

大连海事大学

硕士学位论文

集装箱堆场龙门吊路径优化问题研究

姓名：贺茂英

申请学位级别：硕士

专业：物流工程与管理

指导教师：计明军

20100601

摘要

集装箱港口的竞争力水平受地理位置、港口资源配备、信息技术应用以及经营管理等多种因素的影响。其中, 集装箱堆场装卸资源作为资源配备的一大投入, 其作业效率的高低将会很大程度上影响港口的运营成本甚至其竞争力水平。因此, 本文将对堆场装卸机械龙门吊的路径进行优化研究, 以期能为集装箱堆场乃至整个港口物流系统运作效率的提高提供一定参考价值。

本文以集装箱堆场资源为研究内容, 综合运用了运筹学、优化技术以及计算机仿真学科的理论与方法, 对龙门吊路径的优化问题进行了研究和探讨。首先从理论上分析了集装箱堆场龙门吊路径优化问题, 接着详细介绍了本文采用的模拟退火算法, 并分析总结了模拟退火算法的应用领域。其次, 在前面章节的理论研究基础上, 建立了以最短路径为目标函数、以岸桥装卸要求等为约束条件的单台龙门吊的数学模型, 设计了单台龙门吊可行路径确定策略, 对模拟退火算法进行改进, 并通过编程求得了模型的解。结果分析表明了单台龙门吊路径优化问题数学模型的合理性, 以及采用求解方法的有效性。

但是在集装箱实际装卸操作中, 每一箱区一般配备不低于一台龙门吊进行作业。因此, 本文进一步建立了多台龙门吊的路径优化模型, 并设计多台龙门吊可行路径的确定方法, 且将其应用于模拟退火算法对模型进行了有效求解。最后用两个实例从不同角度验证了模型和算法的合理性。

论文主要创新点之一是: 建立了单台龙门吊路径优化模型, 提出了单台龙门吊可行路径的选择策略, 并设计算法进行了有效的求解。其次, 建立了多台龙门吊的路径优化模型, 通过设计两台龙门吊的可行路径, 改进模拟退火算法对模型进行求解并得到了很好的收敛结果, 这是本文的另一创新点。

本文采用最优化技术以及模拟退火算法等理论和方法对集装箱堆场龙门吊路径优化问题进行了详细研究, 希望能为我国集装箱港口资源配置及优化的理论和实际运作提供一定的指导和决策依据。

关键词: 集装箱; 龙门吊; 路径优化; 模拟退火算法

Abstract

The level of the competitiveness of container ports is affected by geographical location, equipped with port resources, information technology and business management and other factors. Among them ,the handling equipment resources of container yards, as a major investment of these resources, its operating efficiency will significantly affect the level of operating costs and even the competitiveness level of the port .Therefore , this article will research on the routing optimization problem of loading and unloading machines ,gantry cranes ,at container yards ,which can provide a reference value for the operating efficiency improvement of container yards even and the whole port logistics system.

The study contents of this dissertation ,which analysis the routing problem of gantry cranes in depth with comprehensive utilization of operations research, optimization theory and computer simulation of disciplines and methods ,are container yard resources .Firstly, the container yard gantry crane routing problem were analysed from the theoretical ,then the simulated annealing algorithm and its applications are introduced. Secondly, a mathematical model of a single gantry crane on the basis of the theoretical research in the previous chapters, with the objective of the shortest path and the constraints of loading and unloading requirements for quay cranes. A feasible path strategy of single gantry crane is designed and the simulated annealing algorithm is improved, the solution of the model is obtained by programming. The results showed the rationality of the mathematical model of a single gantry path optimization, and the validity of the method.

However, each packaging material zone is generally equipped with not less than a gantry crane for operation in the actual loading and unloading operations at container yards. Therefore, this paper established a optimization model of multiple gantry cranes routing. The feasible method for a determining path of multiple gantry cranes is designed, which is applied to simulated annealing algorithm for solving the model effectively. Finally, effectiveness of the model and the rationality of the algorithm is proved with two examples from different angles.

One of the main innovations of this dissertation is the establishment of a single gantry path optimization model, the feasible path selection strategy of a single gantry ,

and design of effective algorithms. What is more, a optimization model of multiple gantry cranes routing, the possible paths of two gantry cranes are given, a good convergence result with improved simulated annealing algorithm is achieved ,which is another innovation of this article.

This thesis applies the optimization technology and simulated annealing algorithm theory to study the routing problem of gantry cranes at container yard. It provides some valuable suggestions and help for the allocation of resources and the optimization operating at container terminals.

Keywords: Container; Gantry Crane; Routing Problem; Simulated Annealing Algorithm

大连海事大学学位论文原创性声明和使用授权说明

原创性声明

本人郑重声明：本论文是在导师的指导下，独立进行研究工作所取得的成果，撰写成硕士学位论文“集装箱堆场龙门吊路径优化问题研究”。除论文中已经注明引用的内容外，对论文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文中不包含任何未加明确注明的其他个人或集体已经公开发表或未公开发表的成果。本声明的法律责任由本人承担。

学位论文作者签名：贺磊

学位论文版权使用授权书

本学位论文作者及指导教师完全了解大连海事大学有关保留、使用研究生学位论文的规定，即：大连海事大学有权保留并向国家有关部门或机构送交学位论文的复印件和电子版，允许论文被查阅和借阅。本人授权大连海事大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，也可采用影印、缩印或扫描等复制手段保存和汇编学位论文。同意将本学位论文收录到《中国优秀博硕士学位论文全文数据库》（中国学术期刊（光盘版）电子杂志社）、《中国学位论文全文数据库》（中国科学技术信息研究所）等数据库中，并以电子出版物形式出版发行和提供信息服务。保密的论文在解密后遵守此规定。

本学位论文属于： 保 密 ☐ 在 _____ 年解密后适用本授权书。

不保密 ☐ （请在以上方框内打“√”）

论文作者签名：贺磊 导师签名：Zhang

日期：2010年6月29日

第1章 绪论

1.1 论文的背景

随着集装箱运输的迅速发展，集装箱运量不断上升，集装箱船舶日益大型化和高速化，而快速上升的集装箱运输市场使我国现有的不少集装箱码头出现了能力缺口。同时，由于近年来现代物流的快速发展，越来越多的客户要求货物运输能够实现“门到门”的服务，而“门到门”运输方式的实现需要无缝连接的运输链作为支撑。集装箱码头作为运输链上的一个重要环节，不仅要有足够的容量适应快速增加的货源需求，而且要有高水平的集装箱堆场作业能力来快速响应现代物流发展的要求。

1.1.1 现代物流

在科学技术不断进步和经济不断发展的环境下，以运输、储存、装卸搬运、包装、配送、流通加工为一体以及物流信息管理网络化作为主要内容的现代物流，不仅是企业竞争的主要因素。还是一个国家综合国力的重要标志，物流业正成为各企业、各地区乃至国家的支柱产业。货物可以在不同运输方式、各货运站之间实现无缝链接，以最短的时间、最小的费用，安全、准确地实现货物时间和空间的转移，这是现代物流的发展要求。

作为物流系统中重要节点的港口，是远洋、内河船舶以及内陆运输(火车和卡车)的枢纽。港口地区汇集有船东、船舶代理、货主、货运代理、包装公司、陆上运输公司、商品批发零售、海关、商检等各种机构。港口不仅是不同运输方式汇集的重要节点，而且也是经济、技术和各种信息的汇集点。现代物流的迅速发展以及港口本身在现代物流系统中的重要地位，使作为物流平台的港口的发展压力日益增大。为了实现陆路、水路运输方式之间的无缝链接，集装箱港口需要按照现代物流的要求，合理配置港口内部的物流要素，优化物流要素流通渠道，完善物流信息系统，以不断提高港口内部物流网络的运作效率，降低港口运营成本。

1.1.2 集装箱运输量

自从1957年美国将第一艘集装箱船正式投入运营以来，集装箱运输方式便以其装卸运输效率高、港口操作简单、货损货差小等优点在全世界范围内得到了快速

的发展，在运输总量中的比重也在不断提高。一方面发展中国家依靠科技进步发展经济，从传统的单纯原材料进出口转变为科技含量高的工业成品与半成品的出口；另一方面发展中国家不断增加投资，完善集装箱运输系统，件杂货运输逐步实现集装箱化，从而使适箱货比例增加，最终使集装箱运输量不断增长。与此同时，发达国家出口结构更趋高端化，高、精、尖产品不断增加，适箱货源增多，从而促使集装箱运量不断提高，集装箱运输持续发展。

快速增加的运输总量使港口的集装箱吞吐量不断增加，但为了适应全球航运市场的激烈竞争形势，集装箱船舶出现了大型化趋势，世界集装箱船队高速扩张。然而快速增加的集装箱吞吐量和集装箱船舶的大型化，使不少集装箱港口出现了满负荷、超负荷运营的现象。为了能够在激烈的竞争中取胜，港口的管理者不仅要从中长期规划港口的扩建计划，迫在眉睫的一个问题应该是将现有的作业资源优化配置，提高港口的内部运作效率和管理水平，促进集装箱港口向重技术、重管理的方向发展。

1.1.3 集装箱港口

近年来，全球的各大航运公司纷纷组建战略联营体，为的是能将世界航运市场作为自己的经营舞台，并首先从港口开始了从传统物流服务向现代物流服务的转变。为了降低运营成本、提高运作效率、实现规模经济，战略联营实体规模在不断扩大，经营战略也在不断调整。

一般各大航运公司在—个地区只会有—个枢纽港，哪个港口能够最终成为本地区的枢纽港口，除了取决于港口的自然地理条件外，港口的各项技术水平、内部物流运作效率和管理水平、信息交换能力等也是船公司的主要考虑因素。因此为了配合各大航运公司调整经营战略的，在新一轮的竞争中取得竞争优势，港口首先要使其内部物流合理化、系统化，为集装箱船舶停泊、装卸提供—个良好的物流平台，使港口在提高其资源利用率的同时，也要降低集装箱船舶的在港时间。

1.1.4 信息技术

以网络技术、通讯技术、控制技术、感测技术等为代表的现代信息技术的发展以及全球信息网络的兴起，让我们从传统的工业时代进入了—个崭新的信息时代，为集装箱港口进行快速、高效地生产提供了可能。目前信息技术已经在各港

口航运企业得到了广泛的应用。例如,我国沿海一些先进的大型集装箱港口的高频对讲机被无线电通讯替代后,港口的内部物流信息和作业信息的传递变得更为快速、准确。完整的 GIS(地理信息系统)分析软件集成了最短路径模型、车辆路线模型、分配模型、定位模型和网络模型,为最优路径的选择和最佳运输任务的分配提供了决策支持。除此之外,射频技术(RF)、全球定位系统((GPS)、和条码技术等技术的广泛应用都为集装箱港口的内部物流高效运作提供了各种保障。集装箱港口后方堆场是港口中用于堆存集装箱的专设区域,其作为港口主要的组成部分,在集装箱港口的物流作业流程中起着很重要的作用,堆场资源作业效率的高低将直接影响到港口整体的作业效率。

在上述背景下,如何合理高效地利用集装箱堆场的作业资源就成了一个迫切需要解决的课题。

1.2 国内外研究现状

1.2.1 集装箱堆场资源配置研究现状

堆场机械是集装箱港口堆场中用于集装箱装卸、搬运和堆码的专用机械。常用的堆场机械主要有两种,轮胎式龙门吊和轨道式龙门吊,统称为龙门吊。龙门吊可以在堆场的不同箱区和同箱区的不同贝位之间移动。

近年来,国内外已经出现了一些对龙门吊的配置研究。从堆场资源优化配置方面, Richard Linn^[1]研究了几个工班的龙门吊的配置问题。文中给出了一个两阶段模型,首先考虑堆场机械的作业均衡,将到达集装箱按类型分配到箱区;然后再考虑最大限度地降低翻箱率,将集装箱具体放到某一个位置。但是在每个工班之内的移动次数不能超过 1 次,该假设与实际情况存在较大的差距。Amborski^[2]运用了数字仿真研究了堆场存取运作规划。Mosca 等^[3]研究了码头和堆场的空间管理。Kozan and Prseton^[4]建立整数规划模型,使龙门起重机的各箱区总工作时间最大化来解决龙门起重机配置问题。L.MGambardella et al^[5]运用网络流的方法对整个港口的装卸资源进行了仿真研究,在模型中,将每个工作日分为三个工班,同时假设所有的桥吊工作能力均相同,船舶在哪个工班到达为已知,所靠船舶的装卸箱量根据船舶的抵港预报也为已知。需要决定的就是每个工班所需配置的桥吊数量、龙门吊配置数量以及各船每工班装卸箱量,最后将得到一张计划时域内的资

源配置计划表。在国内,徐亚、陈秋双等^[6]为降低集装箱堆场的倒箱率,对翻倒箱落箱位置的确定问题进行了研究,提出了一种改进的启发式算法求解该问题,结果表明算法能够有效的求解该类优化问题。陆志强等^[7]研究了码头的优化模型及其求解算法。计三有和高月文^[8]运用搜索技术提出了出口箱箱位的合理分配及减少倒箱率的合理措施,对如何减少集装箱堆场倒箱率进行了研究,重点研究了集装箱堆场物流系统的优化。上述研究中,对龙门吊配置的研究比较少,尤其是如何确定集装箱港口装卸中的龙门吊配置数量更是少见。

1.2.2 集装箱港口作业机械研究现状

集装箱港口的作业机械有很多,这里只对与堆场龙门吊作业情况密切相关的机卸岸桥和港内车辆进行研究分析。岸桥配备的多少以及作业效率的高低,都会影响到堆场龙门吊的作业效率。从岸桥方面,Gambardella等^[9]研究了岸壁集装箱装卸桥和堆场龙门吊的配置问题以及每个分配的机械装卸进度安排问题。Kozan等^[10]利用遗传算法对接卸设备设置优化问题进行了求解。Böse等^[11]以岸壁集装箱装卸桥的等待时间最短为目标,对岸边有缓冲区的集装箱码头建立了仿真模型,在给定集装箱装卸序列的要求下为桥吊配备跨越车而且还确定了跨越车的作业序列。Peterkofsky等^[12]将桥吊规划看作开放的生产计划问题来对待建立了整数规划模型,并用分支定界法求解了模型。陈璐等^[13]研究了集装箱码头装卸作业的调度控制问题,建立了一个基于柔性化 flow shop 的集成化控制模型,并设计了两类基于优先级规则的启发式跳读算法进行求解。蔡芸^[14]建立了总体船舶在港时间的最小化的仿真优化模型。仿真优化方法采用遗传算法产生和评价泊位分配方案,通过仿真模型获得满足靠泊约束和岸桥调度策略的可行解,最终求得了包括船舶的靠泊时间、靠泊位置和为其服务的岸桥数目等要素的优化解。韩骏等^[15]为了缩短船舶在港停留时间,提出了以船舶在港时间最小化为目标的泊位与岸桥协调调度优化方法,对岸桥分配与泊位调度这两个相互关联的问题进行系统分析与集成,基于免疫遗传算法对所建的模型进行相应的算法开发。目前集装箱码头常用的非自动化的单箱港内运输机械有铲车、集装卸卡车、跨越车等,多集装箱运输有港内轨道系统,而自动化集装箱码头则采用自动导航车辆系统(AGV)。从水平运输机械方面,Vis等^[16]利用了网络优化模型来确定一定任务要求条件下自动导航车辆的

最小数量。Bishop E. K. 等^[17,18,19,20]分别从交通控制、网络优化等角度利用整数规划和启发式算法等研究了车辆分配和路线选择问题。Steenken D 等^[21, 22]多次论述此问题,建立了网络优化等模型,其优化目标是通过组合运输任务最小化空驶距离。Kim 和 Bae^[23]利用整数规划、启发式算法以船舶延误时间和 AGV 车辆消耗时间最小为目标函数为待托运集装箱箱分配 AGV 车辆。Bake^[24]研究了跨运车作业对堆场装卸设备运作效率的影响。吕显强等^[25]对港内车辆调度用整数规划模型进行了初步地探讨。对于岸桥和水平运输资源的研究将会有助龙门吊路径以及龙门吊与其协调作业的优化,可以有效提高整个港口系统的运作效率。

1.2.3 集装箱堆场龙门吊路径优化问题研究现状

在集装箱堆场龙门吊路径优化方面,Chung et al^[26]最早提出了龙门吊路径选择问题,目的是减少龙门吊的不必要移动,提高龙门吊的工作效率。Lai 和 Lam^[27]运用模拟方法比较了不同配置策略下龙门吊的使用效率、作业量及等待时间的问题。Kim 等^[28]研究了单个龙门吊的最优路径问题,文中假设每个龙门吊只为一个箱区服务,并且箱区内每个贝位上的箱子都根据大小、重量和目的港分成几类,同一类的箱子可以分散在不同的贝位。文中给出了混合整数规划模型,其目标是使龙门吊的总处理时间(启动时间和行走时间)最短,并且给出了模型的精确求解方法。Kap 等^[29]考虑了龙门吊操作不同限制条件,运用混合整数规划模型,采用了分支界定法(B&B)进行求解,并用贪婪随机适应搜索过程来克服分支界定法在计算机求解时的不足。W.C.Ng 等^[30,31]研究了龙门吊优化调度问题,目标函数是等待时间最小化,通过物流作业计算实例,应用分支定界算法对调度模型求解。Richard 等^[32]采用了启发式算法对场桥的动态布局进行了研究。Zhang^[33]假定每天每时段区域箱量工作量为已知情况下,利用拉格朗日松弛求解了起重机用时最少的移动路线优化问题。在国内,魏众、申金升^[34]等提出了轮胎式龙门吊优化调度混合整数规划理论,研究了定量作业条件下的龙门吊优化调度问题,通过实际案例分析为码头管理提供决策支持。何军良、亦为建和严伟^[35]采用一种基于整数规划模型和启发式法(爬山算法)的场桥调度混合策略,将场桥调度和堆存空间分配作为整体构建场桥动态调度模型。高鹏等^[36]针对集装箱堆场进口集装箱的提箱作业计划问题,建立了以作业总成本最小为目标的多阶段决策优化模型,构造了内外嵌套两层结

构的优化算法,内层算法实现最短路径搜索了模型,外层算法实现倒箱策略优化了模型,并针对不同的问题规模给出了不同的算法方案。

从以上研究可以发现,目前国内外对龙门吊具体路径优化的研究相对较少,且现在的研究多数针对单台龙门吊的作业情况,而对结合岸壁集装箱桥装卸要求的作业计划也很少有所研究,故本文将针对此方面问题展开探讨,希望能对集装箱码头的作业资源优化理论和业务运作提供一定的指导和决策依据。

1.3 论文的研究意义

1.3.1 理论价值

集装箱码头是资本密集型的经济实体,要使码头作业高效而经济地运行,集装箱码头作业资源的优化配置关系到码头的运作效率及其竞争力。而集装箱码头堆场作为码头一个不可缺少的组成部分,在码头整个作业的流程中起着相当重要作用。为了更好地指导我国集装箱码头的运作实践,本文将以集装箱码头堆场龙门吊路径为研究对象,逐次考虑单作业线、岸壁集装箱装卸要求条件下的龙门吊路径优化问题,希望能为我国集装箱码头的作业资源优化理论和业务运作提供一定的指导和决策依据。

1.3.2 实际意义

另外集装箱运输是一种高效率、大规模生产的运输工艺。集装箱码头是海陆联运的枢纽,是海上运输和陆上运输的链接点之一,提高码头上物流运作水平,对码头现有资源进行合理地控制及决策,不仅可以提高集装箱装卸作业效率,为船舶配载计划提供指导性建议,还可以加速车、船、箱的周转,降低运输成本,提高整个集装箱运输系统的营运效益和综合社会效益。因此,本文的研究具有较强的实践意义。

1.4 论文研究内容及技术路线

本文从实用性、准确性、高效性出发,对集装箱堆场龙门吊的路径进行优化建模、算法设计与实现方法进行研究。结合岸壁集装箱装卸桥的装卸要求,运用模拟退火算法分别对单台龙门吊和多台龙门吊的路径进行建模及分析,并用实际算例验证了模型和算法的有效性。

本文的研究工作按以下步骤进行：

第一章，阐述了课题的研究背景、集装箱堆场资源配置以及龙门吊路径优化问题的研究现状及其不足、研究意义以及本文的研究工作。

第二章，对集装箱堆场龙门吊优化问题进行概述，为接下来的研究作理论准备。

第三章，介绍了模拟退火算法的含义、参数的设置，并总结了算法的应用领域。

第四章，首先对单台龙门吊的路径优化问题进行描述，建立数学模型，设计算法，并通过两个实例验证模型的合理性以及算法的高效性。

第五章，同样先是对多台龙门吊的路径优化问题进行描述，建立相应的数学模型，然后设计两台龙门吊的可行路径及模拟退火算法，并应用同样的算例对比说明模型和算法的最优性。

第六章，总结本文的研究成果以及不足，并指出今后的研究方向。

技术路线如图 1.1 所示。

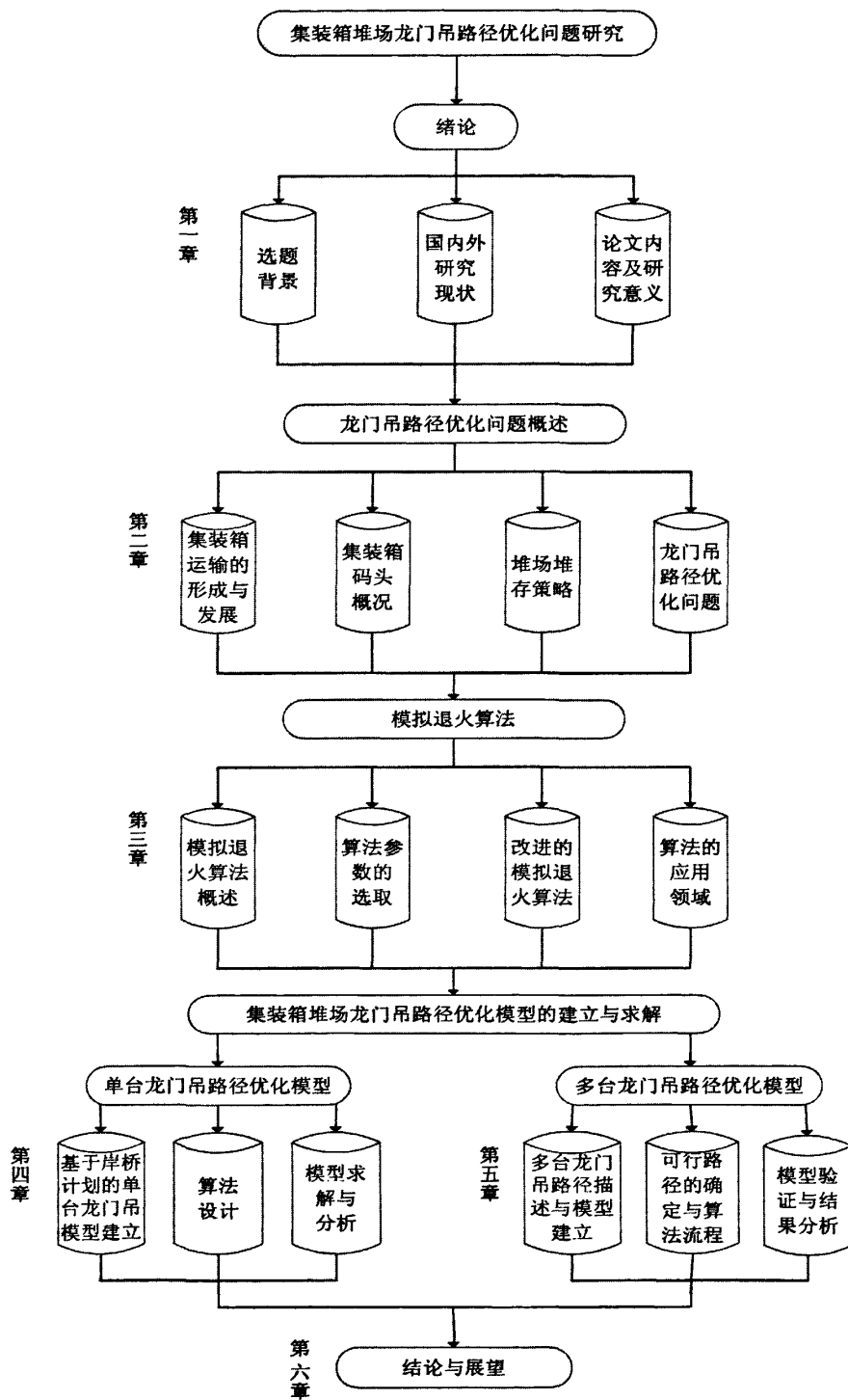


Fig. 1.1 The technology roadmap of the dissertation

第 2 章 集装箱堆场龙门吊路径优化问题概述

2.1 集装箱运输的形成与发展

集装箱 (Container) 是一种便于反复使用、具有足够强度的大型标准化载货容器^[37]。集装箱运输 (Container Transportation) 就是将货物装在集装箱内, 以集装箱作为一个货物集合或成组单元, 进行运输 (包括船舶运输、铁路运输、公路运输、航空运输以及这几种运输方式的联合运输)、装卸、搬运的运输工艺和运输组织形式。

集装箱运输是一种先进的现代化运输方式, 是交通运输现代化的产物和重要标志, 也是件杂货运输的发展方向及运输领域的重要变革, 因此世界各国都把集装箱运输称为 20 世纪的“运输革命”。

2.1.1 集装箱运输产生的原因

第二次世界大战以后, 世界经济得到了迅猛发展, 跨国经营以及国际贸易量不断上升, 国际间的市场竞争愈演愈烈, 致使企业不得不采用大规模的专业化生产, 以降低成本、提高技术水平和生产效率, 争取在国际市场竞争中处于有利地位, 以便获取更大的利润。

国际贸易的上升以及跨国经营的增加, 对国际间货物运输提出了更高的要求。传统的货物运输由于采用件杂货的方式, 货物品种多、包装形式多样、单件重量相差较大, 很难实现全过程的机械化和自动化的运输生产, 也不适应现代大规模专业化生产的要求。为了克服件杂货运输所带来的问题, 扩大运输单元是必然的趋势。在这一变化过程中, 首先出现的是以网络和托盘等成组工具带来的货物运输和装卸的成组化, 但是这种成组方式的集成化程度有限, 外形仍不规整, 由于包装强度所限, 也使得堆码困难。所以, 更大单元的、外形尺寸标准的、包装具有一定强度的集成化运输方式的出现就势在必行, 而集装箱运输方式的诞生正是顺应了这种需求。

2.1.2 国际集装箱运输的发展沿革

萌芽阶段 (1801—1955 年)^[38]。该时期的重要标志是欧美地区的发达国家在国内开始尝试陆上集装箱运输, 运输距离较短, 主要是在欧洲各国之间进行陆

上集装箱运输的合作。由于公路和铁路集装箱运输不统一，制约了陆上集装箱运输的发展，集装箱运输发展缓慢。开创阶段（1955——1966年）。该时期的重要标志是美国首先用油轮、件杂货轮改装成了集装箱船舶在美国沿海从事海上集装箱运输，并获得良好的经济效益。海上集装箱运输获得成功，为实现国际远洋航线的集装箱运输打下了良好的基础。成长阶段（1966——1971年）。这一时期的重要标志是集装箱运输从美国的沿海运输向国际远洋运输发展。从事集装箱运输的船舶为第一代集装箱船，其载箱量在700——1100TEU之间，并有了专用码头。扩张阶段（1971——20世纪80年代末）。扩张阶段的重要标志是集装箱运输迅速发展，世界各主要航线开展了集装箱运输。由于集装箱运输具有装卸效率高、成本低、效益好、运输质量高且便于开展国际多式联运等优点。因此，集装箱运输深受货主、船公司、港口及其他有关部门的欢迎，在1971年至20世纪80年代末发展及其迅速，其国际远洋运输航线从欧美扩展到东南亚、中东及世界各主要航线。成熟阶段（20世纪80年代末至今）。20世纪80年代末以来，国际集装箱运输的发展已进入成熟期。其重要标志是集装箱运输船舶、码头泊位、装卸机械、集疏运的道路桥梁等硬件设施日臻完善，集装箱运输的经营管理、业务管理的方法和手段等软件配置越来越现代化，集装箱运输在全世界普及，多是联运得到进一步发展。船舶大型化、码头深水化、竞争激烈化是发展趋势。

2.1.3 我国集装箱运输发展概况

我国集装箱运输始于20世纪70年代初。1973年9月开始用件杂货船从天津、上海载运小型集装箱（8ft×8ft×8ft）至日本的横滨、大阪、神户^[39]。虽然起步较晚，但发展的速度却是最快的。自1973年天津港接卸了第一个国际集装箱开始，历经了70年代的起步、80年代的稳定发展、90年代的快速发展，到21世纪初我国集装箱运输引起全世界航运界的热切关注。

随着我国对外贸易和外向型经济不断发展，我国集装箱运输发展很快，拥有了一支现代化的集装箱船队，建成了一批集装箱专用深水泊位，建立了较为畅通的集疏运系统，培养了一批集装箱运输经营管理队伍，集装箱化的水平明显上升。我国集装箱目前发展的特点有：船舶运力发展迅速，航线不断增多；基础设施建设不断加强，港口条件和内陆集疏运系统明显改善；加强了法规建设和科技应用，

集装箱运输管理水平有了很大提高。

2.1.4 集装箱运输的发展趋势

目前, 集装箱运输量继续增长; 集装箱船舶大型化、高速化发展; 集装箱码头不断现代化, 港口中转作用日益提高; 集装箱箱型向大型化、专业化发展; 集装箱运输信息管理实现现代化; 集装箱多式联运将进一步发展和完善; 市场竞争激烈, 各大班轮公司经营战略将明显改变; 集装箱运输想绿色化发展。

2.2 集装箱码头的概况及其特征

2.2.1 码头的功能和布局

集装箱码头(Container Terminal)是集装箱运输的枢纽, 它向外延伸国际的远洋运输航线, 向内连接国内的铁路、公路、水路等运输路线^[40]。因此, 集装箱的码头的主要功能有: 集装箱运输系统的集散站; 提供集装箱堆存, 作为转换集装箱运输方式的缓冲池; 水路集装箱运输和陆路集装箱运输的连接点和枢纽。

集装箱码头的高度机械化和高效率的大规模生产方式, 要求集装箱码头头脑更船舶共同形成一个不可分割的有机整体, 从而保证高度严密的流水作业线高效运转, 充分发挥集装箱码头三个主要职能的作用。集装箱码头通常应具备的必要设施有: 泊位、码头前沿、集装箱堆场、货运站、控制室、行政楼、集装箱清洗场、大门、维修车间等。对于集装箱专用码头, 在码头布局上主要要求集装箱泊位岸线长为 300m 以上; 集装箱码头陆域纵深一般为 350m 以上; 码头前沿宽度一般为 40m 左右; 每一集装箱专用泊位, 配置 2 至 3 台岸壁集装箱起重机; 集装箱堆场面积达 105000m², 甚至更大; 集装箱货运站可布置在集装箱大门与堆场之间的地方, 也可布置在集装箱码头以外的地方; 所有通道的布局应根据装卸要求而定。

2.2.2 集装箱码头的作业资源

集装箱的标准化, 为港口码头装卸高效率化和自动化提供了良好的条件。集装箱码头的作业资源主要分为码头岸边装卸机械、水平运输机械、场地装卸机械。

岸边装卸机械主要有: 岸壁集装箱装卸桥(Gantry Crane, Quay Crane)、多用途门座起重机(Tractor)、高架轮胎式起重机(MHC)等。其中岸壁集装箱装卸桥

式现代集装箱码头普遍采用的岸边装卸机械^[41]。岸壁集装箱装卸桥简称“集装箱装卸桥”或“桥吊”，它沿着与码头岸线平行的轨道行走，主要由带行走机构的门架、承担臂架机构的拉杆和臂架等几部分组成。

水平装卸机械主要有：牵引车（Tracter）和和挂车（Trailer）、跨运车（Straddle Carrier，缩写 SC）等。其中，集装箱牵引车又称“拖头”，其本身不具备装货平台，必须和集装箱挂车连接在一起，才能拖带集装箱进行码头内或公路上的运输；挂车又称为“拖车”或“平板”，其本身没有动力，仅仅是一个载箱的平台；跨运车是以门型车架跨在集装箱上，由吊具的液压升降系统吊起集装箱进行搬运和堆码的。

场地装卸机械有：叉车（Fork Lift Truck，缩写 FLK）、轮胎式龙门起重机（Rubber Transtainer Gantry，缩写 RTG）、轨道式龙门起重机（Rail Mounted Gantry，缩写 RMG）、正面吊运机（Reach Stacker，缩写 RS）。集装箱叉车主要用于吞吐量不大的综合性码头进行集装箱的装卸、堆垛、短距离搬运、车辆的装卸作业^[42]。是一种多功能机械，具有正面叉和侧面叉两种。轮胎式龙门起重机是集装箱码头堆场进行装卸、搬运、堆垛作业的专用机械，其由前后两片门框和底梁组成门架，支撑在橡胶轮胎上。装有集装箱吊具的行走小车沿着门框横梁上的轨道运行，配合底盘车进行集装箱的堆码和装卸作业。它不仅能前进、后退，而且还设有转向设置，通过轮子的 90° 旋转，能从一个箱区转移到另一个箱区进行作业。轨道式龙门起重机是集装箱码头和集装箱中转站堆场进行装卸、搬运和堆码集装箱的专用机械。该机由两片悬臂的门架组成，两侧门腿用下横梁连接，门架支承在行走台上，并在轨道上运行。轨道式龙门起重机一般比轮胎式龙门起重机大，堆垛层数多。它是沿着场地上铺设的轨道行走的，所以只能限制在所设轨道的某一场地范围内进行作业。轨道式龙门起重机自身定位能力较强，较易实现全自动化装卸，是自动化集装箱码头比较理想的一种机械。

2.2.3 集装箱码头的装卸工艺

集装箱码头装卸工艺指装卸集装箱的方法。集装箱装卸工艺决定码头装卸机械设备、码头装卸生产组织、劳动定额和劳动生产率，也影响码头装卸作业综合经济效益。目前采用的集装箱码头装卸工艺主要有底盘车装卸工艺、跨运车装

卸工艺、正面吊运机装卸工艺、龙门起重机装卸工艺的系统，其中以龙门起重机工艺系统应用最为广泛。

底盘车工艺系统的码头前沿由集装箱装卸桥进行装卸船作业。卸船时由岸桥直接将集装箱卸到底盘车上，由牵引车拖至堆场停放；装船时由牵引车把装有集装箱的底盘车拖至堆场，装船时再拖至码头前沿装船；底盘车也可以进行陆上水平运输。

跨运车工艺系统一个综合系统，码头前沿由集装箱装卸桥卸船，水平运输、堆场堆码及装卸均由跨运车完成。

正面吊运机工艺系统码头前沿采用岸桥装卸船舶，码头前沿与堆场之间水平搬运、堆场堆码箱及装卸作业均由正面吊运机完成。

龙门起重机工艺系统按其行走方式不同，可分为轮胎式龙门起重工艺系统和轨道式龙门起重机工艺系统。轮胎式龙门起重机工艺系统的装卸船由岸桥完成，轮胎式龙门起重机承担堆场堆码和装卸车作业，水平搬运由集装箱底盘车完成^[43]。轮胎式龙门起重机可堆码集装箱 3 至 4 层高，一般可横跨 6 列集装箱和 1 列火车，它可以从一个箱区移动到另一个箱区。轨道式龙门起重机工艺系统的跨距比轮胎式工艺系统更大可跨 14 列集装箱或更多，可堆码 4 至 5 层集装箱。轨道式龙门起重工艺系统那个适用于堆场面积有限、集装箱吞吐量较大的码头。

2.3 集装箱堆场堆存策略介绍

集装箱码头堆场，不仅起到集装箱装卸场的作用，同时还起着集装箱储存、保管、交接和集疏运作用。堆场管理是集装箱码头现场生产管理的中心环节之一，涉及码头堆场的分类和箱位的安排、堆垛规则及集装箱的堆存、保管、发放、交接、装卸、中转、装箱、拆箱等诸多生产业务。

2.3.1 堆场箱区的划分

堆场箱区可按不同的分类方法分出不同的箱区：按进出口业务可分为进口箱区和出口箱区；按集装箱装载状态可分为空箱区、重箱区；按集装箱货种可分为普通箱区、危险品箱区、冷藏箱区、特种箱区和中转箱区。码头箱管人员在安排箱区时，原则上各箱区堆放哪一类箱式相对固定的，但也可以根据码头进出口箱情况、实际堆存情况、船舶到港情况和船公司用箱情况等，适当调整各箱区的比

例。

2.3.2 堆场箱区的箱位编码方式

集装箱堆放在码头堆场，一般在场地上都要按照集装箱的箱型、尺寸预先划出标准区域，并用一组代码来表示其在堆场内的无理位置，这个位置就是堆场位置，即称“场箱位”，它是组成集装箱堆场的最小单元。在场箱位线端部标出编号，这种号码称做“场箱位号”，又称“场位号”。场箱位由箱区、位（贝）、排、层组成。

（1）箱区。箱区的编码分为两种：一种是用一个英文字母表示；另一种是由一个英文字母和一位阿拉伯数字组成，其中第一个英文字母表示码头的泊位号，第二位阿拉伯数字表示堆场从海侧到陆侧后方堆场的顺序号。国内码头普遍采用一位字母和一位数字组合作为箱区的编码。

（2）位（贝）。一个箱区由若干个位组成，位（贝）的编码一般用两位阿拉伯数字表示，与集装箱船舶箱位（行）号类同，用奇数 01, 03, 05, 07……表示 20ft 箱的位（贝），用偶数 02, 06, 10, 14……表示 40ft 箱或 45ft 箱的位（贝）。位（贝）数与堆场箱区的长度有关，而箱区的长度往往与泊位的长度或纵深相对应。

（3）排。用一位阿拉伯数字表示。堆箱层数是视轮胎吊的跨度而定，一般轮胎吊的跨度为 23.47m，这样箱区的排数就是六排。

（4）层。用一位阿拉伯数字表示。堆箱层数是时轮胎吊的高度而定，不同类型的轮胎吊系统，堆垛高度也不相同，一般是四层或五层。

因此，集装箱的场箱位一般由“五位”或“六位”表示，如“A0111”表示该箱区 01 位（贝）第一排第一层；“A10111”则表示 A1 箱区 01 位（贝）第一排第一层。

2.4 集装箱堆场龙门吊路径优化问题

在集装箱进出口货运业务中，堆场的主要作业流程如下：

（1）提箱作业：首先堆场箱控室接到配载总图、装船顺序单，查看各舱所配卸港，计算箱量；接到调度室的装船通知，联系船边交接员和工班指导员，明确作业意图、提箱顺序、场区和注意事项等，通知相应机械到位，同时船边员通知装

船拖头到场；然后箱控室将电脑显示待装的箱号、场位报给机械司机，机械司机完成装车操作，集卡出场；最后箱控室获得机械司机确认后，将已装船的箱号及操作司机代码输入电脑，堆场提箱作业完成。

(2)卸箱作业:首先堆场箱控室根据卸船堆场计划的场区安排，联系船边交接员，确定卸船作业顺序，通知各接箱场区机械司机做好卸船接箱准备，同时注意各卸船接箱区机械作业的协调进行；箱控室接到卸船清单后，调度员通知卸船作业，待接到船边员通知后，再通知相应机械到位；集卡根据船边交接员的指示将卸船箱拖往相应的卸箱区后，机械司机将箱号报给箱控室，箱控室根据电脑给出的计划卸箱场位，确认符合计划要求后，指导司机卸箱；机械司机完成卸箱操作后将信息报给箱控室，箱控室将箱号、实际卸箱场位、司机代码等输入电脑，堆场卸箱作业完成。堆场龙门吊现场作业示意图如图 1.1 所示。



图 1.1 龙门吊作业示意图

Fig. 1.1 The operating diagram of gantry crane

对于本文所要研究的龙门吊路径具体是指上述提箱作业环节的提箱顺序问题，即满足岸桥装船顺序条件下，龙门吊如何在每一箱区行走，以使龙门吊的作

业时间或是作业路径最短。

2.5 本章小结

本文首先介绍了集装箱运输的形成与发展，然后再从集装箱码头的功能和布局入手，介绍了集装箱码头的作业资源和装卸工艺，其中着重讲解了龙门起重机及其装卸工艺；其次本章给出了集装箱堆场的堆存策略，为下文研究集装箱堆场龙门吊的路径做好了理论准备；最后本章从龙门吊的现实作业角度，分析了龙门吊的路径优化问题，为后续研究奠定了基础。

第3章 模拟退火算法

模拟退火算法思想最早是由 Metropolis 在 1953 年提出的 (Metropolis et al. 1953)。Kirkpatrick et al. (1983) 首先注意到固体退火过程与组合优化问题的类似性 (见图 3.1 与图 3.2), 将 Metropolis 准则引入优化中, 提出了模拟退火算法, 并成功应用在组合优化问题上。理论上, 模拟退火算法是局部搜索算法的扩展, 但不同于局部搜索之处是按一定概率选择邻域中目标函数更优的状态, 可以使算法跳出局部极小点, 保证了算法的全局收敛性。

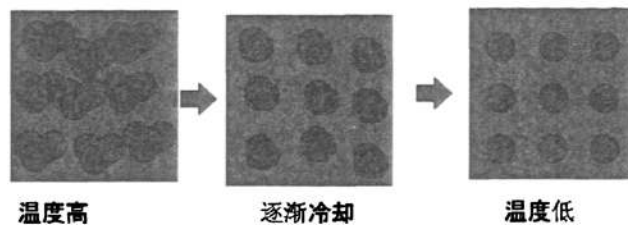


图 3.1 固体退火过程

Fig. 3.1 The annealing process of solid

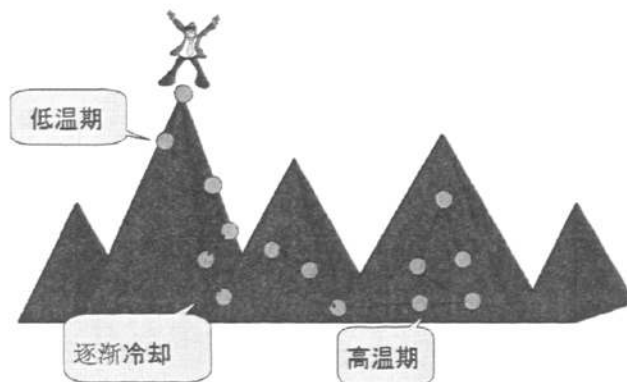


图 3.2 优化寻优过程

Fig. 3.1 The process of optimization

3.1 模拟退火算法概述

模拟退火算法 (Simulated Annealing, SA) 来源于与固体退火的相似性, 同时采用 Metropolis 准则作为接受准则。所谓的固体退火是一个获取固体低能晶格状态

的热处理过程。首先将固体加热，使其熔化为完全无序的液态，此时粒子处于自由运动状态；然后逐渐降温，粒子运动渐渐趋于有序，当温度降低到结晶温度时，粒子运动变为围绕晶格点的微小振动，液体就凝固成固态晶体。退火过程则要求降温过程足够缓慢，使得在每个温度下所有粒子随机排列都达到热平衡，最终达到固体的基态，即系统能量最小的状态。Kirkpatrick 充分利用固体退火原理提出了模拟退火算法，求解优化问题的模拟退火算法对固体退火过程的模拟^[44,45]如下表。

表 3.1 模拟退火算法相对于固体退火算法的模拟

Tab. 3.1 The simulation between simulated annealing algorithm and solid annealing

固体退火	模拟退火算法
粒子的一个状态 i	优化状态的一个解 x
能量最低的一个状态	最优解
系统的能量	目标函数
温度	控制参数
某一恒温状态下趋于热平衡	产生新解-判断-接受/舍弃

模拟退火算法的一般框架如下：

Step1: (初始化) 产生初始温度 t_0 和初始解。

Step2: 如果不满足终止条件，那么

(a) 如果不满足该温度下的内循环停止条件，那么

(1) 按给定方式产生新解，并计算适应函数值增量 ΔE ；

(2) 如果 $\Delta E \leq 0$ ，替代当前解；

(3) 如果 $\Delta E > 0$ ，按概率 $e^{\frac{-\Delta E}{k_b \times t}}$ 接受新解， k_b 是常数。

(b) 按某种准则降低温度 t ，并返回到 Step2。

从模拟退火算法的一般框架可以看出该算法是局部搜索算法的发展，以概率

$e^{\frac{-\Delta E}{k_b \times t}}$ 接受目标函数值大的解。使得算法能够以一定概率跳出局部极小点，这也保证了算法能够以一定概率收敛到全局最优点。理论上，当内循环达到平衡的时候，算法能够收敛到全局最优点。但是关于算法的理论探讨仍然需要进一步研究。

3.2 模拟退火算法参数

模拟退火算法给出的是一个框架性结构, 因此, 在求解优化问题时, 合理选取参数是一个非常重要的问题。本节将讨论实现模拟退火算法时经常遇见的一些技术问题。算法的技术问题一般包含两大类: 其一是解的表示和解的邻域^[46,47]。解的表现形式有多种, 比如二进制编码解、实数解等。合理的解的表现形式可以直接影响算法的收敛性和收敛速度, 解的表现形式不合理, 则可能导致下一个解不是可行解, 盲目搜索。解的邻域结构设置会直接影响算法是否能收敛到全局最优解, 易导致计算时间增加, 若解的邻域过小算法易收敛到局部最优。因此在选取解的表示形式和解的邻域时要和具体的问题相结合, 对问题的本质具有深入的了解, 才能恰当的选取。其二则是算法的结构参数的选取。

3.2.1 初始温度的选取

理论上, 接受新解的概率 $e^{\frac{-\Delta E}{k_b \times t}}$ 越大, 解的遍历性就增加。而影响接受新解的概率包括两方面, 一是初始温度 t_0 , 二是函数增量 ΔE 。从初始得温度看, 希望初始温度 t_0 越高越好, 但实际上温度的过高会浪费大量搜索时间, 即使选取非常高的初始温度, 也不一定能够保障解收敛到全局最优解, 只能增加收敛到最优解的概率。因此, 合理选取初始温度是非常重要的。常用的确定初始温度的方法有:

(1) Kirkpatrick 等于 1983 年提出了确定 t_0 值的选取规则, 给出一个初始温度 t_0 使得初始的接受概率 $e^{\frac{-\Delta E}{k_b \times t}}$ 大于一个给定的初始接收概率 λ_0 。由于受到函数增量的影响, 首先需要确定函数的增量 ΔE , 我们可以通过尝试法获得函数增量。确定函数增量的方法可采用如下做法: 随机给定 m 个初始解, 分别计算目标函数值和并对给定的 m 个初始解进行变化获得的新的目标函数值, 计算增长的目标函数值的增量的平均值 $\overline{\Delta E}$ 。然后用 $\overline{\Delta E}$ 代替接受概率 $e^{\frac{-\Delta E}{k_b \times t}}$ 中的 ΔE 。因此当给定了一个接受概率 λ_0 时, 就可以求出相应的初始温度 $t_0 = \ln(\frac{-\overline{\Delta E}}{k_b \lambda_0})$ 。这种做法可以从概率意义上保证 t_0 是一个较好的初始温度。

(2) 根据具体问题和主观的经验确定一个初始温度 t_0 。这种方法较简单，需要具有很好的模拟退火算法设计的经验和对于问题本质的认识。

综上所述，为使算法进程返回一个高质量的解，应该选取非常大的 t_0 。较高的 t_0 ，初始时的接受概率 $e^{\frac{-\Delta E}{k_b t}}$ 变大，提高了解的遍历性。但是过大的 t_0 又可能会增加过长的 CPU 时间，过高的 CPU 时间同样可能会使算法丧失可行性。因此，合理的选取 t_0 在模拟退火算法是非常重要的。

3.2.2 温度下降的方法

温度的下降函数一般要根据具体的问题而定，不同的问题可以采用不同的下降函数，常见函数形式一般有下面几种情况：

(1) $t_{k+1} = \alpha t_k$ ，其中 α 是介于0和1之间的一个常数，经验取值在[0.5, 0.99]之间。此方法比较直观，当希望温度下降比较慢的时候，可以采用比较大的 α ；希望温度下降的比较快的时候，可以采用相对较小的 α [48]。

(2) $t_{k+1} = \frac{K-k}{K} t_0$ ， $k=1,2,\dots,K$ 。 K 为事先给定的总的温度衰减次数。这个方法中，温度是以等间隔下降的。适合于以迭代次数为停止准则的模拟退火算法。

(3) $t_{k+1} = t_k * e^{-\max\{\Delta \bar{E}_k\}}$ ， $\max\{\Delta \bar{E}_k\}$ 是在 t_k 温度下函数的最大变化值。当函数值变化比较大，让温度下降的速度减小，可以增加跳出局部极小的概率；当函数值变化较小，让温度下降的速度增加，可以减少跳出局部极小的概率。

3.2.3 内循环长度的选取

在某一个温度 t_k 状态下，理论上要求内循环长度 L_i 的选取要满足产生的候选解数目趋于无穷大，使得内循环达到准平衡的状态（马尔科夫链达到平稳概率分布），显然，在实际求解问题时，算法是不能实现的。因此，采用了经验的方法：

(1) 固定的长度 L_i ，在每一个温度状态下，选取固定长度，使得迭代的步数相同。此时的 L_i 可以与问题规模有关，大小可以选取为 n ， n^2 等。

(2) 与温度相关的 L_i 选取，较高的温度状态下，解的接收概率会增加，因此

在高温的温度状态下可以选取小的 L_t 。当温度降低时，解的接收概率将减少，为避免陷入局部极小点，可以采取较大的 L_t 。

3.2.4 算法的停止准则

控制算法结束的方法有很多，常见的方法有：

(1) 根据温度的停止方法，当温度趋于零的时候，固体退火达到稳定状态，对应于函数寻优问题，可以认为寻找到了全局最优解。因此，随着温度趋于零，算法停止。

(2) 根据循环总次数的方法，为了便于计算模拟退火算法的复杂性和 CPU 时间，采用循环总次数是常数的方法控制算法停止。

(3) 根据最优值的精度，
$$e = \left| \frac{f^* - f_{\min}}{f_{\min}} \right|$$
是寻找到最优解的相对误差，当次误差趋于零的时候，算法终止。但是这个方法要求必须先知道算法的最优值 f^* 。而事实上很多问题并不能实现确定最优值。因此，可以采用相应改进的方法。当算

法的解在有限的迭代次数内没有大的变化时，算法终止，即
$$\left| \frac{f_L - f_{L+N}}{f_L} \right| \leq \varepsilon$$
，其中 N 为给定的有限步数。

3.3 改进的模拟退火算法

模拟退火算法提供了一个启发式的框架性结构。随着算法的广泛性应用，很多基于模拟退火算法的改进型被陆续提出。这些算法的改进在提高模拟退火算法的性能方面起了很大的推动作用。下面介绍几类改进的模拟退火算法。

3.3.1 增加记忆功能

模拟退火算法是一类启发式算法。理论上在当温度 t 收敛到零同时内循环步数 k 趋于无穷时，算法能够在概率收敛到全局最优解。但是在实际应用中，不能够满足 $t \rightarrow 0, k \rightarrow \infty$ 的要求。因此，在寻找最优解的时候，并不期望找到全局的最优解，而是希望在有限步数内，找到满意解。

模拟退火算法能够依照 Metropolis 准则接受一些大于当前值的解，因此可以增加算法的“爬坡”功能。但同时也增加了失去当前最好解的可能性。为了保证算

法搜索到的当前的最好解不被丢失。对模拟退火算法可以增加记忆功能。提出了记忆模拟退火算法。算法描述如下：

记忆模拟退火算法 (Memory Simulated Annealing, MSA) [49]；

Step1: 初始化温度 t_0 、变量。计算 $f(x)$ ，令 $x^* := x; f^* = f(x)$

Step2: 是否满足内循环的停止准则，满足，到 Step3；否则

随机产生新解 x_{new} ；

计算目标函数值 $f(x_{new})$ ；

if $f(x_{new}) < f^*$

then $x^* := x_{new}; f^* = f(x_{new})$

if $\Delta f = f(x_{new}) - f(x) < 0$ or $\exp(\frac{-\Delta f}{t}) > random$

then 接受新解 $x := x_{new}$ ；

step3: 计算下一个温度 t ；

step4: 是否满足终止准则，如果满足，则停止计算，输出结果；否则，返回 step2。

从算法中可以看出，在每次迭代过程中，用 x^*, f^* 纪录了当前最优解和最优值被。因此在算法的爬坡过程中当前的最优解并不能丢失。

3.3.2 模拟退火算法的新解产生

模拟退火算法针对不同问题采用不同的产生新解的方法，通常所认为的下一代的解是在上一代解的基础上产生的。此类方法减少了搜索的盲目性，可以有利于解的寻优。下面分两种情况讨论解的产生方式。

3.3.2.1 连续函数优化问题的解的生成

对于连续函数的问题，假设当前解为 x_k ，那么下一代的解为 x_{k+1} 。 x_{k+1} 是在 x_k 基础上产生的。经典的连续可微函数优化问题的新解的产生方式为 $x_{k+1} = x_k + \lambda^* dx$ ，其中， λ 为下降的步长， dx 为负梯度方向，即函数值下降的方向^[50]。但是由于函数是否可微不确定，因此在模拟退火算法中 dx 可以用一个随机向量代替。常见的有：

(1) 均匀分布产生新解

函数解的增量 dx 是一个服从均匀分布的向量。均匀分布的密度函数是

$$D(\xi) = \begin{cases} \frac{1}{b-a}, & \xi \in [a, b] \\ 0, & \xi \notin [a, b] \end{cases}$$

此方法认为在邻域内的每一个解的重要性都相同，没有考虑到函数的本身的特点，试验结果表明，采用此方法的收敛速度较慢。

(2) 正态分布产生新解

函数解的增量 dx 是一个服从正态分布的向量。正态分布的密度函数是

$$D(\xi) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\xi^2}{2\sigma^2}\right), \quad -\infty < \xi < \infty, \quad \sigma \text{ 为方差};$$

采用此种产生新解的方法被称为推广模拟退火算法 (Generated Simulated Annealing, GSA)，该算法表明靠近当前解 x_t 的 dx 被选中的概率大，有利于局部的搜索。

(3) 高斯分布产生新解

函数解的增量 dx 是一个服从高斯分布的向量。标准高斯分布的密度函数是

$$D(\xi) = \frac{1}{\pi} \frac{1}{1+\xi^2}, \quad -\infty < \xi < \infty$$

此方法被称为快速模拟退火算法 (Fast Simulated Annealing, FSA)，此方法具有很好的跳跃性和全局收敛性。

(4) 混沌产生新解

函数解的增量 dx 是一个混沌变量。常用的混沌变量是 Logistic 映射产生的变量。Logistic 映射为

$$x_{k+1} = 4x_k(1-x_k), \quad x_0 \in (0,1)$$

此方法被称为混沌模拟退火 (Chaos Simulated Annealing, CSA)，该算法在求解高维问题具有较好的收敛性。

3.3.2.2 用于离散问题产生新解的结构模式

由于离散问题的类别比较多，不同问题具有不同的解的结构。下面分别对不同的问题设计解的结构。

(1) 0-1 背包问题:

解的表现形式可以是：
$$x_i = \begin{cases} 0 & i\text{物体被选中} \\ 1 & i\text{物体不被选中} \end{cases}, \text{一个解是 } X = (x_1, x_2, \dots, x_m)。$$

(2) 旅行商问题：解的表现形式采用城市的排序，例如，五个城市的旅行商问题，城市为 A B C D E，解可以表示为 $x = (ABCDE)$ 。邻域结构可以由 2-opt 或 3-opt 的方法产生，那个新解可以在邻域中按照某种规则选取^[51]。

新解的产生有多种，比如利用概率的方式可以获得新解，即对于 x_i 在以某种概率方式进行 0 与 1 的变换后得到新解。

由于离散问题很多本节不一一论述。

3.4 模拟退火算法的应用领域

近年来，由于模拟退火算法的稳健性、简明性、全局性和广泛的可应用性，使得模拟退火算法被计算机科学、工程技术、管理科学和社会科学等领域的学者们广泛关注，攻克了一些确定性算法无法解决的非线性复杂的优化问题，并显示巨大的优势和潜力。这些成功的应用包括：

(1) 组合优化：模拟退火算法的最初应用领域。组合优化是研究那些含有有限个可行解的、日常生活中（尤其是工程设计中）大量存在的问题。这其中一个是重要的并且普遍的应用领域就是考虑如何有效利用稀缺的资源来提高生产力。虽然理论上这样问题的最优解可以通过简单的枚举得到，但实际上通常不能实现。特别对于实际规模的问题来说，可行解的数量可能特别巨大。改进的模拟退火算法可以较稳定的使这类问题达到全局最优，且具有较高的效率。

(2) 函数优化：最优化理论是一门实用性很强的学科，优化技术是最优化理论与方法的实际应用，优化问题又往往通过各种数学形式的函数来表示，而模拟退火算法无论是在 Rosenbrnck 函数还是在多峰优化函数等典型优化问题上都可以稳定的达到全局最优^[52]。

(3) 旅行商问题：模拟退火算法是解决旅行商问题的有效算法之一。其解的表现形式可以采用城市的排序，最终可以求得路径总长度最短的城市访问顺序。

(4) 调度问题：调度问题考虑的是如何随着时间的变化，调度有限的资源在执行任务的同时满足特定约束。调度问题几乎在现实环境中无处不再。例如许多制造工业提出的调度问题从本质上讲非常复杂，难以用传统优化方法求解。这些问题

属于 NP 类问题。模拟退火算法是处理这些问题的有效方法。

(5)制造元设计:成组技术中将机器和零部件组成制造单元的过程称为制造元设计,已经证明这是一个 NP—完全问题。应用进化搜索算法对制造元设计问题也作了许多尝试,而模拟退火算法是非常有效的进化探索方法。

(6)网络节点与路径:自从 20 世纪 80 年代以来,计算机网络得到了爆炸性的发展。计算机网络的节点与路径是构建和扩展计算机网络的重要内容之一。运用模拟退火算法可以解决网络系统中的一些问题,包括最短路径问题以及节点选择等问题。

3.5 本章小结

本章首先介绍了模拟退火算法的含义,并将其与固体退火过程相比较说明了模拟退火算法的原理;然后具体介绍了模拟退火算法各项参数的选取方法及其适用情况;其次给出了增加记忆功能的改进模拟退火算法,并详细阐述了算法新解产生的方法,为后续章节中算法的时间奠定了基础;最后总结了模拟退火算法的应用领域。

第4章 基于岸桥作业计划的单台龙门吊路径优化问题

4.1 单台龙门吊路径优化模型

4.1.1 结合岸桥作业计划的单台龙门吊路径优化问题

为了将一个集装箱装至船上，龙门吊移动到目标贝位将集装箱吊起，再移动到箱区一侧的目标集卡上方，将集装箱卸到集卡上。集卡将集装箱运至岸边后，岸壁集装箱装卸桥将集装箱吊起装到船上。

龙门吊的路径优化问题是指在满足岸壁集装箱装卸桥装卸顺序的要求下，龙门吊如何从一个贝位到另一个贝位，再到下一个贝位，依此下去完成装卸任务，来确定龙门吊访问的贝位顺序即龙门吊的路径，及每访问一次贝位所提取的集装箱数量，最终使龙门吊的总的行走路程最短。

要满足岸壁集装箱装卸桥装卸顺序要求，就要已知两个条件：一是根据装船过程中岸边机械装卸顺序，可以将同一尺寸、重量和目的港的集装箱的工作任务归为一类，称为子任务，然后给出个子任务的装卸顺序表，也是龙门吊的提取顺序单；另一条件是指堆场箱区集装箱的分布状态，即每一类集装箱（和岸桥装卸顺序的集装箱分类相同，同一尺寸、重量和目的港的集装箱为一类）在每一箱区的分布状态。子任务具体是指龙门吊在完成提取顺序单中每一次序的提取任务中所行走的路线。龙门吊从一个子任务有序的进入下一个子任务，当完成所有子任务时，将所有子任务中的行走路线、以及每两个子任务之间的路线连接起来即是龙门吊的最终行走路径。龙门吊的作业路径图如图4.1所示。其中S点、E点分别是虚拟的初始点和终点，代表龙门吊的初始位置和最终位置，它节点上的数字代表贝位号。

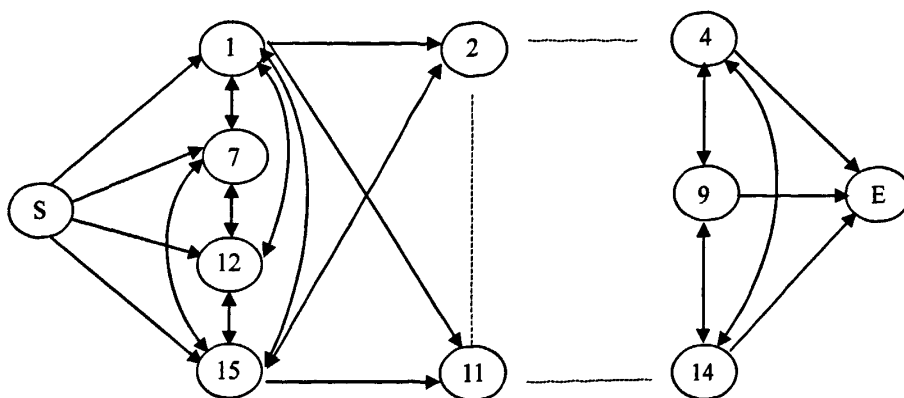


图 4.1 龙门吊作业路径图

Fig. 4.1 The operation roadmap of gantry crane

另外，对于本文的研究还基于以下条件：（1）龙门吊的工作顺序必须满足岸壁集装箱装卸桥装卸顺序；（2）对于每一类集装箱而言，从所有贝位所提取的该类集装箱总数必须等于堆场在初始状态下所堆存的该类集装箱总数，而且满足任务要求数量；（3）所有集装箱分布在同一箱区或是共线的相邻箱区，满足本文对轨道式龙门吊研究要求。

本章假设只有 1 台龙门吊作业。

4.1.2 符号表示

N : 堆场出口箱箱区中 Bay 的个数；

M : 龙门吊提取顺序单中子任务总数；

l : 集装箱类型的数量；

t : 龙门吊提取顺序单中的子任务编号, $t=1, 2, \dots, m$;

B : 堆场出口箱箱区中 Bay 组成的集合, $B=\{1, 2, \dots, n\}$;

G : 集装箱类型编号组成的集合, $G=\{1, 2, \dots, l\}$;

g_t : 在第 t 个子任务中龙门吊需作业的集装箱类型的编号；

r_t : 在第 t 个子任务中龙门吊需作业的集装箱的数量；

$b(h)$: 由存放 h 类型集装箱的 Bay 编号所组成的集合；

$s(h)$: 由需要提取 h 类型集装箱的子任务编号所组成的集合; $s(h)=\{t | g_t=h, t=1, \dots, m\}$;

c_{hj} :初始状态下, Bay j 拥有 h 类型集装箱的数量;

$d_{ij}=|i-j|$: Bay i 与 Bay j 之间的间隔倍数;

决策变量:

$x'_j \geq 0$, 表示龙门吊在第 t 个子任务中从 Bay j 提取的集装箱数量。

$y'_{ij} \in \{0,1\}$, 表示如果龙门吊完成第 t 个子任务后由 Bay i 进入 Bay j , 则等于 1, 否则等于 0, 其中, $i \in b(g_{t-1})$, $j \in b(g_t)$ 。

$z'_{ij} \in \{0,1\}$, 表示如果龙门吊在第 t 个子任务中由 Bay i 进入 Bay j , 则等于 1, 否则等于 0, 其中 $i, j \in b(g_t)$ 。

$y^0_{sj} = 1$, 表示龙门吊首先从 Bay j 开始。

$y^m_{jE} = 1$, 表示龙门吊在 Bay j 结束。

4.1.3 模型建立

$$\text{目标函数: } \min F = \sum_{t=0}^m \sum_{i,j \in B} d_{ij} y'_{ij} + \sum_{t=1}^m \sum_{i,j \in B} d_{ij} z'_{ij} \quad (4.1)$$

$$\text{约束条件: } \sum_{j \in b(g_1)} y^0_{sj} = 1 \quad (4.2)$$

$$- \sum_{j \in b(g_m)} y^m_{jE} = -1 \quad (4.3)$$

$$\sum_{j \in b(g_{t-1})} y^{t-1}_{ji} + \sum_{k \in b(g_t)} z^t_{ik} = \sum_{l \in b(g_{t+1})} y^t_{il} + \sum_{k \in b(g_t)} z^t_{ik}, \quad i \in b(g_t), \quad t=1, \dots, m \quad (4.4)$$

$$\sum_{i,j \in b(g_t)} z^t_{ij} \leq |N|-1, \quad N \subseteq b(g_t), \quad t=1, \dots, m \quad (4.5)$$

$$\sum_{i \in s(h)} x^t_j = c_{hj}, \quad h=1, \dots, p, \quad j \in b(h) \quad (4.6)$$

$$\sum_{j \in b(g_t)} x^t_j = r_t, \quad t=1, \dots, m \quad (4.7)$$

$$x^t_j \leq M * \left(\sum_{k \in b(g_t)} z^t_{kj} + \sum_{i \in b(g_{t-1})} y^{t-1}_{ij} \right), \quad j \in b(g_t), \quad t=1, \dots, m \quad (4.8)$$

$$y'_{ij} \in \{0,1\}, \quad i, j \in B, \quad t=1, \dots, m-1,$$

$$y^0_{sj} \in \{0,1\}, \quad j \in B,$$

$$y_{iE}^m \in \{0,1\}, i \in B,$$

$$z_{ij}^t \in \{0,1\}, i, j \in B, t=1, \dots, m,$$

$$x_j' \geq 0, j \in B, t=1, \dots, m.$$

目标函数 (4.1) 龙门吊行走的总路程最短, 其中 d_{ij} 表示 Bay i 与 Bay j 之间的间隔倍数, 由于堆场贝位之间的实际距离一般是定值, 所以本文直接简化用贝位之间的间隔表示距离不影响龙门吊的路径优化。对于约束条件, 式 (4.2) 和式 (4.3) 分别表示龙门吊从 S 点开始、到 E 点结束; 式 (4.4) 表示中间点的流入量必须等于流出量; 式 (4.5) 表示在每一子任务中避免回路, N 是集合 $b(g_t)$ 的子集, $|N|$ 表示集合 N 中组成元素的个数; 式 (4.6) 表示在所有子任务中从 Bay j 提取的集装箱总数应等于在 Bay j 上堆存的原始集装箱数量; 式 (4.7) 表示在每一子任务中提取的集装箱数量满足工作计划要求; 式 (4.8) 表示当 $x_j' > 0$ 时, 节点 j 一定有流入量, 即保证 x_j' 与流入量一致, 其中 M 是一个足够大的数。

4.2 算法设计

4.2.1 基本原理

模拟退火算法充分利用了固体退火原理: 将粒子的一个状态模拟成优化状态的一个解 x , 系统的能量相当于目标函数, 用控制参数表示温度变化, 当能量达到最低的状态时相当于函数的最优解, 而在某一恒温状态下趋于热平衡时, 用产生新解——判断——接受或是舍弃来模拟。模拟退火算法是局部搜索算法的扩展, 但是模拟退火算法是按一定的概率选择邻域中目标函数更优的状态, 可以使算法跳出局部极值点, 保证了算法的全局收敛性。另外根据实践经验, 模拟退火算法具有很强稳定性和高效性的特点, 所以本文选用模拟退火算法对数学模型进行求解。

4.2.2 单台龙门吊行走路径的确定

假设已知在第一个任务过程中需要提取第 I 类集装箱, 将堆放该类集装箱的 Bay 号组成集合, 且将 Bay 号由小到大排序组成矩阵的一行; 其次, 第二个任务中需要提取第 II 类集装箱, 再将堆放第 II 类集装箱的 Bay 号组成集合, 对 Bay 号

由小到大排序，接在第一个任务对应的 Bay 号的后面，组成新的一行；依此下去到第 m 个任务结束，得出一行满足任务顺序要求的 Bay 号，称为第一行。每一 Bay 位都对应一个所堆存的集装箱数量的原始数，将其按位置对应放在 Bay 号的下方，组成第二行。每一任务都有一个工作计划要求数，将这个数对应放在第三行。由上述三行，得到一个已知矩阵 a 。下面从初始矩阵 a 变化求解龙门吊的路径的可行解。

按任务顺序要求将矩阵 a 分成 m 个小模块，例如，对于第一个任务对应第一个小模块。由第一个小模块的列数组成一个 for 循环，判断第二行第一列数是否大于第三行的任务要求数，大于则将对应的 Bay 号和第三行的数记忆到解矩阵的第一列内，并将第二行内所有（即矩阵 a 内）具有相同 Bay 号的集装箱数更新，其次还要将第一小模块的第三行数重置成 0，跳出循环；如果小于且第二行的数不等于零，则将对应的 Bay 号和第二行的数记忆到解矩阵的第一列内，并将第二行内所有（即矩阵 a 内）具有相同 Bay 号的集装箱数更新，其次将模块的第三行任务数相应减少，然后进入下一循环。依次下去完成 m 个模块的搜索后，解矩阵内就记录了龙门吊依次访问的 Bay 号及相应提取的集装箱数量。由于初始矩阵 a 和求解过程的特殊设计，使得此解既满足任务顺序要求、计划任务量也避免了子回路等要求，可知此解是可行解。

4.2.3 算法流程

以下给出模拟退火算法的步骤，算法流程图如图 4.2 所示。

第一步，根据上一小节单台龙门吊路径的确定方法产生初始解，并给出初始温度。

第二步，如果不满足终止条件，那么

(1) 如果不满足该温度下内循环的终止条件，那么 (a) 按给定的方式产生新解，本文采用类似遗传算法中的变异方法，即对初始矩阵 a 随机选择可换位置的两列进行交换，产生新的矩阵 b ，在矩阵 b 的基础上再确定可行解，并计算适应函数值的增量 ΔE ；(b) 如果 $\Delta E \leq 0$ ，替代当前解看；(c) 如果 $\Delta E > 0$ ，按概率 $e^{\frac{-\Delta E}{k_b \cdot t}}$ 接受新解， k_b 是常数。

(2) 按 $t_{k+1} = \alpha t_k$ 的准则降低温度 t ，其中 α 是介于 0 和 1 之间的一个常数，

返回第二步。

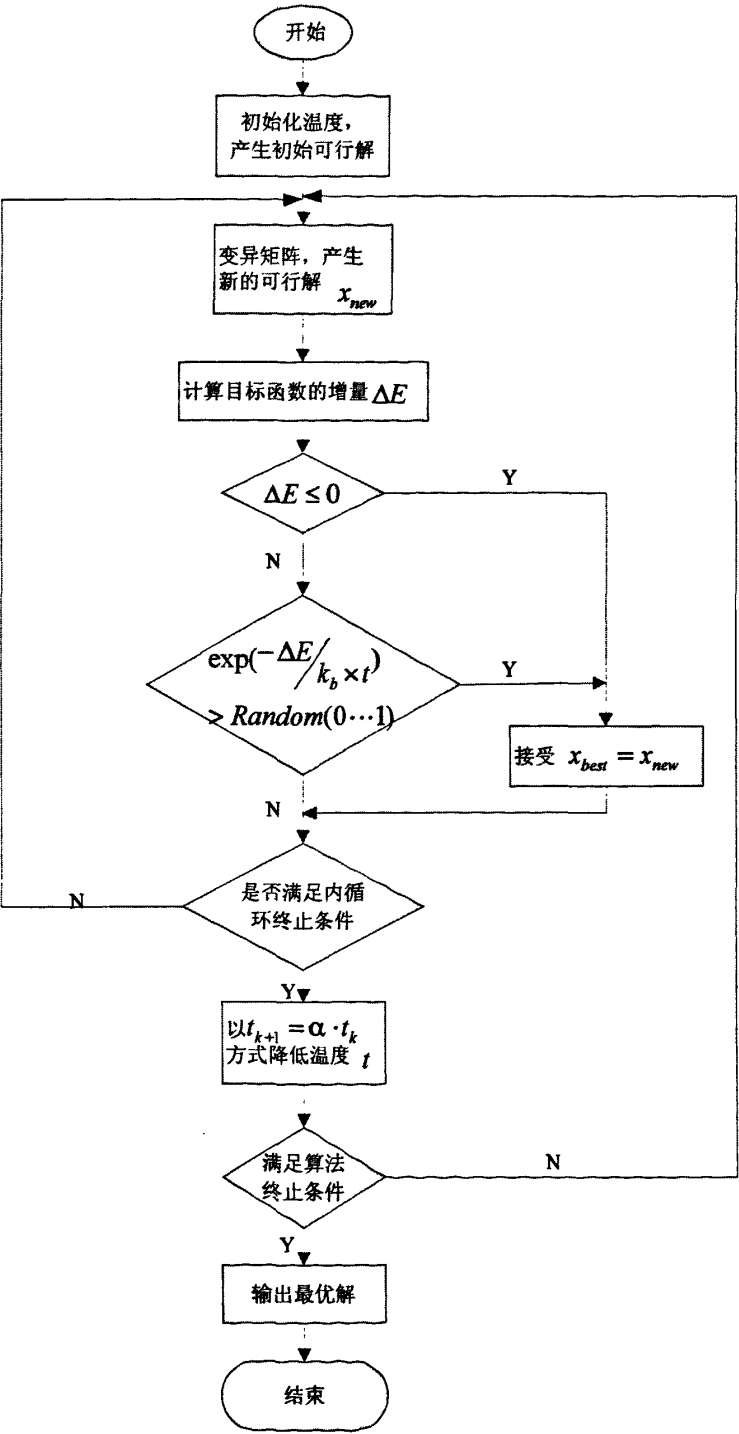


图 4.2 模拟退火算法示意图

Fig. 4.2 The sketch map of simulated annealing algorithm

4.3 模型求解

(1) 一台龙门吊从 15 个贝位上提取集装箱来完成任务要求。已知龙门吊的提取顺序单如表 4.1，堆场出口箱区集装箱的分布状态如表 4.2。

表 4.1 算例一的龙门吊提取顺序单

Tab. 4.1 The pick-up order of gantry crane of the first example

子任务标编号 t	1	2	3	4	5	6	7	8
箱子类型	I	II	I	III	II	I	II	III
箱子数量	20	15	14	25	16	18	12	15

表 4.2 算例一的堆场集装箱分布状态

Tab. 4.2 The distribution of containers in container yard of the first example

Bay	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
箱子类型	I	II		III	II		I	II	III		II	I		III	I
箱子数量	9	19		13	9		12	7	12		8	18		15	13

将已知条件代入本章模型，结合算法，利用 Matlab 软件对其进行编程求解。
首先给出本例题的模拟退火算法。

- (1) 根据单台龙门吊行走路径的确定方法，给出初始可行解；
- (2) 初始温度， $T_{\max}=100$ ；
- (3) 内循环次数， $L=100$ ；
- (4) 适应值函数为目标函数 F ；
- (5) 温度下降策略： $T_{\max}=0.95\times T_{\max}$ ；
- (6) 终止准则， $T_{\max}<10^{-4}$ 。

本章编程代码如附录 A。

利用模拟退火算法求得本例的最优解，即龙门吊的提取方案，用表 4.3 表示。

表 4.3 算例一单台龙门吊提取方案

Tab. 4.3 The pick-up programme of single gantry crane for the first example

	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	堆存量
Bay1			9						9
Bay2		15					4		19
Bay4				13					13
Bay5					8		1		9
Bay7	7		5						12
Bay8							7		7
Bay9				12					12
Bay11					8				8
Bay12						18			18
Bay14								15	15
Bay15	13								13
任务量	20	15	14	25	16	18	12	15	

其中贝位号是 0 时, 表示龙门吊没有从此贝位提取集装箱。由此可得龙门吊的行走路径是: (Bay15) → (Bay7) → (Bay2) → (Bay1) → (Bay7) → (Bay9) → (Bay4) → (Bay11) → (Bay5) → (Bay12) → (Bay8) → (Bay2) → (Bay5) → (Bay14), 行走路程是 49。龙门吊的行走路径示意图见图 4.3, 其中圆圈表示节点, 圆内数字表示贝位号, 一个灰色矩形表示一个子任务。另作说明的是, 优化结果路程内没有包括到虚拟初始点和终点的距离, 因为在实践中可以从优化路径的第一个贝位开始工作, 同时也没有必要回到终点, 所以结果是符合实际要求的。

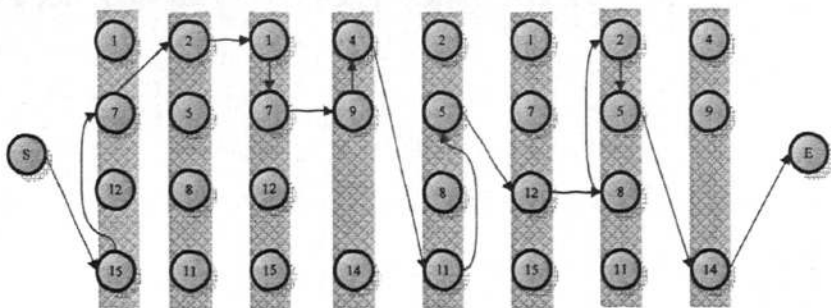


图 4.3 算例一单台龙门吊作业路径图

Fig. 4.3 The operation roadmap of single gantry crane for the first example

该值要好于前期文献给出的结果：(Bay1) → (Bay7) → (Bay2) → (Bay7) → (Bay15) → (Bay9) → (Bay4) → (Bay2) → (Bay5) → (Bay11) → (Bay12) → (Bay11) → (Bay8) → (Bay14)，行走路程是 57^[53]。

本章算例函数收敛图像如图 4.4 所示。

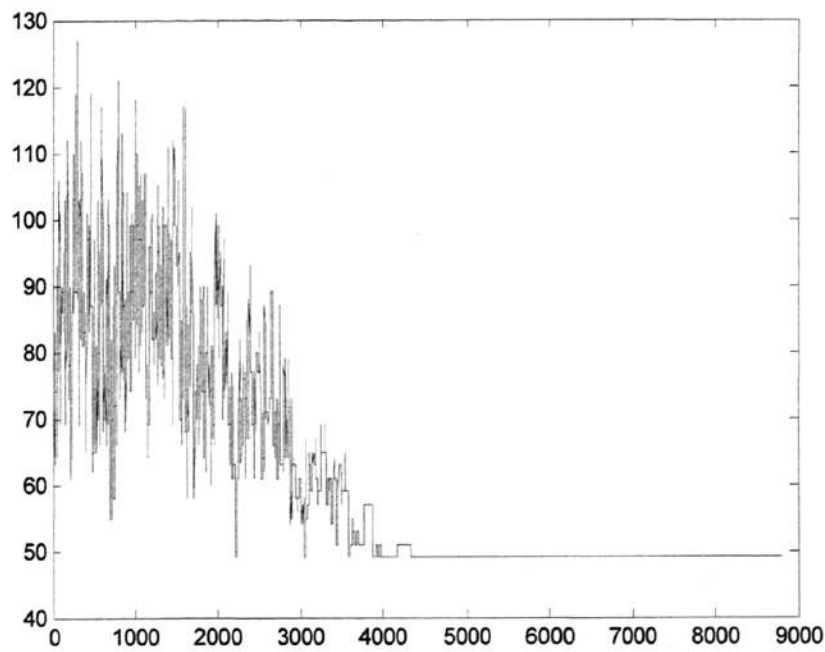


图 4.4 单台龙门吊算例一目标函数收敛图

Fig. 4.4 The convergence diagram of object function for the first example

(2) 一台龙门吊从 25 个贝位上提取集装箱来完成任务要求。已知龙门吊的提取顺序单如表 4.4，堆场出口箱区集装箱的分布状态如表 4.5。

表 4.4 算例二的龙门吊提取顺序单

Tab. 4.4 The pick-up order of gantry crane of the second example

子任务标编号 t	1	2	3	4	5	6	7	8	9
箱子类型	I	II	III	I	IV	II	I	V	III
箱子数量	25	15	13	7	21	19	16	17	23

表 4.5 堆场集装箱分布状态

Tab. 4.5 The distribution of containers in container yard of the second example

Bay	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
类					I	I	II		I	V	IV	IV	II			V	III		II		III	III			
型																									
数					18	21	16		9	7	15	6	5			10	7		13		20	9			
量																									

将已知条件代入本章模型，调整模拟退火方法的参数，利用 Matlab 编程求解，得到算例二龙门吊的提取方案如表 4.6 所示，且龙门吊的行走路径是（Bay5）→（Bay6）→（Bay7）→（Bay17）→（Bay22）→（Bay9）→（Bay11）→（Bay12）→（Bay13）→（Bay19）→（Bay7）→（Bay6）→（Bay9）→（Bay10）→（Bay16）→（Bay21）→（Bay22），目标函数值即龙门吊的行走路程是 67。龙门吊的行走路线图如图 4.5 所示。

表 4.6 算例二单台龙门吊的提取方案

Tab. 4.6 The pick-up program of single gantry crane for the second example

	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	堆存量
Bay5	18									18
Bay6	7						14			21
Bay7		15				1				16
Bay9				7			2			9
Bay10								7		7
Bay11					15					15
Bay12					6					6
Bay13						5				5
Bay16								10		10
Bay17			7							7
Bay19						13				13
Bay21									20	20
Bay22				6					3	9

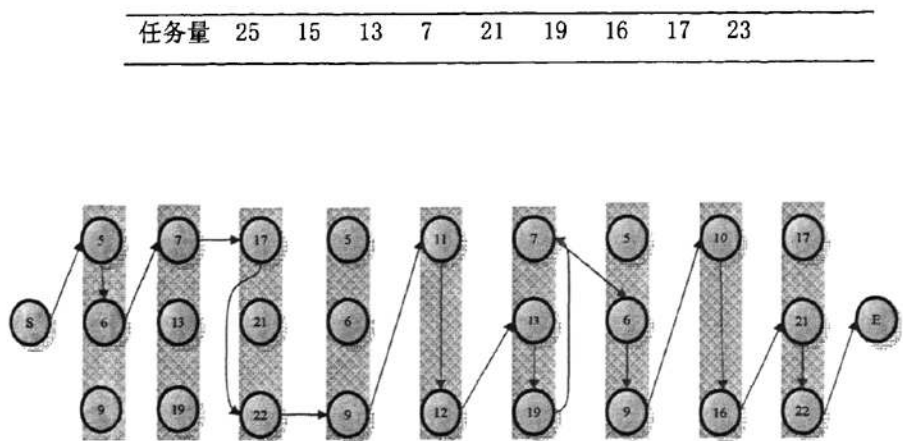


图 4.5 算例二单台龙门吊作业路径图

Fig. 4.5 The operation roadmap of single gantry crane for the second example

目标函数的收敛状况如图 4.4 所示。

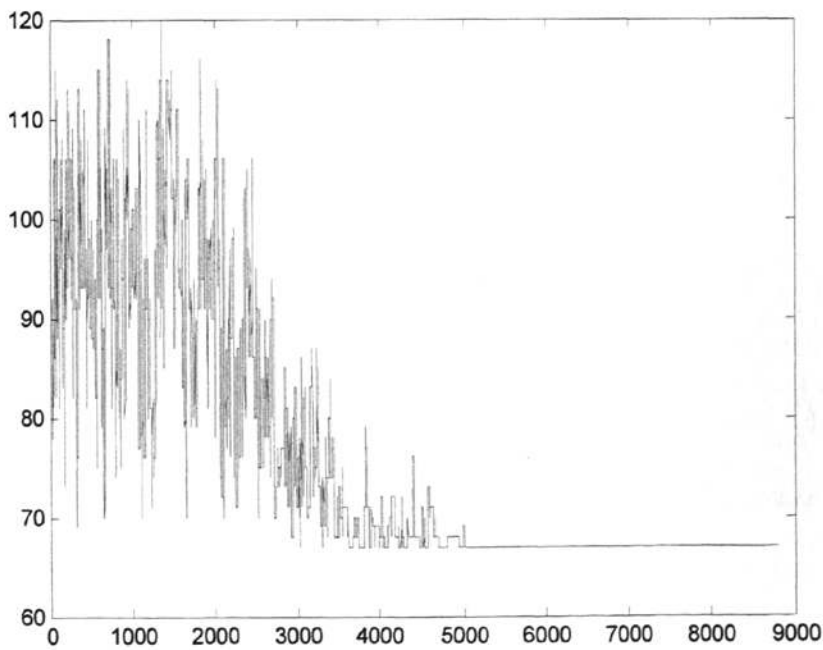


图 4.4 单台龙门吊算例二目标函数收敛图

Fig. 4.4 The convergence diagram of object function for the second example

从数值结果看，本章算法具有较好的收敛性，能够收敛到全局最优点，也验

证了单台龙门吊模型的合理性。

算例一和算例二的主要区别之一是集装箱堆场贝位数量的不同，相差十个贝位。从收敛结果可以看出，无论对于具有少量贝位的堆场箱区还是对于具有较多贝位堆场箱区，本章所设计的算法都具有很好的收敛性。另一区别在于集装箱种类数量由三种升至五种，同样得到了很好的优化结果，使得模型验证更趋于实际情况，可为后续研究提供更好的参考价值。

4.4 本章小结

本章首先描述了结合岸桥装卸要求的单台龙门吊路径优化问题，建立了该优化问题的数学模型；然后设计了单台龙门吊行走路径确定策略，并将其策略应用到堆场龙门吊行走路径的求解算法中，给出了求解该模型的模拟退火算法；最后给出两个不同的具体算例和求解结果，证明了本章模型的合理性和算法的有效性。

第5章 满足岸桥装卸要求的多台龙门吊路径优化问题

5.1 多台龙门吊路径优化模型

5.1.1 多台龙门吊路径优化问题

问题同上一章单台龙门吊的路径优化问题的理论基础类似，只是现实实践中为了加快装卸速度，缩短船舶的在港时间，一个贝位一般配备多台龙门吊同时进行作业。所以本章所要研究是在满足岸壁集装箱装卸桥装卸顺序的要求下，两台龙门吊分别如何从一个贝位到另一个贝位，再到下一个贝位，依此下去完成装卸任务，来确定两台龙门吊访问的贝位顺序即各自龙的路径，及每访问一次贝位所提取的集装箱数量，最终使两台龙门吊的总的行走路程最短。

类似上一章要满足岸壁集装箱装卸桥装卸顺序要求，龙门吊的提箱顺序单及堆场箱区的分布状态已知。本章的路径优化具体是指两台龙门吊分别从不同的贝位开始作业，然后按照提箱顺序单从一个子任务有序的进入下一个子任务，而且两台龙门吊式对同一个任务同时作业，只有当上一个任务全部完成后，龙门吊才能进入下一个子任务。当所有子任务都完成后，将两台龙门吊访问的贝位分别连接起来，即是它们各自的行走路径。当然两台龙门吊（分别称做 a 龙门吊、b 龙门吊）的路径优化的一个前提是：任一台龙门吊都不能越过另一台龙门吊提取集装箱。两台龙门吊的作业路径也类似上一章图 4.1，只是对于 a 龙门吊从 S 点开始又回到 S 点，b 龙门吊从 E 点开始又回到 E 点。而且在实际计算过程中，对于按任务顺序要求排序的贝位，只需让 a 龙门吊从箱区一个方向开始，b 龙门吊从箱区另一方向开始，待完成所有任务后都无需回到初始点。

5.1.2 符号表示

N: 堆场出口箱箱区中 Bay 的个数;

M: 龙门吊提取顺序单中子任务总数;

l: 集装箱类型的数量;

t: 龙门吊提取顺序单中的子任务编号, $t=1, 2, \dots, m$;

B: 堆场出口箱箱区中 Bay 组成的集合, $B=\{1, 2, \dots, n\}$;

G: 集装箱类型编号组成的集合, $G=\{1, 2, \dots, l\}$;

g_t : 在第 t 个子任务中龙门吊需作业的集装箱类型的编号;

r_t : 在第 t 个子任务中龙门吊需作业的集装箱的数量;

$b(h)$: 由存放 h 类型集装箱的 Bay 编号所组成的集合;

$s(h)$: 由需要提取 h 类型集装箱的子任务编号所组成的集合, $s(h)=\{t|g_t=h, t=1, \dots, m\}$;

c_{hj} : 初始状态下, Bay j 拥有 h 类型集装箱的数量;

$d_{ij}=|i-j|$: Bay i 与 Bay j 之间的间隔倍数;

决策变量:

$x_j^{ta} \geq 0$, 表示 a 龙门吊在第 t 个子任务中从 Bay j 提取的集装箱数量。

$x_j^{tb} \geq 0$, 表示 b 龙门吊在第 t 个子任务中从 Bay j 提取的集装箱数量。

$y_{ij}^{ta} \in \{0,1\}$, 表示如果 a 龙门吊完成第 t 个子任务后由 Bay i 进入 Bay j , 则等于 1, 否则等于 0, 其中, $i \in b(g_{t-1})$, $j \in b(g_t)$ 。

$y_{ij}^{tb} \in \{0,1\}$, 表示如果 b 龙门吊完成第 t 个子任务后由 Bay i 进入 Bay j , 则等于 1, 否则等于 0, 其中, $i \in b(g_{t-1})$, $j \in b(g_t)$ 。

$z_{ij}^{ta} \in \{0,1\}$, 表示如果 a 龙门吊在第 t 个子任务中由 Bay i 进入 Bay j , 则等于 1, 否则等于 0, 其中 $i, j \in b(g_t)$ 。

$z_{ij}^{tb} \in \{0,1\}$, 表示如果 b 龙门吊在第 t 个子任务中由 Bay i 进入 Bay j , 则等于 1, 否则等于 0, 其中 $i, j \in b(g_t)$ 。

$y_S^{0a} = 1$, 表示 a 龙门吊首先从虚拟 Bay S 进入 Bay j 开始工作。

$y_{jS}^{ma} = 1$, 表示 a 龙门吊从 Bay j 回到虚拟 Bay S 结束工作。

$y_{Ej}^{0b} = 1$, 表示 b 龙门吊首先从虚拟 Bay E 进入 Bay j 开始工作。

$y_{jE}^{mb} = 1$, 表示 b 龙门吊从 Bay j 回到虚拟 Bay E 结束工作。

5.1.3 模型建立

$$\text{目标函数: } \min F = \sum_{t=0}^m \sum_{i,j \in B} d_{ij} (y_{ij}^{ta} + y_{ij}^{tb}) + \sum_{t=1}^m \sum_{i,j \in B} d_{ij} (z_{ij}^{ta} + z_{ij}^{tb}) \quad (5.1)$$

$$\text{约束条件: } \sum_{j \in b(g_t)} y_S^{0a} = 1 \quad (5.2)$$

$$-\sum_{j \in b(g_m)} y_{jS}^{ma} = -1 \quad (5.3)$$

$$\sum_{j \in b(g_1)} y_{Ej}^{0b} = 1 \quad (5.4)$$

$$-\sum_{j \in b(g_m)} y_{jE}^{mb} = -1 \quad (5.5)$$

$$\sum_{j \in b(g_{t-1})} y_{ji}^{(t-1)a} + \sum_{k \in b(g_t)} z_{ki}^{ta} = \sum_{l \in b(g_{t+1})} y_{il}^{ta} + \sum_{k \in b(g_t)} z_{ik}^{ta}, \quad i \in b(g_t), \quad t=1, \dots, m \quad (5.6)$$

$$\sum_{j \in b(g_{t-1})} y_{ji}^{(t-1)b} + \sum_{k \in b(g_t)} z_{ki}^{tb} = \sum_{l \in b(g_{t+1})} y_{il}^{tb} + \sum_{k \in b(g_t)} z_{ik}^{tb}, \quad i \in b(g_t), \quad t=1, \dots, m \quad (5.7)$$

$$\sum_{i,j \in b(g_t)} z_{ij}^{ta} \leq |N|-1, \quad N \subseteq b(g_t), \quad t=1, \dots, m \quad (5.8)$$

$$\sum_{i,j \in b(g_t)} z_{ij}^{tb} \leq |N|-1, \quad N \subseteq b(g_t), \quad t=1, \dots, m \quad (5.9)$$

$$x_j^{ta} \leq M * (\sum_{k \in b(g_t)} z_{kj}^{ta} + \sum_{i \in b(g_{t-1})} y_{ij}^{(t-1)a}), \quad j \in b(g_t), \quad t=1, \dots, m \quad (5.10)$$

$$x_j^{tb} \leq M * (\sum_{k \in b(g_t)} z_{kj}^{tb} + \sum_{i \in b(g_{t-1})} y_{ij}^{(t-1)b}), \quad j \in b(g_t), \quad t=1, \dots, m \quad (5.11)$$

$$\sum_{j \in b(g_t)} x_j^{ta} + \sum_{\substack{q \in b(g_t) \\ q > j}} x_q^{tb} = r_t, \quad t=1, \dots, m, \quad (5.12)$$

$$\sum_{t \in s(h)} (x_j^{ta} + x_j^{tb}) = c_{hj}, \quad h=1, \dots, p, \quad j \in b(h) \quad (5.13)$$

$$y_{ij}^{ta} \in \{0,1\}, \quad i, j \in B, \quad t=1, \dots, m-1,$$

$$y_{ij}^{tb} \in \{0,1\}, \quad i, j \in B, \quad t=1, \dots, m-1,$$

$$y_{Sj}^{0a} \in \{0,1\}, \quad j \in B,$$

$$y_{iS}^{ma} \in \{0,1\}, \quad i \in B,$$

$$y_{Ej}^{0b} \in \{0,1\}, \quad j \in B,$$

$$y_{iE}^{mb} \in \{0,1\}, \quad i \in B,$$

$$z_{ij}^{ta} \in \{0,1\}, \quad i, j \in B, \quad t=1, \dots, m,$$

$$z_{ij}^{tb} \in \{0,1\}, \quad i, j \in B, \quad t=1, \dots, m,$$

$$x_j^{ta} \geq 0, \quad j \in B, \quad t=1, \dots, m,$$

$$x_j^{ab} \geq 0, \quad j \in B, \quad t = 1, \dots, m。$$

目标函数(5.1)表示两台龙门吊总的行走总路程最短。对于约束条件,式(5.2)和式(5.3)分别表示 a 龙门吊从 S 点开始、到 S 点结束;式(5.4)和式(5.5)分别表示 b 龙门吊从 E 点开始、到 E 点结束;式(5.6)和式(5.7)分别表示两台龙门的中间点的流入量分别等于其流出量;式(5.8)式和(5.9)分别表示 a、b 龙门吊在在每一子任务中都要避免回路, N 是集合 $b(g_t)$ 的子集, $|N|$ 表示集合 N 中组成元素的个数;式(5.10)表示当 $x_j^{ab} > 0$ 时, 节点 j 一定有流入量;式(5.11)表示当 $x_j^{ab} > 0$ 时, 节点 j 一定有流入量, 其中 M 是一个足够大的数;式(5.12)表示两台龙门吊在所有子任务中从 Bay j 提取的集装箱总数应等于在 Bay j 上堆存的原始集装箱数量;式(5.13)表示在每一子任务中两台龙门吊提取的集装箱总数满足工作计划要求。

5.2 算法设计

5.2.1 多台龙门吊可行路径的确定

本节只以两台龙门吊的路径为研究对象。首先同确定单台龙门吊的可行路径方法, 先根据已知条件生成一个初始矩阵 a , 然后对矩阵 a 变化求解两台龙门吊路径的可行解。

首先给 a 龙门吊一个初始贝位 1, 给 b 龙门吊一个初始贝位 n , 并按任务顺序要求将矩阵 a 分成 m 个小模块, 如第一个 for 循环是基于第一个小模块的所有列数进行的操作。仍以第一个 for 循环为例, 判断第二行第一列的数是否大于第三行的任务要求数, 大于则继续判断新遍历到的贝位号和 a 龙门吊及 b 龙门吊的初始贝位号的距离差, 然后将对应的 Bay 号和第三行的数记忆到距离差更小的龙门吊的解矩阵第一列内, 并将第二行内所有(即矩阵 a 内)具有相同 Bay 号的集装箱数以及移动过的龙门吊的贝位位置更新, 其次还要将第一小模块的第三行数重置成 0, 跳出循环; 如果小于且第二行的数不等于零时, 继续判断新遍历到的贝位号和 a 龙门吊及 b 龙门吊的初始贝位号的距离差, 然后将对应的 Bay 号和第二行的数记忆到距离差更小的龙门吊的解矩阵第一列内, 并将第二行内所有(即矩阵 a 内)具有相同 Bay 号的集装箱数以及移动过的龙门吊的贝位位置更新, 其次将模块的

第三行任务数相应减少（第三行的数减去第二行的数），进入下一循环。依此下去完成 m 个模块的搜索后， a 、 b 龙门吊的解矩阵内就记录了其依次访问的 Bay 号及相应提取的集装箱数量。由于初始矩阵 a 和求解过程的特殊设计，使得此解满足任务顺序、计划任务量及两台龙门吊不能互相越过贝位取箱等要求，同时也避免了子回路，可知此解是可行解。

5.2.2 算法流程

本章模型仍采用模拟退火算法求解，第一步是根据本章设计的多台龙门吊路径的确定方法产生初始解，并给出初始温度。第二步之后的算法类似 4.2.3 节，算法流程图也大致相似，故本章不再赘述。

5.3 模型求解

(1) 为了比较分析优化结果，本节算例数据和 4.3.1 小节是同一数据，不同的是两台龙门吊从 15 个贝位上提取集装箱来完成任务要求。提取顺序单及堆场出口箱区集装箱的分布状态分别见表 4.1 和表 4.2。

将已知条件代入本章模型，结合设计的算法，利用 Matlab 对软件其进行编程求解。

首先要给出本例题的模拟退火算法。

- (1) 根据多台龙门吊行走路径的确定方法，给出初始可行解；
- (2) 初始温度， $T_{\max} = 150$ ；
- (3) 内循环次数， $L = 150$ ；
- (4) 适应值函数为目标函数 F ；
- (5) 温度下降策略： $T_{\max} = 0.95 \times T_{\max}$ ；
- (6) 终止准则， $T_{\max} < 10^{-4}$ 。

本章编程代码如附录 B。

利用模拟退火算法优化本章算例，两台龙门吊的提取方案如表 5.1。

表 5.1 算例一两台龙门吊的提取方案

Tab. 5.1 The pick-up programme of double gantry cranes for the first example

t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	堆存量
-----	-----	-----	-----	-----	-----	-----	-----	-----

Bay1						9			9
Bay2					7		12		19
Bay4			13						13
Bay5					9				9
Bay7	12								12
Bay8		7							7
Bay9				12					12
Bay11		8							8
Bay12	8		10						18
Bay14								15	15
Bay15			4			9			13
任务量	20	15	14	25	16	18	12	15	

需要说明的是对于没有集装箱堆存的贝位，此表没再显示。另外表中蓝色数字表示 a 龙门吊在每个任务中相应贝位提取的集装箱数量，黑色粗体数字表示 b 龙门吊在每个任务中相应贝位提取的集装箱数量。由此可得 a 龙门吊的行走路径是：(Bay7) → (Bay8) → (Bay9) → (Bay4) → (Bay5) → (Bay2) → (Bay1) → (Bay2)，行走路程是 13；b 龙门吊的行走路径是：(Bay12) → (Bay11) → (Bay12) → (Bay15) → (Bay15) → (Bay14)，行走路程是 6。行走路线如图 5.1 所示，其中蓝色表示 a 龙门吊的路线，黑色表示 b 龙门吊的行走路线，另同上一章，在实际操作过程中龙门吊可以在直接在最优路径的第一贝位开始工作，同时也没有必要回到初始贝位，所以本例所计算的路程也没有包括到模型中虚拟的初始点和终点的距离，图中虚线只是路径设计需要。故目标函数是 19，比单台龙门吊的结果优化很多。

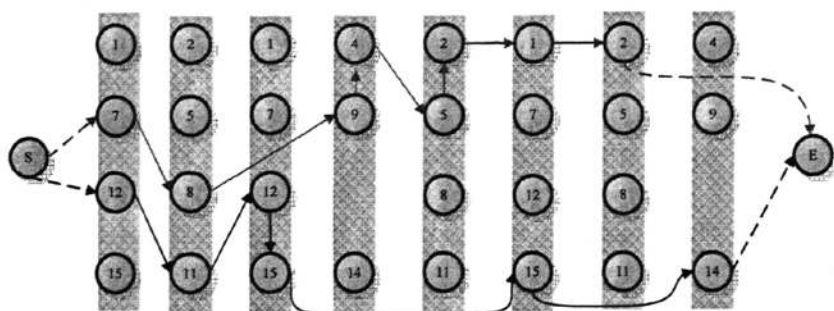


图 5.1 算例一两台龙门吊作业路径图

Fig. 5.1 The operation roadmap of double gantry cranes for the first example

本章算例一函数收敛图像如图 5.2 所示。

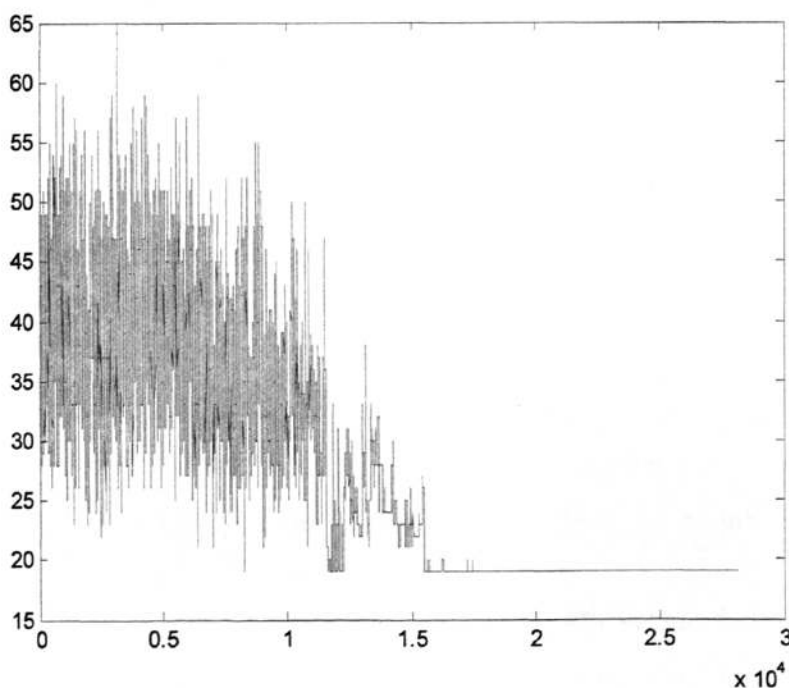


图 5.2 算例一中多台龙门吊的目标函数收敛图

Fig. 5.2 The convergence diagram of object function for two gantry cranes of the first example

(2) 同样为了对比分析优化结果，本节采用与 4.3.2 小节同样的算例数据不同的是两台龙门吊同时从 25 个贝位上提取集装箱来完成任务要求。提取顺序单及

堆场出口箱区集装箱的分布状态分别见表 4.4 和表 4.5。

将已知条件代入本章模型，不断调整模拟退火方法的各项参数，利用 Matlab 软件编程求解，得到算例二两台龙门吊的提取方案如表 5.2 所示，其中蓝色字体数字表示 a 龙门吊在每个任务过程中每访问一个贝位所提取的集装箱数量，黑色粗体数字表示 b 龙门吊在每个任务过程中每访问一个贝位所提取的集装箱数量。并且有 a 龙门吊的行走路径是 (Bay5) → (Bay6) → (Bay7) → (Bay9) → (Bay11) → (Bay12) → (Bay13) → (Bay7) → (Bay9) → (Bay6) → (Bay10)，b 龙门吊的行走路径是 (Bay22) → (Bay21) → (Bay19) → (Bay16) → (Bay17) → (Bay21)，目标函数值是 30。本例龙门吊的作业路径图如图 5.3 所示，其中蓝色表示 a 龙门吊的行走路线，黑色表示 b 龙门吊的行走路线，虚线表示虚拟的路径。

表 5.2 算例二两台龙门吊的提取方案

Tab. 5.2 The pick-up program of double gantry cranes for the second example

	t=1	t=2	t=3	t=4	t=5	t=6	t=7	t=8	t=9	堆存量
Bay5	18									18
Bay6	7						14			21
Bay7		15				1				16
Bay9				7			2			9
Bay10								7		7
Bay11					15					15
Bay12					6					6
Bay13						5				5
Bay16								10		10
Bay17									7	7
Bay19						13				13
Bay21			4						16	20
Bay22			9							9
任务量	25	15	13	7	21	19	16	17	23	

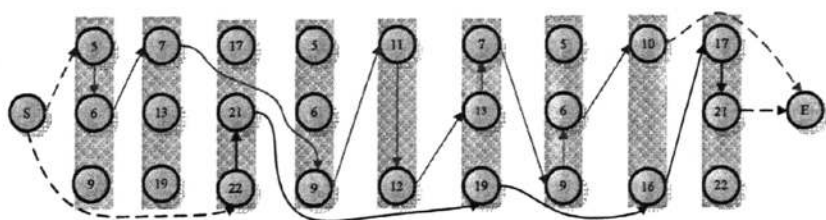


图 5.3 算例二两台龙门吊作业路径图

Fig. 5.3 The operation roadmap of double gantry cranes for the second example

目标函数的收敛状况如图 5.4 所示。

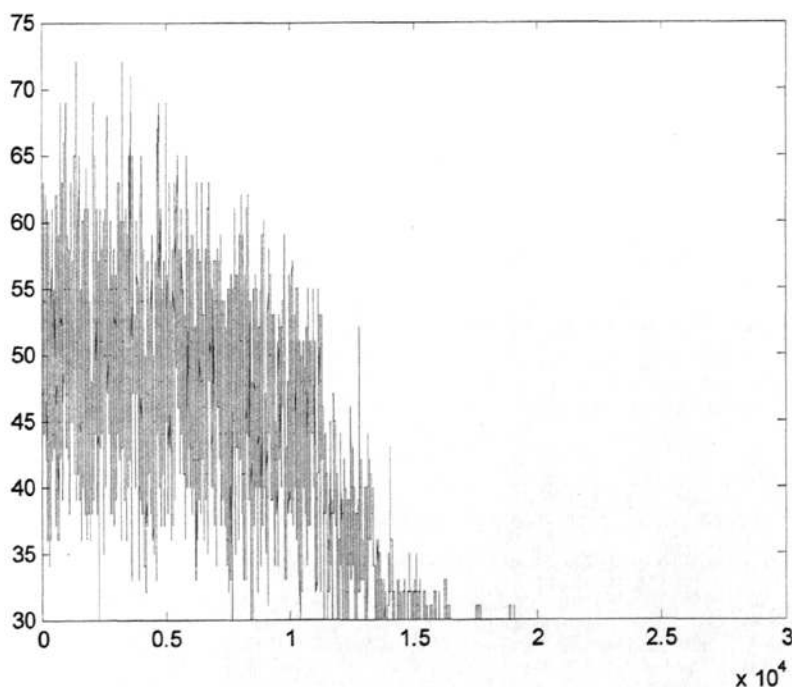


图 5.4 算例二中多台龙门吊的目标函数收敛图

Fig. 5.4 The convergence diagram of object function for two gantry cranes of the second example

从本章数值结果及收敛图看，算法具有较好的收敛性，可以稳定的收敛到全局最优点，同时也验证了本章模型的合理性。

本章算例都选取与第三章同样的已知数据，为的是证明本文所研究的两台龙门吊模型是合理的，而且如果不考虑龙门吊供不应求的情况，两台龙门吊同时作业一个箱区，可以更节约成本。另外算例二比算例一也多出了十个贝位、两种集

装箱，为的是说明本章多台龙门吊的模型也适合于多贝位、多种集装箱种类的情况，而且比单台龙门吊所获得的结果要优化的多。

5.4 本章小结

本章首先介绍了多台龙门吊的路径优化问题，建立了该优化问题的数学模型；然后结合各种约束条件设计了两台龙门吊的可行路径，并运用模拟退火算法调整各项参数后给出了求解该模型的算法流程；最后将模型和算法应用到两个不同的实际算例中，得到了很好的收敛结果后对算例进行比较分析，以验证本章多头龙门吊模型的有效性，为提高龙门吊乃至整个堆场的作业效率提供了一定的参考依据。

第6章 结论与展望

本文以集装箱码头堆场龙门吊为研究对象,主要研究了堆场单台龙门吊的路径优化以及多台龙门吊同时作业的路径优化问题,以最小化龙门吊的行走路径为目标,行了建模与优化。在优化方法上本文采用了模拟退火算法,并分别设计了单台龙门吊的以及多台龙门吊的可行路径,在设置不同参数后编程求解,然后通过实例证明了算法的有效性,实现了全局优化。集装箱堆场的调度人员可依据此结论对系统进行研究,以便合理配置龙门吊,安排工作计划,降低码头运营成本,提高其服务水平和竞争力。

本文主要工作如下:

(1) 在单台龙门吊的路径优化问题中,建立了优化模型,设计了求解单台龙门吊可行路径的方法,然后利用模拟退火算法实现了单台龙门吊的路径优化。

(2) 在多台龙门吊的路径优化问题中,建立了两台龙门吊的优化模型,给出了确定两台龙门吊可行路径的方法,然后调整模拟退火算法的各项参数,实现了多台龙门吊的路径优化。

(3) 针对龙门吊的路径优化问题,本文创新性的设计模拟退火算法求解,其结果的全局收敛,不仅证明了模型的合理性,还验证了其算法的高效稳定性。

论文不足之处有:对于算例中参数的确定上经验性较大,集装箱堆场的箱子堆放不是一个贝位至堆放一类集装箱,实际中更具有多变性,而且在整个龙门吊的工作过程中需要更全面地考虑堆场的某些特殊情况。

在本课题研究的基础上,下一步研究工作将沿着以下几个方面展开:

(1) 将贝位上堆存的箱子类型以及龙门吊每一子任务中的箱子类型扩展到多种箱子,更接近实际的堆存状况以及龙门吊的作业情况。

(2) 把龙门吊的作业范围扩展到整个堆场区域,对堆场系统内的龙门吊进行同时优化,实现整个龙门吊资源的有效利用以及系统整体成本最优。

(3) 结合时间限制要求对龙门吊的作业路径进行研究,即满足不同的箱子在不同的时刻之前必须提取,使龙门吊的路径动态化,这样更能真实反映龙门吊的工作特点。

参考文献

- [1]Richard Linn., Chuqian Zhang. A Heuristic for Dynamic Yard Crane Deployment in a Container Terminal. IIE Transactions, 2000, February 12:2-4.
- [2]K. Amborski, Optimization of container storage place operation, Modelling, Simulation&control C: Environmental, Biomedical, Human&Social Systems, 1984, 1: 1-8.
- [3]R. Mosca, P. Giribone, A. G. Bruzzone, Simulation of dock management and planning in a port terminal, International Journal of Modelling and Simulation, 2000, 20:153-157.
- [4]Kozan, Preston. Genetic algorithm to schedule container transfer at multimodal terminals. International Transaction in Operational Research, 1999, 6:311-328.
- [5]L. M. GAMBARDILLA, M. MASTEROLILLI, A. E. RIZZOLI and M. ZAFFALON An optimization methodology for intermodal terminal management Journal of intelligent manufacturing, 12, 521-534.
- [6]徐亚, 陈双秋, 龙磊等. 集装箱倒箱问题的启发式算法研究. 系统仿真学报, 2008, 20:3666-2647.
- [7]L. Yu, L. Xi, Z. Lu, Optimal model and its heuristic algorithm research of container yard planning, Proceeding, 2007, October, Egypt, p. p. 1415-1425.
- [8]计三有, 高悦文. 集装箱堆场减少倒箱率方法的研究. 水运工程, 2006, 8:28-29.
- [9]L. M. Gambardella, M. Mastrolilli, A. E. Rozzoli, An optimization methodology for intermodal terminal management, Journal of Intelligent Manufacturing, 2001, 12:521-534.
- [10]E. Kozan, P. preston, Genetic algorithm to schedule container transfers at multimodal terminals, International Transactions in Operational Research, 1999, 6:311-329.
- [11]Böse J., Vehicle dispatching at seaport container terminals using evolutionary algorithms, Proceeding of 33th Hawaii International Conference of System Sciences, Hawaii, 2000:33-36.
- [12]Peterkofsky R I, Daganzo C F, A branch and bound solution method for the crane scheduling problem, Transportation Research B, 1990, 24(3):159-172.
- [13]陈璐等, 集装箱码头装卸作业的调度控制模型及算法设计, 控制理论与应用, 2006, 23:874-878.
- [14]蔡芸, 同时求解泊位分配及岸桥调度问题的仿真优化方法, 可持续发展的中国:2003,

11:22-26.

[15]韩骏, 孙晓娜, 靳志宏. 集装箱码头泊位与岸桥协调调度优化. 大连海事大学学报, 2008, 34 (2): 117-121.

[16]Vis, I F A, Koster R, Roodbergen K J, Peeters L W P. Determination of the number of automated guided vehicles required at a semi-automated container terminal. Journal of the Operational Research Society, 2001, 52:409-417.

[17]Evers J J M, Koppers S A J. Automated guided vehicle traffic control at a container terminal, Transportation Research A, 1996, 30(1):21-34.

[18]Bish E K, Leong T, Li C, Simchi-Levi D. Analysis of a new vehicle scheduling and location problem, Naval Research Logistics, 2001, 48(5):363-385.

[19]Van M. R., Operational control of internal transport, ERIM Ph. D. theries Research in Management, 2000, 1:13-17.

[20]Chen Y, Leong Y T, Ng J W C, Demir E K, Nelson H L, Simchi-Levi D. Dispatching automated guided vehicles in a mega container terminal, paper presented at INFORMS Montreal, 1998:19-22.

[21]Steenken D, Henning A, Freigang S, Voss S. Routing of straddle carriers at a container terminal with the special aspect of internal moves., OR Spelctrum, 1993, 15 (3):167-172.

[22]Bodin L D, Golden B L, Assad A A., Hall M O, Routing and scheduling of vehicles and crews, Computers and Operations Research, 1983, 10 (2):63-211.

[23]Kim K H, Bae J W. A dispatching method for automated guided vehicles to minimize delays of containership operations. International Journal of Management Science, 1999, 5(1):1-25.

[24]Baker C. High time for straddles. Cargo systems, 1998, 10:23-26.

[25]吕显强, 张宏伟. 集装箱码头分派车辆的整数规划模型. 大连水产学院学报, 2004, 19(6)

[26]Chung YG, Randhawa S. U. , McDowell E. D. A simulation analysis for a transtainer-based container handling facility. Computers & Industrial Engineering, 1988, 14(2):113-125.

[27]Lai, K. K, Lam, K., A study of container yard equipment allocation strategy in Hong Kong[J] International Journal of Modeling and Simulation .1994 14(3):134-138.

[28]Kim K. H., Kim K. Y. An optimal routing algorithm for a transfer crane in port container terminals. Transportation Science, 1999, 33(1):17-33.

[29]Kap Hwan Kim, Young-Man Park. A Crane Scheduling Method for Port Container

- Terminals, European Journal of Operational Research Part E, 2004. 156:752-768.
- [30] W. C. Ng, K. L. Mak, Yard crane scheduling in port container terminals [J]. Applied Mathematical Modelling, 2005, 29:263-267.
- [31] W. C. Ng, K. L. Mak, YX. Zhang. Scheduling trucks in container terminals using a genetic algorithm [J]. Engineering Optimization 2007 39(1):33-37.
- [32] J. L. Richard, C. Q. Zhang, A routing algorithm for a single transfer crane to load export containers onto a containership, Computers&Industrial Engineering, 1997, 33:673-676.
- [33] C. Zhang, Resource Planning in container storage: (博士论文). 香港: 香港大学, 2000, 4.
- [34] 魏众, 申金升等. 港口集装箱码头轮胎式龙门吊优化调度研究. 中国工程科学, 2007. 9: 22-24.
- [35] 何军良, 亦为建, 严伟. 基于爬山算法的集装箱堆场场桥调度. 上海海事大学学报, 2007. 12.
- [36] 高鹏, 金淳, 韩庆平. 提箱作业优化问题的嵌套启发式算法. 系统管理学报, 2008. 4, 17(2):203-209.
- [37] 王鸿鹏. 集装箱运输管理. 北京: 电子工业出版社, 2008:35-40.
- [38] 王鸿鹏, 许路, 邓丽娟. 国际集装箱运输与多式联运. 大连: 大连海事大学出版社, 2004:37-42.
- [39] 武德春. 集装箱运输实务. 北京: 机械工业出版社, 2003:110-119.
- [40] 陈戌源. 集装箱码头业务管理. 大连: 大连海事大学出版社, 2009:21-56.
- [41] 陈洋. 集装箱码头操作. 北京: 高等教育出版社:2001:37-69.
- [42] 于汝民. 集装箱码头经营管理. 北京: 人民交通出版社:1999:160-192.
- [43] 杨志刚. 国际集装箱多式联运实务与法规. 北京: 人民交通出版社, 2001:203-224.
- [44] 靳利霞. 蛋白质结构预测算法: (博士论文). 大连: 大连理工大学, 2002.
- [45] 计明军. 若干随机性全局优化算法的研究: (博士论文). 大连: 大连理工大学, 2004.
- [46] 项宝卫. 结构优化中的模拟退火算法研究和应用: (硕士论文). 大连: 大连理工大学, 2004.
- [47] 唐焕文. 实用最优化方法. 大连: 大连理工大学出版社, 2003:145-173.
- [48] 吴祈宗. 运筹学与最优化方法. 北京: 机械工业出版社, 2003:301-342.
- [49] 闫平凡, 张长水. 人工神经网络与模拟退火算法. 北京: 清华大学出版社, 2000:205-216.
- [50] 潘彦. Levitin A. 算法设计与分析基础. 北京: 清华大学出版社, 2004:325-341.
- [51] 王凌. 智能优化算法及其应用. 北京: 清华大学出版社, 2004:121-159.
- [52] 庞峰. 模拟退火算法的原理及算法在优化问题上的应用: (硕士论文). 吉林: 吉林大学,

2006. 12.

[53] 韩晓龙. 集装箱港口龙门吊的最优路径问题. 上海海事大学学报, 2005. 6, 26(2): 39-41.

附录 A 单台龙门吊的模拟退火算法程序代码

```
clear all                                %求解4.3节例题的主函数
a=[1 7 12 15 2 5 8 11 1 7 12 15 4 9 14 2 5 8 11 1 7 12 15 2 5 8 11 4 9 14;
   9 12 18 13 19 9 7 8 9 12 18 13 13 12 15 19 9 7 8 9 12 18 13 19 9 7 8
  13 12 15;
   20 20 20 20 15 15 15 15 14 14 14 14 25 25 25 16 16 16 16 18 18 18 18
  12 12 12 12 15 15 15];
Tmax=100;
N=100;
n=30
set=[];
x=zeros(2,30);
[x,fx]=Routing(x,a);% 调用子函数
z=x;
fz=fx;
while Tmax>10^(-2)
    for L=1:N
        if fx<=fz
            z=x;
            fz=fx;
        end
        i=fix(n*rand+1);
        switch i
            case {1,2,3,4}
                j=fix(4*rand+1);
            case {5,6,7,8}
                j=fix(8*(0.5+0.5*rand)+1);
            case {9,10,11,12}
```

```

        j=fix(12*(2/3+(1/3)*rand)+1);
    case {13, 14, 15}
        j=fix(15*(4/5+(1/5)*rand)+1);
    case {16, 17, 18, 19}
        j=fix(19*(15/19+(4/19)*rand)+1);
    case {20, 21, 22, 23}
        j=fix(23*(19/23+(4/23)*rand)+1);
    case {24, 25, 26, 27}
        j=fix(27*(23/27+(4/27)*rand)+1);
    otherwise %case {28, 29, 30}
        j=fix(30*(9/10+(1/10)*rand)+1);
    end
    if i~=j
        b=a;
        yt=[];
        yt(:,i)=b(:,i);%      解的变异
        b(:,i)=b(:,j);
        b(:,j)=yt(:,i);
        [x,fb]=Routing(x,b);
        if fb<=fx | exp(-(fb-fx)/Tmax)>rand %rand是一个大于0小于
1的随机数
            a=b;
            fx=fb;
        end
    end
    set=[set,fx];
    L
end
Tmax=0.90*Tmax

```

```

end
plot(set)                                %作出结果的收敛图
z
fz

function [x, fr]=Routing(x, a)           % 确定可行解并求出目标函数值的子函数
x1=zeros(2, 30);
c=a;
for p=1:4
    if c(2, p)>=c(3, p)
        x1(:, p)=[c(1, p)                %记忆访问的贝位号及提取的集装箱数量
                    c(3, p)];
        c(2, p)=c(2, p)-c(3, p);          %更新贝位上堆存的集装箱数量
        for k=p:30                        %更新后面同类集装箱在同一贝位的集装箱数量
            if c(1, k)==c(1, p)
                c(2, k)=c(2, p);
            end
        end
    end
    for l=p:4
        c(3, l)=0;                        %更新任务要求
    end
    break;
elseif c(2, p)<c(3, p) & c(2, p)~=0       %继续循环, 取箱
    x1(:, p)=[c(1, p)
                c(2, p)];
    for l=p+1:4
        c(3, l)=c(3, p)-c(2, p);
    end
    c(2, p)=0;
end

```

```

        for k=p:30                                %更新后面同类集装箱在同一贝位的集装箱数量
            if c(1,k)==c(1,p)
                c(2,k)=0;
            end
        end
    end
end
for p=5:8
    if c(2,p)>=c(3,p)
        x1(:,p)=[c(1,p)
                c(3,p)];
        c(2,p)=c(2,p)-c(3,p);
        for k=p:30                                %更新后面同类集装箱在同一贝位的集装箱数量
            if c(1,k)==c(1,p)
                c(2,k)=c(2,p);
            end
        end
        for l=p:8
            c(3,l)=0;
        end
        break;
    elseif c(2,p)<c(3,p)&c(2,p)~=0                %继续循环, 取箱
        x1(:,p)=[c(1,p)
                c(2,p)];
        for l=p+1:8
            c(3,l)=c(3,p)-c(2,p);
        end
        c(2,p)=0;
    end
end

```

```

        for k=p:30      %更新后面同类集装箱在同一贝位的集装箱数量
            if c(1,k)==c(1,p)
                c(2,k)=0;
            end
        end
    end
end
for p=9:12
    if c(2,p)>=c(3,p)
        x1(:,p)=[c(1,p)
                c(3,p)];
        c(2,p)=c(2,p)-c(3,p);
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=c(2,p);
            end
        end
        for l=p:12
            c(3,l)=0;
        end
        break;
    elseif c(2,p)<c(3,p)&c(2,p)~=0      %继续循环，取箱
        x1(:,p)=[c(1,p)
                c(2,p)];
        for l=p+1:12
            c(3,l)=c(3,p)-c(2,p);
        end
        c(2,p)=0;
        for k=p:30

```

```

        if c(1,k)==c(1,p)
            c(2,k)=0;
        end
    end
end
end
for p=13:15
    if c(2,p)>c(3,p)
        x1(:,p)=[c(1,p)
            c(3,p)];
        c(2,p)=c(2,p)-c(3,p);
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=c(2,p);
            end
        end
    end
    for l=p:15
        c(3,l)=0;
    end
    break;
elseif c(2,p)<c(3,p) & c(2,p)~=0
    x1(:,p)=[c(1,p)
        c(2,p)];
    for l=p+1:15
        c(3,l)=c(3,p)-c(2,p);
    end
    c(2,p)=0;
    for k=p:30
        if c(1,k)==c(1,p)

```

```

                                c(2,k)=0;
                                end
                            end
                        end
                    end
                for p=16:19
                    if c(2,p)>c(3,p)
                        x1(:,p)=[c(1,p)
                                c(3,p)];
                        c(2,p)=c(2,p)-c(3,p);
                        for k=p:30
                            if c(1,k)==c(1,p)
                                c(2,k)=c(2,p);
                            end
                        end
                    end
                    for l=p:19
                        c(3,l)=0;
                    end
                    break;
                elseif c(2,p)<c(3,p)&c(2,p)~=0
                    x1(:,p)=[c(1,p)
                            c(2,p)];
                    for l=p+1:19
                        c(3,l)=c(3,p)-c(2,p);
                    end
                    c(2,p)=0;
                    for k=p:30
                        if c(1,k)==c(1,p)
                            c(2,k)=0;

```



```
        end
    end
end
for p=20:23
    if c(2,p)>=c(3,p)
        x1(:,p)=[c(1,p)
            c(3,p)];
        c(2,p)=c(2,p)-c(3,p);
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=c(2,p);
            end
        end
    end
    for l=p:23
        c(3,l)=0;
    end
    break;
elseif c(2,p)<c(3,p)&c(2,p)~=0
    x1(:,p)=[c(1,p)
        c(2,p)];
    for l=p+1:23
        c(3,l)=c(3,p)-c(2,p);
    end
    c(2,p)=0;
    for k=p:30
        if c(1,k)==c(1,p)
            c(2,k)=0;
        end
    end
end
```

```

        end
    end
end
for p=24:27
    if c(2,p)>=c(3,p)
        x1(:,p)=[c(1,p)
                c(3,p)];
        c(2,p)=c(2,p)-c(3,p);
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=c(2,p);
            end
        end
        for l=p:27
            c(3,l)=0;
        end
        break;
    elseif c(2,p)<c(3,p)&c(2,p)~=0
        x1(:,p)=[c(1,p)
                c(2,p)];
        for l=p+1:27
            c(3,l)=c(3,p)-c(2,p);
        end
        c(2,p)=0;
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=0;
            end
        end
    end
end

```

```

end
end
for p=28:30
    if c(2,p)>=c(3,p)
        x1(:,p)=[c(1,p);
                c(3,p)];
        c(2,p)=c(2,p)-c(3,p);
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=c(2,p);
            end
        end
        for l=p:30
            c(3,l)=0;
        end
        break;
    elseif c(2,p)<c(3,p)&c(2,p)~=0
        x1(:,p)=[c(1,p);
                c(2,p)];
        for l=p+1:30
            c(3,l)=c(3,p)-c(2,p);
        end
        c(2,p)=0;
        for k=p:30
            if c(1,k)==c(1,p)
                c(2,k)=0;
            end
        end
    end
end
end

```

```

end
x1                                %显示每次解的结果
c                                %显示求出解后，矩阵变化结果
fr1=0;
for i=1:29                        %求解目标函数
    for j=i+1:30
        if x1(2,i)~=0 & x1(2,j)~=0
            fr1=fr1+abs(x1(1,i)-x1(1,j));
            break;
        end
    end
end
fr1
x=x1;
fr=fr1;

```

附录 B 多台龙门吊的模拟退火算法程序代码

```

clear all          %求解第五章算例的主函数
a=[1 7 12 15 2 5 8 11 1 7 12 15 4 9 14 2 5 8 11 1 7 12 15 2 5 8 11 4 9 14;
   9 12 18 13 19 9 7 8 9 12 18 13 13 12 15 19 9 7 8 9 12 18 13 19 9 7 8
13 12 15;
   20 20 20 20 15 15 15 15 14 14 14 14 25 25 25 16 16 16 16 18 18 18 18
12 12 12 12 15 15 15];
Tmax=150;
N=100;
n=30
set=[];
%x0=[1 7 0 0 2 0 0 0 0 7 0 15 9 4 0 2 5 0 11 0 0 12 0 0 0 11 8 0 0 14;
%   9 11 0 0 15 0 0 0 0 1 0 13 12 13 0 4 9 0 3 0 0 18 0 0 0 5 7 0 0 15];
%fx0=57;
x1=zeros(2,30);
x2=zeros(2,30);
[x1,x2,fx]=twoR(x1,x2,a);
z1=x1;
z2=x2;
fz=fx;
while Tmax>10(-2)
    for L=1:N
        if fx<=fz
            z1=x1;
            z2=x2;
            fz=fx;
        end
        i=fix(n*rand+1);

```

```

switch i
    case {1, 2, 3, 4}
        j=fix(4*rand+1);
    case {5, 6, 7, 8}
        j=fix(8*(0.5+0.5*rand)+1);
    case {9, 10, 11, 12}
        j=fix(12*(2/3+(1/3)*rand)+1);
    case {13, 14, 15}
        j=fix(15*(4/5+(1/5)*rand)+1);
    case {16, 17, 18, 19}
        j=fix(19*(15/19+(4/19)*rand)+1);
    case {20, 21, 22, 23}
        j=fix(23*(19/23+(4/23)*rand)+1);
    case {24, 25, 26, 27}
        j=fix(27*(23/27+(4/27)*rand)+1);
    otherwise %case {28, 29, 30}
        j=fix(30*(9/10+(1/10)*rand)+1);
end
if i~=j
    b=a;
    yt=[];
    yt(:, i)=b(:, i); %解的变异
    b(:, i)=b(:, j);
    b(:, j)=yt(:, i);
    [x1, x2, fb]=twoR(x1, x2, b);
    if fb<=fx | exp(-(fb-fx)/Tmax)>rand %rand是一个大于0小于1的
        随机数
        a=b;
        fx=fb;

```

```
        end
    end
    set=[set, fx];
    L
end
Tmax=0.90*Tmax
end
plot(set)
z1
z2
fz
```

攻读学位期间公开发表论文

[1]Ji Mingjun,He Maoying. Application of a Novel Simulated Annealing in Flow Shop Scheduling Problem. Fifth International Conference on Natural Computation ICNC 2009,2009

[2]贺茂英, 计明军. 基于双层规划的动态腹地二级港口物流网络优化. 交通运输系统工程与信息(核心期刊, 二审状态)