鉴于以上，“五位一体”总体布局的概念现如今被国际提出，目的是努力建设现代化的经济体系，同时坚持供给侧改革。由原来的以价格取胜，过度生产等模式，转变为以质量取胜，合理控制生产。同时，科学技术的发展，现如今，云计算，云服务，云同步的概念，不断兴起，这也帮助港口的建设使用更智能化。

现如今，进行对外进出口贸易的港口的建设，已经如雨后春笋般撒满了祖国的各个沿海发达城市。港口城市方面，北有天津港——中国北方的第一个5亿吨港口，青岛港——环渤海湾集“航运、物流、金融”为一体的港口，中部有洋山深水港（首个全自动化集装箱码头），南部又有厦门港（东南沿海的门户）和广州港等，分别承担各个区域的港口作业。

在港口作业的具体实际业务层面，随着经济的发展，科技的进步，港口码头集装箱装卸作业面临的新的选择，新的标准，新的要求和新的挑战。一方面，港口码头的装卸作业，通过结合当前图像识别、机器学习，极大的加快了港口码头集装箱作业的运行效率以及作业准确度，另一方面，通过结合当前的新型的管理信息系统，实物仿真调度系统，Unity 3D 画面动态呈现等功能，帮助了港口码头调度人员进一步做更好的决策，在处理问题的过程中，引入新的技术，在引入新的技术的过程中，尝试新的方案，在尝试新的方案的过程中，让港口码头集装箱装卸作业由半自动化走向全自动化。

谈到港口码头集装箱装卸作业，无论是装载还是卸除，船舶配载问题又是重中之重，那么提高港口码头中集装箱的装卸作业的运行效率过程中要做好的其中一个重要任务，就是处理好船舶的配载问题。

### 1.2 研究目的及意义

现实船舶配载作业中，通常要处理很多问题，由于实际集装箱尺寸上存在20尺、40尺甚至45尺；按照集装箱的功能划分，又分为危险品类型箱，普通型箱，冷藏型箱，特种类型箱等；按照重量划分，又分为轻型箱，中型箱，重型箱，超重箱。实际作业中，种类多变的集装箱的类型，导致船舶货物配载工作难度加大，复杂度大大提升。并且装卸过程中要考虑多泊位，多贝位，多堆场的情况，现有的国内港口码头装卸工艺，具有代表性的即洋山四期深水港，其规定的具体装卸规则中，又要考虑岸桥不可交叉作业的约束限制，组贝问题（20尺贝位和40尺贝位的合并），船舱位置对岸桥调度作业的影响，舱盖板的数量，舱盖板的位置，甲板层上集装箱层数，甲板层下集装箱层数等一系列问题。与此同时，传统的C/S架构实现的船舶装配决策支持系统，已经无法满足需求日益增长的港口操作业务，现如今新型的基于互联网模式的B/S 架构下的船舶装配决策支持系统的建设问题，有待解决。实际的开发过程中，很多技术问题会限制相关功能的正常实现，由于内部核心的引擎在不同的浏览器之间有很大的区别，比如Chrome浏览器的Blink内核，IE的 Trident内核，Firefox浏览器的Gecko内核，Safari的Webkit内核以及Opera浏览器的Blink内核这几个主流浏览器。内核不同，导致具体做不同浏览器的兼容适配会出现问题，现有方案就是，提倡并建议用户进行浏览器系统升级，放弃落后的IE6、IE7等现如今微软官方不支持的浏览器，转而去使用现如今的主流浏览器，方便相关功能的适配；由于船舶配载问题的特殊性，现如今主流的UI(User Interface)框架，无法满足其相关功能的开发，严格讲，关于船舶配载信息监控，堆场和船舶之间的装载卸载问题，人工手动配载、自动配载问题，不同泊位不同船舶的不同结构加载问题，这要求前端的实现只能在需要在充分了解相关业务的前提下，去客户化定制或生成相关的功能页面设计。另外，船舶配载过程中，包含了大量数据的传输，监控信息的实时性更新，这背后，不仅需要硬件部分（如传感器，控制器等）进行支持，还需要网络传输部分的稳定执行，这在具体页面的实现上，需要考虑页面的发送请求问题，脚本加载速度，以及数据发送接收后的解析问题，页面渲染速度等。在前端部分实现的同时，同样的，要充分考虑后端的处理压力，实际船舶配载问题，管理上，会涉及到不同级别管理员对信息进行增删改查，分权限管理问题，很难产生清晰的管理方案，又因为船舶配载问题的复杂性（不同业务之间的关联），导致很多业务功能之间具有较高的耦合性，这对后台的功能模块化开发，又增加了实现难度；数据库方面，较大数据量的输入输出，导致传统的增删改查模式可能无法满足其现实的需求，更多的，如引入数据库开发概念中的存储过程，再比如引入视图机制（不同管理权限的管理员对数据库的操作），在处理事务并发（配载过程中可能会产生的资源抢占而导致的输入输出问题）过程中，对乐观锁，死锁等问题的处理等。

实际上，以上问题，更多是船舶配载的监控层面的功能实现，但这种监控层面的功能实现，只是决策支持系统的一部分，具体船舶配载问题的决策优化，还需要算法模型的引入，如何建立算法库，模型库，以及如何建立船舶配载问题中决策支持的知识库，如何在建立该决策支持系统的同时，处理整个系统的输入输出问题（接口信息的处理），如何保证整个系统的建立可以随着实际船舶配载操作。

通过对船舶配载问题的决策支持系统的研究，在人机交互层面，实现船舶配载信息监控以及船舶装配的手工指定； 在算法模型层面，引入方法库、模型库，生成配载调度的决策信息，方便仿真系统的仿真实施； 在数据库存储层面，引入关系模型，关联存储船舶配载相关信息，并及时进行增删该查，有利于港口码头调度作业中其他子系统的信息获取，以及该系统内部信息的关联变化。

### 1.3 国内外研究现状

基于船舶配载相关决策问题，很多研究已经在国内外展开。国内方面，大连海事大学杜嘉立曾提出以专家系统为基础去产生智能化控制船舶配载的策略，该方案可以提供船舶在装卸的每一个子进程中，集装箱数量的多少，装入船舱的时间先后，顺序等性能指标[1]。计明军通过对集装箱船舶配载策略展开并进行比较分析，以四种配载方式划分，通过对比不同模式下船舶的倒箱率大小、稳定性、强度、吃水深度差，分析出不同配载方式的优势和不足，从而为减少实际船舶倒箱率问题提供辅助决策支持[2]。近几年的研究中，上海海事大学的物流研究中心的丁一等人，以堆场箱区的作业均衡性为前提，去研究港口码头的船舶配载问题， 减少了总装船时间的平均值，翻箱数量，以及不均衡程度，同时可以满足箱区作业的最大程度的均衡[3]。杨蔚等人在研究集装箱班轮运输的配载优化时，把船舶的稳定性作为条件约束，求解目标设定为整条航线最小倒箱量之和，数学模型的构建是基于0-1整数规划，并通过改进遗传算法，从而提升了配载计划的稳定性，以及船舶运营的安全性[4]。

### 1.4 文章内容及结构

#### 1.4.1 主要内容

本文前面主要讲船舶配载决策支持系统，所需要的技术框架，工具，概念，作为整个研究的引入，后面主要讲决策支持系统各个功能的定义与开发过程，同时探讨并尝试实现算法库模型库引入船舶配载决策支持系统。

#### 1.4.2 论文结构

第一章，绪论部分，背景、现状和研究意义的介绍。

第二章，理论概述部分，论文研究以及决策支持系统开发过程中涉及的重要技术功能以及船舶配载业务的展开。

第三章，实体分析部分，对船舶进行实体分析，分析其属性，以及其和外部实体之间的联系。

第四章，系统规划部分，描述船舶配载部分的系统实施目标。

第五章，系统边界部分，定义本系统的边界问题，便于进一步引入并分析系统的输入输出问题。

第六章，系统设计部分，包含整个管理信息系统的业务功能的设计实现。

第七章，结论和展望部分，分析本次研究的优势与不足。

结尾附录部分，决策支持系统开发过程中的核心要素。

## 第二章、 相关技术理论概述

### 2.1 决策支持系统

决策支持系统（英文全称叫Decision-Support-Systems，简称DSS），它可以利用收集的现有信息帮助需要决策的决策者进行辅助决策，协助进行商业级或组织级别或者其他级别的决策活动的信息系统。DSSs通常服务于中高层面管理人群，作用于组织机构内部操作、管理、和规划级的决策，帮助决策者对快速变化同时决策者自身很难提前确定的问题进行决策分析，提出相应意见或者建议。决策支持系统的建立，既要依赖知识库、方法库、模型库，又会不同程度的涉及商业事务中的数据仓库、数据挖掘等问题。需要运用决策支持系统处理的问题，通常是非结构化（NonStructured）或者半结构化（SemiStructured）的，也就是在实际管理工作中，中级管理人员和高级管理人员要面对的问题，理论上，它的最终发展目标是智能化决策。决策支持系统既可以是完全脱离外在干预的自动化决策，也可以是相关信息反馈给决策者之后的完全人工决策，或者两者兼有[5]。

决策支持系统按照分类的不同维度有以下几种：通过该系统支持的管理层次划分，分为高层决策支持系统和中层决策支持系统；通过持赞同建议的决策者的数目，分为组织决策支持系统和个体决策支持系统；通过系统的智能程度，分为智能决策支持、传统决策支持以及专家系统[6]。

### 2.2 自动化集装箱码头

随着港口经济的发展，传统的集装箱码头的作业方式，以及很难再去满足当前巨大的港口装卸作业需求。不难想象，滞后的集装箱码头作业方式，将会大量的耗费人力以及物力， 假如不做改变，这里将会损失巨大的经济效益。为此，一种基于自主装卸操作的集装箱码头(英文缩写为ACT)的概念被提出。这里，我们把它称为自动化集装箱码头。 实际上，到目前为止,自动化集装箱码头技术的发展的时间已经超过了30年。 上个世纪80年代的中期，荷兰、英国、日本分别开始提出建设规划自动化集装箱码头这一概念，但由于资金投入的不足，日本方面的建设被搁置。1993年，荷兰的鹿特丹港口，建成第一个自动化集装箱码头。在充分总结了Delta Sealand码头建设和使用经验的基础上, ECT的Delta Dedicated East (DDE)和Delta Dedicated West (DDW)也分别于1997和2000年建成投产。与此同时，新加坡的Pasir Panjang码头在1997年建成远程操控的高架行车系统，堆场的半自动操作得以实现。1999年，香港在国际货柜码头实现堆场的半自动化作业。进入21世纪后的今天，依靠自动化技术的快速发展，自动化码头的数量不断增加。2002年10月，德国汉堡的CTA码头正式运行。 2008年，荷兰ECT码头投入试运营。多年以来，进行运营自动化码头的，不止有以上，还有HHLA-CTA码头、SETO-MSC码头、TTI码头、DPW 比利时泽布勒赫码头。

### 2.3 集装箱码头的配载

#### 2.3.1 定义

集装箱码头的配载，即把预先划定好装载出口的集装箱，制定每一个集装箱的装载计划，参考标准按照港口码头的业务要求和船舶运输过程中的适配需要。码头配载员需根据船公司的预配要求以及实际收箱状况来确定集装箱在船舶中具体的装载位置和装载顺序[7]。

##### a. 集装箱船舶资料内容

船舶箱位容量和箱位分布 船舶堆积负荷强度 船舶的尺寸和吃水情况 冷藏箱位置 危险箱的装载限制 空船重量大小和常数设置 稳性和吃水差计算书

##### b. 堆场集装箱资料

集装箱装箱单信息 装货单信息 特种箱清单信息 危险货物箱清单信息 危准单信息 预配船图船舶结构 集装箱的堆场位置明细

##### c. 船舶配载的基本原则

满足船舶的运输要求 (1)确保船舶拥有良好的稳性 (2)保证满足船舶适当的吃水差 (3)不超过船舶规定的强度规范 (4)考虑挂港作业同时合理配载 (5)满足特殊箱运输要求标准

符合码头的作业要求 (1)遵守堆场取箱作业规则 (2)遵守船舶装载作业计划要求

保证机械合理有序地移动

### 2.4 系统技术概念

#### 2.4.1 前端

利用 HTML，CSS，JavaScript，实现船舶配载的功能操作。HTML（HyperText Markup Language），它的中文名叫超文本标记语言， 它是一种可以用来动态化创建出网页页面的标记语言。后台服务器发送HTML文件，以供作为客户端的浏览器读取并解析，之后把该文件渲染生成相应的可视化网页。HTML 元素是构建网站的基础。开发者可以把图像，视频，音频等需要的对象嵌入其中，同时也可以利用其去构建交互式的表单信息，这样便于去结构化信息处理，具体例如列表，标题，段落等。

CSS(Cascading Style Sheets)，它的中文翻译为层叠样式表。它是一种可以为结构化的文档添加样式（包括颜色，边距，间距，字体等）的一门计算机语言。具体由World Wide Web 进行维护升级。

JS(Javascript)， 它通常写作缩写形式，JS， 它是比较高级的，解释性语言的编程语言。 这门语言基于原型（prototype）、函数先行（function），它也是一门多范式（multi Norm Forms）的语言。 可以支持命令式（Imperative）编程， 面向对象（object oriented）编程， 以及函数式(Functional)编程。 该编程语言通过语法来操控数组、字符串、浮点数、日期以及正则表达式等形式。 但不能够直接支持输入输出（I/O），具体如存储、网络和图形等操作， 但实际处理上可以通过该语言的宿主环境，来进行功能上的支持。与此同时，该语言已经被 欧洲计算机制造商协会（ECMA）通过ECMAScript实现语言的标准化。其中，它已经被主流浏览器（Safari、IE、Chrome、Opera、Firefox）支持。在客户端中，传统意义上，JavaScript被实现为一种解释语言。如今，它可以被即时编译。同时，最新的HTML5和CSS3语言标准让Js适应不同的应用程序的开发，以及不同的服务端网络运行环境。

关于三种语言之间的关系，HTML相当于人类的骨架，CSS相当于人类的肌肉部分，以及头发，服饰等外在视觉元素，JAVASCRIPT相当于人类的各种动作任务结合身体四肢五官的执行。

#### 2.4.2 后端

##### a. 后端框架

Django。Django 是一个Web应用框架, 这个框架通过Python这门语言进行编写，现在最新的版本是 2.2.1。该框架的设计模是模型层Model，视图层View和模板层Template三种概念的组合，即MVT的软件设计模式。该框架创立的主要目标是为了简化由数据库驱动的，复杂的网站的开发流程。其中，该框架特别注重以下原则：Don't Repeat Yourself（DRY），敏捷开发原则，组件的可以复用性，组件的可插拔。 该框架的组件部分主要包括路由分发、视图显示、模板渲染、连接数据库关系模型的中间。并且，该框架具有独立的Web服务器，这使得开发者便于进行开发和功能测试；处理表单序列化的系统以及验证防止CSRF的系统，这便于前端页面数据和后端数据库部分的数据转化；各种中间件的实现，开发者可以对一个http请求的每一个阶段做单独处理；独特的序列化系统，减轻了处理数据传输过程中，读取或者生成相关数据的代码压力。

##### b. 数据库

数据库MySQL。通常，数据库MySQL 特别擅长管理关系型数据库。而且在实际使用上，MySQL也具有很多优点，比如性能强大（支持高并发查询，支持数据库日志操作等）和成本低廉（该软件属于开源，可以很容易找到官网下载使用。丰富文档资料，降低了该产品的上手使用或者进一步深入学习的门槛）。MySQL 的管理工作。传统的方案，利用命令行工具操作MySQL, 亦可利用MySQL编辑器，即GUI工具Workbench进行数据库的管理操作。第三方集成管理工具：Navicat。Navicat支持连接到本地或远程数据库并进行可视化显示，支持query查询等基本数据库管理功能。 MySQL连接方式。 python有一套开源的连接MySQL的驱动程序：mysqlclient。

##### c. 关系模型

数据库设计，主要是数据结构层面。起初，层次模型被引入数据库系统。层次模型通过一种“有向树”的数据结构表示系统内各个实体以及各个实体之间的关系。之后，网状模型——用来表达无法用层次模型表达的非层次关系的模型的数据结构关系，在20世纪70年代由CODASYL提出。但这两种模型的诞生，具有很多不足，如对数据插入和删除的操作限制，DDL（数据定义语言）的过度复杂等，终于在1970年，IBM提出的基于实体--属性的ER模型，完善了上面的不足，关系模型的设立，是基于严格的数学关系推导。关系模型涉及关系（Relation），元组（Tuple），属性（Attribute），关键字（Key），域（Domain），分量，关系模式这些基本概念。具体数据库开发过程中，会涉及ER模型的建立，ER模型的转换， 完整性约束等[8]。

关系型数据库具有以下三类完整性，模型的用户定义的完整性，模型的参照的完整性（ReferentialIntegrity）以及模型实体的完整性（EntityIntegrity）。在完成关系模型的设计之后，通常还要考虑关系模式的设计是否符合相应的规范，包括数据冗余问题，属性间的联系是否合理，函数依赖的设定中，是否合理区分部分函数依赖和完全函数依赖，多值依赖是否要保留，以及是否要运用更高级别的范式理论。

#### 2.4.3 版本控制

##### a. 版本控制平台

平台GitHub。GitHub的前身，是通过Linux组织相关成员发起并建设。它具有充当远程工作仓库的功能。现如今，很多知名的开源项目，都可以在GitHub上找到并进行Star（标记追踪）或Fork(本地克隆该项目并进行分支开发)。

GitHub 通过git bash 操作，实现代码的版本控制服务。

##### b. 版本控制工具

控制工具git。git 是一个可以进行分布式版本控制代码的软件。起初是为了管理Linux内核代码的版本。

该工具采用了一套分布式存储的版本库。在进行版本控制的过程中，开发者不必单独使用一些其他的服务端软件，即可实现不同版本的稳定管理。

##### c. 版本控制结构

（1）git 库目录： （2）hooks：存储钩子 （3）logs： 存储日志 （4）refs：存储指向各分支的指针 （5）objects： 存储git对象 （6）config: 存储设置文档 （6）HEAD： 存储当前开发所在分支的文件路径指向

##### d. 版本控制用例

（1）git pull #连接远程代码库，把相关更新同步到本地代码库 （2）git status #当前项目对应远程分支代码更新状态 （3）git add . #把当前修改工作暂存 （4）git commit #输入当前修所要记录的基本信息，并保存 （5）git push #把当前commit后的修改，提交到远程分支

### 2.5 船舶配载规范

#### 2.5.1 集装箱船舶箱位命名规则

实际船舶配载过程中，为了能够准确描述集装箱在船舶上的具体位置，这里引入船舶箱位的概念。根据洋山四期深水港规划建设的标准，船舶上的每一个箱位，都由一个由六位字符组成的字符串表示，这六位字符串，前后分别包含了贝位（BAY），列数（ROW）,层数（TIER）。对于贝位，初始船舶预配载前只有奇数贝位，且奇数贝位有两位字符显示，小于10的奇数，需要添加字符“0”，组合成诸如“01”，“03”，“05”，“07”这种，大于10的奇数，即表示位相应的字符形式。对于列数，由于实际船舶配载过程中，列数有奇数列，也有偶数列。另外，由于箱子堆放问题，箱子 有时在“00线”两侧堆放，有时覆盖“00线”堆放，故层数设定，一侧以字符串“01”、“03”、“05”、“07”....依次表示，另一侧以“02”、“04”、“06”、“08”....依次显示。中间线位置有箱位覆盖时，即用“00”表示，否则，该字符串不表示任何箱位。层数表示的设定上，由于船舶装卸，分为甲板舱内和甲板舱外，从船身侧面观察，即舱盖板以上和舱盖板以下。具体的，甲板舱内层数，从下往上，不同层数依次是“02”，“04”，“06”呈递增形式的字符串表示，甲板舱外层数，从下往上，依次是“82”，“84”，“86”，“88”，同样呈递增形式。

#### 2.5.2 集装箱船舶装卸规则

一艘船舶在进行装卸作业前，首先要进行预配载信息设定。这里包括组贝设置、箱位信息设置等。组贝设置上，只有两个相邻的20尺贝位方可组合（组贝）变成40尺贝位。

岸桥调度过程中，要考虑船舶船舱位置，对于船舱位置的处理，通常有两种方式，一种是船舱占据一定数目的贝位，同时贝位分布是连续的；另一种是船舱不占据任何贝位，同时贝位分布是不连续的。具体装卸顺序上，考虑到整体船舶的装船和卸船，通常是先卸船，再装船，具体顺序是，甲板舱外卸箱——甲板舱内卸箱——甲板舱内装箱——甲板舱外装箱。实际操作中，如装箱的时候，会涉及到倒箱，即目标箱位当前存放的集装箱不满足当前最新的调度装卸方案而进行的集装箱之间的箱位变换。岸桥在进行装卸作业过程中，也要遵循不可跨越当前作业区的原则，最大程度避免了岸桥与岸桥之间的作业冲突。

## 第三章、 实体分析

### 3.1 船舶实体分析

集装箱船舶的属性包括了航次信息、船舶结构信息、积载信息。

（1）船舶航次信息：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | Vessel | 船名 | String |
| 2 | ImpVoy | 进口航次 | String |
| 3 | ExpVoy | 出口航次 | String |
| 4 | VesType | 船型 | String |
| 5 | PlaBerThgTim | 计划靠泊 | datetime |
| 6 | PlaUnbThgTim | 计划离泊 | datetime |
| 7 | ReaBerThgTim | 实际靠泊（时间） | datetime |
| 8 | ActUnbTim | 实际离泊 | datetime |
| 9 | PlaBerThgPos | 计划靠泊位置 | String |
| 10 | ActBerPos | 实际靠泊位置 | numeric |
| 11 | BerThgDir | 靠泊方向 | String |
| 12 | VesDrauMax | 进出最大吃水 | Real |
| 13 | PlaClosCustTim | 计划截关时间 | datetime |
| 14 | ClosCustTim | 截关时间 | datetime |
| 15 | OpAssSign | 已安排作业 | String |
| 16 | EntPlanMakSig | 已安排集港 | String |
| 17 | TaskFiniSig | 作业完成否 | String |
| 18 | UnBSta | 离泊情况 | boolean |
| 19 | PlaLoaGPCtnFotNum | 计划进重40 | Integer |
| 20 | PlaLoaEmpCtnFotNum | 计划进空40 | Integer |
| 21 | PlaLoaDraCtnFotNum | 计划进危40 | Integer |
| 22 | PlaLoaGPCtnFotFivNum | 计划进重45 | Integer |
| 23 | PlaLoaEmpCtnFotFivNum | 计划进空45 | Integer |
| 24 | PlaLoaDraCtnFotFivNum | 计划进危45 | Integer |
| 25 | PlaUloGPCtnFotNum | 计划出重40 | Integer |
| 26 | PlaUloEmpCtnFotNum | 计划出空40 | Integer |
| 27 | PlaUloDraCtnFotNum | 计划出危40 | Integer |
| 28 | PlaUloGPCtnFotFivNum | 计划出重45 | Integer |
| 29 | PlaUloEmpCtnFotFivNum | 计划出空45 | Integer |
| 30 | PlaUloDraCtnFotFivNum | 计划出危45 | Integer |

备注：背景色为黄色，表示为关键字（主键）。 备注：“作业完成情况”：表示装卸船情况，1表示正在卸船（未装船），2表示正在装船（未卸船）、3表示正在装卸船(两者同时开展)，4表示卸船完成，正在装船；5表示装卸船都完成； 备注：“离泊情况”：1表示装卸船已经完成，正在离泊准备；2表示正在离泊作业，3表示离泊完成； 备注：进出港最大吃水：是船舶进港和出港时，船舶的最大吃水深度，是为了制订船舶靠泊计划用。

（2）船舶结构信息

表4-1-2船舶结构（针对某一船舶型号）：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | Vessel | 船名 | String |
| 2 | ImpVoy | 进口航次 | String |
| 3 | ExpVoy | 出口航次 | String |
| 4 | VesType | 船型 | String |
| 5 | PlaBerThgTim | 计划靠泊 | datetime |
| 6 | PlaUnbThgTim | 计划离泊 | datetime |
| 7 | ReaBerThgTim | 实际靠泊（时间） | datetime |
| 8 | ActUnbTim | 实际离泊 | datetime |
| 9 | PlaBerThgPos | 计划靠泊位置 | String |
| 10 | ActBerPos | 实际靠泊位置 | numeric |
| 11 | BerThgDir | 靠泊方向 | String |
| 12 | VesDrauMax | 进出最大吃水 | Real |
| 13 | PlaClosCustTim | 计划截关时间 | datetime |
| 14 | ClosCustTim | 截关时间 | datetime |
| 15 | OpAssSign | 已安排作业 | String |
| 16 | EntPlanMakSig | 已安排集港 | String |
| 17 | TaskFiniSig | 作业完成否 | String |
| 18 | UnBSta | 离泊情况 | boolean |
| 19 | PlaLoaGPCtnFotNum | 计划进重40 | Integer |
| 20 | PlaLoaEmpCtnFotNum | 计划进空40 | Integer |
| 21 | PlaLoaDraCtnFotNum | 计划进危40 | Integer |
| 22 | PlaLoaGPCtnFotFivNum | 计划进重45 | Integer |
| 23 | PlaLoaEmpCtnFotFivNum | 计划进空45 | Integer |
| 24 | PlaLoaDraCtnFotFivNum | 计划进危45 | Integer |
| 25 | PlaUloGPCtnFotNum | 计划出重40 | Integer |
| 26 | PlaUloEmpCtnFotNum | 计划出空40 | Integer |
| 27 | PlaUloDraCtnFotNum | 计划出危40 | Integer |
| 28 | PlaUloGPCtnFotFivNum | 计划出重45 | Integer |
| 29 | PlaUloEmpCtnFotFivNum | 计划出空45 | Integer |
| 30 | PlaUloDraCtnFotFivNum | 计划出危45 | Integer |

备注：“40组贝”是一个字符串，长度为20尺贝位数，从船首开始往后计组贝情况，如第一个组贝为第2和3号20尺贝位，那第2、3位置标注为1，第4、5号20尺贝位为第二个40尺组贝，那在第4、5位置标注上2，依次类推。 备注：“中间贝处理”是为了确定船舶中间贝是00行，还是最中间就是一条线，左右两边对称放箱。

（3）船舶贝位结构信息

表4-1-3船舶贝位结构

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | VesType | 船型 | String |
| 2 | BayNo | 贝号 | String |
| 3 | BaySiz | 贝尺寸 | String |
| 4 | BayCom | 贝组 | String |
| 5 | DeckHeg | 甲板层高 | Integer |
| 6 | DeckWidMax | 甲板最宽 | Integer |
| 7 | CabHeg | 舱内层高 | Integer |
| 8 | CabWidMax | 舱内最宽 | Integer |
| 9 | HatCovKind | 舱盖板类型 | String |
| 10 | HatCovSiz | 边舱盖板宽 | Integer |
| 11 | BayX | 贝X坐标 | Integer |
| 12 | BayY | 贝Y坐标 | Integer |

备注：“舱盖板类型”，0表示是自开式，1~4表示舱盖板数量；“边舱盖板宽”：这里假定舱盖板是左右对称的，结合“舱盖板类型”，可以确定具体尺寸，边舱盖板是最边上的舱盖板，宽度就是该舱盖板的宽度。 备注：贝尺寸：是20尺还是40尺，是否可以存放45尺。 备注：贝组：只有20尺贝需要组贝，0表示不组贝，1表示向右组贝，2表示向左组贝。 备注：贝X坐标和贝Y坐标：表示该贝位靠近船头一侧的面和船中心线的交点所在的坐标值。 备注：甲板最宽：表示船舶甲板上堆放集装箱的排数。

（4） 船舶贝位层结构信息

表4-1-4船舶贝位层结构

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | VesType | 船型 | String |
| 2 | BayNo | 贝号 | String |
| 3 | TireNo | 层号 | String |
| 4 | DeckCagSig | 甲板or舱内 | boolean |
| 5 | BayWid | 宽度 | Integer |
| 6 | BayHigh | 离底层 | Integer |
| 7 | CellSpeSig | 箱位特标 | String |
| 8 | RefSig | 箱位冷冻标 | String |

备注：“离底层”表示该层离开舱底和甲板的层数。 备注：上述三张表应该可以完整表示船舶结构。

（5）船舶积载（进口箱的积载）信息

表4-1-5船舶积载（进口箱的积载）：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | Vessel | 船名 | String |
| 2 | Voyage | 航次 | String |
| 3 | VesCellNo | 船箱位 | String |
| 4 | ColNo | 列号 | tinyint |
| 5 | DeckCagSig | 甲板or舱内 | String |
| 6 | CtnNo | 箱号 | String |
| 7 | CtnTyp | 箱型 | String |
| 8 | Size | 尺寸 | String |
| 9 | Status | 状态 | String |
| 10 | CtnWegt | 箱重 | Real |
| 11 | Owner | 持箱人 | String |
| 12 | GForce | 货特要求 | String |
| 13 | StaPort | 起运港 | String |
| 14 | UnloadPort | 卸货港 | String |
| 15 | DesPort | 目的港 | String |
| 16 | StrPickAwayCtn | 直提箱 | String |
| 17 | CtnOpeSta | 箱作业状态 | String |
| 18 | PlaCtnCel | 计划箱位 | String |

（6）出口箱配载计划信息

表4-1-6出口箱配载计划：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | Vessel | 船名 | String |
| 2 | Voyage | 航次 | String |
| 3 | VesCellNo | 船箱位 | String |
| 4 | ColNo | 列号 | tinyint |
| 5 | DeckCagSig | 甲板or舱内 | String |
| 6 | CtnNo | 箱号 | String |
| 7 | CtnTyp | 箱型 | String |
| 8 | Size | 尺寸 | String |
| 9 | Status | 状态 | String |
| 10 | CtnWegt | 箱重 | Real |
| 11 | GForce | 货特要求 | String |
| 12 | StaPort | 起运港 | String |
| 13 | UnloadPort | 卸货港 | String |
| 14 | DesPort | 目的港 | String |
| 15 | CtnOpeSta | 箱作业状态 | String |
| 16 | YardCel | 堆场箱位 | String |

### 3.2 岸桥实体分析

岸桥的设备类型分很多种，主要有单小车、双小车、三小车岸吊，同时吊具有双吊和双四十尺吊，所以总共有六种岸桥，需要对岸桥建立统一的类型分析。

岸桥的实体信息包括：岸桥的结构信息，岸桥的当前状态信息，岸桥的作业计划信息，岸桥的作业统计信息。

岸桥的结构信息主要用来描述岸桥的类型，岸桥的作业效率，岸桥大车的移动速度等信息。

岸桥信息主要用来确定岸桥的当前状态信息。

（1）岸桥信息

表4-2-1 岸桥信息

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | QCNo | 岸桥编号 | String |
| 2 | CellSpeSig | 类别 | String |
| 3 | QcPos | 岸桥位置 | Real |
| 4 | QCSta | 岸桥状态 | String |
| 5 | OpVes | 作业船舶 | String |
| 6 | BegBayPoi | 起贝定位 | Real |
| 7 | OpBerthPos | 当前作业岸边位置 | String |
| 8 | NowOpVesCel | 当前作业船箱位 | String |
| 9 | OpCtnNo | 当前作业箱号 | String |
| 10 | OpAgvFir | 当前作业AGV一 | String |
| 11 | OpAgvSec | 当前作业AGV二 | String |
| 12 | OpAgvThr | 当前作业AGV三 | String |
| 13 | BetLegTraBlk | 故障胯下阻断 | String |
| 14 | OpAgvFou | 当前作业AGV四 | String |
| 15 | QcNoLeft | 左侧岸桥编号 | String |
| 16 | QcNoRight | 右侧岸桥编号 | String |
| 17 | VesOPTimFir | 船上第一时间 | datetime |
| 18 | VesOPTimSec | 船上第二时间 | datetime |
| 19 | AgvTimFir | AGV第一时间 | datetime |
| 20 | AgvTimSec | AGV第二时间 | datetime |
| 21 | PlfOpTimFir | 平台第一时间 | datetime |
| 22 | PlfOpTimSec | 平台第二时间 | datetime |
| 23 | PlfOpTimThr | 平台第三时间 | datetime |
| 24 | PlfOpTimFou | 平台第四时间 | datetime |
| 25 | PlaQcMtnFiniTim | 岸吊预期修复时间 | datetime |
| 26 | FauTailBreak | 故障尾部阻断 | String |

备注：岸桥状态：0、空闲；1、卸船；2、装船；3、大车移动；4、故障；5、起大梁；6、起大梁大车移动；

备注：起贝定位：是指岸吊小车到船上作业时，最靠近岸边的第一贝的中心离开泊位垂直面的距离；

备注：第一时间是吊具着箱或放箱落地时间，第二时间是吊具起吊时间；平台是考虑到岸桥可能是多小车，往往上面有中转平台，所以设定该参数；平台第三时间是在平台上的第二吊箱的着箱或放箱落地时间，平台第四时间是第二次吊具起吊时间。

备注：当前作业岸边位置，是指AGV所在的车道，当前作业AGV是该集装箱要指派给的AGV编号。

备注：岸桥预期修复时间：是用来确定岸桥故障后，最大可能的修复时间，该数据项可能影响到船舶的靠泊，AGV的作业等情况。

备注：故障胯下阻断：是为了后续AGV的路径，或者外部集卡的行走路径。胯下是指岸桥轨道中间的空间，数据项类型为char，长度为车道数量，比如车道为5，那char的长度为5位，每一位用0表示畅通，1表示阻断。

备注：故障尾部阻断：同故障胯下阻断，长度也是后续的车道数。

（2）岸桥结构类别信息

岸桥结构类别信息主要是用来为了后续画出岸桥的图示，这些信息保存在表中，在落实到具体岸桥后，可以把这些信息直接写入到岸桥的实体中。

表4-2-2 岸桥结构类别信息

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | Qckind | 类别 | String |
| 2 | QcLengFr | 外伸距 | Real |
| 3 | QcOpLengFr | 外伸作业距 | Real |
| 4 | QcHeg | 岸桥高度 | Real |
| 5 | OpHeg | 作业高度 | Real |
| 6 | QcLengTr | 后伸距 | Real |
| 7 | OpLengTr | 后伸作业距 | Real |
| 8 | PlfPos | 平台位置 | Real |
| 9 | BetLegLeng | 大车轨道跨距 | Real |
| 10 | BetLegLanNum | 大车跨下车道数 | Integer |
| 11 | LifSpdDolFr | 前置小车起升速度 | Object |
| 12 | DowVelDolFro | 前置小车下降速度 | Object |
| 13 | HorSpdDolFr | 前置小车水平速度 | Object |
| 14 | OpStyDolFro | 前置小车作业方式 | Integer |
| 15 | LifSpdDolTr | 后置小车起升速度 | Object |
| 16 | DowVelDolTr | 后置小车下降速度 | Object |
| 17 | HorSpdDolTr | 后置小车水平速度 | Object |
| 18 | OpKindDolTr | 后置小车作业方式 | Integer |

备注：类别：1、单小车双吊；2、单小车双四十次吊；3、双小车双吊；4、双小车双四十尺吊；5、三小车双吊；6、三小车双四十尺吊；

备注：外伸距：从泊位垂直面到最前面的距离；后伸距，从内侧轨道到最后距离；

备注：外伸作业距：从泊位垂直到最外侧可以作业的距离；后伸作业距：从内侧轨道到最后可以作业的距离；

备注：前置小车作业方式：是门字形作业还是抛物线形作业；

备注：小车\*\*速度是分布函数，主要体现该活动服从什么分布。

（3）岸桥作业计划信息

根据上述信息能确定岸桥的外形动画以及已经当前作业状态，另外还需要为每一台岸桥做装卸任务列表。首先是岸桥的作业计划，该计划说明了每一台岸桥需要作业的贝位及其先后关系。然后是每一个贝位内的每一个具体作业任务。

表4-2-3 岸桥作业计划（该计划是岸桥调度模型的输出结果）

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | QCNo | 岸桥编号 | String |
| 2 | OpVes | 作业船舶 | String |
| 3 | OpBay | 作业贝位 | String |
| 4 | OpKind | 作业类别 | String |
| 5 | OpPos | 作业位置 | Real |
| 6 | PreOpVes | 紧前作业船舶 | String |
| 7 | PreOpBay | 紧前作业贝位 | String |
| 8 | PreOpPos | 作业紧前位置 | String |
| 9 | PreOpKind | 作业紧前类别 | String |

备注：作业紧前贝位：就是岸桥在当前作业贝位紧前的作业贝位，其需要与作业紧前船舶联合才能确定；

备注：作业类别（同“作业紧前类别”）：1表示甲板卸；2表示卸舱盖板；3表示舱内卸；4表示舱内装；5表示装舱盖板；6表示甲板装。

（4）岸桥作业贝位计划信息

表4-2-4 岸桥作业贝位计划（是在岸桥作业计划的基础上，为岸桥分配集装箱）

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 英文名称 | 中文名称 | 类型 |
| 1 | QCNo | 岸桥编号 | String |
| 2 | OpSta | 作业状态 | String |
| 3 | OpVes | 作业船舶 | String |
| 4 | OpBay | 作业贝位 | String |
| 5 | OpKind | 作业类别 | String |
| 6 | NowOpVesCelFir | 当前作业船箱位一 | String |
| 7 | CtnNoFir | 箱号一 | String |
| 8 | CtnSizFir | 箱一尺寸 | String |
| 9 | BerPosFir | 岸边位一 | String |
| 10 | NowOpVesCelSec | 当前作业船箱位二 | String |
| 11 | CtnNoSec | 箱号二 | String |
| 12 | CtnSizSec | 箱二尺寸 | String |
| 13 | BerPosSec | 岸边位二 | String |
| 14 | NowOpVesCelThr | 当前作业船箱位三 | String |
| 15 | CtnNoThr | 箱号三 | String |
| 16 | CtnSizThr | 箱三尺寸 | String |
| 17 | BerPosThr | 岸边位三 | String |
| 18 | NowOpVesCelFou | 当前作业船箱位四 | String |
| 19 | CtnNoFou | 箱号四 | String |
| 20 | CtnSizFou | 箱四尺寸 | String |
| 21 | BerPosFou | 岸边位四 | String |
| 22 | PreVesCelFir | 紧前作业船箱位一 | String |
| 23 | PreVesCelSec | 紧前作业船箱位二 | String |
| 24 | PreVesCelThr | 紧前作业船箱位三 | String |
| 25 | PreVesCelFou | 紧前作业船箱位四 | String |
| 26 | PreOpVes | 紧前作业船舶 | String |
| 27 | PreOpBay | 紧前作业贝位 | String |
| 28 | NowOpBay | 当前作业贝 | String |
| 29 | DesBay | 目标贝位 | String |

备注：这里的箱号从一到四，是考虑到双四十尺吊具。

备注：作业状态：见表4-2-7

备注：作业类别：根据表4-2-7,4-2-8,4-2-9来确定，卸船一类该数据项为11，卸船二类为12，第一个数表示作业大类，比如卸船就为1，第二数表示具体明细类。作业类别A：表示大车移动。

备注：当前作业贝，目标贝位：表示岸吊大车移动的指令。

## 第四章、 系统规划

### 4.1 船舶结构

船舶结构具体包含以下：

### 4.2 船舶配载过程

码头配载依据——预配船图： 预配是指由船公司确定的各港口集装箱在船舶上的装载区域。

#### a. 配载原则和作业流程

（1）收集并核对配载单证资料 （2）制作配载船图 （3）配载图的审核 （4）配载图的签发 （5）退关箱复关的处理

#### b. 配载的数据准备

（1）制作船舶规范 （2）确定航次挂靠港 （3）场站收据放关确认 （4）退关箱、复关箱处理 （5）出口箱整船换装

#### c. 制作船舶规范

（1）船舶轮廓尺寸参数 （2）倍内单元设定 （3）特殊箱位设定 （4）组倍设定

#### d. 船舶配载的一般步骤

（1）分类 （2）索箱 （3）选倍 （4）划块 （5）配箱

#### e. 配载应遵循的原则

满足船舶的运输稳性要求： （1）保证船舶良好的稳性 （2）保持船舶适当的吃水差 （3）满足船体强度要求 （4）避免配载不当造成沿线挂港作业困难 （5）满足特种箱的配载要求

符合码头的作业要求： （1）符合堆场取箱规则 （2）符合单船作业计划要求 （3）确保机械合理、有序地移动

#### f. 配载的主要决策内容

实配的主要决策内容。根据船公司的预配要求以及实际收箱状况来确定集装箱在船舶中具体的装载位置和装载顺序。

#### g. 船舶适航要求

配载结果——出口船图——船公司配载确认——船舶适航要求

——大副确认

## 第五章、系统分析

### 5.1 船舶相关的信息分析

### 5.2 船舶配载业务功能调查

#### 5.2.1 背景介绍

配载是把要装船的集装箱按船舶的运输要求以及码头的作业要求而制定的具体装载计划，最终的呈现形式是把已经在堆场的每一个待装集装箱（那些直装的集装箱则是直接运到岸桥下，直接装船，但是也要做配载计划）计划到具体的船箱位。配载必须满足船舶的运输要求，即船舶的船行安全、集装箱及其货物的运输质量船舶营运的经济效益，同时也要兼顾集装箱码头装卸工艺和操作方式，使码头能合理、有效地组织生产。

船舶配载一般有预配船图，预配船图是由船公司提供，作为集装箱码头实配的依据，该信息主要包括：船上每一具体船箱位将要装箱的规格信息（尺寸、箱重、卸货港、箱型---该信息主要包括是GP箱还是其他例如框架箱之类的信息），一般情况下，只要符合这些箱重的要求，那么船舶的重心、航行的要求都能得到满足。

#### 5.2.1 规则制定

集装箱船舶预配是配载员最重要的配载依据，其制定过程需要满足预配中定义的放箱约束。预配限定的是某一类箱子应该放那些区域；实配限定的是某个箱子应该放在哪个位置上。如下图所示，是某个贝位的预配图。在预配图中定义了船上某个堆存位可以装载哪个卸货港的箱子，在此基础上还可以设定所需要的箱子的箱型、尺寸、是否超箱、是否是危险品、是否可放45ft等特殊属性。

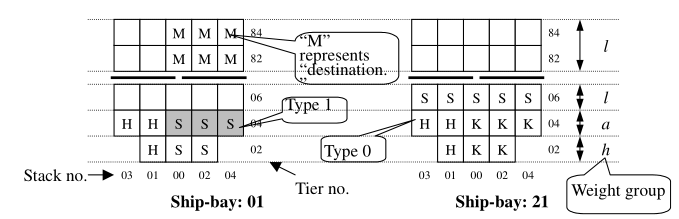


图1.船舶预配图（某一贝位）

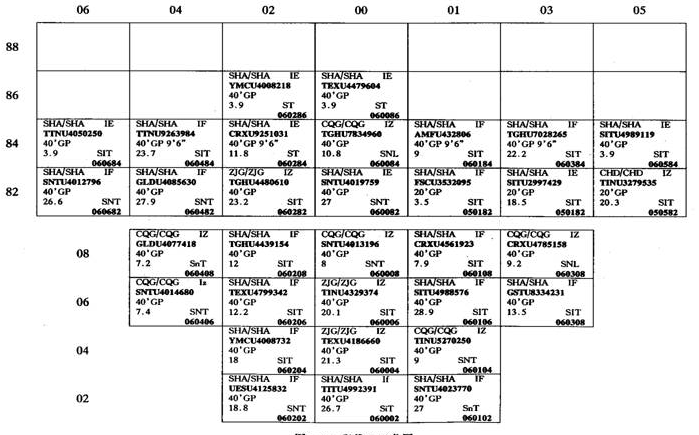


图2实配图（某一贝位）

配载的一般制作过程：配载制作过程一般分为五步进行，分别是分类、索箱、选倍、划块、配箱。1）分类：是指将对该航线出口箱按照箱子的卸货港、尺寸、箱型、箱重等属性来划分。2）索箱：是指配载员选择某类箱子，然后检索出该类箱子在场地里面的堆存位置，并且要查看和分析箱子分布在几个箱区以及在箱区的集中程度等场地状态信息，从而辅助配载员构思配载方案。3）选倍：选倍是指为刚刚索箱的这组集装箱选择想要配载的船舶倍位。4）划块：划块是指在已选倍内划出一个目标配载区域。5）配箱：即要场地内的各个区位上的某些箱子按照一定的配载顺序配载到刚刚划块作业所圈定的船箱位上。

配载遵循的原则：第一、满足预配船图的要求；第二、满足后续装船作业的有序开展（对于自动化集装箱码头来说，要满足这个要求，一般要包括：堆场的装船作业和其他能尽可能分开，堆场发箱顺序能够和后续的岸吊计划能匹配，堆场机械的作业能力能满足后续装船的需要）。

### 5.3 船舶配载业务流程调查

### 5.4 数据汇总

#### 5.4.1 ER模型

#### 5.4.2 数据项

#### 5.4.3 U/C 矩阵

### 5.5 数据流程图

### 5.6 数据字典

#### 5.6.1 数据结构

#### 5.6.2 数据流

#### 5.6.3 数据存储

#### 5.6.4 外部实体

#### 5.6.5 逻辑处理

## 第六章、 系统设计

### 6.1 功能结构图设计

#### 6.1.1 层次模块结构图

#### 6.1.2 H图

#### 6.1.3 IPO 图

### 6.2 系统流程图

### 6.3 数据存储设计

#### 6.3.1 关系模型

### 6.4 处理流程图

#### 6.4.1 处理流程--

### 6.5 程序设计

#### 6.5.1 配载程序设计

#### 6.5.2 输出设计

#### 6.5.3 输入设计

## 第七章、 结论和展望

## 参考文献

[1] 丁一,龚杰,林国龙.考虑箱区作业均衡的ACT船舶配载研究[J].计算机应用与软件,2018,35(09):7-15.

[2] 杨蔚,韩晓龙.集装箱班轮运输配载优化问题[J].大连海事大学学报,2018,44(01):72-79.

[3] 计明军,曲田.集装箱船舶配载策略的比较分析[J].航海技术,2012(01):28-32.

[4] 杜嘉立,杨盐生,郑云峰.基于专家系统的船舶配载智能控制策略[J].交通运输工程学报,2005(03):64-68.

[5] 维基百科.决策支持系统.https://zh.wikipedia.org/wiki/决策支持系统/.

[6] 薛华成.管理信息系统:第六版[M].清华大学出版社,2017:279.

[7] 智库百科.集装箱船舶配载.https://wiki.mbalib.com/wiki/集装箱船舶配载/.

[8] 周屹,李艳娟,崔琨,姜晓宏.数据库原理及开发应用:第二版[M].清华大学出版社,2013:29.

[9] 维基百科.遗传算法.https://zh.wikipedia.org/wiki/遗传算法/.

## 致谢

本次毕业的顺利完成，首先要感谢我的导师，是导师每周的耐心的指导与严格的督促，才使得毕业设计的工作正常的推进。

其次，我要感谢共同一起完成本次毕设的小组同学，这其中，由于一开始本人对船舶配载调度的具体业务较为生疏，经过孙同学的耐心讲解，我对整体业务逻辑的理解才更加清晰；在算法部分，刘同学比较熟悉数学模型的搭建以及算法模型的处理，在文章核心算法库的实现过程中，刘同学给予了很多帮助，包括参数处理，接口设置等问题； 同时，也要感谢刘同学在船舶配载功能实现过程中，堆场部分相关概念与功能的支持。

另外，本次毕设编码部分的实现，还要感谢本科阶段很多老师的教学以及身边同学、学长的帮助。前端部分，要感谢石学长，大概两年前，帮助我引入前端的概念；另外，还要感谢我的导师在管理信息系统这门课上对相关概念的引入，督促我进一步深入学习。后端部分 ，首先要感谢赵老师，在数据库方面的耐心教学；另外也要感谢沈同学在课程设计阶段的帮助，帮助我进一步认识并掌握后端开发的思想与方法。算法模型部分，要感谢本科阶段的胡老师，让我从此认识了运筹模型的概念，也要感谢本科阶段重视调度规划设计教学的添老师，让我对运筹模型的认识由理论上升到了实践，同时，也要感谢吴同学，本科阶段带我参加数学建模竞赛。

还有很多同学或者老师或者工作上的同事，对本次毕业设计的完成，有着或多或少的帮助，这里无法一一致谢，这里只好统一表达感谢。

## 附录

**代码仓库**

https://github.com/ZhaoYitong/LE-FINAL-DESIGN/

**文件结构**

LE-FINAL-DESIGN  
 ├─.idea  
 ├─document  
 ├─server  
 └─web  
 ├─css  
 │ └─tool  
 │ └─fontawesome-free  
 │ ├─css  
 │ ├─sprites  
 │ ├─svgs  
 │ │ ├─brands  
 │ │ ├─regular  
 │ │ └─solid  
 │ └─webfonts  
 ├─images  
 ├─js  
 │ └─tool  
 │ ├─bootstrap  
 │ │ └─js  
 │ ├─jquery  
 │ └─jquery-easing  
 └─template  
 └─subTemplate  
 ├─loadOrUnload  
 │ ├─loadView  
 │ └─unloadView  
 ├─vessel  
 │ ├─vesselSheet  
 │ └─vesselView  
 └─yard  
 ├─yardSheet  
 └─yardView

### 代码部分

#### HTML

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="utf-8">  
 <meta http-equiv="X-UA-Compatible" content="IE=edge">  
 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">  
 <link href="../../../../css/tool/jquery-ui.css">  
 <link href="../../../../css/tool/fontawesome-free/css/all.min.css" rel="stylesheet" type="text/css">  
 <link href="../../../../css/tool/google-api.css" rel="stylesheet">  
 <link href="../../../../css/main.css" rel="stylesheet">  
 <link href="../../../../css/custom.vessel.css" rel="stylesheet">  
</head>  
<body id="page-top">  
<div id="IMO-001" class="vessel">  
 <div id="control-buttons">  
 <div id="buttons" class="d-flex flex-row">  
 <button type="button" class="btn btn-outline-primary startCombine" onclick="combineToStart()">编辑组贝</button>  
 <button type="button" class="btn btn-outline-primary confirmCombine" disabled onclick="combineToConfirm()">确认组贝</button>  
 <button type="button" class="btn btn-outline-primary reStartCombine" onclick="combineReset()">重新组贝</button>  
 <button type="button" class="btn btn-outline-success createVessel" onclick="createVesselSide()">生成船舶</button>  
 <button type="button" class="btn btn-outline-success createStowage" onclick="createStowageInfo()">配载信息</button>  
 <button type="button" class="btn btn-outline-success createLoadOrUnload" onclick="createLoadOrUnloadInfo()">待装卸信息</button>  
 </div>  
 </div>  
 <div class="zoomToggle d-flex flex-row">  
 <input id="test" min="1" max="10" value='10' step="1" onchange="showVal(this.value)" type="range"/>  
 </div>  
 <div class="mainArea">  
 <div class="bayArea"></div>  
 <div class="newBayArea"></div>  
 <div class="vesselAreaSide">  
 </div>  
 <div class="vesOperationInfo">  
 <div class="above">  
 <div class="operationInch40Num">  
  
 </div>  
 <div class="unload">  
 <div class="aboveUnloadInch40"></div>  
 <div class="aboveUnloadInch20"></div>  
 </div>  
 <div class="load">  
 <div class="aboveLoadInch40"></div>  
 <div class="aboveLoadInch20"></div>  
 </div>  
 <div class="operationInch20Num">  
  
 </div>  
 </div>  
 <div class="board"> </div>  
 <div class="below">  
 <div class="unload">  
 <div class="belowUnloadInch40"></div>  
 <div class="belowUnloadInch20"></div>  
 </div>  
 <div class="load">  
 <div class="belowLoadInch40"></div>  
 <div class="belowLoadInch20"></div>  
 </div>  
 </div>  
 </div>  
 <div class="vesselStowageInfo">  
 </div>  
 </div>  
</div>  
<script src="../../../../js/tool/jquery.js"></script>  
<script src="../../../../js/tool/bootstrap/js/bootstrap.bundle.min.js"></script>  
<script src="../../../../js/tool/jquery-easing/jquery.easing.min.js"></script>  
<script src="../../../../js/tool/jquery-ui.js"></script>  
<script src="../../../../js/main.min.js"></script>  
<script src="../../../../js/vessel.js"></script>  
</body>  
</html>

#### CSS

<style>  
 body {  
 margin:0;  
 padding:0;  
}  
  
#control-buttons {  
 margin-top: 20px;  
 margin-bottom: 40px;  
}  
  
#control-buttons > div > button {  
 margin: 5px;  
}  
  
.zoomToggle {  
 margin-top:10px;  
 margin-bottom: 40px;  
}  
  
.mainArea {  
 width: 100%;  
}  
  
.bayArea .bayZone\_inch20 {  
 background:#2f96b4;  
 width:38px;  
 height:30px;  
 float:left;  
 margin:2px;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
}  
/\* combined bay area \*/  
.bayArea .bayZone\_inch40 {  
 width:38px;  
 height:30px;  
 float:left;  
 background-color:lightgrey;  
 margin:2px;  
 position: relative;  
}  
  
.newBayArea .newBay20 {  
 background:#2f96b4;  
 width: 38px;  
 height: 60px;  
 float:left;  
 margin:2px;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
}  
  
.newBayArea .comBay20\_40 {  
 width: 80px;  
 height:60px;  
 float: left;  
 margin: 2px;  
}  
  
.newBayArea .comBay20\_40 .newBay40InCom {  
 width: 80px;  
 height: 30px;  
 float:left;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
 background-color: yellow;  
}  
  
.newBayArea .comBay20\_40 .newBay20InComParent {  
 width: 80px;  
 height: 30px;  
 display:flex;  
 background:#2f96b4;  
}  
  
.newBayArea .comBay20\_40 .newBay20InComParent .newBay20InComLeft {  
 width: 40px;  
 height: 30px;  
 float:left;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
 border-right: 1px solid silver;  
}  
  
.newBayArea .comBay20\_40 .newBay20InComParent .newBay20InComRight {  
 width: 40px;  
 height: 30px;  
 float:left;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
 border-left: 1px solid silver;  
}  
  
.vesOperationInfo {  
 margin-top: 30px;  
}  
  
.vesOperationInfo .above {  
 margin-bottom: 20px;  
}  
  
.unload > div,  
.load > div {  
 display: flex;  
}  
  
.unloadInch20 {  
 background:#dd569e;  
 width: 38px;  
 height: 20px;  
 float:left;  
 margin-left: 2px;  
 margin-right: 2px;  
 border: 1px solid black;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
}  
  
.loadInch20 {  
 background:#373db2;  
 width: 38px;  
 height: 20px;  
 float:left;  
 margin-left: 2px;  
 margin-right: 2px;  
 border: 1px solid black;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
}  
  
.unloadInch40 {  
 background:#dd569e;  
 width: 80px;  
 height: 20px;  
 float: left;  
 margin-left: 2px;  
 margin-right: 2px;  
 border: 1px solid black;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
}  
  
.loadInch40 {  
 background:#373db2;  
 width: 80px;  
 height: 20px;  
 float: left;  
 margin-left: 2px;  
 margin-right: 2px;  
 border: 1px solid black;  
 display:flex;  
 justify-content:center;  
 align-items:center;  
}  
  
.vesselBodyInLoading {  
 background:black;  
}  
/\* vesselOperation \*/  
/\* selected bay control when selecting \*/  
.bayArea\_40 .leftBaySelected {  
 border-left:5px solid red;  
 border-bottom: 3px solid red;  
 border-top: 3px solid red;  
}  
  
.bayArea\_40 .rightBaySelected {  
 border-right:5px solid red;  
 border-bottom: 3px solid red;  
 border-top: 3px solid red;  
}  
  
/\* selected bay control when selecting \*/  
.bayArea .bayArea\_20,  
.bayArea .bayArea\_40,  
.mainArea .newBayArea {  
 display: inline-flex;  
}  
  
.mainArea .vesselAreaSide {  
 margin-top: 25px;  
}  
  
.conZoneBayAbove\_inch20 {  
 width:38px;  
 height:80px;  
 float: left;  
 background-color:#d0ddf2;  
 margin:2px;  
}  
  
.boardSide\_inch20 {  
 width:42px;  
 height:10px;  
 float: left;  
 background-color:#775614;  
}  
  
.conZoneBayBelow\_inch20 {  
 width:38px;  
 height:50px;  
 float:left;  
 background-color:#d0ddf2;  
 margin:2px;  
}  
  
.conZoneBayLayerAbove\_inch20,  
.conZoneBayLayerBelow\_inch20 {  
 width:38px;  
 height:10px;  
 border-bottom: 1px solid white;  
 border-top: 1px solid white;  
 border-left: 1px solid white;  
 border-right: 1px solid white;  
 background-color: white;  
}  
  
.vesselBody\_inch20 {  
 width:38px;  
 height:10px;  
 border-bottom: 1px solid black;  
 border-top: 1px solid black;  
 border-left: 1px solid black;  
 border-right: 1px solid black;  
 background-color: black;  
}  
  
.vesselConZone\_inch20 {  
 width:38px;  
 height:10px;  
 border-bottom: 1px solid black;  
 border-top: 1px solid black;  
 border-left: 1px solid black;  
 border-right: 1px solid black;  
 background-color: white;  
}  
  
.onBoardSide,  
.belowBoardSide{  
 width: 100%;  
 display: inline-flex;  
}  
/\* jquery ui-select \*/  
#feedback {  
 font-size: 1.4em;  
}  
  
#selectable .ui-selecting {  
 background: #FECA40;  
}  
  
#selectable .ui-selected {  
 background: #F39814;  
 color: white;  
}  
/\* jquery ui-select \*/  
/\* character in span \*/  
.bay20Index,  
.newBay20Index,  
.newBay20IndexInCom,  
.loadOrUnloadNum {  
 color: #cdd8f6;  
}  
/\* character in span \*/  
.newBay40IndexInCom,  
.vesselBodyBayIndex{  
 color: black;  
}  
  
.leftBaySelected .bay40Index {  
 float: right;  
 margin-right: 2px;  
}  
  
.rightBaySelected .bay40Index {  
 float: left;  
 margin-left: 2px;  
}  
/\* custom \*/  
@-webkit-keyframes blinker {  
 from {opacity: 1.0;}  
 to {opacity: 0.0;}  
}  
/\*.blink{\*/  
/\* text-decoration: blink;\*/  
/\* -webkit-animation-name: blinker;\*/  
/\* -webkit-animation-duration: 0.6s;\*/  
/\* -webkit-animation-iteration-count:infinite;\*/  
/\* -webkit-animation-timing-function:ease-in-out;\*/  
/\* -webkit-animation-direction: alternate;\*/  
/\*}\*/

#### JavaScript

function numToIdString(num) {  
 return num < 10 ? "0" + num.toString() : num.toString();  
}  
function isExist(array,value){  
 for(let i=0;i<array.length;i++){  
 if(array[i] === value){  
 return true;  
 }  
 }  
 return false;  
}  
function toAbsent(value) {  
 return value>0?value:-value;  
}  
/\*\*  
 \* initialize bay area  
 \*/  
function initAreaForInline() {  
 $(`.bayArea`).append(`<div class="bayArea\_40"></div>`);  
 $(`.bayArea`).append(`<div id="selectable" class="bayArea\_20"></div>`);  
}  
function BayNumToRealIndexList(bayNum) {  
 let bay = {};  
 bay.inch20 = [];  
 for (let i=0;i<bayNum;i++){  
 bay.inch20[i] = {  
 "id":i+1,  
 "bayRealIndex":numToIdString(i\*2+1)  
 };  
 }  
 return bay;  
}  
// layer above  
// layer below  
function layerNumToRealIndexList(layerNumAbove,layerNumBelow) {  
 let layer = {};  
 layer.above = [];  
 layer.below = [];  
 for (let i=0; i<layerNumAbove; i++){  
 layer.above[i] = {  
 id: i+1,  
 layerRealIndex: numToIdString((i+41)\*2),  
 }  
 }  
 for (let j=0; j<layerNumBelow; j++){  
 layer.below[j] = {  
 id: j+1,  
 layerRealIndex: numToIdString((j+1)\*2),  
 }  
 }  
 return layer;  
}  
function insertBay(bayLists){  
 let bay\_num = bayLists.inch20.length;  
 for(let i=bay\_num-1;i>=0;i--){  
 let bayIndex = bayLists.inch20[i].bayRealIndex;  
 let bayId = bayLists.inch20[i].id;  
 $('.bayArea\_40').append(`<div id= ${bayId} class="bayZone\_inch40"></div>`);  
 $('.bayArea\_20').append(`<div id= ${bayId} title=${bayIndex} class="bayZone\_inch20"><span class="bay20Index">${bayIndex}</span></div>`);  
 }  
}  
/\*\*  
 \* redraw bayArea after confirm  
 \*/  
function createBayAfterOperation(newList) {  
 let newBay\_num = newList.data.length;  
 let dataList = newList.data;  
 for(let i=newBay\_num-1;i>=0;i--) {  
 let itemId = dataList[i].id;  
 if(dataList[i].type == "single"){  
 $(`.newBayArea`).append(  
 `<div id= ${itemId} class="newBay20">`+  
 `<span class="newBay20Index">${dataList[i].bayInch20[0].index}</span>`+  
 `</div>`);  
 }  
 else {  
 $(`.newBayArea`).append(  
 `<div id= ${itemId} class="comBay20\_40">`+  
 `<div class="newBay40InCom"><span class="newBay40IndexInCom">${dataList[i].bayInch40[0].index}</span></div>`+  
 `<div class="newBay20InComParent">`+  
 `<div class="newBay20InComLeft"><span class="newBay20IndexInCom">${dataList[i].bayInch20s[1].index}</span></div>` +  
 `<div class="newBay20InComRight"><span class="newBay20IndexInCom">${dataList[i].bayInch20s[0].index}</span></div>` +  
 `</div>`+  
 `</div>`);  
 }  
 }  
}  
/\*\*  
 \* select bay  
 \*/  
function selectToInch40(leftBay,rightBay,comBayIndex){  
 $(`.bayArea\_40 div[id=${leftBay}]`).addClass("leftBaySelected");  
 $(`.bayArea\_40 div[id=${leftBay}]`).addClass("combined");  
 $(`.bayArea\_40 div[id=${rightBay}]`).addClass("rightBaySelected");  
 $(`.bayArea\_40 div[id=${rightBay}]`).addClass("combined");  
  
 $(`.bayArea\_40 div[id=${leftBay}]`).append(`<span class="bay40Index">${comBayIndex}</span>`);  
 $(`.bayArea\_40 div[id=${rightBay}]`).append(`<span class="bay40Index">${comBayIndex}</span>`);  
}  
function clearSelected(){  
 $(`.bayZone\_inch20.ui-selected`).children().removeClass("ui-selected");  
 $(`.bayZone\_inch20.ui-selected`).removeClass("ui-selected");  
}  
/\*\*  
 \* vessel creation  
 \*/  
function createVesselSide(){  
 $(`.vesselAreaSide`).append(`<div class="onBoardSide"></div>`);  
 $(`.vesselAreaSide`).append(`<div class="belowBoardSide"></div>`);  
 let bayLists = BayNumToRealIndexList(numOfBay);  
 let layerLists = layerNumToRealIndexList(layerNumAbove,layerNumBelow);  
 let conZone\_bay\_num = bayLists.inch20.length;  
 let conZone\_layerAbove\_num = layerLists.above.length;  
 let conZone\_layerBelow\_num = layerLists.below.length;  
 for(let i=conZone\_bay\_num-1;i>=0;i--){  
 let conZoneBayIndex = bayLists.inch20[i].bayRealIndex;  
 $(`.onBoardSide`).append(`<div point-x=${conZoneBayIndex} class="conZoneBayAbove\_inch20"></div>`);  
 $(`.belowBoardSide`).append(`<div point-x=${conZoneBayIndex} class="conZoneBayBelow\_inch20"></div>`);  
 }  
 for(let j=0;j<conZone\_bay\_num;j++){  
 let conZoneBayIndex = bayLists.inch20[j].bayRealIndex;  
 for(let k=conZone\_layerAbove\_num-1;k>=0;k--){  
 let conZoneLayerIndex = layerLists.above[k].layerRealIndex;  
 $(`.onBoardSide div[point-x=${conZoneBayIndex}]`).append(`<div class="conZoneBayLayerAbove\_inch20" pointx=${conZoneBayIndex} pointz=${conZoneLayerIndex}></div>`);  
 }  
 for(let m=conZone\_layerBelow\_num-1;m>=0;m--){  
 let conZoneLayerIndex = layerLists.below[m].layerRealIndex;  
 $(`.belowBoardSide div[point-x=${conZoneBayIndex}]`).append(`<div class="conZoneBayLayerBelow\_inch20" pointx=${conZoneBayIndex} pointz=${conZoneLayerIndex}></div>`);  
 }  
 }  
 for(let t=0;t<VIEW\_SIDE.vessel.length;t++){  
 for(let u=0;u<VIEW\_SIDE.vessel[t].bayIndexList.length;u++){  
 $(`[pointx=${VIEW\_SIDE.vessel[t].bayIndexList[u]}][pointz=${VIEW\_SIDE.vessel[t].layerIndex}]`).addClass("vesselBody\_inch20");  
 }  
 for(let v=0;v<VIEW\_SIDE.vessel[t].conZoneIndexList.length;v++){  
 $(`[pointx=${VIEW\_SIDE.vessel[t].conZoneIndexList[v]}][pointz=${VIEW\_SIDE.vessel[t].layerIndex}]`).addClass("vesselConZone\_inch20");  
 }  
 }  
 $(`.createVessel`)[0].disabled = true;  
}  
/\*\*  
 \* stowage info  
 \*/  
function createStowageInfo() {  
 $(`.createStowage`)[0].disabled = true;  
  
}  
function createLoadOrUnloadInfo() {  
 $(`.createLoadOrUnload`)[0].disabled = true;  
 let bayList = vesselOperationInfo.data.List;  
 let bayListNum = vesselOperationInfo.data.List.length;  
 for(let i=bayListNum-1;i>=0;i--){  
 if(bayList[i].type == "single"){  
 let bayIndex20 = bayList[i].bayInch20[0].index;  
 $(`div[class="aboveUnloadInch40"]`).append(  
 `<div bayIndex=${bayIndex20} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div>`);  
 $(`div[class="belowUnloadInch40"]`).append(  
 `<div bayIndex=${bayIndex20} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="aboveLoadInch40"]`).append(  
 `<div bayIndex=${bayIndex20} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowLoadInch40"]`).append(  
 `<div bayIndex=${bayIndex20} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 }  
 else {  
 let bayIndex40 = bayList[i].bayInch40[0].index;  
 $(`div[class="aboveUnloadInch40"]`).append(  
 `<div bayIndex=${bayIndex40} class="unloadInch40"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowUnloadInch40"]`).append(  
 `<div bayIndex=${bayIndex40} class="unloadInch40"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="aboveLoadInch40"]`).append(  
 `<div bayIndex=${bayIndex40} class="loadInch40"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowLoadInch40"]`).append(  
 `<div bayIndex=${bayIndex40} class="loadInch40"><span class="loadOrUnloadNum">0</span></div></div>`);  
 }  
 }  
 for(let j=bayListNum-1;j>=0;j--){  
 if(bayList[j].type == "single"){  
 let bayIndex20 = bayList[j].bayInch20[0].index;  
 $(`div[class="aboveUnloadInch20"]`).append(  
 `<div bayIndex=${bayIndex20} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowUnloadInch20"]`).append(  
 `<div bayIndex=${bayIndex20} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="aboveLoadInch20"]`).append(  
 `<div bayIndex=${bayIndex20} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowLoadInch20"]`).append(  
 `<div bayIndex=${bayIndex20} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 }  
 else {  
 let bayIndex20\_first = bayList[j].bayInch20s[1].index;  
 let bayIndex20\_second = bayList[j].bayInch20s[0].index;  
 $(`div[class="aboveUnloadInch20"]`).append(`<div bayIndex=${bayIndex20\_first} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="aboveUnloadInch20"]`).append(`<div bayIndex=${bayIndex20\_second} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowUnloadInch20"]`).append(`<div bayIndex=${bayIndex20\_first} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowUnloadInch20"]`).append(`<div bayIndex=${bayIndex20\_second} class="unloadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
  
 $(`div[class="aboveLoadInch20"]`).append(`<div bayIndex=${bayIndex20\_first} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="aboveLoadInch20"]`).append(`<div bayIndex=${bayIndex20\_second} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowLoadInch20"]`).append(`<div bayIndex=${bayIndex20\_first} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 $(`div[class="belowLoadInch20"]`).append(`<div bayIndex=${bayIndex20\_second} class="loadInch20"><span class="loadOrUnloadNum">0</span></div></div>`);  
 }  
 }  
 let vesselBodyBayList = ["17","19",];  
 for(let i=0;i<vesselBodyBayList.length;i++){  
 $(`div[bayIndex=${vesselBodyBayList[i]}]`).addClass("vesselBodyInLoading");  
 $(`div[bayIndex=${vesselBodyBayList[i]}] span`).addClass("vesselBodyBayIndex");  
 }  
}  
/\*\*  
 \* selectable  
 \*/  
let combinedBay20inch = [];  
function setStopOfSelectable() {  
 $(`#selectable`).selectable({  
 stop: function() {  
 let selectedBay = $(`.bayZone\_inch20.ui-selected`);  
 let isNumSelectRight = selectedBay.length===2 ? true : false;  
 if(isNumSelectRight){  
 let isReselect = (isExist(combinedBay20inch,selectedBay[0].id) || isExist(combinedBay20inch,selectedBay[1].id))? true:false;  
 let isNextTo = toAbsent(parseInt(selectedBay[0].id) - parseInt(selectedBay[1].id)) == 1?true:false;  
 if(isReselect || !isNextTo) {  
 alert("请重新选择");  
 clearSelected();  
 }  
 else {  
 let leftBayId = selectedBay[0].id;  
 let rightBayId = selectedBay[1].id;  
 combinedBay20inch.push(leftBayId);  
 combinedBay20inch.push(rightBayId);  
 let combinedBayInch40Index = numToIdString((leftBayId\*2-1+rightBayId\*2-1)/2);  
 selectToInch40(leftBayId,rightBayId,combinedBayInch40Index);  
 }  
 }  
 else {  
 alert("请重新选择");  
 clearSelected();  
 }  
 }  
 });  
}  
function disableSelectable() {  
 $(`#selectable`).selectable({  
 disabled: true  
 });  
}  
function enableSelectable() {  
 $(`#selectable`).selectable({  
 disabled: false  
 });  
}  
/\*\*  
 \* combination buttons  
 \*/  
function combineToStart (){  
 // initial bay  
 initAreaForInline();  
 insertBay(BayNumToRealIndexList(numOfBay));  
  
 $(`.startCombine`)[0].disabled = true;  
 setStopOfSelectable();  
 $(`.confirmCombine`)[0].disabled = false;  
}  
function combineToConfirm (){  
 disableSelectable();  
 $(`.startCombine`)[0].disabled = false;  
 $(`.confirmCombine`)[0].disabled = true;  
 clearSelected();  
 $(`.bayArea`)[0].style.display = 'none';  
 createBayAfterOperation(newBayList);  
 $(`.confirmCombine`)[0].disabled = true;  
}  
function combineReset (){  
 alert("确认重新组贝?");  
 combinedBay20inch = [];  
}  
/\*\*  
 \* zoom in and zoom out  
 \*/  
function setZoom(zoom,el) {  
  
 let transformOrigin = [0,0];  
 el = el || instance.getContainer();  
 let p = ["webkit", "moz", "ms", "o"],  
 s = "scale(" + zoom + ")",  
 oString = (transformOrigin[0] \* 100) + "% " + (transformOrigin[1] \* 100) + "%";  
  
 for (let i = 0; i < p.length; i++) {  
 el.style[p[i] + "Transform"] = s;  
 el.style[p[i] + "TransformOrigin"] = oString;  
 }  
  
 el.style["transform"] = s;  
 el.style["transformOrigin"] = oString;  
  
}  
function showVal(a){  
 let zoomScale = Number(a)/10;  
 setZoom(zoomScale,document.getElementsByClassName('mainArea')[0])  
}  
/\*\*  
 \* main  
 \*/  
let numOfBay = VIEW\_SIDE.max\_bay\_number;  
let layerNumAbove = VIEW\_SIDE.max\_layer\_above\_number;  
let layerNumBelow = VIEW\_SIDE.max\_layer\_below\_number;