

分类号:

学校代号: 11845

UDC:

密级:

学 号: 2111303041

广东工业大学硕士学位论文

(工学硕士)

三维虚拟实验室中装配序列规划问题研究

张秋云

指导教师姓名、职称: 刘东峰 教授

专业或领域名称: 信息与通信工程

学生所属学院: 信息工程学院

论文答辩日期: 2016年5月

A Dissertation Submitted to Guangdong University of Technology
for the Degree of Master
(Master of Engineering Science)

Study of Assembly Sequence Planning in 3D Virtual
Laboratory

Candidate: Zhang Qiuyun
Supervisor: Prof. Liu Dongfeng

May 2016
Faculty of Information Engineering
Guangdong University of Technology
Guangzhou, Guangdong, P. R. China, 510006

摘 要

实验教学是教学活动中的一个重要组成部分。传统实验教学存在着各种各样的问题，如实验设备昂贵，实验仪器易损耗且难以操作，有些实验的操作过程也存在一定危险性，受到时间和空间的制约，一些抽象性的实验在真实的情景中难以实现或观察。虚拟实验室综合运用了虚拟现实、多媒体、人机交互、数据库以及网络通信等技术，力求搭建一个逼真、精确、生动、可交互的虚拟实验环境，从而作为传统实验教学强有力的补充手段。

本文首先分析了目前国内外三维虚拟实验室的发展和研究现状，发现虽然虚拟实验室解决了传统实验教学存在的各种各样的问题，但是目前已有的大多数虚拟实验室仿真系统仅仅实现了对实验现象或结果的展示，缺少对实验仪器的操作和组装过程的学习及评估功能，而这些在真正的实验教学活动中是非常重要的。接着又介绍了虚拟装配方法和装配序列规划方法的发展历程及其特点，发现它们常用于工业领域和军事培训领域，可以使产品的组装过程和受培训者的学习过程变得更加具体、易于理解与接受，但是为了实现更好的实验教学目的将它们应用于虚拟实验室中的例子却鲜有，现有的许多装配序列规划方法也存在着一些问题。

在综合分析之后，本文针对虚拟实验仪器装配过程的特点和现有装配序列规划方法存在的一些问题，提出了一种装配序列规划方法。该方法综合利用几何信息和人的经验知识来进行规划活动，克服了单一利用几何信息或人的经验知识来进行规划的局限性，规划质量更高；利用矩阵来对装配信息建模，而非用树或图来建模，简化了装配信息模型和建模过程，便于存储和计算；综合利用传统的基于精确计算的方法和基于启发式算法的方法来进行装配序列规划，既能克服前者易出现的组合爆炸问题和后者易出现的大量不可行解的问题，也能更快地获得更高质量的装配序列；针对传统遗传算法的易出现早熟现象和过快收敛的问题，提出了一种改进的遗传算法，保证了算法快速收敛于全局最优解的能力；针对虚拟环境中近距离装配操作难的特点，实现了零件近距离时的自动装配功能；同时为了让操作者能够更直观高效地学习实验仪器的操作和组装过程，实现了对装配操作的实时信息提示和引导，输出保存装配过程信息以及装配过程动画回放等功能。

最终将本文提出的装配序列规划方法在自主开发的三维虚拟物理实验室实验仪器

的虚拟装配过程中进行了仿真实现，达到了实验仪器可行装配序列的生成和优化的目的，装配序列的相关应用也有助于操作者更好地掌握实验仪器的操作和装配，从而更好地达到实验教学的目的。最后以常用实验仪器天平的虚拟装配过程为例，展示了利用本文提出的装配序列规划方法进行装配序列规划的效果。

关键词：三维虚拟实验室；虚拟装配；装配序列规划；遗传算法；装配信息建模

Abstract

Experimental teaching is an important part of teaching activities. There are various problems in traditional experimental teaching, such as expensive laboratory equipment, laboratory equipment easy to wear and tear and difficult to operation, risk in doing some experiments, it is restricted by time and space, and it is difficult to achieve or observe some abstract experiments in a real scenario. Virtual laboratory integrated use of virtual reality, multimedia, human computer interaction, databases and network communication technology to build a virtual lab environment that realistic, accurate, vivid and interactive, as a highly complementary for traditional experimental teaching.

At first, this paper analyzes the development and the current research status of 3D virtual laboratory at home and abroad and finds that although it can solve a variety of problems existing in the traditional experimental teaching, but at present most of the virtual laboratory simulation system only realizes the show of experimental phenomena and results, and lacks the operations of experimental apparatus and the learning and evaluation functions of assembly process, which is very important in a real experiment teaching. Then this paper introduces the development and characteristics of virtual assembly methods and assembly sequence planning methods, which are often used in industrial areas and military training, and has the advantages to make the assembly process of products and the trainee's learning process become more specific, easy to be understood and accepted. But there are few examples that apply the virtual assembly and assembly sequence planning methods to virtual labs in order to achieve better experimental teaching effect. And many existing assembly sequence planning methods also have some problems.

After a comprehensive analysis, in this paper, according to the characteristics of the assembly process of virtual experiment instrument and the existing problems of existing assembly sequence planning methods, a new method of assembly sequence planning is proposed. This method integrating uses the geometric information and experience knowledge to conduct the planning activities. It can overcome the limitations of the method that only uses the geometric information or experience knowledge, and has a higher quality

of planning. It takes advantage of matrix, not tree or graph structure, to achieve the modeling of assembly information and simplify the model of assembly information and the process of modeling. It is also convenient for storage and computing. Comprehensive utilization of the traditional method based on precise calculation and the method based on heuristic algorithm for assembly sequence planning can overcome the combinatorial explosion problem of former method and the problem of a large number of infeasibility of latter method, and can obtain assembly sequence of higher quality and less time. For the prone to premature and fast convergence of the traditional genetic algorithm, an improved genetic algorithm is proposed and ensures the ability to quickly converge to global optimal solution. Because it is difficult to achieve the assembly operation of a short distance in virtual environments, this paper achieves the function of automatic assembly in short distance. At the time, in order to allow the operator to be more intuitive and efficient to learn the operation and assembly of laboratory instruments, this paper realizes the popup of real-time message to guide assembly operation, saving and outputting assembly process information as well as the animation playback of assembly process.

Finally, a simulation of the assembly sequence planning method proposed in this paper is conducted in the independently developed 3D virtual physics laboratory. It achieves the purposes of generation and optimization of feasible assembly sequence. And the related applications of assembly sequence can help the operator master the operation and assembly of experimental apparatus and achieve a better effect of experimental teaching.

Keywords: 3D Virtual Laboratory; Virtual Assembly; Assembly Sequence Planning; Genetic Algorithm; Assembly Information Modeling

目 录

摘 要.....	I
Abstract.....	III
目 录.....	V
Contents.....	VII
第一章 绪论.....	1
1.1 课题研究背景和意义.....	1
1.2 国内外研究现状.....	1
1.2.1 三维虚拟实验室.....	1
1.2.2 虚拟装配.....	3
1.2.3 装配序列规划.....	5
1.3 本文研究内容和结构安排.....	6
第二章 三维虚拟仪器装配信息建模.....	8
2.1 装配信息.....	8
2.1.1 几何约束.....	9
2.1.2 人为约束.....	9
2.2 装配信息建模.....	9
2.2.1 全局装配干涉矩阵.....	10
2.2.2 约束邻接矩阵.....	13
2.2.3 位置姿态矩阵.....	14
2.2.4 重力支撑矩阵.....	15
2.2.5 装配方向矩阵.....	16
2.3 本章小结.....	17
第三章 三维虚拟仪器装配序列规划方法.....	18
3.1 三维虚拟仪器装配序列生成.....	18
3.1.1 几何约束满足判定.....	19
3.1.2 人为约束满足判定.....	21

3.2 三维虚拟仪器装配序列评价	23
3.2.1 装配过程稳定性	24
3.2.2 装配方向改变次数	25
3.2.3 人为约束满足程度	25
3.3 三维虚拟仪器装配序列优化	26
3.3.1 染色体编码	28
3.3.2 适应度函数	29
3.3.3 初始种群生成	30
3.3.4 遗传算法的进化操作	31
3.3.5 进化过程的控制参数	34
3.4 本章小结	34
第四章 三维虚拟仪器装配序列规划应用	36
4.1 三维虚拟实验室平台	36
4.2 仪器装配序列规划	42
4.2.1 仪器装配序列生成	42
4.2.2 仪器装配序列评价与优化	45
4.3 本章小结	47
总结与展望	48
参考文献	49
攻读硕士学位期间发表的论文	53
学位论文独创性声明	54
学位论文版权使用授权声明	54
致 谢	55

Contents

Abstract	III
Contents.....	VII
Chapter 1 Introduction	1
1.1 Background and Significance Topics	1
1.2 Research Status at Present	1
1.2.1 3D Virtual Laboratory	1
1.2.2 Virtual Assembly	3
1.2.3 Assembly Sequence Planning	5
1.3 Main Content and Structure Arrangement in the Paper.....	6
Chapter 2 Assembly Information Modeling of 3D Virtual Instrument.....	8
2.1 Assembly Information	8
2.1.1 Geometric Constraint	9
2.1.2 Artificial Constraint.....	9
2.2 Assembly Information Modeling.....	9
2.2.1 Global Assembly Interference Matrix	10
2.2.2 Constraint Adjacent Matrix	13
2.2.3 Position Attitude Matrix	14
2.2.4 Gravity Support Matrix	15
2.2.5 Assembly Direction Matrix	16
2.3 Chapter Brief Summary.....	17
Chapter 3 Assembly Sequence Planning Method of 3D Virtual Instrument	18
3.1 Assembly Sequence Generation of 3D Virtual Instrument.....	18
3.1.1 Satisfaction Determination of Geometric Constraint	19
3.1.2 Satisfaction Determination of Artificial Constraint.....	21
3.2 Assembly Sequence Evaluation.....	23
3.2.1 Assembly Process Stability	24
3.2.2 Change Times of Assembly Direction.....	25

3.2.3 Satisfaction Degree of Artificial Constraint	25
3.3 Assembly Sequence Optimization.....	26
3.3.1 Chromosome Encoding	28
3.3.2 Fitness Function	29
3.3.3 Initial Population Generation	30
3.3.4 Evolutionary Operation of Genetic Algorithm.....	31
3.3.5 Control Parameters of Evolutionary Process.....	34
3.3 Chapter Brief Summary.....	34
Chapter 4 Assembly Sequence Planning Application of 3D Virtual Instrument.....	36
4.1 Platform of 3D Virtual Laboratory	36
4.2 Assembly Sequence Planning of Instrument	42
4.2.1 Assembly Sequence Generation of Instrument	42
4.2.2 Assembly Sequence Evaluation and Optimization of Instrument	45
4.3 Chapter Brief Summary.....	47
Conclusion and Forecast.....	48
Reference	49
The Published Papers During Master Period	53
Declaration	54
Acknowledgements.....	55

第一章 绪论

1.1 课题研究背景和意义

社会在不断进步，科技发展也日新月异，这些都离不开教育发展的支持，而实验教学则是教学活动中重要组成部分^[1]，它对于培养学生们的实际动手能力、学习理解能力、解决问题能力以及思维创造能力等都有着非常积极的作用，虽然目前教育改革仍在不断进行，政府和社会对教育事业投入的力度仍在不断加大，使得传统实验室的设施和环境有了很大提高，但是实验教学活动的有效进行仍然受到一些限制^[2]，例如，很多地区仍然存在实验器材损耗严重及维护难，专业的实验教学人员少和实验活动所需时间和空间受到制约的问题，另外，许多实验存在危险性导致实验教学无法进行或者受限制^{[3][4]}。

近些年来，随着虚拟现实、多媒体、人机交互、数据库以及网络通信等技术的出现和迅猛发展，三维虚拟实验室作为传统实验教学强有力的补充手段应运而生^[1]，它可以结合在工业领域迅猛发展的虚拟装配和装配序列规划技术，搭建一个逼真、精确、生动、可交互的虚拟实验环境，从而可以让操作者能够更直观高效地学习和掌握实验仪器的结构和操作以及实验原理和过程，打破了传统实验室对操作者和实验教学活动的限制^[2]，提供了生动逼真的仪器模型和仿真环境，实时方便的交互手段等，最终较好地达到了实验教学目的^[3]。

1.2 国内外研究现状

1.2.1 三维虚拟实验室

在 20 世纪 80 年代末，美国的一名教授第一次提出了“虚拟实验室”的概念，近些年来，由于计算机、多媒体以及网络等信息技术的飞速发展，虚拟实验室的应用领域也越来越广泛，它目前已经应用在了军事、农业、医疗、教育以及工程等领域^[3]。虚

拟实验室的目的是创建一个可以进行仿真实验活动并能实现交互的虚拟仿真实验环境,但最初它只是用于某一门学科的仿真和计算。然而随着科技的不断进步,以及人们对仿真环境真实感的需求越来越高,要求通过虚拟实验室不仅能够获得正确的实验数据和实验结果,也需要在实验仿真的过程中得到更良好的人与虚拟实验环境之间的交互感,并希望最终能够逐渐实现人与人之间很好的交互功能^[5]。

虚拟实验室是一个由一系列的输入、输出、控制条件、约束规则以及参数等构成的复杂系统^[3]。一般的虚拟实验室仿真系统,都能够支持仿真模型的建立或导入、实验数据的生成与获取,并能够提供人与人或人与环境之间的交互。根据仿真模型的呈现形式不同,目前存在的虚拟实验室可分为二维虚拟实验室和三维虚拟实验室,前者提供更高的仿真效率,后者则侧重提供更好的仿真效果即“真实性”;根据构建虚拟实验室环境使用工具的不同,可分为软件构建方式和软硬件结合构建方式。因虚拟实验室具有很好的发展前景,虚拟实验室的研究与建设吸引了社会各界的注意,它逐渐成为国内外很多科技研究工作者青睐的研究对象。

国外方面很早就对虚拟实验室进行了大量的研究,并进行了实现。1988年,麻省理工学院针对电路设计以及微电子学等课程的实验教学问题,研发了微电子在线实验室系统,该系统利用Java进行编程,搭建了一个二维的仿真实验环境,实现了对实验过程和现象的展示功能,目前,它也已经从1.0版本更新到了5.0版本,并增加了手机远程控制等先进功能^[3];针对物理课程的实验教学问题,美国北卡罗来那大学Greensboro分校开发了一个基于Web的探索式的LAA Physics虚拟物理实验室^[6],该实验室由虚拟实验、在线交流以及学习评价等几个模块组成^[7];同样针对物理课程实验教学的问题,DesignSoft公司设计开发了一款虚拟教学软件叫作Newton^[8],它搭建了用户可以进行物理实验的交互环境;而美国Michigan大学针对化学课程的实验教学问题,开发了一个叫作VRChEL的实验室系统,实现了化学课程的实验仿真功能;出于无机化学实验教学的需要,CamegieMellon大学研发了一款叫作IrYdium的虚拟化学实验室系统^[9];HHMI(Howard Huglles医学院)也开发了一款用于生物医学课程实验教学的软件,即BioimeraCtive系统^[10];由于先前的虚拟实验室都是针对单一学科的限制性,为了实现多个领域的虚拟实验教学目的,Houston大学联合了NASA Johnson航天中心共同研发了支持多个学科和领域的培训系统,即VETL实验室^[3]。

近些年来,国内学者在利用虚拟实验室进行虚拟实验教学方面也做了许多研究。国内第一套虚拟的实验教学软件,即“几何光学实验设计平台”^[11],是由中国科技大

学研发的，它可以进行多种物理实验过程和现象的展示以及交互；北京大学研发实现了一个名为 3WNVLAB 的网上虚拟实验室^[12]，能够进行 CACHE 的结构设计与流水线规划设计，用户能够通过客户端来设置虚拟实验相关的各种参数信息，实验仿真和实验结果则在服务器端进行和提供；为了构建一个用于三维电子线路实验仿真的虚拟环境，北京师范大学基于 QTVR 和 VRML 技术研发了 Elva 系统^[13]，实现了对电路知识的高效学习和实验验证目的；另外还有由中科院的上海有机化学研究所研发的 ChemLab 虚拟化学实验室^[14]，由北方交通大学开发的 RECEL 虚拟仿真实验系统，以及由华中科技大学搭建的网上工程测试虚拟实验室等等^[15]。

作为传统实验教学的有力补充，虚拟实验室有效解决了传统实验教学难以满足实验教学需求的问题，利用各种信息技术、网络技术和虚拟现实技术等来模拟实验过程和现象，提供了较为逼真和形象的实验场景，把抽象的概念和理论变得更生动有趣，从而使知识更易于理解和掌握，同时通过虚拟实验室也降低了实验成本和损耗，解决了一些传统实验所存在的安全问题，从长远发展来看，它有利于配合课堂教学以及传统实验教学手段来提高整体的教学水平。但是，目前现有大多数的仿真软件还存在或多或少的缺陷，比如沉浸感、交互操作、操作灵活性、实验现象逼真度等方面，另外目前大多数的系统仅仅提供了对实验现象或者结果的展示，大多缺少对实验过程的引导和评估模块，无法根据学习者的学习情况智能推荐相应内容等问题^[3]。

1.2.2 虚拟装配

针对虚拟现实技术的相关研究日益深入，使得面向机械产品的虚拟装配技术成为了虚拟制造领域相关研究的热点问题^[16]。数字化和自动化是现代化制造技术的显著特点，而虚拟装配则是数字化过程中一个重要的实现环节^[17]，也是虚拟制造过程的一个重要组成部分。虚拟装配是实际装配活动在计算机上的仿真再现，它综合利用了计算机仿真、虚拟现实等技术，通过辅助设计软件、计算机和一系列交互设备来构建一个能够提供较强真实感的虚拟环境，并在其中实现建模、仿真以及交互功能，用户能够进行虚拟漫游或虚拟拆装操作^[16]，并实现装配过程的可视化，从而在计算机上模拟实际装配过程。目前，该技术主要被用于工业领域中。在这种虚拟的环境中，用户可进行产品的设计、装配、装配规划，并验证和评估装配性能，从而能够及早地发现装配过程中的问题，获得最为经济、合理和实用的装配方案^[18]。因此，虚拟装配技术在优

化产品的设计,减少甚至避免资源的浪费,缩短产品设计、制造的周期,对装配操作人员进行培训,新产品的开发与维护等方面具有很大的优势^{[19][20]}。最初,虚拟装配技术主要被应用于军事方面的研究中,最终在工业领域获得了迅速发展,各个科研机构及个人争相进行了虚拟装配系统的研发^[3]。

在国外的一些发达国家,如美国、日本以及德国等,虚拟装配技术被广泛应用在国防工业、航空航天业、船舶制造业、汽车制造业以及电子产品制造业等很多领域。Rossler 等为了辅助产品设计者进行零件的虚拟拆装,基于空间约束算法开发了一个虚拟装配系统;Antonishek 等则借助水平显示设备开发了一个半沉浸式的虚拟装配系统;Heriot-Watt 大学也进行了多个虚拟装配相关的项目,其中有两个分别用于构建一个可以获取装配规划相关知识的虚拟环境和用于电缆连接设计与规划,并研发了一个虚拟装配规划系统即 UVAVU^[18];美国的国家标准技术研究所 NIST 和华盛顿州立大学联合开发了用于装配规划和装配评价的虚拟装配设计环境 VADE^{[18][21]},该系统已成功通过了工业测试。

国内针对虚拟装配研究与开发的探索则开始于上个世纪 90 年代的后期。利用 CAD 软件 Pro/E,国内著名大学清华大学研发了一个叫作 VASS 的虚拟装配支持系统^[18],可通过仿真三维实体模型的拆装过程,生动直观地对产品的可装配性进行验证、分析,从而规划和改进装配工艺,最终生成装配工艺相关文件实现对实际装配的指导;西安飞机制造公司所研发的一个数字化产品设计制造系统则能实现对零件三维模型的设计、预装配及空间干涉检查等功能;另外,孙军华等人基于虚拟现实建模语言(VRML)搭建了一个交互式的虚拟装配平台;浙江大学万华根、高曙名等人则设计了一个虚拟设计与虚拟装配系统即 VDVAS 和一个多模态沉浸式虚拟装配系统即 MIVAS^[22];上海交通大学也开展了对虚拟装配相关技术的探索与研究,提出了一个分布式的虚拟制造系统即 DVMS 的框架体系;刘振宇等人对装配设计过程中的装配语义识别以及约束转化问题进行了研究^[23]。

虚拟装配技术广泛应用于工业领域,从而降低了工业产品的设计与开发成本,给制造业带来了巨大的经济与社会效益;它可以方便地完成对装配过程的评估以及对装配工艺选择,从而提高产品的设计、研发与制造的效率,同时通过的实现良好的仿真性能来提供比较真实的仿真数据以及生动、逼真的展示效果,使得产品的设计过程和受培训者的学习过程变得更加具体、易于理解与接受^[3]。但是虚拟装配有关的大多数研究与应用实例都存在或多或少的问题,虚拟装配技术需要结合具体的应用场景来实现

进一步的完善。而且目前针对虚拟装配技术的研究与应用主要集中于工业制造与军事培训领域，将虚拟装配技术与三维虚拟实验室结合起来，应用于实验教学领域的研究与应用却做得比较少。

1.2.3 装配序列规划

装配序列规划（ASP）是产品装配过程中的一个重要环节，而产品装配序列的好坏则直接影响着产品的装配质量和成本^[24]。ASP 是以获得一个可行且低成本的装配序列为目的，并借助装配信息模型，根据某种推理方法求解出一条或多条装配序列用于目标产品的装配，再依据一定的评价标准来筛选获得符合要求且可行的最优序列。装配序列规划本质上是一个 NP-hard 组合优化的问题^[25]。完成产品的装配序列规划必须解决两个关键问题：第一，如何对零部件之间存在的约束进行合理的表示；第二，怎样获得最优的或接近最优的可行装配序列^[26]。在过去几十年里，国内外学者对 ASP 问题进行了大量研究与应用。

比较典型的装配序列规划方法则是对装配相关的几何信息进行数学描述，根据零部件之间的空间位置关系以及邻接关系来获得装配干涉信息以及装配优先关系，进而通过规划获得可行装配序列，建立合适的用于装配序列评价的目标函数，获得最优的可行装配序列^[24]。Bourjault 提出使用联接图来表示装配体的方法，但要求用户必须回答很多问题才能获得可行的装配序列^[26]；HomemdeMello 等人则在使用联接图表示装配体方法的基础上，通过应用割集理论和定义简化规则来减少需要用户回答的问题数量^[26]；Baldwin 等人则通过询问在给定联接之前以及之后必须建立的联接关系来进一步减少问题的规模，但很难保证用户能够正确地回答问题^[26]；为了克服以上联接图方法所带来的巨大计算量，Chakrabarty 等人提出了利用层次图表示装配体零部件之间的结构关系的方法^[26]；Tseng 等人则提出将装配体中的联接件作为装配序列规划过程中的基本装配单元^[27]，最终得到一个基于联接件的多路径的装配序列图的方法，这种方法可以有效降低表示装配体的复杂度，最终能够简化装配模型；王俊峰等人则提出根据优先关系以及联接矩阵获得可行装配序列的方法^[24]。

后来，各种启发式算法也被用于装配序列的规划。Wanget 等人将蚁群算法（AA）应用于装配序列规划^[26]；Cheng 等人则考虑了装配资源对装配序列的影响，综合应用了遗传算法（GA）和蚁群算法来实现装配序列的优化^[28]；Chang 等人则将人工免疫算

法（AIS）应用在了装配序列规划领域^[29]，来实现装配序列的生成与优化；一些其它的启发式算法，如粒子群优化算法（PSO）等也已经被用于装配序列规划^[27]。

从现在已有的装配序列规划方法来看，其主要分成两类：一是精确计算的方法，二是启发式方法；其中属于精确计算方法的例子有树搜索方法和图搜索方法，它们通常会应用割集算法，所以利用该方法获得的装配序列能够确保是全局最优的，但当装配体较复杂即零件数目比较多的时候，使用该方法将会造成组合爆炸，进而求解难度增大，最后很难得到理想的序列；因此，当进行复杂装配体的或者需要智能性的装配序列规划活动时，必须引入启发式算法，即遗传算法、蚁群算法、模拟退火算法以及粒子群算法等^[24]。另外，根据装配序列的获得过程直接与否也可以分为直接法和间接法。直接法，顾名思义，是通过直接对装配活动进行分析建模以及规划，最终获得可行装配序列的方法；间接法即“逆向拆卸法”，也叫作“可拆即可装”方法，该方法以假设零件是刚性零件以及拆卸序列与装配序列可逆为前提^[30]，通过进行拆卸序列规划获得拆卸序列，将顺序颠倒之后即作为装配序列，目前提出的很大一部分装配序列规划方法都是基于间接法的，显然该方法的应用前提是基于许多的假设与约束，也不够直接和易于理解。

1.3 本文研究内容和结构安排

针对目前国内外已有的大多数虚拟实验室仿真系统仅仅实现了对实验现象和结果的展示，缺少操作者对实验仪器和实验过程的学习以及缺乏教学和评估功能等问题；而常用于工业领域和军事培训领域的虚拟装配技术和装配序列规划技术恰好具备使产品的设计、装配、装配规划过程和受培训者的学习过程变得更加具体、易于理解与接受，利于验证和评估装配性能，适合于对装配操作人员进行培训，并能提供比较真实的仿真数据以及生动、逼真的展示效果等优点；所以，为了提供一个逼真、精确、生动、可交互的虚拟实验环境，让操作者更加直观和高效地学习和掌握实验仪器的结构和操作以及实验原理和过程，并获得数据用于教学以及评估，最终实现更好的实验教学效果，本文将虚拟装配技术和装配序列规划技术应用到虚拟实验室仿真系统中，并对装配序列规划的关键问题进行了详细研究，针对虚拟实验室的特点提出了一种装配序列规划方法，最终通过自主开发的三维虚拟物理实验室仿真系统进行了仿真验证。

本文的研究目标是结合自主开发的三维虚拟物理实验室系统，将本文提出的装配

序列规划相关算法嵌入到实验仪器的手动装配过程中，让操作者能够更直观高效地学习和掌握实验仪器的结构和操作以及实验原理和过程，并能进行创造性的实验。利用开源的游戏仿真引擎 Delta3D 搭建了三维虚拟实验室的仿真平台，结合开源的三维图形渲染引擎 OSG 和物理仿真引擎 ODE 开发了三维虚拟实验仪器的装配和实验原理及过程仿真环境^[3]。针对实验教学的要求，实现良好的装配仿真、实验仿真及人机交互等功能。

本文对三维虚拟实验仪器装配序列规划相关的关键技术进行研究，主要包括以下几个方面：

- (1) 装配序列规划相关信息的建模；
- (2) 可行装配序列的判定与生成；
- (3) 可行装配序列的评价与优化；
- (4) 装配序列规划的应用；
- (5) 装配过程以及装配序列规划的物理仿真。

本文结构安排总共分为四章，具体安排如下：

第一章是绪论，详细介绍了虚拟实验室、虚拟装配和装配序列规划技术背景以及它们的研究现状，为本文中算法和方法的研究与提出提供理论和事实依据。并在最后给出了本文的研究内容和结构安排。

第二章介绍了进行装配序列规划需要的装配信息，重点分析了装配信息建模方法以及研究现状，最终提出通过建立五种相关的矩阵来对装配信息建模，为装配序列规划活动做好准备。

第三章提出了根据第二章建立的装配信息模型对装配序列可行性进行判定，最终获得可行的装配序列的具体算法；并针对本文研究对象的特点提出了合适的评价因子，介绍了遗传算法并作出了改进，建立了合适的适应度函数，最后结合改进了的遗传算法提出了装配序列优化算法。

第四章对本文提出的装配序列规划相关方法在实例中的具体应用进行了详细的描述，并对仿真效果进行了展示。

第二章 三维虚拟仪器装配信息建模

2.1 装配信息

装配信息建模的目的是作为装配序列规划活动的信息来源并为其提供装配信息的存取机制^[31]。装配信息模型要能够处理外界的输入信息以及装配序列规划过程的中间和结果信息，所以装配信息应随着规划过程的推进而趋于完善^[31]。尽管各个虚拟装配系统进行装配建模的所需信息不尽相同，但较为完整的装配信息应包含：几何信息、拓扑信息、工程语义信息、装配工艺信息、装配资源信息和管理信息^{[20][31][32]}。

装配信息可以通过自动推理或者人工输入的方式获取^[33]，为了高效地获取高质量的装配信息，满足装配序列规划的需求，本文采取两种方式相结合的方法，首先从零件的 CAD 几何模型中的配合特征或配合信息，自动推理生成几何信息如零件之间联接和阻碍关系、几何信息、层次信息以及位姿信息等，然后通过用户输入获得自动推理方式难于提取的非几何信息如装配约束邻接关系等^[33]。装配信息的获取方式如图 2-1 所示，有了装配信息，就可以对装配信息进行建模。

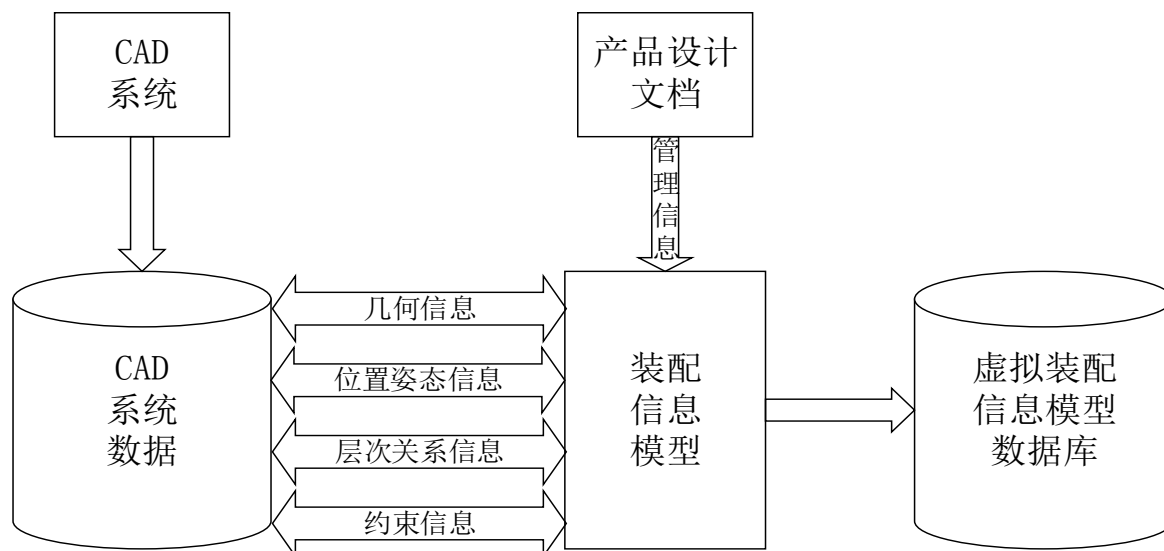


图 2-1 装配信息获取

Fig.2-1 Acquisition of assembly information

本文针对虚拟实验室中装配序列规划的特点，提出根据装配干涉信息和约束邻接信息来生成可行的装配序列，根据零件的位置姿态信息实现零件的装配定位，根据零

件之间的重力支撑信息和装配方向信息来实现装配序列的优化。

一个可行的装配序列首先必须满足零件和装配体本身的存在各种约束，即是几何可行的，其次还要尽可能地满足一些人为施加的约束。本文把这两类约束分别定义为几何约束和人为约束。

2.1.1 几何约束

几何可行指零件装配过程中以及零件装配到位后都不存在阻碍和干涉现象^[30]。零件装配过程中的干涉主要是由不合理的装配路径或者装配顺序导致的；零件装配好之后产生的干涉是由零件的尺寸或者几何形状上的缺陷导致，属于零件设计和制造错误，可以通过进行静态干涉检查来解决^[30]。

本文只考虑由于装配顺序以及装配方向不合理产生的干涉问题，提出通过建立并依据全局装配干涉矩阵来分析、判断装配顺序的几何可行性，并生成装配顺序中各个零件的可行装配方向^[30]，并用装配方向矩阵来表示多个装配顺序中零件的可行装配方向。

2.1.2 人为约束

人为约束即人们根据装配经验、统计数据或者个人喜好等通过强制手段指定的任意两个零件间的装配优先关系^[34]。针对实验室中组成实验仪器的零件数目有限的特点，本文提出一种约束邻接矩阵^[3]来表示人为约束。

2.2 装配信息建模

装配信息建模即建立一种合理的模型来表示装配信息^[35]。一个好的装配信息模型有助于装配序列规划活动的高效进行，它必须能够正确以及完整地提供装配体零部件之间的有用信息，而完整的装配信息应包含各个零部件之间的层次关系信息以及空间位置信息等^[30]；一个产品的 CAD 模型只能提供某些用于图形显示和交互的信息，如零部件之间的几何以及空间位置关系，这些信息必须经过提取和转换才可用直接用于装配序列规划^[30]；目前针对装配序列规划问题提出的装配信息模型主要有：层次结构关

系模型、图结构关系模型和基于虚拟链的混合关系模型^[30]。

对装配信息进行建模是实施装配序列规划活动的必要前提。在虚拟环境中，装配零部件模型的表达方式不同于 CAD 系统，CAD 系统是进行精确的几何描述，而虚拟环境中则是采用三角面片的模型来表达装配体的^[36]，减少了数据量，从而满足了仿真的实时性要求，但是却损失了精确的几何、拓扑以及工程设计等信息，难以进行装配过程的分析、评价以及对装配约束的定义^[32]。传统的 CAD 建模系统往往只注重产品信息的几何表示，并不能很好地支持产品的设计与装配规划，将参数化、特征造型和变量化技术以及约束关系与 CAD 系统结合，把装配体当作一个由约束构成的系统，这样就可以很好地表达装配信息^[31]。装配零部件的约束信息可以表示零部件的空间位置和状态以及零部件之间的约束关系等，是进行装配和装配序列规划的重要信息，由于组成实验仪器的零部件数目不多，所以使用矩阵可以方便、有效地表示零部件之间的相关约束信息。本文针对三维虚拟实验室中实验仪器构造简单、组成零件数目少的特点，并结合装配序列规划的特点，提出一种用矩阵来表示各种约束信息的方法，并且选取了几何约束信息和人为约束信息用于装配序列规划，建立了五种装配信息相关矩阵，即全局装配干涉矩阵、约束邻接矩阵、位置姿态矩阵、重力支撑矩阵和装配方向矩阵来分别描述装配干涉信息、约束邻接信息、零件的位置姿态信息、零件之间的重力支撑信息和装配方向信息，实现对装配信息的建模，从而为后续的装配序列规划打下良好的基础。

2.2.1 全局装配干涉矩阵

本文利用全局装配干涉矩阵^{[25][36]}，通过推理和算法来判断装配序列的几何可行性即满足几何约束。全局装配干涉矩阵描述了装配体中每个零件沿全局坐标系的六个坐标轴方向从无穷远处移动到最终装配位置过程中与其他零件的装配干涉情况^[30]，其中假设其他零件全部装配到最终位置，它所包含的信息足以准确描述装配零部件之间的几何约束关系^[25]。其定义如下：

若一个装配体是由 n 个零件（索引分别为 $0, 1, 2, \dots, n-1$ ，本文中用到的所有零件索引均从 0 开始编号）构成的，其全局装配干涉矩阵可以表示为一个 n 阶方阵，如下所示：

$$I_G = \begin{bmatrix} I_{00X} I_{00/X} I_{00Y} I_{00/Y} I_{00Z} I_{00/Z} & \cdots & I_{0,n-1X} I_{0,n-1/X} I_{0,n-1Y} I_{0,n-1/Y} I_{0,n-1Z} I_{0,n-1/Z} \\ \vdots & \ddots & \vdots \\ I_{n-1,0X} I_{n-1,0/X} I_{n-1,0Y} I_{n-1,0/Y} I_{n-1,0Z} I_{n-1,0/Z} & \cdots & I_{n-1,n-1X} I_{n-1,n-1/X} I_{n-1,n-1Y} I_{n-1,n-1/Y} I_{n-1,n-1Z} I_{n-1,n-1/Z} \end{bmatrix} \quad (2.1)$$

I_G 的每个元素均由六位二进制数构成, $I_{ijX} I_{ij/X} I_{ijY} I_{ij/Y} I_{ijZ} I_{ij/Z}$ 代表零件 i 分别沿着 $+X, -X, +Y, -Y, +Z, -Z$ 六个全局坐标轴方向装配到最终位置时与零件 j 的干涉情况。若不发生干涉, 则对应位取值为 1, 发生干涉则取值为 0。约定当 $i=j$ 时, 对应位取值为 0。根据装配零件之间的几何运动相互性可知^[25], 当 I_{ijD} 与 $I_{ji/D}$ 相异时, $I_{jiD} I_{ji/D} = I_{ij/D} I_{ijD}$, 否则 $I_{jiD} I_{ji/D} = I_{ijD} I_{ij/D}$, 其中 $D \in \{\pm X, \pm Y, \pm Z\}$, “/” 表示负号。为了便于计算机处理, 本文将全局装配干涉矩阵的每个元素即六位二进制数均以十进制数存储, 即将各个六位二进制数转化为十进制数。

下面以图 2-2 所示装配体为例来对全局装配干涉矩阵的建立方法进行详细说明。检查图示装配体的零件沿全局坐标系的六个坐标轴方向从无穷远处移动到最终装配位置的过程中与其他零件(假设其他零件已装配到最终位置)的干涉情况^{[30][37][38]}。图示装配体由 4 个零件组成, 所以其全局装配干涉矩阵应是一个 4 阶矩阵^[37]。先考察装配零件 0 时, 零件 1 对零件 0 的干涉情况^[38], 从而确定矩阵中第 1 行第 2 列的元素 $I_G(0,1)$ 即 $I_{01X} I_{01/X} I_{01Y} I_{01/Y} I_{01Z} I_{01/Z}$ 的值^[37]。可以依次检查六个方向上的干涉情况^{[37][38]}, 例如当零件 0 沿着坐标轴的 $+X$ 方向从无穷远处移动至其最终装配位置时, 零件 0 与零件 1 发生干涉即零件 1 阻碍零件 0 的装配, 所以对应的 I_{01X} 应取值为 0, 同理可以依次确定其他五个方向上的装配干涉情况, 得到 $I_{01X} I_{01/X} I_{01Y} I_{01/Y} I_{01Z} I_{01/Z}$ 的值为 010000。同理也可以确定全局装配干涉矩阵中其它元素的值, 最终得到图示装配体的全局装配干涉矩阵的二进制和十进制表示如图 2-3 和图 2-4 所示。

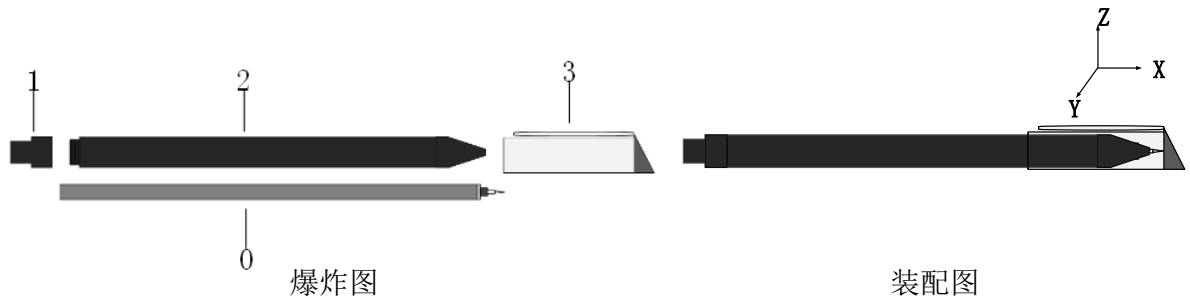


图 2-2 圆珠笔模型

Fig.2-2 Model of Ball point pen

$\begin{bmatrix} 000000 & 010000 & 100000 & 100000 \\ 100000 & 000000 & 100000 & 101111 \\ 010000 & 010000 & 000000 & 100000 \\ 010000 & 011111 & 010000 & 000000 \end{bmatrix}$	$\begin{bmatrix} 0 & 16 & 32 & 32 \\ 32 & 0 & 32 & 47 \\ 16 & 16 & 0 & 32 \\ 16 & 31 & 16 & 0 \end{bmatrix}$
--	--

a) 二进制表示

b) 十进制表示

a) Binary representation

b) Decimal representation

图 2-3 圆珠笔的全局装配干涉矩阵

Fig.2-3 Global assembly interference matrix of ball point pen

同样也可以获取图 2-4 所示天平模型的全局装配干涉矩阵如图 2-5 所示。天平的零件索引号和名称的对应关系如表 2-1 所示。

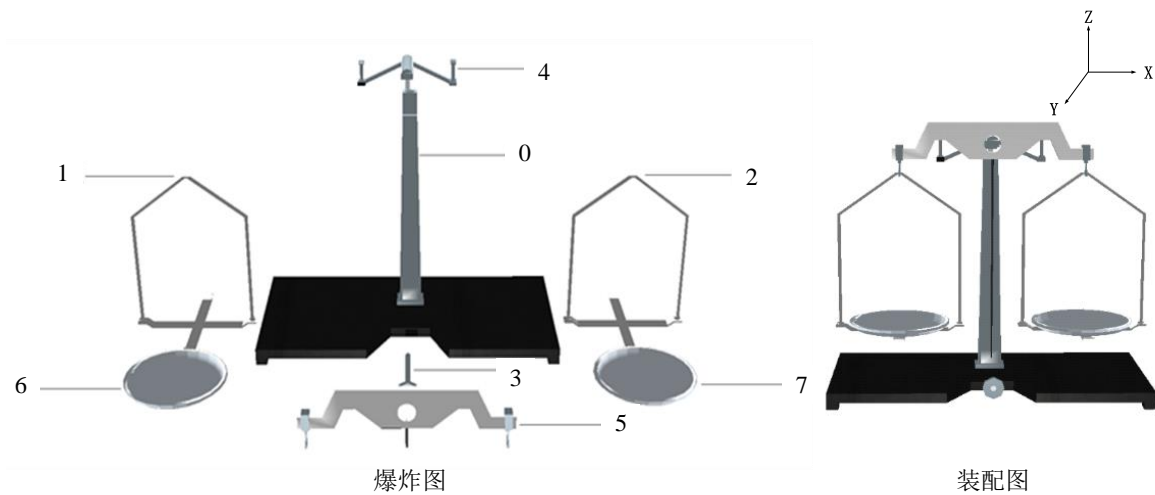


图 2-4 天平的模型

Fig.2-4 Model of balance

表 2-1 天平的零件名称和索引号

Table2-1 Name and index number of part

索引号	零件名称	索引号	零件名称
0	base0	4	gear_up
1	bracker_left	5	ruler
2	bracket_right	6	tray_left
3	gear_down	7	tray_right

$$\begin{bmatrix} 000000 & 111111 & 111111 & 001000 & 000110 & 111111 & 111111 & 111111 \\ 111111 & 000000 & 111111 & 111111 & 111111 & 000001 & 000010 & 111111 \\ 111111 & 111111 & 000000 & 111111 & 111111 & 000001 & 111111 & 000010 \\ 000100 & 111111 & 111111 & 000000 & 111111 & 111111 & 111111 & 111111 \\ 001001 & 111111 & 111111 & 111111 & 000000 & 001000 & 111111 & 111111 \\ 111111 & 000010 & 000010 & 111111 & 000100 & 000000 & 111111 & 111111 \\ 111111 & 000001 & 111111 & 111111 & 111111 & 111111 & 000000 & 111111 \\ 111111 & 111111 & 000001 & 111111 & 111111 & 111111 & 111111 & 000000 \end{bmatrix}$$

a) 二进制表示

a) Binary representation

$$\begin{bmatrix} 0 & 63 & 63 & 8 & 6 & 63 & 63 & 63 \\ 63 & 0 & 63 & 63 & 63 & 1 & 2 & 63 \\ 63 & 63 & 0 & 63 & 63 & 1 & 63 & 2 \\ 4 & 63 & 63 & 0 & 63 & 63 & 63 & 63 \\ 9 & 63 & 63 & 63 & 0 & 8 & 63 & 63 \\ 63 & 2 & 2 & 63 & 4 & 0 & 63 & 63 \\ 63 & 1 & 63 & 63 & 63 & 63 & 0 & 63 \\ 63 & 63 & 1 & 63 & 63 & 63 & 63 & 0 \end{bmatrix}$$

b) 十进制表示

b) Decimal representation

图 2-5 天平的全局装配干涉矩阵

Fig.2-5 Global assembly interference matrix of balance

在本文建立的虚拟装配环境中，导入零件的三维模型文件后，可以根据各个零件的几何位置信息，使用两两零件模型的包围盒沿各个全局坐标轴方向上的投影的重合情况来判断当一个零件沿着坐标轴方向移动到最终装配位置时与另一个零件是否会发生干涉情况^[36]，从而获得全局装配干涉矩阵；也可通过对三维建模环境进行二次开发，充分利用三维建模软件系统内集成的静态干涉求解器（如 UG 的 UFClear 模块），来快速获取建立全局装配干涉矩阵所需的信息。

2.2.2 约束邻接矩阵

本文提出利用约束邻接矩阵^[3]，通过推理和运算来判断装配序列的人为可行性即满足人为约束。约束邻接矩阵是根据人的经验或特别指定的待装配对象的零件间优先装配关系即指定的两个零件之间紧邻的先后装配顺序获取的。包含 n 个零件的装配体的约束邻接矩阵表示为 n 阶方阵，其模型如下：

$$P = \begin{bmatrix} p_{00} & p_{01} & \cdots & p_{0,n-1} \\ p_{10} & p_{11} & \cdots & p_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n-1,0} & p_{n-1,1} & \cdots & p_{n-1,n-1} \end{bmatrix} \quad (2.2)$$

其中 p_{ij} 代表零件 i 是否优先于零件 j 装配。若零件 i 优先于零件 j 装配，即 j 紧邻 i 之后装配，则 p_{ij} 取值为 1，否则取值为 0。约定当 $i=j$ 时， $p_{ij}=0$ 。易知当 $p_{ij}=1$ 时， $p_{ji}=0$ 。

通过人工输入的方式来获得零部件之间的装配优先关系，从而建立约束邻接矩阵。下面仍以图 2-2 所示圆珠笔为例，介绍装配优先矩阵的建立过程。图示圆珠笔包含 4 个零件，则建立约束邻接矩阵为一个 4 阶的方阵；然后确定矩阵中各个元素值，例如根据经验，知道零件 3 常常紧邻着零件 1 或 2 之后装配，则对应元素 p_{13}, p_{23} 的值均为 1；可以通过同样的方法来依次确定矩阵中其它元素的值，最后得到圆珠笔的约束邻接矩阵如图 2-6 所示。同理可以获得图 2-4 所示天平的约束邻接矩阵如图 2-7 所示。

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

图 2-6 圆珠笔的约束邻接矩阵

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

图 2-7 天平的约束邻接矩阵

Fig.2-6 Precedence constraint matrix of ball point pen

Fig.2-7 Priority constraint matrix of balance

2.2.3 位置姿态矩阵

在虚拟装配环境中对零件的装配过程，是对零部件模型施加约束使定位至目标位置的过程，该过程的实质就是通过零部件的位置姿态矩阵不断调整零件的空间位姿最终使零部件移动到目标位置的过程^{[3][39]}。零部件的位置姿态矩阵定义如下：

$$PA = \begin{bmatrix} p_{11} & p_{12} & p_{13} & 0 \\ p_{21} & p_{22} & p_{23} & 0 \\ p_{31} & p_{32} & p_{33} & 0 \\ x & y & z & 1 \end{bmatrix} \quad (2.3)$$

它与装配体的零件数目无关，始终是一个四阶方阵。其中元素 p_{ij} 包含着当前零部件的空间位姿信息，即零部件的位置和姿态信息，其值是在零部件的局部坐标系下获得的，只与零部件自身以及局部坐标系的三个坐标轴相对于全局坐标系的方向矢量有关^[39]。元素 x, y, z 则表示当前零部件的位置信息，代表零部件的局部坐标系原点与全局坐标系原点之间的相对位置。在本文中，各个装配零部件的位置姿态矩阵由 Delta3D 引擎获得^[3]，有了各个零部件的位置姿态矩阵就可以调整零部件模型的位置和姿态，具体表现形式如旋转、平移等，从而完成各个零部件的装配。获取的图 2-4 所示天平模型中零件 3 和零件 4 的位置姿态矩阵如图 2-8 所示。

$$\begin{aligned} \text{a) 零件 3} \quad & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0.399377 & -0.680108 & 0.25072 & 1 \end{bmatrix} \\ \text{a) Part 3} \quad & \text{b) 零件 4} \quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0.383734 & 1.43354 & 11.2126 & 1 \end{bmatrix} \\ & \text{b) Part 4} \end{aligned}$$

图 2-8 天平部分零件的位置姿态矩阵

Fig.2-8 Position and attitude matrix of some parts in balance

2.2.4 重力支撑矩阵

装配序列评价涉及到采用哪些参数评价一个装配序列好坏，优化则通过遗传算法进行进化来获取更优的装配序列。本文选取装配体不稳定性和装配方向改变次数作为装配序列评价参数，以获得更低的不稳定性和更少的装配方向改变次数作为优化目标。其中不稳定性由装配序列和重力支撑矩阵通过运算获得，装配方向改变次数则由装配序列和全局装配干涉矩阵通过运算获得。通过分析零件间的重力支撑关系获取重力支撑矩阵^[30]，包含 n 个零件的装配体重力支撑矩阵为 n 阶方阵，即：

$$S = \begin{bmatrix} s_{00} & s_{01} & \cdots & s_{0,n-1} \\ s_{10} & s_{11} & \cdots & s_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{n-1,0} & s_{n-1,1} & \cdots & s_{n-1,n-1} \end{bmatrix} \quad (2.4)$$

其中 s_{ij} 代表零件 i 和零件 j 的重力支撑关系。若零件 i 重力支撑零件 j ，则取值为 1，否则取值 0。约定当 $i=j$ 时， $s_{ij}=0$ 。根据定义可以依次确定重力支撑矩阵中所有元

素的值。

为了更详细地解释重力支撑矩阵的获得过程，仍以图 2-4 所示天平为例，因其包含 8 个零件，所以重力支撑矩阵为一个 8 阶的方阵；接下来先确定矩阵中元素 s_{01} 的值，假设在装配体的上面把零件 0 移走，没有了零件 0 的支撑作用，零件 1 将不再稳定，受到重力作用会下落，所以零件 0 对零件 1 有重力支撑关系，那么元素 s_{01} 的值应取值为 1；同理可以依次确定图 所示装配体的其它零件之间的重力支撑关系，进而建立重力支撑矩阵如图 2-9 所示。

$$\begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

图 2-9 天平的重力支撑矩阵

Fig.2-9 Gravity support matrix of balance

2.2.5 装配方向矩阵

方向矩阵是由多个装配序列中各个零件的可行装配方向为元素组成的，是装配序列规划过程中生成的中间数据，矩阵的每一行对应一个装配序列的每个零件的可行装配方向，其随着装配序列的变化而变化。对于 n 个零件组成的 m 个装配序列来说，其方向矩阵是一个 m 行 n 列的矩阵。其模型如下所示：

$$D = \begin{bmatrix} d_{11X}d_{11/X}d_{11Y}d_{11/Y}d_{11Z}d_{11/Z} & \cdots & d_{1nX}d_{1n/X}d_{1nY}d_{1n/Y}d_{1nZ}d_{1n/Z} \\ \vdots & \ddots & \vdots \\ d_{m1X}d_{m1/X}d_{m1Y}d_{m1/Y}d_{m1Z}d_{m1/Z} & \cdots & d_{mnX}d_{mn/X}d_{mnY}d_{mn/Y}d_{mnZ}d_{mn/Z} \end{bmatrix} \quad (2.5)$$

它是由全局装配干涉矩阵通过一定算法运算获得的，所以其组成元素的形式与全局装配干涉矩阵的元素结构类似，也是由六位二进制数组成的。其中， d_{ijD} 是一个二进制数，取值范围为 0 和 1，代表第 i 个装配序列中第 j 个零件在 D 方向上的装配可行性，其中 $D \in \{\pm X, \pm Y, \pm Z\}$ ，/表示负号，则在六个方向上的装配可行性就组成了装配方向矩阵的一个元素即 $d_{ijX}d_{ij/X}d_{ijY}d_{ij/Y}d_{ijZ}d_{ij/Z}$ 。根据定义可以依次获得矩阵的全部元素，

从而建立方向矩阵。同全局装配干涉矩阵一样，为了便于表示和计算，装配方向矩阵的元素也是以十进制数的形式表示和存储的。

对于一个含有 8 个零件的装配体，它的一个装配序列为：0—>4—>3—>5—>1—>2—>6—>7，通过全局装配干涉矩阵判断该序列是几何可行性的，同时可以获得它每一步的装配可行性函数的输出结果依次为：111111、001001、000100、000100、000001、000001、000001、000001，对应十进制表示为：63、9、4、4、1、1、1、1，则可以构成装配方向矩阵的一行元素，由装配方向矩阵定义可知每个零件的可行装配方向集依次为： $\{+X-X+Y-Y+Z-Z\}$ 、 $\{+Y-Z\}$ 、 $\{-Y\}$ 、 $\{-Y\}$ 、 $\{-Z\}$ 、 $\{-Z\}$ 、 $\{-Z\}$ 、 $\{-Z\}$ ，可以由若干个这样的可行装配序列来获得对应的装配方向矩阵。

2.3 本章小结

本章对虚拟装配信息建模技术以及虚拟装配信息建模所需信息进行了介绍和分析，并结合三维虚拟实验室及其中虚拟装配环境的特点，选取了合适的装配建模信息以及装配信息模型，提出可行装配序列的定义以及需要满足的约束条件即几何约束和人为约束，综合利用了几何信息和相关专业人员的经验知识，相比传统的单一依靠几何信息或者经验知识的方法具有效率高，规划所得装配序列质量高，实现了“个性定制”的功能；最终通过建立全局装配干涉矩阵、约束邻接矩阵、位置姿态矩阵、重力支撑矩阵以及装配方向矩阵五种矩阵来共同表示装配信息，相比传统的基于树或图的建模方法，本文提出的用五种矩阵对装配信息建模的方法具有很大的优势，简化了装配信息模型，提高了装配信息建模效率，用矩阵表示装配信息也便于计算机处理，降低后续装配序列规划的复杂度，并提高装配序列规划效率，有利于克服当装配零件较多时易出现组合爆炸的缺点。文中给出了五种矩阵的具体定义，并举出几种装配体的实例来详细地描述了各类装配信息的建模过程和建模结果，从而为后续的装配序列生成、装配序列评价与优化以及装配序列的应用等环节提供良好的信息储备。

第三章 三维虚拟仪器装配序列规划方法

3.1 三维虚拟仪器装配序列生成

在前面对装配序列规划技术分析的基础上, 本文提出利用精确计算的方法来获得可行装配序列。装配序列生成主要是以生成一个或者多个可行的装配序列为目的^[40], 根据装配信息或模型通过一定的推理或者算法获得几何可行以及人为可行的装配序列, 即满足几何约束和人为约束的装配序列, 而进行的一系列的活动的。在本文的虚拟装配环境中, 选取全局装配干涉矩阵、位置姿态矩阵和约束邻接矩阵作为进行装配操作的约束条件, 即依据这三种矩阵来获得可行的装配序列并进行装配操作, 不需要对大量的装配信息进行存储, 这样就可以选用任意格式的三维模型作为数据输入^[3]。

本文在虚拟装配过程中嵌入装配序列生成算法, 实现对操作者装配操作的引导。当操作者开始组装实验仪器, 并用鼠标点击选中某个零件, 准备拖拽移动零件时, 系统会判断当前选中的零件与之前已经装配好的零件构成的新的装配序列是否满足装配干涉矩阵和约束邻接矩阵的约束, 即判断新序列是否是几何可行和人为约束可行的, 并给出相应的提示信息。操作者就可以根据提示信息, 实施下一步正确的装配动作, 手动拖拽零件实现装配, 如果移动过程中零件的位置与最终装配位置的距离小于设定的阈值, 则突出显示选中零件并根据位置姿态矩阵自动调整其位置状态完成该零件的装配, 并更新当前序列和当前装配方向表。等待下一次的装配操作。这样就通过装配序列生成算法达到装配序列的实时生成与对装配操作的实时引导的目的。虚拟装配操作流程如图 3-1 所示。

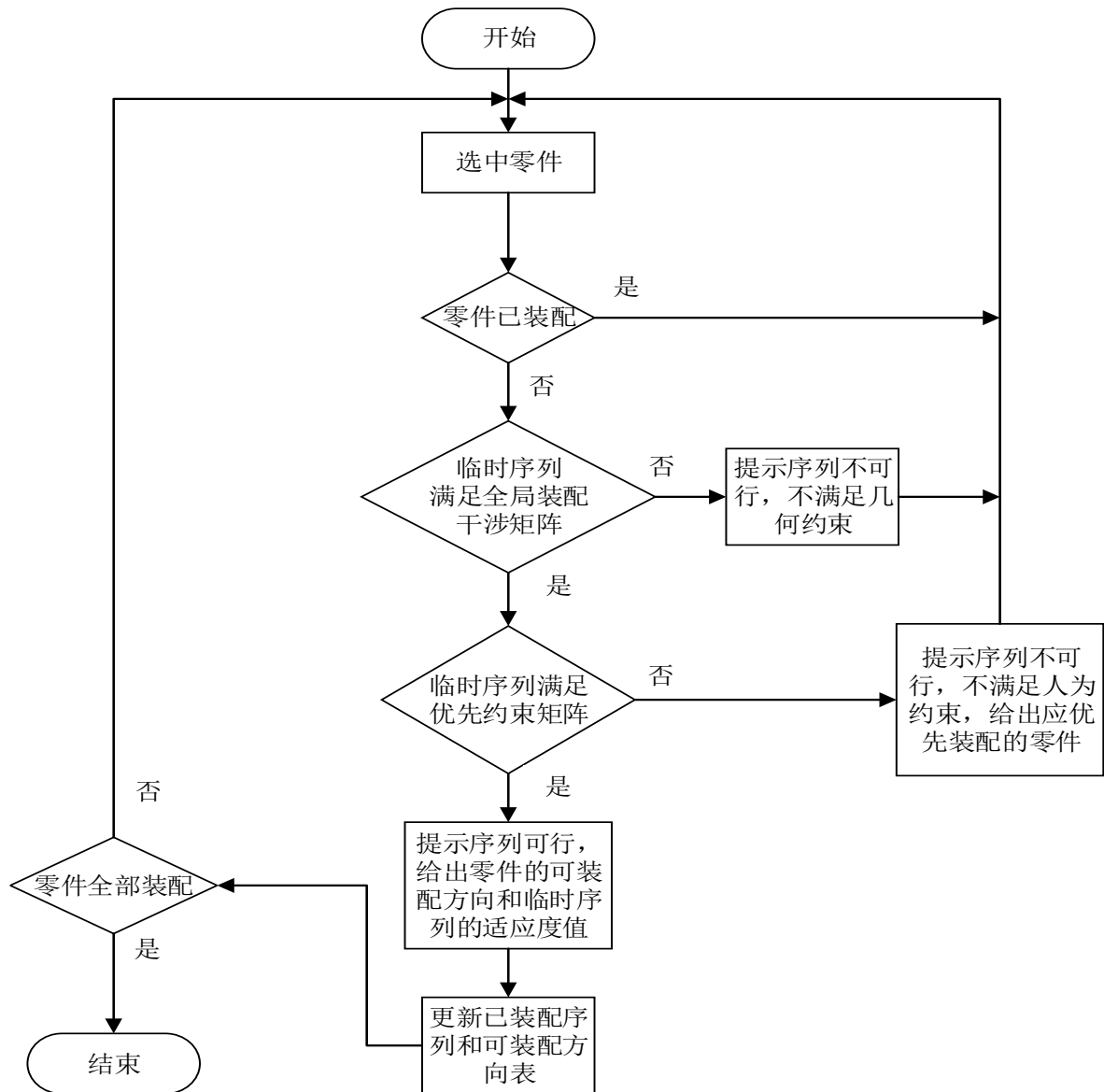


图 3-1 虚拟装配操作流程

Fig.3-1 Operation process of Virtual assembly

3.1.1 几何约束满足判定

首先,一个可行的装配序列必须满足装配体零部件之间的几何方面的装配约束关系,即装配序列必须在几何上是可行的,几何可行是指零件的装配过程中不应有干涉情况^[33],即零件是可装配的^[41],装配序列的几何可行性可以通过装配序列中的所有零部件是否都存在可行的装配方向来确定^[41],若存在则序列几何可行,否则几何不可行。本文根据全局装配干涉矩阵来获得装配序列中各个零件的可行装配方向,从而判断装配序列的几何可行性。与仿真结合的算法具体实现形式是每次选择一个零件就执行如图

3-2 所示的程序流程来判断已装配零件与当前选中零件组成的临时装配序列是否满足几何约束，这样最终实现对所有的装配序列的几何可行性的判定。其中，已装配零件必须也是经过几何约束判定的，即满足几何约束。

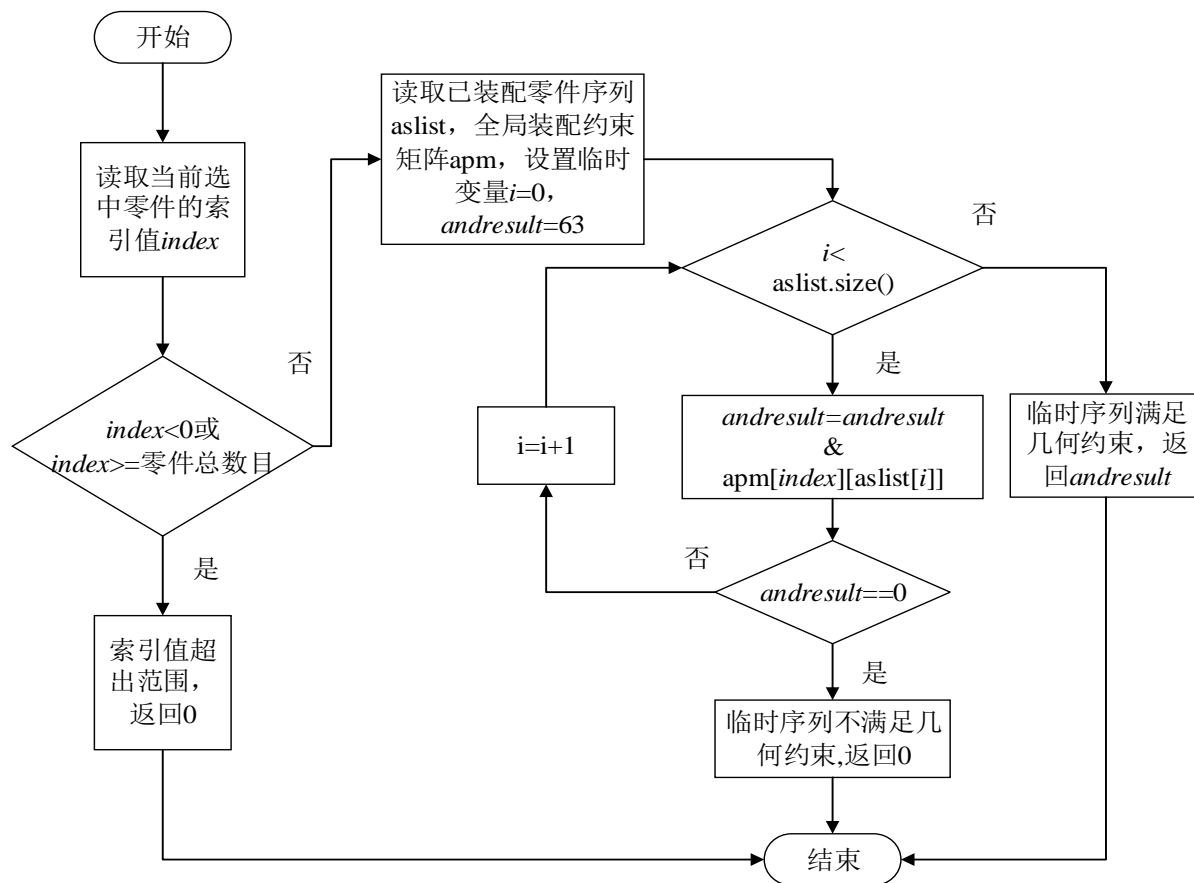


图 3-2 临时装配序列的几何可行性判定

Fig.3-2 Geometric feasibility determination of temporary assembly sequence

本文提出的三维虚拟实验室系统中实现图 3-2 所示流程的对应函数为：int InstrumentObject::isProbable(size_t index)，当选中一个物体时，该函数判断并返回选中物体与之前装配好的物体组成的装配可行性对的相与结果，结果若不为 0 则可行，否则不可行。

以上是对一个由满足几何约束的已装配序列和待装配的选中零件组成的临时序列是否满足几何约束的判定流程，对于一个给定的装配序列，它可以看作是多次选择零件进行装配的结果，所以对于一个给定的装配序列几何可行性的判定，则可以通过对多次选择零件时形成的多个临时装配序列执行图 3-2 所示的流程来判断其几何可行性，若有任何一个临时装配序列不满足几何约束，则给定装配序列几何不可行，否则给定装配序列几何可行。对给定装配序列的几何可行性判定具体流程如图 3-3 所示。

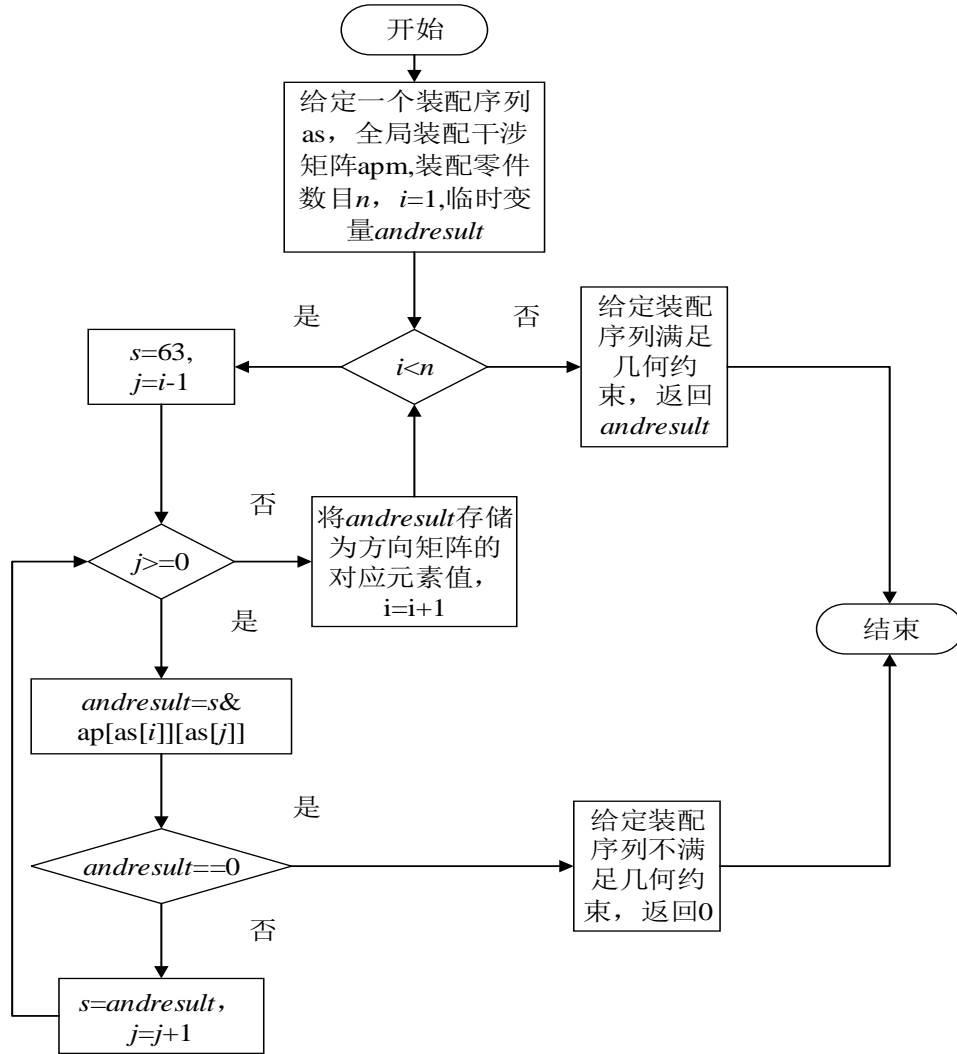


图 3-3 给定装配序列的几何可行性判定

Fig.3-3 Geometric feasibility determination of a given assembly sequence

对于图 2-2 所示圆珠笔的装配, 给定它的两个包含全部零件的装配序列 0->2->1->3 和 3->0->1->2, 根据前面章节获得的圆珠笔的全局装配干涉矩阵以及图 3-3 所示算法容易得知第一个给定序列是满足几何约束的, 而第二个给定序列是不满足几何约束的。

3.1.2 人为约束满足判定

人为可行序列即人们允许的装配序列, 有些序列不符合个人的喜好或者与实际装配过程的经验不符, 不是一个性能好的序列或者不为个人所喜爱。本文通过人为输入获得人为指定的装配优先关系, 建立约束邻接矩阵, 在获得几何可行序列的基础上, 根据该矩阵来对装配序列的人为可行性进行判断, 进而获得一个即满足几何约束又满足

人为约束的装配序列，即可行的装配序列，实现了可行装配序列的生成。如图 3-4 和图 3-5 所示分别为判断虚拟装配过程中构成的临时装配序列和给定装配序列是否满足人为约束的流程图。

虚拟装配过程中对临时装配序列的人为约束进行判定的对应实现函数为：
std::string InstrumentObject::_findUnfinishedAdjacentPart(size_t index)，当选中一个物体时，该函数判断与该物体相关的应该事先安装的物体是否安装妥当，如果没有安装妥当则返回该物体的名称，否则返回空字符，进而可以根据函数的返回值判断选中零件与已装配零件序列构成的临时序列是否满足人为约束。

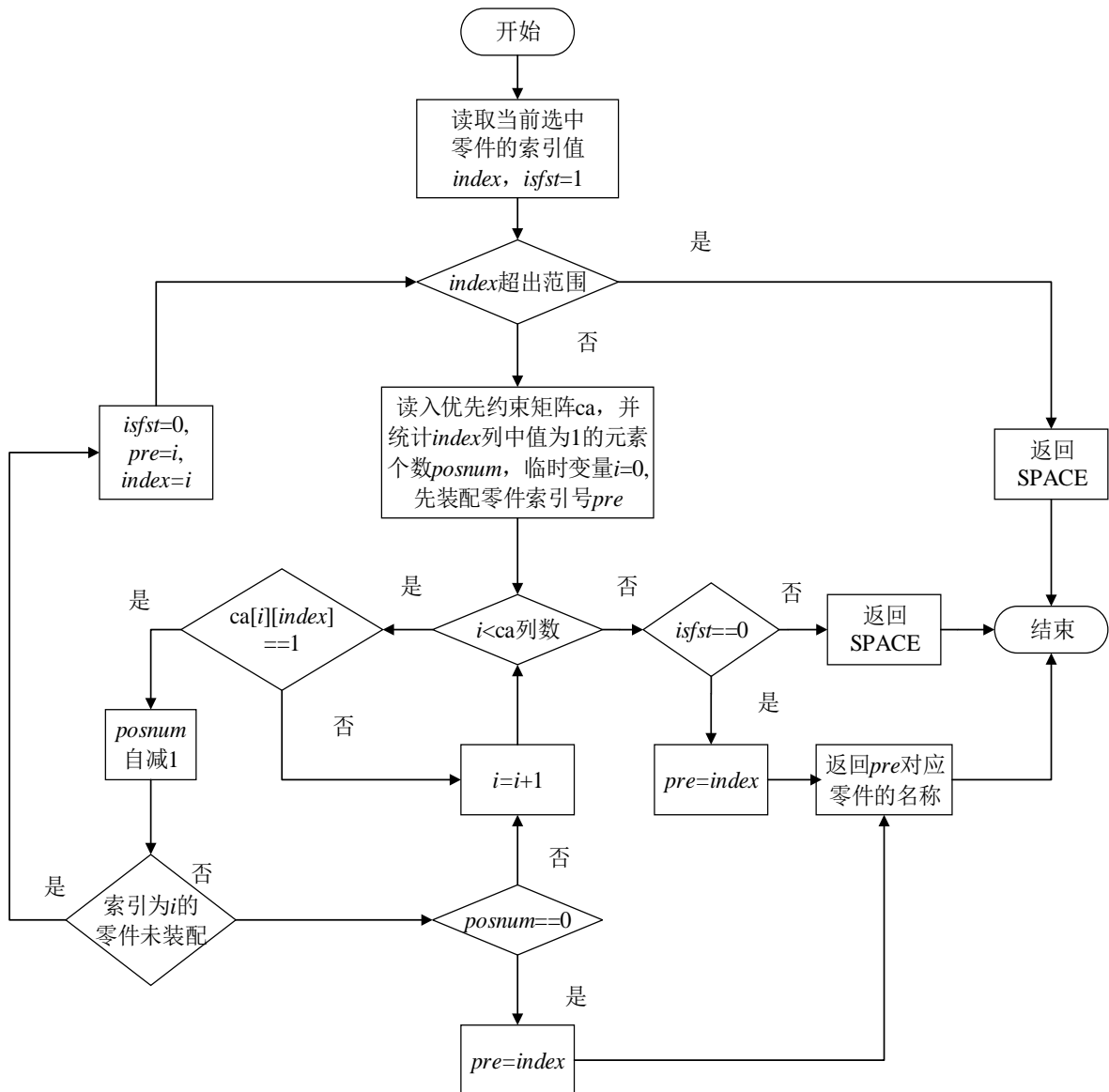


图 3-4 临时装配序列的人为约束判定

Fig.3-4 Artificial constraint determination of temporary assembly sequence

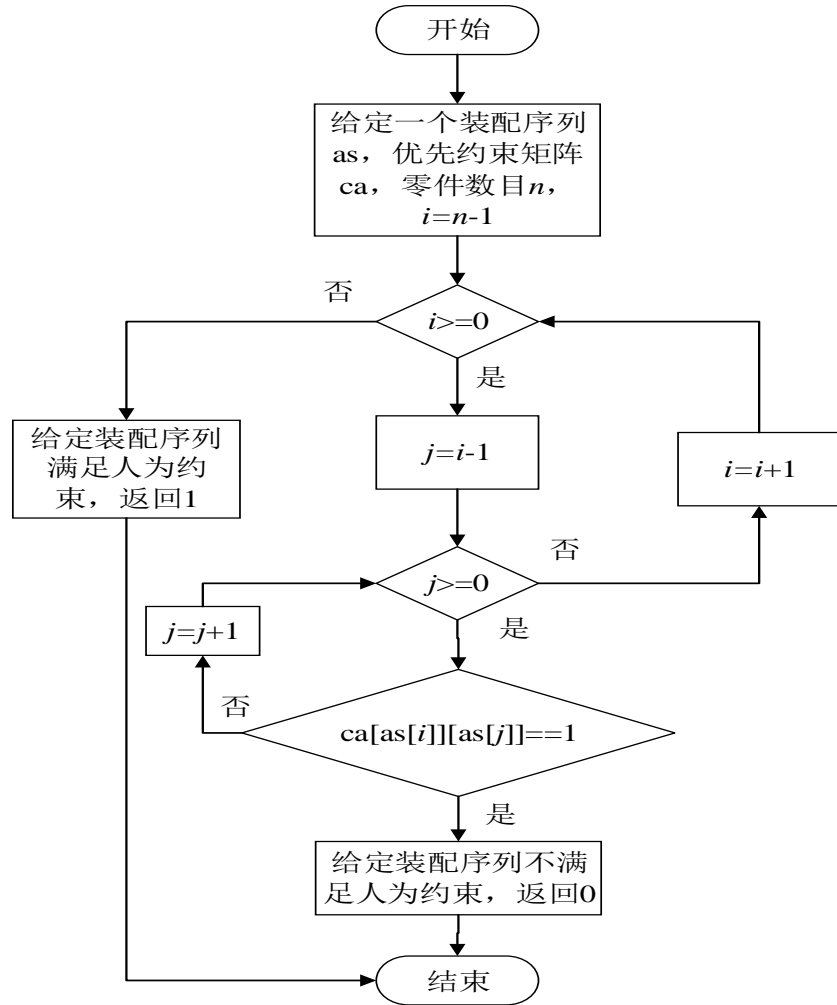


图 3-5 给定装配序列的人为约束判定

Fig.3-5 Artificial constraint determination of a given assembly sequence

3.2 三维虚拟仪器装配序列评价

装配序列评价与优化是以获得最优的或者接近最优的可行装配序列为目的, 为实际的产品设计、规划、装配以及制造活动提供参考或指导的活动^[17]。进行装配序列评价的主要目的就是为后续的装配序列优化活动建立优化的目标, 最终通过装配序列优化获得各方面性能最优或者接近最优的序列, 最终可以对实际的装配过程进行指导。装配序列评价是在生成一个或者多个可行的装配序列的前提下, 选取合适的参数, 来评价这些可行装配序列的好坏程度。对于一个装配体, 往往存在着多个可行装配序列^[17], 尤其是对于一个复杂的装配体, 其可行的装配序列更是不计其数^[42], 所以有必要做进一步的筛选从而获得装配成本较少、装配效率较高的装配序列^[17], 这就涉及到采用什

么样的评价方法或者评价标准的问题。一个好的装配序列除了首先要满足可行性要求以外，还需要考虑人们对装配成本以及装配稳定性等的要求^[42]。一般，装配过程应该满足以下几个要求：（1）可行性^[42]，即可装性，装配过程中不应发生干涉，它是基本前提条件；（2）并行度^[42]，对于一个给定的装配体，可以同时进行的装配操作越多则并行度越大，并行度越大则装配序列越好；（3）稳定性^[42]，与装配过程中需要人为施加约束才能保持已装配好的零部件稳定的操作数有关，这样的操作数越少稳定性越好，则装配序列也越好；（4）聚合性^{[17][42]}，是指一个装配序列反映的装配过程中相似装配操作的执行是否集中，越集中说明聚合度越高，其集中程度决定着一些装配工具的更换次数，次数越少装配序列越好，即聚合度越高装配序列越好；（5）重定向次数^{[17][42]}，执行一个装配操作需要对零件进行翻转等来调整角度的次数，重定向次数越少装配序列越好^[25]，一个装配序列的装配方向改变次数与重定向次数成正比，两者可以互换。本文的实现的虚拟装配是串行装配，不存在并行装配操作，且装配对象是实验仪器，所以很少涉及装配工具的更换，所以并行度和聚合性对本文实现的虚拟装配影响不大。本文主要以获得较高的装配过程稳定性以及较少的装配方向改变次数为评价目标，同时在虚拟装配过程中结合装配序列的人为约束满足程度一起实现装配序列的优化，并以装配过程稳定性、装配方向改变次数和人为约束满足程度为参数建立了优化活动的目标函数即遗传算法的适应度函数。

3.2.1 装配过程稳定性

在实际的装配过程中，装配过程的稳定性具有很大的意义，装配过程的稳定性影响着装配操作的可靠性以及装配工具的复杂性^{[33][43]}。在传统的虚拟装配环境中，一般实现的仿真效果都缺乏对物理现象的动态仿真，即仿真环境中的零部件失去了在现实环境中的物理属性如重力下的下沉现象、碰撞之后的反弹现象等，实现的是整个虚拟装配过程都是稳定且失重的状态，为了使装配序列规划后的结果对实际的装配过程产生指导作用，必须考虑装配过程的稳定性，一个装配序列体现了实际装配过程中的零件装配步骤，为了将装配序列的装配过程稳定性量化表示^[43]，本文从各个零件在装配到最终位置后是否有稳定的支撑这一角度来分析^[30]，用所有零件不存在稳定支撑的总数目来反映一个装配过程的稳定性，具体可以由装配序列和重力支撑矩阵通过一定运算获得。对于一个可行装配序列，其装配过程越稳定越好。通过重力支撑矩阵来表示

装配零件之间的重力支撑关系，然后依次计算装配序列中的各个零件与支撑它的且在其后面装配的所有零件的装配顺序间隔数目，并累加，来表示装配序列的装配过程的稳定性。一个序列的装配过程稳定性定义如下所示：

$$f_1 = \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} t_{i,j}, t_{i,j} = \begin{cases} j-i, \text{第} j+1 \text{个零件重力支撑第} i+1 \text{个零件} \\ 0, \text{其他} \end{cases} \quad (3.1)$$

其数值越小代表装配序列越稳定，反之，装配序列越不稳定。相比其他文献中的方法^[30]，本文提出的装配序列评价方法只需建立重力支撑矩阵一次，不需要对重力支撑矩阵进行不断的更新，极大地减少了计算量。

3.2.2 装配方向改变次数

为了提高装配的效率，应尽可能减少装配方向改变次数^[44]。对于一个可行的装配序列，由装配序列和全局装配干涉矩阵，通过前面介绍的装配方向矩阵的建立方法可以获得一个装配序列的各个零件的可装配方向组成的集合^[43]，进而对序列中所有两两相邻零件的可装配方向求交集，若交集为空，则相邻两个零件的装配方向必然发生了改变，否则，没有发生改变，每发生改变一次则累加 1，最后累加得到的次数即为该序列的装配方向改变次数。具体公式如下：

$$f_2 = \sum_{i=1}^{n-1} t_i, t_i = \begin{cases} 1, \text{第} i \text{个零件与第} i+1 \text{个零件的可行装配方向集交集为空} \\ 0, \text{其他} \end{cases} \quad (3.2)$$

3.2.3 人为约束满足程度

一个好的装配序列，在不违背几何约束的前提下，需要其尽量满足人为约束的要求。为了将装配序列人为约束满足程度量化表示，本文从装配序列中各个零件和与其存在约束邻接关系的零件是否满足约束邻接关系即人为约束关系的角度来分析，具体可由装配序列和装配约束邻接矩阵通过一定运算获得。对于一个几何可行的装配序列，可以依次计算装配序列中的各个零件与优先于它的且在其后面装配的所有零件的装配顺序间隔数目，并累加，来表示一个装配序列的人为约束满足程度。其具体定义如下：

$$f_3 = \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} t_{i,j}, t_{i,j} = \begin{cases} j-i, \text{第} j+1 \text{个零件优先于第} i+1 \text{个零件装配} \\ 0, \text{其他} \end{cases} \quad (3.3)$$

其数值越小代表装配序列的人为约束满足程度越高，越符合人为约束的要求，即越符合人们的特定需求。本文提出的装配序列评价方法只需要建立约束邻接矩阵一次，不需要对装配约束邻接矩阵进行不断的更新，极大地减少了计算量。

3.3 三维虚拟仪器装配序列优化

在前面对装配序列规划技术分析的基础上，本文采用启发式算法来进行装配序列的优化，目前已经应用于装配序列规划领域的启发式算法主要有蚁群算法、粒子群算法、模拟退火算法和遗传算法。其中蚁群算法普遍较为复杂，参数多，参数的选择往往也比较困难，搜索最优解的效率也比较低，容易陷入局部最优解，所获得解的质量往往不高，适合可用图表描述的组合优化问题，如路径优化和电路布线等问题；粒子群算法的参数比较少，收敛速度比较快，但是极易出现早熟收敛的问题，也容易获得不可行解；模拟退火算法的一些环节难以在实际设计算法时实现，往往不能够得到全局最优解，算法的稳定性也比较差；而遗传算法则具备算法原理和步骤直观易懂，收敛速度快，编程实现时的代码简洁，容易获得全局最优解，解的质量高，易实现，实用性强，易扩展的特点。相比其他启发式算法而言，遗传算法往往更稳定，更高效，所以遗传算法最常被用于装配序列规划问题的研究，所以本文也采用遗传算法进行装配序列的优化，但是传统的遗传算法会产生大量的不可行解从而降低算法的搜索效率，且易出现早熟现象，为了克服这些缺点，本文对传统遗传算法做出了改进，提出了一种改进的遗传算法用于实现更好的装配序列优化效果。

装配序列优化是在装配序列评价活动的基础上，用评价装配序列好坏所采用的参数建立一个合适的适应度函数模型，以每个装配序列和其装配信息为输入，将每个装配序列的好坏程度量化表示，并通过合适的算法和操作对可行装配序列进行优化，最终得到最好或者接近最好的装配序列。本文以获得更高的稳定性、更少的装配方向改变次数和更高的人为约束满足程度为装配序列优化目标，并根据以上三个参数建立适应度函数，利用改进的遗传算法对装配序列进行进化，最终获得一个或多个最优或者接近最优的装配序列。为了保持种群的多样性和算法的效率，采用自动随机生成若干个初始种群和手动输入初始种群相结合的方法^[33]，自动生成若干个随机的满足几何约束的装配序列，若生成失败，则请求手动输入初始序列种群；然后对当前种群进化，保留适应度值最高的两个序列直接作为下一代的两个序列，下一代种群的余下成员则

通过随机选择当前种群的两个序列，以一定概率执行交叉，突变等操作来生成，这样既能保证每次进化都能保留最优的序列，即确保了进化过程的收敛性，又能保证进化过程种群的多样性，遗传算法也不会过早收敛；由于在执行交叉以及突变的过程中会生成大量的不可行序列，降低遗传算法的效率^[25]，为了淘汰大量的不可行序列，加快遗传算法的搜索速度，所以本文结合全局干涉矩阵和约束邻接矩阵，将前面章节所述的可行装配序列的判定算法嵌入到了种群进化的过程中，从而实现种群的不断进化。本文的装配序列优化算法如图 3-6 所示。

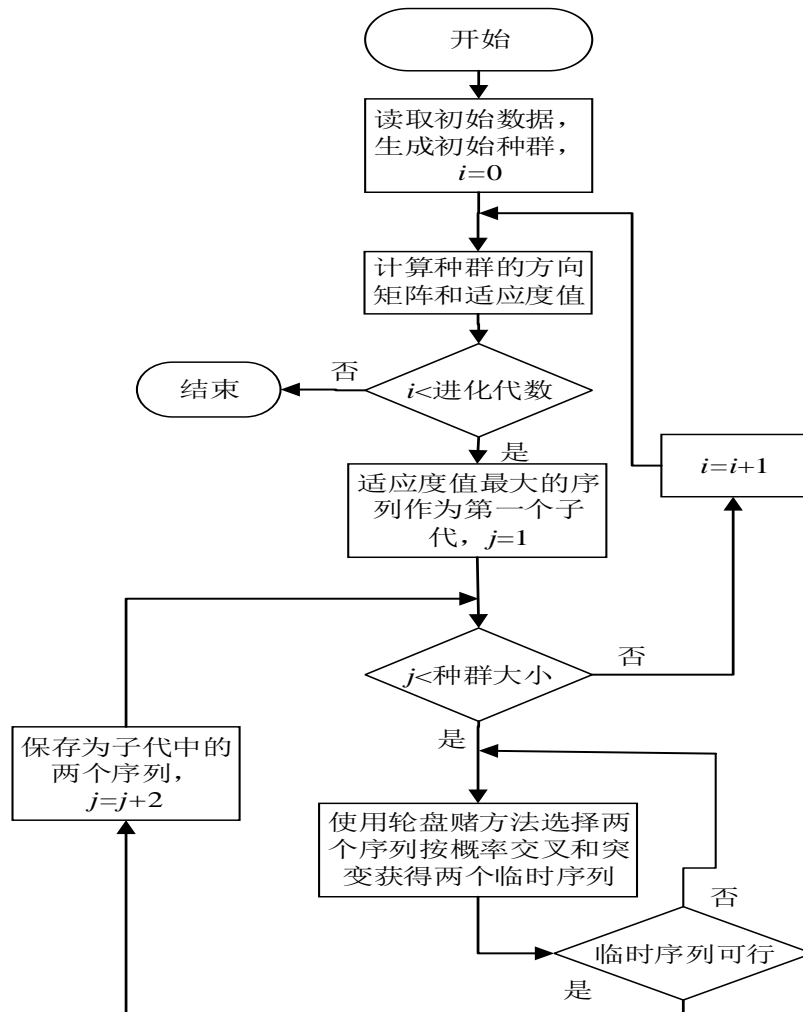


图 3-6 装配序列优化算法

Fig.3-6 Assembly sequence optimization algorithm

本文为了保持装配序列的多样性并防止遗传算法过快收敛产生局部最优解^[45]，把必须满足的几何约束作为初始种群中和进化过程中装配序列的基本筛选条件，而将需要尽量满足的装配序列的人为约束满足程度作为优化目标之一，并赋给较大的权值，从而保证装配序列优化算法向着人为可行的目标搜索最优解。即其中装配序列优化算

法即本文提出的改进的遗传算法中可行的装配序列是指满足几何约束即全局装配干涉矩阵的装配序列。

遗传算法是一种自适应的随机搜索算法^{[25][30]}，其模拟自然界中的遗传和进化，是一个迭代过程，有很好的鲁棒性以及全局优化搜索能力，常常适用于组合优化问题的求解^[25]。遗传算法提供了一种通用的算法框架，其五个关键要素是染色体编码、适应度函数、初始种群、进化操作以及迭代进化过程的控制参数^[43]，凭借其高效、简单、实用的优点，在许多领域得到了应用且取得了很好的效果，本文即是采用改进的遗传算法来优化装配序列，对其中的五个关键要素详细介绍如下：

3.3.1 染色体编码

首先需要针对装配序列提出一种遗传算法中染色体的编码方式，其在一定程度上决定了遗传算法中进化的方式和效率^{[25][46]}。本文根据装配序列的特点，采用整数型数组来对装配序列进行编码表示和存储，假若一个装配体有 n 个零件（为了与矩阵和数组的下标表示一致，零件索引从 0 开始， n 个零件的索引号分别为 $0, 1, 2, \dots, n-1$ ），则其对应的完全装配序列编码之后是一个大小为 n 且由数字 $0, 1, 2, \dots, n-1$ 组成的排序数组，针对装配序列的特点即装配序列中装配元素的唯一性，要求编码之后的同一个个体中同一个数字不可重复出现^{[33][45]}，假若数组的标号为 i 的元素值是 j ，则代表第 $i+1$ 个装配的零件索引号为 j ，如一个染色体是 $5\ 0\ 2\ 1\ 3\ 4$ ，则代表一个由 6 个元素组成的装配体中 6 个零件（零件索引号分别为 $0, 1, 2, 3, 4, 5$ ）的装配序列为 $5 \rightarrow 0 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4$ ，编码之后的染色体的长度为 n ，包含 n 个基因，每一个基因的位置对应零件的装配顺序，基因值对应零件的索引号，并且种群进行初始化以及各种进化操作时一个染色体中各零件的索引号必须出现且仅出现一次^{[33][43]}。此种编码方式易于理解，编码方式简单，符合人们书写装配序列的习惯，且很适合计算机处理和存储，也容易进行解码。如下图所示为具有 8 个零件的装配体的一个序列对应的染色体。

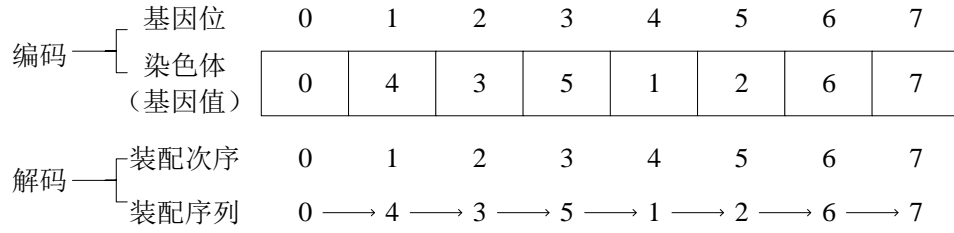


图 3-7 染色体编码与解码格式

Fig.3-7 Encoding and decoding format of chromosome

3.3.2 适应度函数

其次需要建立合适的适应度函数，适应度的大小代表着个体的优劣程度，从而将装配序列的评价指标量化，评估种群中的生成的各个序列，并给遗传算法提供一个合适的进化依据，适应度函数直接决定着遗传算法的收敛性。一个个体的适应度越大，其生存能力越强，所以其遗传到下一代的概率也越大^[45]。而且适应度函数的具体表达形式与具体的应用场景密切相关，且不唯一。本文选取了装配过程稳定性、装配方向改变次数和装配序列人为约束满足程度为评价参数，根据前面章节对装配序列评价以及三个评价参数的分析可知，本文中遗传算法的适应度函数应是关于以上三个评价参数为自变量的函数，其定义如下：

$$benefit = Max - (\alpha \times f_1 + \beta \times f_2 + f_2 + \gamma \times f_3) \quad (3.4)$$

其中 f_1 为装配序列的不稳定性， f_2 为装配方向改变次数， f_3 为装配序列的人为约束满足程度， $f_1 + f_2 + f_3 = 1$ ； Max 的设置是为了将问题的求解转换为求函数的最大值问题，且根据具体应用中装配体的复杂性和零件的数目设置，较好的 Max 值是与 $\alpha \times f_1 + \beta \times f_2 + f_2 + \gamma \times f_3$ 的最大值最为接近且向上取整的数值，使得仿真结果可以较明显地反映进化过程中种群的整体优劣程度的变化规律，比如，本文最终进行仿真时取 Max 值为 100 时装配序列优化对最优解的趋近效果比较理想^[44]； α 、 β 与 γ 分别为装配序列的稳定性、装配方向改变次数和装配序列人为约束满足程度的权重系数^[45]。一个装配序列首先必须是几何可行的，其次要以满足人为约束为首要目标，然后再寻求获得更好的装配过程稳定性以及更少的装配方向改变次数，所以本文根据需要设置 $\alpha = 0.2$ ， $\beta = 0.1$ ， $\gamma = 0.7$ 。本文提出的计算种群中个体的适应度值的具体定义如下：

若 n 个零件组成的一个装配序列按照本文方法编码后对应的染色体存储在数组

$chr[n]$ 中, 且重力支撑矩阵为 bm , 方向矩阵为 adm , 约束邻接矩阵为 cm , 则染色体即种群中的个体的装配序列稳定性和装配方向改变次数对应表达式分别为:

$$f_1 = \sum_{i=0}^{n-2} \sum_{j=i+1, bmch[chr[j]][chr[i]]==1}^{n-1} (j-i) \quad (3.5)$$

$$f_2 = \sum_{i=1, adm[i] \& adm[i-1]==0}^{n-1} 1 \quad (3.6)$$

$$f_3 = \sum_{i=0}^{n-2} \sum_{j=i+1, cm[chr[j]][chr[i]]==1}^{n-1} (j-i) \quad (3.7)$$

对应的具体实现函数为: `int compute_benefit(vector<vector<int>> bmatrix, vector<vector<int>> cmatrix, int* admatrix, int * chromosome, int num_of_part)`, 其中函数的五个形参依次为装配体的重力支撑矩阵、约束邻接矩阵、装配序列对应的装配方向矩阵的一行、染色体即装配序列和装配体的零件数目。

3.3.3 初始种群生成

初始种群生成的方式以及其规模和质量很大程度影响着未来种群的进化方向, 也影响着进化的收敛性和全局优化搜索能力。如前面章节所述, 首先保证初始种群是由几何可行装配序列组成的, 否则将没有实际意义, 其次本文采用人工输入和自动生成相结合的方式来获得可行装配序列组成的初始种群^[33], 可以综合随机方式所带来的多样性以及人工输入方式带来的人的先验知识保证收敛速度的优点^[33], 同时可以应对装配序列可行解太少以致生成种群失败的异常情况, 也利于计算机编程实现^[33]。生成初始种群的算法流程如图 3-8 所示。

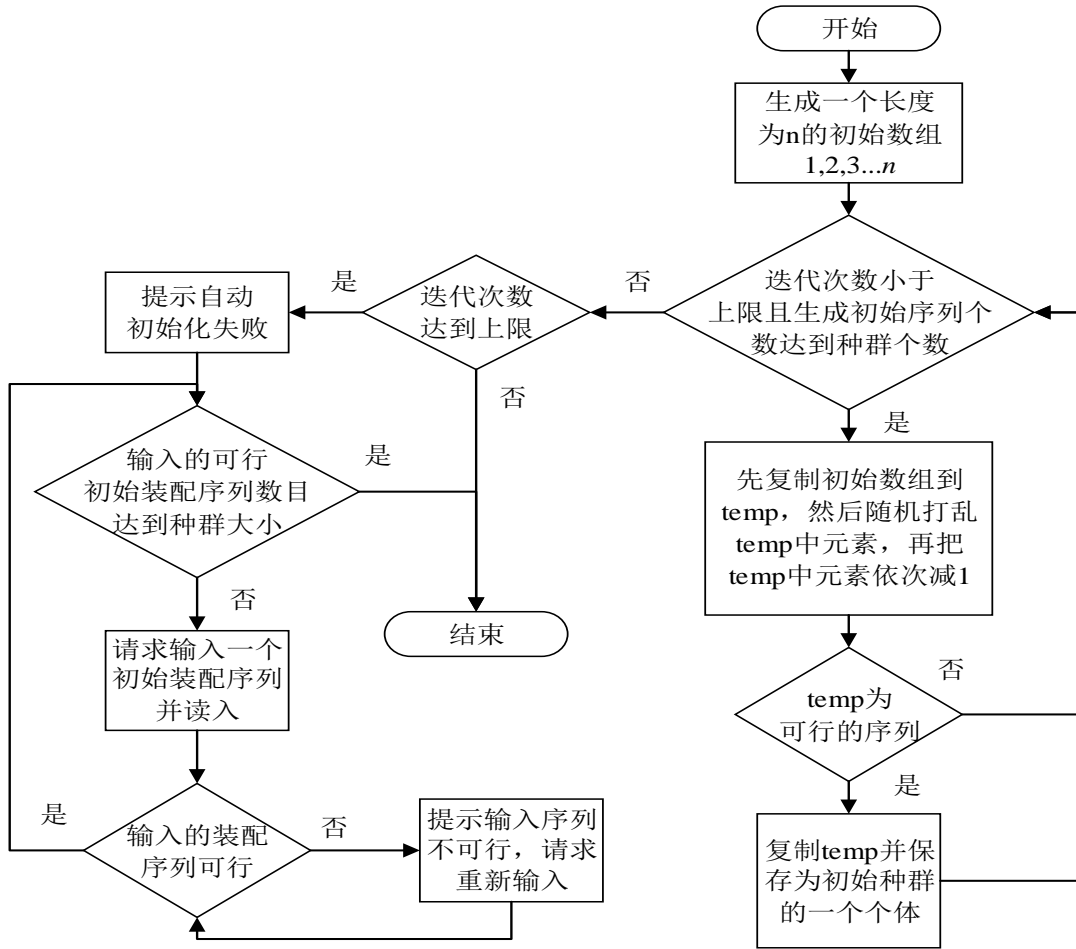


图 3-8 初始种群生成算法

Fig.3-8 Initial population generation algorithm

实现该功能的函数为: `void generate_colony (int** colony, int scale_of_colony, int num_of_part, vector<vector<int>> cmatrix, vector<vector<int>> apmatrix, int**admatrix)`。

3.3.4 遗传算法的进化操作

遗传算法的进化操作是整个迭代进化过程的最关键的部分。本文设计了三类操作来实现该过程, 包括选择操作、交叉操作和突变操作。在本文的遗传算法中交叉操作和突变操作是按概率执行的, 对应的概率分别为交叉概率 P_e , 突变概率 P_v 。

实现整个进化过程的函数为: `void ERV_colony (int **colony, float pe, float pv, int num_of_part, int scale_of_colony, vector<vector<int>> cmatrix, float* benefit_array, vector<vector<int>> apmatrix, int** admatrix)`。

(1) 选择操作

选择操作是遗传算法的基本操作，它的目的是模拟自然界中的“优胜劣汰”自然选择原则，尽可能地使优良的个体存活到下一代，使其基因得到更多的遗传，并且使得下一代种群数量与父代一致。由于遗传算法是一种随机搜索算法，而实际的装配序列规划问题的规模即种群的大小不会是理想的无穷大，为了减小选择误差，防止遗传算法早熟或者错误收敛^[47]，本文采用保留父代中适应度值最大即性能最优的个体直接作为子代的一个个体，即最优保存策略^[47]，这样可以保证进化过程的收敛性，然后进行这样一个迭代过程，即：根据“轮盘赌”方法随机选取父代中的两个个体^{[43] [45]}，然后对它们按概率依次执行交叉和突变操作，实现两个个体的进化和变异，“轮盘赌”方法中每个个体被选中的概率与其适应度值成正比，所以适应度值越高，个体被选中的概率也越大，从而使适应度较高即较优的个体的基因更能获得继承，保留在子代个体中。

“轮盘赌”算法流程如图 3-9 所示。实现“轮盘赌”方法的对应函数为：`int GetIndexofChromoRoulette(int scale_of_colony, float* benefit_array)`。

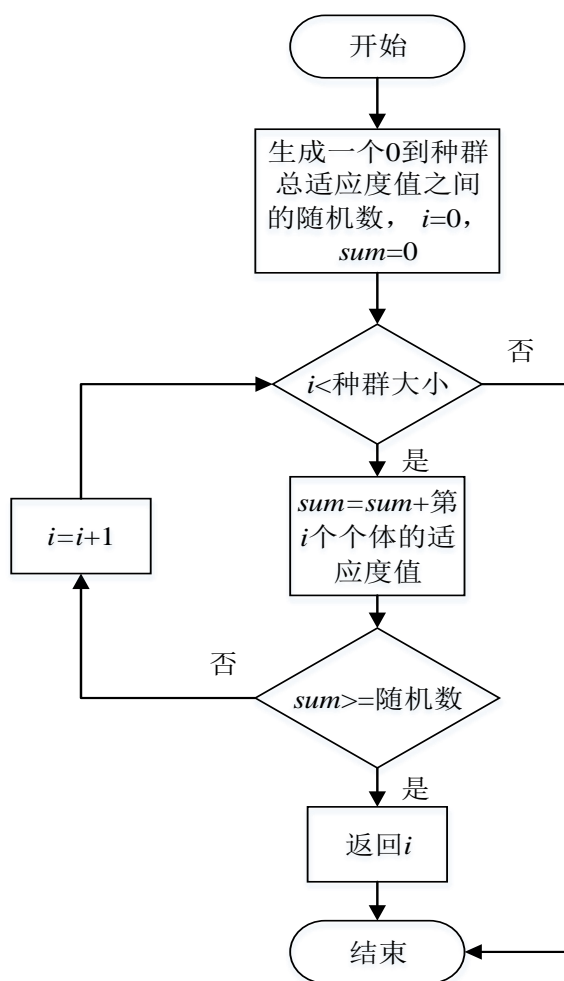


图 3-9 “轮盘赌”算法

Fig.3-9 Roulette algorithm

(2) 交叉操作

交叉操作模拟自然界中的基因重组过程^[47]，是遗传算法中进化过程的主要操作，它通过染色体对等位置的基因交叉操作，获得两个新的个体，即完成了将两个父代的染色体中优良基因遗传到子代中的过程。它是以两个染色体为操作单位的，交叉操作一般要满足以下几条原则：第一，必须保证优良的基因获得更多的遗传机会；第二，通过交叉操作生成优良基因是可行的；第三，交叉操作要保证染色体结构完整，因此要针对编码方式即染色体结构来设计，且要具有较高的效率^[47]。本文结合以上三条原则，并针对装配序列的特点，即序列中零件不可重复，每个零件必须出现且仅出现一次的特点，设计了一种改进的交叉操作。对应的实现函数为：`void exchange_gene(int * chromosome_first, int * chromosome_second, int num_of_part)`。为了详细地说明本文提出的交叉操作方法，下面以具有 8 个零部件的装配体为例，对其按本文方法进行染色体编码，并执行交叉操作，具体步骤如下：

①在父代种群中依据轮盘赌方法随机选择两个不同序号的个体 `parent1`、`parent2`，随机生成一个大于等于 0 且小于零件数目 8 的数字，即交叉点索引 `exchange_pos`，当 `exchange_pos=2` 时，用竖线标定交叉点位置如下。

`parent1=2 6 | 0 5 1 7 3 4`

`parent2=2 5 | 1 6 7 0 3 4`

②把 `parent1` 的交叉点位置之后的基因用 `parent2` 的全部基因代替，同理，把 `parent2` 的交叉点位置之后的基因用 `parent1` 的全部基因代替，获得的两个临时的染色体如下：

`parent1'=2 6 | 2 5 1 6 7 0 3 4`

`parent2'=2 5 | 2 6 0 5 1 7 3 4`

③将上一步生成的两个临时染色体各自的交叉点位置之后与交叉点位置之前相同的基因全部删除，得到的两个序列即为交叉操作之后的子代。

`children1=2 6 5 1 7 0 3 4`

`children2=2 5 6 0 1 7 3 4`

(3) 突变操作

突变操作模拟了生物进化过程中染色体上的某个或者多个基因发生突变，从而改变染色体的基因组成结构的现象^[47]。其保证了种群的多样性，本文的选择操作减小了算法的搜索空间，降低了种群的多样性，而突变操作则相反，其保证了遗传算法的全

局搜索能力，并和选择算法互补，一起保证算法可以收敛到最优解^[47]。突变是以一个染色体为操作单位的，鉴于遗传算法的应用对象是装配序列，序列中元素不重复，所以本文提出的突变操作做了改进，通过随机生成一个突变基因位和突变后的基因值，并寻找与突变后基因值相同的基因位，然后对两个基因位上的基因来进行交换从而实现染色体的突变。实现该功能的对应函数为：`void gene_mutate(int *chromosome, int num_of_part)`。以具有 8 个零部件的装配体对应的一个染色体为例执行突变操作，具体步骤如下：

- ①针对装配序列规划的特点，随机生成变异点和变异后的基因值，它们均为大于等于 0 且小于零件数目 8 的数字，设要进行突变操作的父代染色体为 2 1 0 7 4 5 6 3，随机生成的变异点为 3，目前变异点的基因值为 7，随机生成的变异后的基因值为 1；
- ②寻找父代中与变异后基因值相同的基因位置，易知序号为 1 的基因值与变异基因值 1 相同，则序号为 1 的基因为与变异点的交换点；
- ③将交换点与变异点的值互换，可得突变之后的子代为 2 7 0 1 4 5 6 3。

3.3.5 进化过程的控制参数

需要确定合适的种群规模和终止代数，以及各种进化操作的概率来保证遗传算法能够实现较好的收敛性以及种群多样性，并能获得更加优良的种群。种群规模即大小和终止代数即进化代数的设置取决于装配体中零部件的数目。根据变异概率与进化代数和种群平均适应度值以及种群多样性的关系，基因交叉概率一般取值在 0.4-0.99 之间比较合适，基因突变概率设置在 0.05-0.3 之间较合适^[48]。本文通过设置有效性检验即装配序列是否满足几何约束、种群数目检验以及迭代上限的方式来控制整个遗传算法的进化过程。比如，整个种群的进化过程的终止条件是进化代数达到设置的值^[42]，每次的进化操作通过判断生成的可行子代数达到种群大小和达到循环上限相结合的控制方式等。

3.4 本章小结

传统的基于精确计算的装配序列规划方法生成的可行装配序列大多数质量比较低，且当零件较多时易出现组合爆炸的问题；而基于启发式算法尤其是遗传算法的装配序

列规划方法所得的装配序列质量高,容易获得全局最优解,但是会产生大量不可行解,降低算法的搜索效率。针对以上问题,本文将传统的基于精确计算的方法和基于启发式算法的方法结合起来,分别实现可行装配序列的生成和装配序列的优化功能,综合利用两者的优点,从而更加高效地获得更高质量的装配序列,同时避免组合爆炸问题的出现。本章对如何生成可行的装配序列问题展开了研究,提出了通过依次判定装配序列的几何可行性以及人为可行性的方法,以淘汰不可行装配序列保留可行装配序列的形式,实时生成可行装配序列的虚拟装配流程,并给出了根据全局装配干涉矩阵和约束邻接矩阵判定虚拟装配环境中形成的临时装配序列和给定装配序列可行性的具体算法。通过本章的可行装配序列生成方法,结合前一章节的装配信息以及装配信息模型,就可以获得一个或者多个可行的装配序列,用于装配体的装配活动,并为之后的装配序列评价与优化活动提供合理的研究对象^[49]。接着对装配序列评价方法进行了研究,并提出根据装配序列的可行性、装配过程稳定性以及装配方向的改变次数三个因素对装配序列进行评价^[49],并建立了合适的适应度函数,为之后的装配序列优化奠定基础,然后对遗传算法进行了介绍,并对其五个关键要素进行了详细介绍和讨论,给出了本文改进的遗传算法中所采用的具体方法,即选用易于理解的整数数组对染色体进行编码和存储;采用自动生成和人工输入相结合的方式生成初始种群;使用最优保存策略及“轮盘赌”方法选择父代,保证了遗传算法的收敛性,针对装配序列的特点设计了合适的交叉操作和突变操作,保证了遗传算法中种群的多样性,三种进化操作的设计即避免了算法的早熟,又确保了算法能收敛于全局最优解;最终按概率执行交叉和突变操作以及通过全局装配干涉矩阵和约束邻接矩阵来完成种群的进化操作,并通过设置合适的各类相关参数,来实现装配序列的优化。

第四章 三维虚拟仪器装配序列规划应用

4.1 三维虚拟实验室平台

本文提出的装配序列规划方法的最终仿真验证是在自主开发的三维虚拟物理实验室平台上开展的,该平台的主要逻辑框架是利用开源游戏引擎 Delta3D 搭建的,各种实验场景和虚拟装配中的模型是利用著名的三维建模软件 3DMax 建立并由 Delta3D 提供的 3D 地图编辑器 STAGE 导入到虚拟环境中的,虚拟装配过程中的场景和零部件的组织及管理是通过 Delta3D 封装的开源渲染引擎系统 Open Scene Graph(OSG)实现的,对虚拟装配过程以及实验现象的物理仿真和碰撞检测则通过其封装的开源物理仿真引擎 Open Dynamic Engine (ODE) 实现的,界面是用 Delta3D 集成的免费图形用户界面开发工具 Crazy Eddie's GUI (CEGUI) 开发的^{[3][4]}。自主开发的三维虚拟实验室搭建和虚拟装配系统设计方法见文献[3]介绍。

利用天平进行测量是物理及化学领域较常见的实验操作,所以本文将所提出的装配序列规划方法在如图 2-6 所示的三维虚拟仪器天平的虚拟装配过程中给予实现。通过三维虚拟物理实验室系统菜单^[1],还可进行装配过程动画的回放,自动装配以及重新装配等操作。对天平进行装配序列规划所需的装配信息模型即全局装配干涉矩阵、约束邻接矩阵、位置姿态矩阵、支撑矩阵和装配方向矩阵可见本文第二章。

如图 4-1 所示是虚拟装配的具体应用场景三维虚拟实验室的系统仿真操作流程。

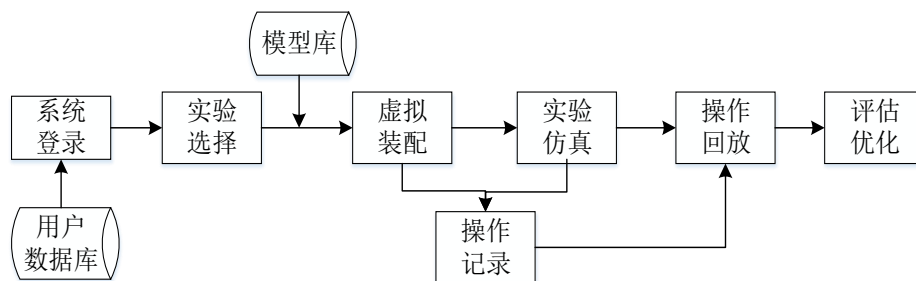


图 4-1 三维虚拟实验室系统仿真操作流程

Fig.4-1 Operating procedure of 3D virtual simulation laboratory system

系统登录首先需要用户在登录界面输入相关信息;然后系统会对用户输入的信息

在数据库进行检索，若有匹配用户信息则进入系统开始界面；在该界面可以实现对语言的选择以及对系统其他参数的设置，可以通过按钮进入实验室环境中，也可以返回上一个界面；用户成功登录之后选择进入实验室环境中，系统会检查该用户的历史信息，提示之前的历史学习记录，然后进入实验选择环节^[3]；在实验选择环节，选择要进行的实验，系统会将对应的实验场景以及零部件导入到虚拟实验环境中，进入到虚拟装配和实验仿真环节中；在虚拟装配和实验仿真环节，用户可以通过交互模块实现虚拟装配操作，也可以对实验进行仿真，在交互的过程中，系统记录操作相关数据信息，以供后续的操作回放和评估等使用^[3]。

本文以实验仪器天平为例进行了虚拟装配仿真。其对应的实验场景和零件模型导入到虚拟装配环境中的初始状态如图 4-2 所示。初始的实验场景是一个包含角色对象和场景地形的一个集合，由 Delta3D 提供的地图编辑器 STAGE 制作^[3]。

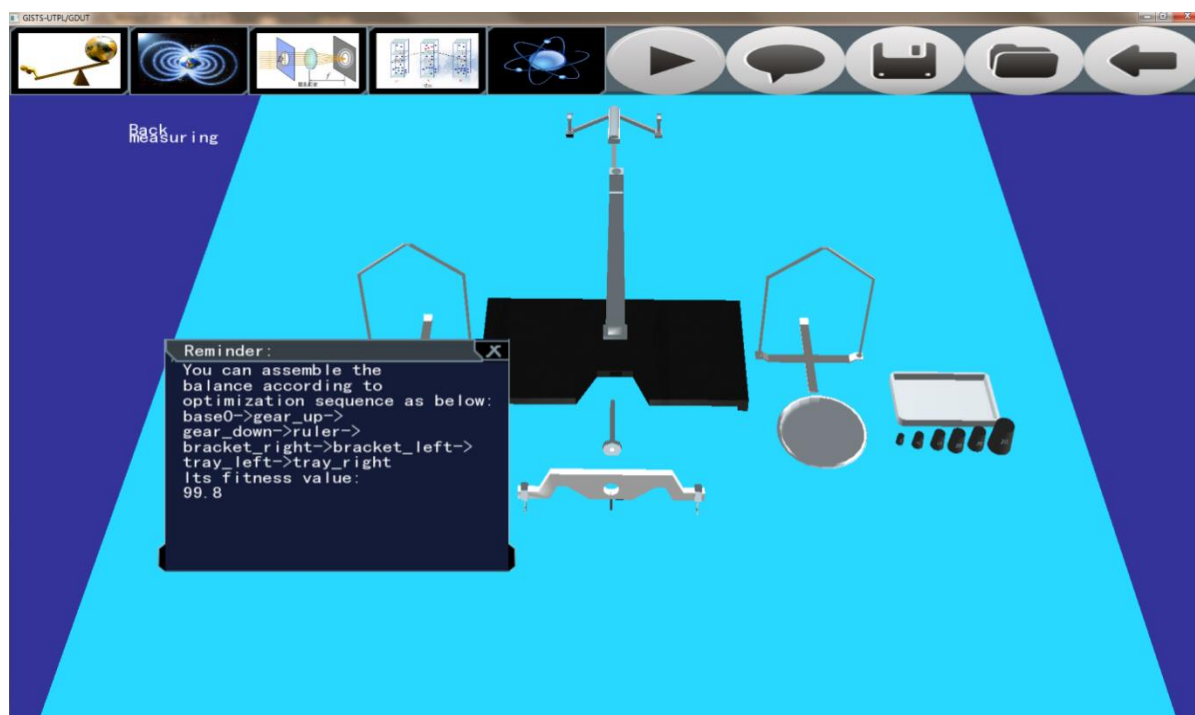


图 4-2 天平初始的实验场景

Fig. 4-2 Initial experiment scene of balance

本文在实现时主要定义了三个类即：VirtualLabComponent 类，AssemblyMotionModel 类和 InstrumentObject 类来实现虚拟装配系统中的实验仪器的组装与模拟。本文提出的装配序列规划算法主要在 InstrumentObject 类中实现，所以下面将详细介绍 InstrumentObject 类。其类定义如下：

```
class GA_EXPORTS InstrumentObject:public dtCore::Base
```

```

{
public:
    InstrumentObject(dtGame::GameManager*gm,const std::string& name=
    "InstrumentObject");
    static const std::string SPACE;
    class GA_EXPORTS Show_Command
    {
    public:
        std::string name;//节点名称
        osg::Matrix mat; //转移矩阵
        double      time;//发生的时间
    };
    class GA_EXPORTS NodeData
    {
    public:
        std::string NodeName;
        osg::Matrix mat;
    };
    typedef NodeData NodeDataType;
    typedef std::deque<NodeDataType> NodeDataList;
    typedef std::vector<Show_Command> Move_List;
    typedef std::map<std::string,size_t> PartMap;
    typedef std::vector<std::vector<int> > AssemblyMatrix;
    typedef std::vector<int> AssemblyList;
    virtual void OnMessage(MessageData*);
    virtual void onTick(PartModelActor* selected);//装配序列规划算法在其中实现
    virtual void initPartModelActor();
    virtual void assemblyCBFunc(const dtCore::ODESpaceWrap::CollisionData& data);
    virtual void createJoints()=0;
    float compute_benefit( );//计算临时装配序列的适应度值

```



```

virtual void enableDynamics()=0;

void getAssemblySequenceIntoFile(std::vector<int> aslist,std::vector<int> adlist,
const std::string& outfilename);//输出装配序列信息到文件

void outsequence(AssemblyList aslist,AssemblyList adlist);//输出装配序列信息

int isProbable(size_t index);//返回临时装配序列的可行性

void readAssemblyProbableMatrix(const std::string & filename,const std::string&
outfilename);//读取全局装配干涉矩阵

void readConstraintAdjacencyMatrix(const std::string& fileName,const std::string&
outfilename);//读取约束邻接矩阵

void readAssembledModel(const std::string& path); //从文件中读取模型

void autoAssembly(); //自动装配

void restore(PartModelActor* pma); //恢复未装配时各个零件的初始位置

void restorePosForPlayRecord();

void restorePositionForAutoAssembly();

void readAssemblyPAM(); //开发时读取装配好的各个零件的位置姿态矩阵

void savePAM(const std::string& file); //保存装配好各个零件的位置姿态矩阵

void readPAM(const std::string& file); //仿真时读取装配好的零件位置姿态矩阵

PartMap getPartMap(){return _partMap;}//获得待装配仪器对应的场景地图

std::string getCollisionInf(){return _collisionInf;} //获得实时碰撞信息

protected:

virtual ~InstrumentObject();

void _init();

void _updatePAMList(const osg::Vec3& basePos);

void _updatePAMList(const osg::Matrix& baseMat);

void _updatePAMList();

dWorldID _getWorldID(){return _scene->GetWorldID();}

dJointGroupID _getJointGroupID(){return _scene->GetContactJointGroupID();}

int _findIndex(const std::string& name); //通过零件的名称找到序号

std::string _findName(size_t index); //通过零件的序号找到名称

ga::PartModelActor* _findPartActor(size_t index);

```

```

void _freshCollision(PartModelActor* selectedPartActor);

std::string _findUnfinishedAdjacentPart(size_t index); //返回应先装配的零件名称
protected:

dtGame::GameManager* _gm; //游戏管理器
dtCore::Scene* _scene; //场景
PartMap _partMap; //场景地图
AssemblyList _aslist; //实时记录装配序列
AssemblyList _adlist; //实时记录装配序列对应的可行装配方向集
AssemblyMatrix _apmatrix; //全局装配干涉矩阵
AssemblyMatrix _camatrix; //约束邻接矩阵
Move_List* _moveList; //记录手工移动物体过程
Move_List* _autoplayList; //记录自动演示数据
NodeDataList* _restoreList; //记录还原位置数据
NodeDataList* _pamList; //记录装配好的各个部件的位置姿态矩阵
osg::ref_ptr<osg::Group> _playRecordGroup; //动画记录的组节点
osg::ref_ptr<osg::Group> _autoPlayGroup; //动画记录的组节点
osg::ref_ptr<osg::Group> _moveGroup; //手工移动零件模型的组节点
osg::ref_ptr<osg::Switch> _odeModelRoot; //模型根节点
PartModelActor* _selectedPma; //当前选中零件对应角色
PartModelActor* _pre_pma; //前一个选中零件对应角色
std::string _collisionInf; //碰撞信息
dJointGroupID _jointGroup;
dtCore::ODESpaceWrap::CollisionCBFunc _collisionCBFunc;

};

```

InstrumentObject 类主要用于派生各种具体的对象描述各种仪表仪器，其中本文所用的示例实验仪器天平即是一种仪表仪器，它就是通过 InstrumentObject 类派生的 Scale 类的 _scaleInstrument 对象来描述的。每个 InstrumentObject 对象都拥有一个用于管理系统中各个零件角色的游戏管理器，实际使用时，要先在 VirtualLabComponent 类中初始化一个 InstrumentObject 对象，并在 InputComponent 类中导入对应的场景地图即 map。然后就可以通过 InstrumentObject 对象完成如下工作：

(1) 实验系统中各个仪器零部件之间的碰撞之后的反应。每当系统中的零部件被选中，由 `AssemblyMotionModel` 类执行碰撞检测算法，然后发送 `collision` 消息。`InstrumentObject` 对象通过读取该消息，来执行相应的动作；

(2) 初始化各种数据结构如用于记录初始零件位置的 `_restoreList`，用于动画演示手工移动物体过程的 `_moveList`，主要在 `_init()` 和 `initPartModelActor()` 函数中完成；

(3) 初始化零件间的约束关系，该约束关系决定了哪些零件可以组装在仪器，和零件的部分装配顺序，该工作在 `void readAssemblyProbableMatrix()` 和 `readConstraintAdjacencyMatrix()` 函数中完成，在该类的 `OnMessage()` 函数中会对零件之间一旦发生碰撞，进行判断，如果顺序不对或组装错误，则弹出提示信息。

(4) 读取零件的 PAM 矩阵并记录到文件中作为基础数据用于组装过程的近距离自动装配。系统开发时该 PAM 矩阵的读取和保存在 `readAssemblyPAM()` 和 `savePAM()` 函数中完成。在实际应用时在 `initPartModelActor()` 函数中导入开发时保存的各个零件的 PAM 矩阵，具体实现在 `InstrumentObject` 派生的对象中的 `initPartModelActor()` 函数中调用 `readPAM("data/pam/mechanism/scale_pam.txt")`，以读取 PAM 值。

(5) 组装训练时当零件间处于一个临界值时的自动装配，在 `onTick()` 函数中完成。如图 4-3 所示是正在对天平的一个零件进行移动来实现装配的仿真效果。

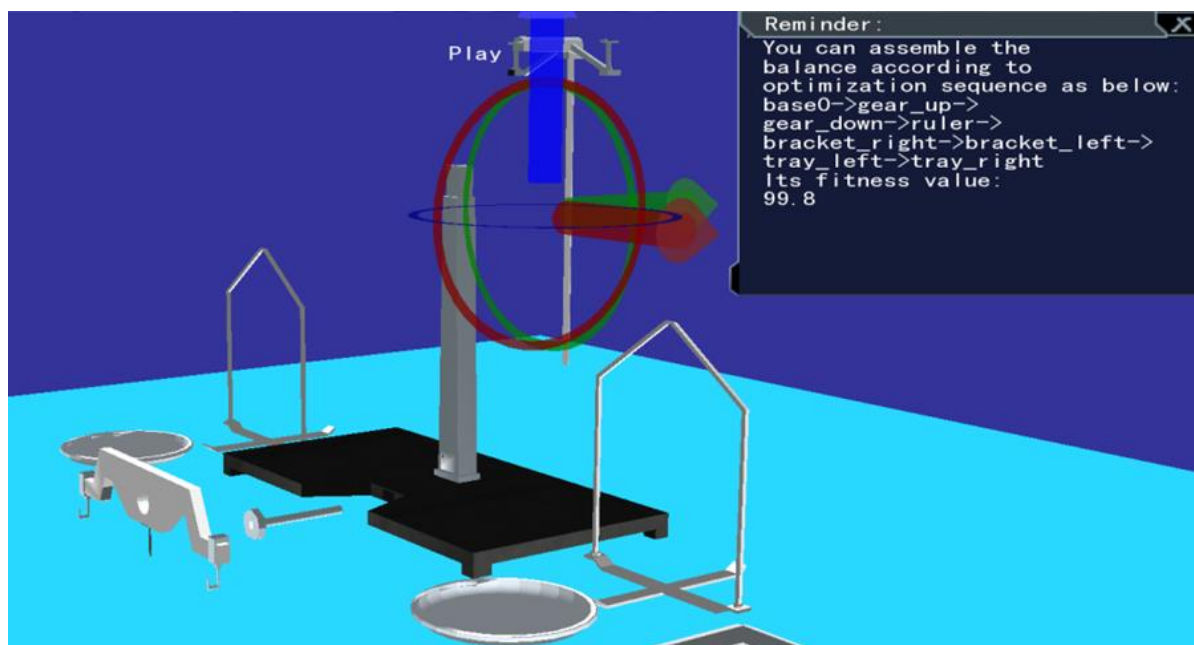


图 4-3 移动零件进行装配

Fig.4-3 Moving parts for assembly

装配操作过程数据记录在 `_moveList` 中，可以用于装配过程的评估和回放。通过如图 4-4 所示的菜单栏中四个选项可以依次实现装配过程动画播放功能、自动装配、重新装配的场景恢复和普通装配功能。自动装配功能可以跳过手动的装配过程，通过开发时读取保存的装配好的各个零件位置姿态矩阵来调整它们的位置和姿态，最终实现将所有零件一步装配到位，获得一个装配好的实验仪器天平，供实验仿真实验。重新装配的场景恢复功能可以根据保存的零件初始状态文件 `_restoreList` 来调整零件位置姿态到最初状态。

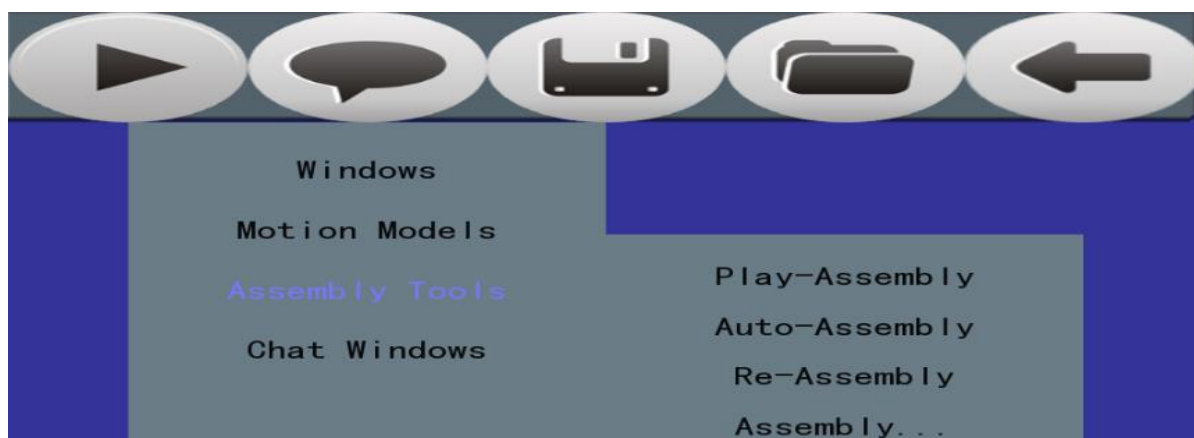


图 4-4 菜单栏的装配工具选项

Fig.4-4 Assembly tool options of menu bar

4.2 仪器装配序列规划

将本文第三章中所述装配序列生成以及装配序列评价和优化方法进行了编程实现，以图 2-4 所示的三维虚拟仪器天平为例，下面将对具体仿真结果进行详细说明。

4.2.1 仪器装配序列生成

当装配进行到某一步时，选中一个新的未装配的零件，系统将根据本文的装配序列生成算法来判断选中零件与之前已装配零件组成的装配序列是否是可行的，若可行，则将弹出提示信息，提示当前序列可行，并给出选中零件的可装配方向和当前序列的适应度值，如图 4-5 所示为安装完 `gear_up` 之后，点击选中 `gear_down` 提示当前序列是可行的，并给出了可行装配方向和序列适应度值的提示信息；若不可行，则将提示当前序列不可行，且给出当前序列违背了哪些约束，若不满足人为约束，还将给出前一

个应该先于选择零件装配好的零件名称，如图 4-6 所示为装配好 gear_up 之后，点击选中 ruler 提示当前序列不可行，不满足人为约束，gear_down 应该先装配。

装配序列生成算法是在 void InstrumentObject::onTick (PartModelActor* selected) 函数中实现的，信息的提示则通过在 void VirtualLabComponent::_freshTick(float dt) 函数中调用 std::string InstrumentObject::getcollisionInf() 实现。

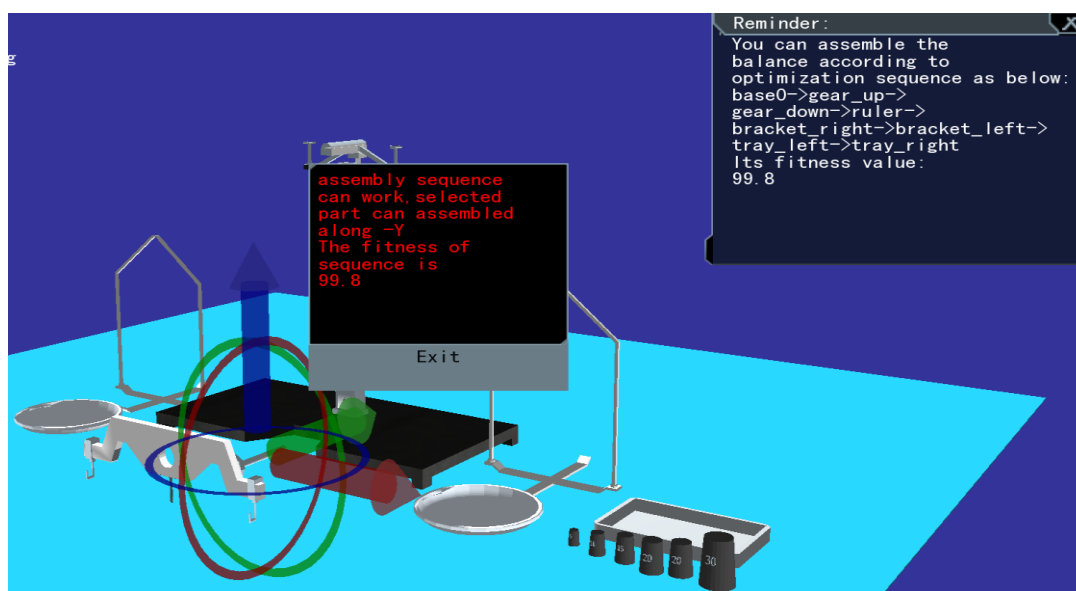


图 4-5 当前装配序列可行

Fig.4-5 Current feasible assembly sequence

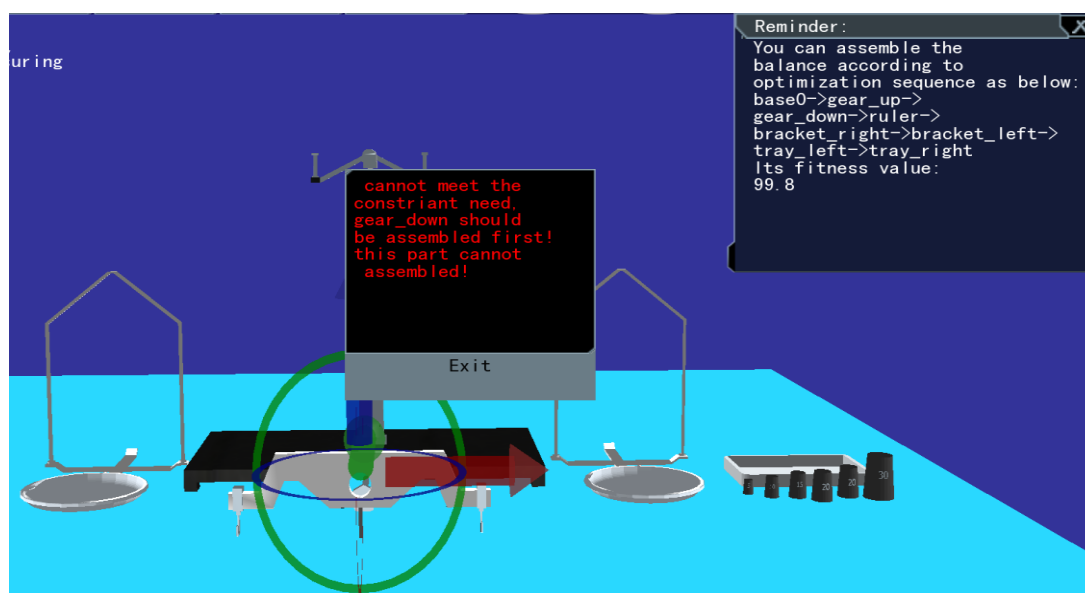


图 4-6 当前序列不满足人为约束

Fig.4-6 Current sequence against the artificial constraint

在一个虚拟装配系统中，人没有了真实装配过程才有的各种感官作用等的帮助，

对空间运动的控制比较难,且不精确,需要借助辅助设备才可以提供较好的装配效果^[50],但是现在大多数虚拟装配系统由于受到硬件技术和成本等的制约,很难实现比较精确的装配定位效果^[50]。为了让操作者可以自如、精确地对装配零件进行定位和装配,本文通过提供一些智能提示信息,以及实现近距离的自动装配功能,来实现对装配动作的引导^[3],最终实现装配零部件的精确定位和快速装配^[50]。当选中零件与其最终装配位置的距离小于设定的阈值,则突出显示选中零件并根据零件的位置姿态矩阵自动调整其位置和姿态从而完成该零件的自动装配^[50],如图 4-7 所示,并更新和保存已装配零件序列,继续等待下一次的装配操作。具体实现代码为:

```
if((pos-translation).length()<2.5f)//设定近距离自动装配的阈值
{
    _selectedPma->RenderCollisionGeometry(true);//突出显示
    dtCore::Transform xform;
    xform.Set (mat);
    _selectedPma->SetTransform(xform);//自动装配到最终位置
    _selectedPma->setFinished(true);//设置选中零件为已装配
}
```

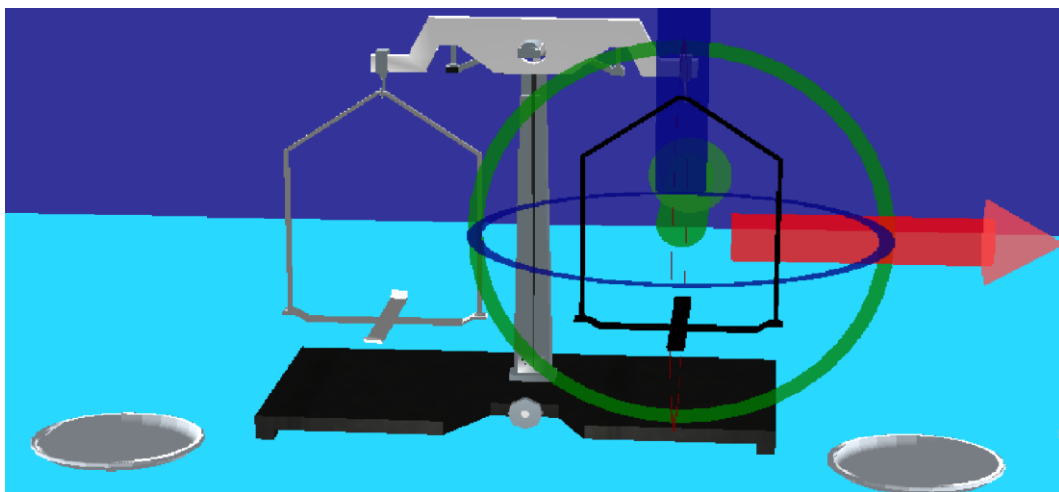


图 4-7 近距离的自动装配和突出显示

Fig.4-7 Automatic assembly and highlighting in close

对在虚拟装配环境中实际操作获得的符合各种约束的可行的尤其是好的装配序列进行实时记录和保存以供装配序列分析和评估是非常重要的,本文实现的装配序列规划方法中有对可行装配序列的实时记录和保存输出功能,还记录了时间、装配原始数据和可行装配方向等重要数据,为后续的相关应用提供了较为全面的信息。获得的部

分输出保存的仿真数据如图 4-8 所示。其中装配仿真过程中对实际装配序列的实时记录和输出是由函数 `void InstrumentObject:: getAssemblySequenceIntoFile(std::vector<int> aslist, std::vector<int> adlist,const std::string& outfilename)` 实现的。

```

本次记录时间: 2016/01/22 21:49:22 Friday 本年第022天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 1 {+X-X+Y-Y+Z-Z} —— 6 {-Z}
本次记录时间: 2016/01/23 17:14:30 Saturday 本年第023天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 3 {-Y} —— 5 {-Y}
本次记录时间: 2016/03/02 19:41:54 Wednesday 本年第062天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 1 {+X-X+Y-Y+Z-Z}
本次记录时间: 2016/03/22 22:49:43 Tuesday 本年第082天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 1 {+X-X+Y-Y+Z-Z}
本次记录时间: 2016/03/22 22:58:29 Tuesday 本年第082天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 5 {-Y} —— 3 {-Y} —— 1 {-Z} —— 6 {-Z}
本次记录时间: 2016/03/31 21:23:28 Thursday 本年第091天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 2 {+X-X+Y-Y+Z-Z}
本次记录时间: 2016/04/01 20:04:02 Friday 本年第092天 中国标准时间
实际装配序列及各零件的装配方向为:
0 {+X-X+Y-Y+Z-Z} —— 4 {+Y-Z} —— 3 {-Y} —— 5 {-Y} —— 2 {-Z} —— 1 {-Z} —— 6 {-Z} —— 7 {-Z}

```

图 4-8 部分仿真数据

Fig.4-8 Part of simulation data

根据装配过程中系统记录的若干关于时间以及装配零件的位置姿态的信息，可以调用函数 `void PlayRecordCallback::operator()(osg::Node* node,osg::NodeVisitor* nv)` 来遍历这些数据并对零件的位置姿态恢复，从而实现装配过程动画回放功能，也为后续装配过程评价提供了依据，还可以通过提供好的装配动画对操作者进行教学和培训。

4.2.2 仪器装配序列评价与优化

为了验证本文第三章中提出的装配序列评价和优化算法的有效性^[45]，对改进的遗传算法进行了仿真实验。经过多次反复试验，发现将种群规模设置为 7，交叉概率和突变概率分别设置为 0.7 和 0.1 时算法能够获得较快的收敛速度和较高适应度值的装配序列，每次获得的最优装配序列和收敛代数也比较稳定。下面以进化 150 代时的一次仿真数据为例来对装配序列优化算法的有效性进行说明。最终的仿真效果如表 4-1 和图 4-9 所示。可以看出整个进化过程中种群适应度的平均值是不断波动的，但整体也随着适应度最大值呈增长趋势。种群适应度最大值则随着进化过程呈现单调递增的稳定增长趋势，在进化到 95 代左右时取得最大适应度值 99.8 的个体为 0 4 3 5 2 7 1 6，对应零件装配序列为 `base0->gear_up->gear_down->ruler->bracket_right->bracket_left->tray_left->tray_right`。适应度平均值的波动说明了种群具备良好的多样性，保证了全局搜索的能

力。适应度最大值的单调递增说明了种群一直在向着最优解收敛，和种群的多样性一起保证了整个装配序列优化算法能够搜索到全局最优解的能力。其中种群的多样性是由交叉和突变操作实现的，种群的收敛性是由选择操作和可行性检验操作实现的。每一代个体对应的装配序列经过判定也都是几何可行的，且最后一代种群的最优个体也是满足人为约束且具备最高的适应度值，说明通过本文算法获得了良好的优化效果。

表 4-1 首代和末代种群（进化 150 代，耗时 0.67s）

Table4-2 First and last generation of the population (evolution 150 generations, cost 0.67s)

个体序号	首代个体	首代适应度值	末代个体	末代适应度值
1	2 5 1 7 0 3 4 6	87.2	0 4 3 5 2 7 1 6	99.8
2	7 2 1 6 3 5 0 4	78.3	2 3 5 0 7 1 6 4	89
3	6 2 1 0 5 3 4 7	84.3	0 4 3 5 2 7 1 6	99.8
4	4 5 2 1 0 6 7 3	86	2 3 5 0 7 1 6 4	89
5	1 2 7 3 0 5 6 4	81.1	0 4 3 5 2 7 1 6	99.8
6	6 3 2 1 5 0 7 4	82.4	0 4 3 5 2 7 1 6	99.8
7	1 6 2 0 3 5 7 4	82.7	0 4 3 5 2 7 1 6	99.8

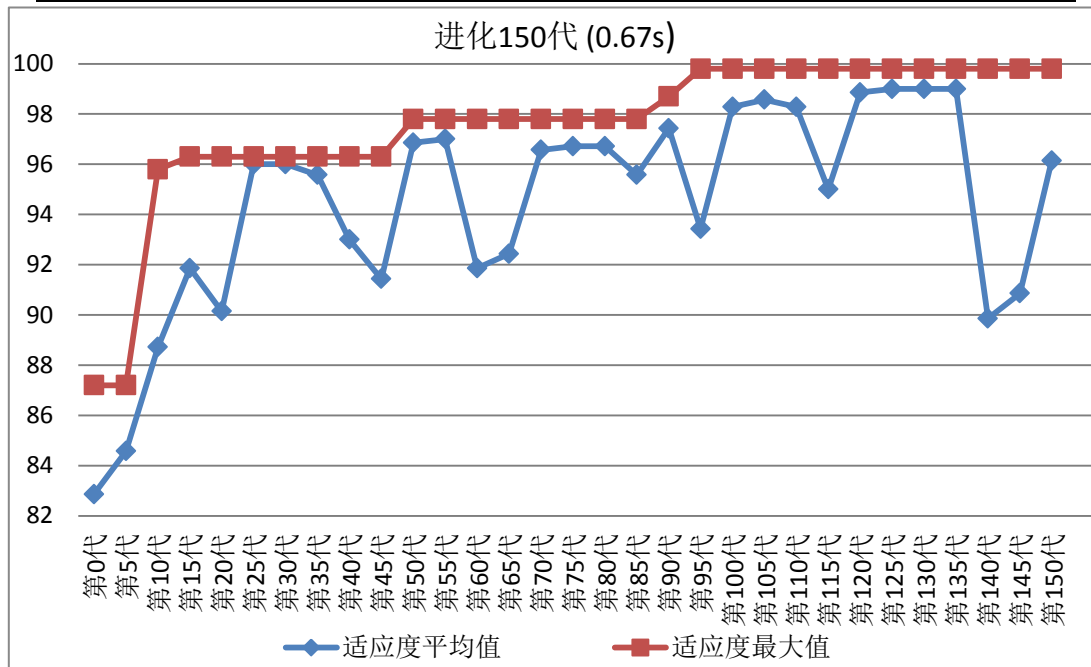


图 4-9 进化 150 代的适应度值变化趋势

Fig.4-9 Trend of fitness value during the process of evolution 150 generations

如图 4-5 和图 4-6 所示，在虚拟装配环境中，当点击选中一个零件准备装配时，系统不仅判断临时序列的可行性并给予提示，也会计算和提示临时装配序列的适应度值，

给操作者参考，引导操作者对装配序列的进行调整，从而达到优化的目的。

本文对通过装配序列优化算法获得的装配序列、时间以及一些初始数据进行了输出和保存到文本文件中，并将其读入到虚拟装配环境中作为提示信息进行显示，从而和其他智能提示信息一起引导装配操作的进行。当选择实验进入到对应实验的仪器虚拟装配初始界面之后，将弹出优化的装配序列相关的提示信息，供实验操作者参考，仿真效果见图 4-2。当操作者按照图 4-2 中提示的优化的装配序列进行天平的装配操作时，当装配到最后一个零件 tray_right 时，即点击选中该零件时，也会弹出实时的提示信息，提示当前序列可行，给出可装配方向和当前序列的适应度值，如下图所示。

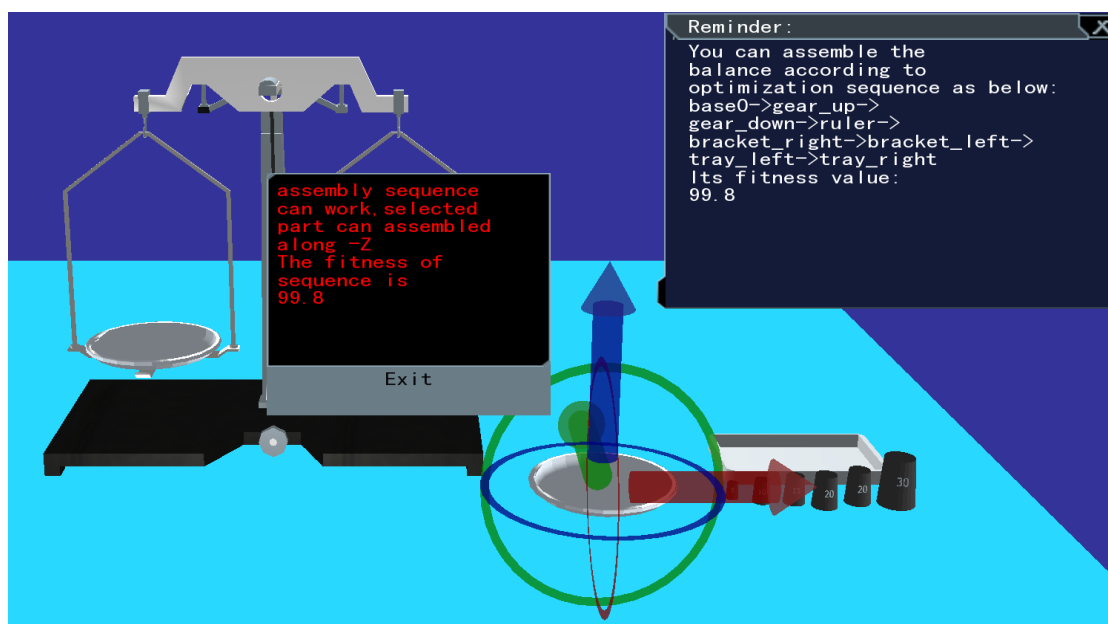


图 4-10 根据提示的优化序列进行装配

Fig.4-10 Assembled according to the optimization sequence in initial tips

可以看出实时弹出的提示信息中当前序列的适应度值和进入天平装配环境的初始界面时弹出的提示信息中给出的最优装配序列的适应度值是相等的，说明装配序列优化算法嵌入到虚拟装配环境中获得的仿真结果是正确的。

4.3 本章小结

本章主要对装配序列规划应用相关的仿真结果进行了分析，以三维实验仪器天平的虚拟装配过程为例进行了仿真验证，得到了满意的仿真效果，并给出了具体的仿真效果图，证明了本文所提出的装配序列规划方法的有效性及可靠性，同时通过对装配序列规划应用的仿真证明了本文提出的方法对装配过程具有良好的实际指导意义^[45]。

总结与展望

虚拟现实技术的应用越来越广泛，也受到了各国政府以及研究结构和个人的重视，由于传统实验教学的限制以及各种信息技术的发展，应运而生的虚拟实验室可以辅助甚至有逐渐代替传统的实验教学活动的趋势，研究虚拟实验室中的相关技术和内容变得很有意义，本文即是针对大多数虚拟实验室存在的几个主要问题进行了研究，为了实现更好的实验教学效果，提出了将工业领域中的装配序列规划方法与虚拟实验室中的实验仪器的虚拟装配过程结合起来的解决方法，从而可以通过虚拟装配操作加强对实验原理以及仪器的使用和组成结构等的学习和掌握，而不再只是简单的展示实验，以便更好地达到实验教学的目的。

本文介绍了虚拟实验室、虚拟装配以及装配序列规划技术的现状，并提出了自己的装配序列规划方法，进行了详细的阐述和仿真，实现了更好的虚拟装配效果。具体包括用矩阵表示约束从而对装配信息建模，并通过这些约束实现了装配零件的位置和姿态的调整，装配序列可行性的实时判定与装配动作的实时引导等，最终应用在了零件的自动装配、装配过程的动画回放、装配序列的优化以及输出和保存等方面。本文所介绍的三维虚拟实验室的虚拟装配环境克服了工业领域虚拟装配序列规划对零件模型过于依赖、模型的重用性低以及模型信息量较大的缺点，从而具备了更好的可扩展性；综合利用位置姿态矩阵以及 ODE 的碰撞检测功能，能够实现快速的碰撞检测和零件定位，甚至在一定条件下能够实现零件的自动装配，最终进行快速的装配；通过实时保存各个装配零件的位置和姿态信息，最终通过读取这些信息对装配过程实现了动画回放，有利于对虚拟装配过程的评价及对操作者的培训。以常用的实验仪器天平为例，进行了虚拟装配过程的仿真，实现了虚拟装配序列的生成、优化。验证了本文方法的有效性，很好地满足了实验教学的需要。本文实现的仿真效果，基本达到了预定的研究目标，但是还有很多可以改善及补充的地方，这些也是未来研究的方向。

- (1) 物理仿真的对象以及效果都可以做进一步的完善；
- (2) 可以增加零件装配路径的自动规划功能；
- (3) 可以开发三维虚拟实验室的网络功能，从而实现用户的远程操作、多用户之间的实时交互以及多人协同装配。

参考文献

- [1] 陈学军,徐明铭. 基于虚拟仪器的综合实验室平台探索[J]. 实验室研究与探索. 2014(10):250-253.
- [2] 曹颖. 虚拟实验室在仪器分析教学中的应用探讨[J]. 实验科学与技术. 2015(02):36-37.
- [3] 孙艺萌. 三维虚拟实验仪器装配过程中关键技术的研究[D]. 广东工业大学,2015.
- [4] 王良,刘东峰. 三维虚拟物理实验仿真平台的界面设计与实现[J]. 计算机技术与发展. 2014(09):238-240.
- [5] 张晓玲. 基于 Falsh 技术的大学物理仿真实验的研究[D]. 华中师范大学,2007.
- [6] 余春华. 基于认知灵活性理论的高中物理虚拟实验教学平台的设计与开发[D]. 广西师范学院,2011.
- [7] Dumitrescu C, Olteanu R L, Gorghiu L M, et al. Using virtual experiments in the teaching process[J]. Procedia - Social and Behavioral Sciences, 2009, 1(1):776-779.
- [8] Newton 虚拟教学软件. <http://www.newtonlab.com/English/newton/>
- [9] IrYdium 虚拟化学实验室. <http://ir.chem.cmu.edu/>
- [10] BioimeraCtive 系统. <http://www.hhmi.org/biointeractive/>
- [11] Ramat E, Preux P. “Virtual laboratory environment” (VLE): a software environment oriented agent and object for modeling and simulation of complex systems[J]. Simulation Modelling Practice & Theory, 2003, 11(1):45-55.
- [12] 余鹏程,李晶. 基于网络的虚拟实验室的研究[J]. 数字技术与应用. 2011(12): 217-218.
- [13] 裘伟廷. 基于 WEB 的虚拟实验室构建[J]. 远程教育杂志,2002(06):39-42.
- [14] 瞿伟娜. 基于虚拟现实技术的虚拟实验室的研究与实现[D]. 大庆石油学院,2003.
- [15] Gabrielli S, Hodapp M, Ranon R. Designing a Multipurpose Virtual Laboratory to Support Communities of Practice in Physics[C] IEEE International Conference on E-Science & Grid Computing. IEEE Computer Society, 2006:139-139.
- [16] 李东旭,扶晓,刘劲. 虚拟装配的几种中心技术研究[J]. 电脑知识与技

- 术,2010(10):2506-2507.
- [17] 吕凤县,张桂香,张庆洪. 基于拆卸和拆卸约束矩阵的装配序列规划[J]. 现代制造工程. 2014(01):21-24+63.
- [18] 李建广,夏平均. 虚拟装配技术研究现状及其发展[J]. 航空制造技术,2010(03):34-38.
- [19] 王先元,何汉武,吴悦明,王勇,杨英. 港口起重机虚拟装配仿真与评估系统[J]. 装备制造技术,2015(04):67-70.
- [20] 刘子建,王平,艾彦迪. 面向过程的产品信息虚拟装配建模技术研究[J]. 中国机械工程. 2011(01):60-64.
- [21] Jayaram S, Jayaram U, Wang Y, et al. VADE: a Virtual Assembly Design Environment [J]. Computer Graphics & Applications IEEE, 1999, 19(6):172-179.
- [22] Wan H, Gao S, Peng Q, et al. MIVAS: A Multi-Modal Immersive Virtual Assembly System[C] ASME 2004 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. American Society of Mechanical Engineers, 2004:113-122.
- [23] 刘振宇,谭建荣,张树有. 面向虚拟装配的产品层次信息表达研究[J]. 计算机辅助设计与图形学学报. 2001(03):223-228.
- [24] 王丰产,孙有朝,李娜. 多工位装配序列粒子群优化算法[J]. 机械工程学报. 2012(09):155-162.
- [25] 李海军,姜捷,王云飞. 基于扩展干涉矩阵和遗传算法的拆卸序列规划[J]. 计算机工程与设计. 2013(03):1064-1068.
- [26] Chen Z Z, Wang L, Wang Z. A hybrid honey-bees mating optimization algorithm for assembly sequence planning problem[C] Natural Computation (ICNC), 2012 Eighth International Conference on. IEEE, 2012:1135-1140.
- [27] Tseng Y J, Chen J Y, Huang F Y. A particle swarm optimisation algorithm for multi-plant assembly sequence planning with integrated assembly sequence planning and plant assignment[J]. International Journal of Production Research, 2010, 48(48):333-345.
- [28] Hui C, Yuan L, Zhang K F. Efficient method of assembly sequence planning based on GAAA and optimizing by assembly path feedback for complex product[J]. International

- Journal of Advanced Manufacturing Technology, 2009, 42(11):1187-1204.
- [29] Chang C C, Tseng H E, Meng L P. Artificial immune systems for assembly sequence planning exploration[J]. Engineering Applications of Artificial Intelligence, 2009, 22(8):1218-1232.
- [30] 张鹤娜. 基于子装配体识别和遗传算法的装配顺序规划[D]. 哈尔滨工程大学, 2011.
- [31] 马小凉. 基于虚拟环境的典型机械产品装配序列优化方法研究[D]. 长春理工大学, 2009.
- [32] 夏平均, 陈鹏, 姚英学, 唐文彦. 层次约束结构的虚拟装配建模技术[J]. 哈尔滨工业大学学报. 2009(07):40-45.
- [33] 韩晓东, 蔡勇, 蒋刚. 基于改进的遗传算法的装配序列规划[J]. 机械设计与制造. 2009(03):212-214.
- [34] Chrysosolouris G, Mavrikios D, Fragos D, et al. A virtual reality-based experimentation environment for the verification of human-related factors in assembly processes[J]. Robotics and Computer-Integrated Manufacturing, 2000, 16(4):267-276.
- [35] 张兴华. 基于 CATIA 的数字化装配信息建模与序列规划研究[D]. 武汉理工大学, 2013.
- [36] 张闻雷, 曲蓉霞, 许美蓉, 罗小川. 复杂产品装配干涉矩阵自动生成方法[J]. 机械工程学报. 2016(01):139-148.
- [37] 龚德刚. 面向装配序列规划的遗传模拟退火算法研究[D]. 华中科技大学, 2003.
- [38] 王松, 孙振忠, 郭建文, 张智聪. 基于混合蛙跳算法的复杂产品装配序列规划[J]. 计算机集成制造系统. 2014(12):2991-2999.
- [39] 张志贤, 刘检华, 宁汝新. 虚拟装配中的装配约束实现技术研究[J]. 系统仿真学报. 2011(09):1878-1883+1888.
- [40] Zhang J, Sun J, He Q. An Approach To Assembly Sequence Planning Using Ant Colony Optimization[C] International Conference on Intelligent Control and Information Processing. 2010:230-233.
- [41] 邢宇飞, 王成恩, 于嘉鹏. 基于蚁群算法的爆炸图自动生成方法[J]. 东北大学学报(自然科学版). 2012(06):883-886.
- [42] 陈天凡, 汤文成. 基于干涉矩阵的装配序列规划[J]. 机械制造与自动化.

2007(01):36-38+41.

- [43] 周开俊,李东波,黄希. 基于遗传算法的装配序列规划研究[J]. 机械设计. 2006(02):30-32.
- [44] 刘亚杰,古天龙,徐周波. 基于加权启发式遗传算法的装配序列规划[J]. 桂林电子科技大学学报. 2012(02):129-133.
- [45] 蒋超,吴波,李明宇,赵勇力,王小成. 基于遗传算法的产品装配序列规划研究[J]. 机械与电子. 2012(04):7-11.
- [46] 张超群,郑建国,钱洁. 遗传算法编码方案比较[J]. 计算机应用研究. 2011(03):819-822.
- [47] 王敬品,胡志勇,蔺吉顺,温玉春. 基于遗传算法的装配序列规划[J]. 机械工程与自动化. 2008(02):9-11.
- [48] Yu Guo, Xibo Lin, Weili Hu. Study on Premature Convergence in Assembly Sequence Genetic Planning [C] Proceedings of the 6th World Congress on Intelligent Control and Automation. 2006:6470-6474.
- [49] 张耀. 三维环境下装配序列的生成、评价与仿真[D]. 华中科技大学,2006.
- [50] 朱理劲,刘林. 机械产品虚拟装配智能导航技术的研究[J]. 工程图学学报. 2009(05):183-187.

攻读硕士学位期间发表的论文

- 1、张秋云，刘东峰. 三维虚拟实验室中装配序列规划问题研究[J]. 信息通信，已录用待发表

学位论文独创性声明

本人郑重声明：所呈交的学位论文是我个人在导师的指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明，并表示了谢意。本人依法享有和承担由此论文所产生的权利和责任。

论文作者签名：

日期：

学位论文版权使用授权声明

本学位论文作者完全了解学校有关保存、使用学位论文的规定，同意授权广东工业大学保留并向国家有关部门或机构送交该论文的印刷本和电子版本，允许该论文被查阅和借阅。同意授权广东工业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，可以采用影印、缩印、扫描或数字化等其他复制手段保存和汇编本学位论文。保密论文在解密后遵守此规定。

论文作者签名：

日期：

指导教师签名：

日期：

致 谢

转眼之间，三年的研究生生活就要结束，在这三年的时间里，无论是学习科研方面，还是工作生活方面，我都收获了很多，包括实践能力、思维方式、科研写作以及为人处世能力等，也遇到了许多的良师益友，受到了他们的关心与帮助，故想借论文完稿之际，对曾给予过我关心和帮助的人们表示最真诚的感谢。

首先我要感谢我敬爱的导师刘东峰教授，是他在我课题的选题、研究以及撰写过程给予了最全面、最用心的指导，也是他在我的为人处世和工作求职方面提供了最有益的建议和引导，对我的工作生活以及成长给予了帮助。在此我对刘老师送上最诚挚的谢意。

感谢各个实验室的同学们以及我的师兄师姐，还有实验室的何家峰老师和程昱老师，无私地分享自己的学识和经验，为我解惑答疑，给了我很多学习科研以及工作方面的很好的指导和建议，让我时刻都感到身处在一个温馨快乐的大家庭里。感谢我宿舍的三个姐妹们，感谢她们在这三年研究生生活的陪伴和帮助，感谢你们，使我的研究生生活如此精彩！感谢我亲爱的爸爸妈妈，感谢他们对我多年的养育和支持。感谢三年中一直陪伴我帮助我的一群可爱的河南老乡们，似兄长似姐妹的关怀，使我充实快乐，从而让我顺利走出了过于恋家带来的困扰。感谢我的男朋友，谢谢你默默地陪伴。尤其感谢我的外公外婆多年的养育之恩，无法回报，但你们永远在我心中！感谢所有一直陪伴在我的身边，不断关心我和鼓励我的亲友们，是他们给了我不断进步的动力和挑战困难的勇气。

最后感谢在百忙之中能够抽出时间的评审老师们。谢谢你们！

张秋云

2016年4月于广州