

基于多约束信息融合的特定网络检测方法设计

覃俊

(南京理工大学, 南京 210094)

摘要: 针对当前特定网络检测方法中没有对数据粒度进行过滤, 数据粒度过于粗糙, 检测过程的单一, 致使检测效率低、检测正确性偏差等问题; 提出一种基于多约束信息融合的特定网络检测方法, 利用 Windows 中的 Wpcap.dll 获取特定网络中 NIC 相关信息, 构建特定网络侦听, 制定过滤条件实现特定网络数据的获取; 根据 Rough 集理论对特定网络数据粒度进行过滤, 减小数据粒度的粗糙程度; 构建特定网络检测模型, 结合 D-S 证据理论得到基本置信函数值并确定值的权重, 代入 D-S 合成公式获得检测结果, 引入群体信任法对检测结果再次过滤, 实现网络异常数据的彻底检测和清除, 解决检测方法的单一性; 实验表明, 该方法提高了网络检测的效率和正确性, 有效解决了当前网络检测方法中存在的问题。

关键词: 多约束; 信息融合; 特定网络; 检测;

Design of Specific Network Detection Method Based on Multi-Constrained Information Fusion

Qin Jun

(Nanjing University of Science & Technology, Nanjing 210094, China)

Abstract: In view of the current specific network detection method, there is no data granularity filtering, the data granularity is too rough, the detection process is single, resulting in low detection efficiency, detection correctness deviation and other issues. Put forward a specific network detection method for multi constraints based on information fusion, to obtain the specific network NIC information using Windows Wpcap.dll in the construction of a specific network interception, making filtering conditions to achieve access to specific network data; according to the theory of Rough sets of specific network data filtering granularity, reduce the roughness of the construction of a specific data granularity; the network detection model, combined with D-S evidence theory are basic belief function value and determine the value of weight by D-S synthesis formula get the test results, the introduction of group trust method filtered again on the test results, the complete network anomaly detection data and clear, single solution detection method. The experiment shows that this method improves the efficiency and correctness of network detection, and effectively solves the problems existing in the current network detection methods.

Keywords: multi constraint; information fusion; specific network; detection

0 引言

特定网络指的是专门针对一个应用或者一个区域的流量。特定网络异常检测方法通常能够分为两类: 滥用检测与异常检测^[1]。其中的滥用检测一般是依据专家的所学知识和经验来判别异常的产生, 这导致了该方法对新入侵模式以及伪装之后入侵模式的检测率低。由此, 研究人员试着通过异常检测解决该问题^[2]。在进行异常检测时, 和正常的行为有出入的行为就会被检测出, 同时生成入侵警报。

为提高检测的整体性能, 引入了大规模的人工智能技术, 用于滥用检测与异常检测。在网络异常检测的领域中, 应用的比较早的就是专家系统^[3]。随之机器学习技术以及神经网络技术等也被陆续应用于网络异常检测中, 但在实际应用中发现, 由于审计数据规则限制, 有些入侵模式稍

离审计数据就很难被检测出来, 有些正常运行中具有较小变动的行为又可能致使虚假的警报^[4]。由此提出一种基于多约束信息融合的特定网络检测方法。

1 特定网络数据获取

基于多约束信息融合的特定网络检测中, 应对特定网络数据进行获取, 再对其进行后续处理。采用 WinPcap 对特定网络数据进行获取, 通过 WinPcap 对网络数据进行获取不仅能够利用调用 Packet.dll 中的 API 实现, 也能够通过调用 Wpcap.dll 中的 API 完成^[5]。利用 Windows 中的 Wpcap.dll 实现特定网络数据的获取。特定网络数据获取的过程为: 获取特定网络中 NIC 相关信息; 构建特定网络侦听; 制定过滤条件; 循环获取。过程如图 1 所示。

1) 获取特定网络中 NIC 相关信息。这里有两种方法可以获取 NIC 信息: 用户自己选定; 将 Pcap 接口函数进行调用, 自动检索可用的网络接口。则获取的 NIC 有关信息函数为: Pcap-lookupdev 以及 Pcap-lookupnet。其中, Pcap-lookupdev 作用是表示本地的网络接口; 而 Pcap-lookupnet 是为了得到本地的 IP 地址与子网掩码。如果函数调

收稿日期: 2018-03-02; 修回日期: 2018-04-04。

作者简介: 覃俊(1982-), 男, 广西南宁人, 工学学士, 工程师, 主要从事智能化设备管理、现工作于财政部信息中心方向的研究。

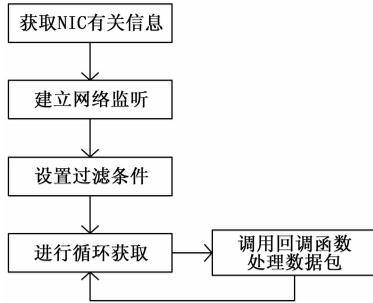


图 1 特定网络数据获取流程

用失败，则返回-1；

2) 网络侦听的启动。上述中得到了网络的接口，用到的函数是 Pcap-open-live，其责任为按指定的参数初始化 Wpcap。其返回值类型是 Pcap-t 型指针，该指针就是侦听句柄^[6]。

3) 对于数据过滤条件的设置。详细步骤为：将用户所输入字符类型定义的过滤条件，转变为系统所认可的 bpf-program 类型，然后把转换之后的过滤条件发送到侦听句柄。

4) 循环获取。以上步骤均为对 NPF（数据包过滤器）的初始化，实现初始化后，对特定网络数据进行循环获取。Pcap 库中有两个函数能够实现该功能：Pcap-loop 与 Pcap-dispath，它们功能大致相同，另外在调用成功的时候，将会返回获取到的字节数，相反，返回的则为 0。其中最大的区别就在于 Pcap-loop 获取超时，不会返回，Pcap-dispath 获取超时会返回 0。数据包所有操作均在回调函数内完成，将回调函数定义为全局函数，针对每个循环中获取的数据包，均按特定网络数据获取的定义操作。

2 特定网络数据预处理

数据粒度也可以叫作信息粒度，是指信息单元的粗糙程度^[7]。在特定网络检测中，如果数据粒度过于粗糙，会降低网络检测效率。特定网络数据的预处理中，以降低数据粒度粗糙程度为目标，利用 Rough 集理论对特定网络数据粒度进行过滤，减小信息粒度的粗糙程度，以达到提高网络检测效率的目的。通常在进行网络检测时，会利用一个规则对信息粒度进行预测，当排列分布为 1 时，则预测是正确的。不过，一个正确的高质量规则不能够保证预测的准确性。举例说明：假设通过 Rough 集法于少量对象信息系统内，发现了规则，就算其近似质量很高，不过因为支持的对象少，该种近似质量有可能是因为偶然的因素引起的，致使其在分类新的对象时，预测能力比较低^[8]。由此规则预测的高效性需要通过统计意义实现测试。

如果信息系统内的信息粒度比较高，规则的统计价值 T 通常很高。所以提出利用 Rough 集法对数据的粒度进行过滤。其基本思想为：通过决策属性值 D 确定无差别关系 θ_0 等价类的合并来提升规则统计意义，进而降低信息粒度，也就是数据粒度的粗糙程度。也就是减少等价类数量，同

时保持数据不受损失。信息粒度粗糙程度过滤的详细步骤为：

输入：数据集 $I = [U, K, V_q, f_q]_{q \in K}$ ，其中 $K = C \cup D$ 且 $C \cap D = \phi$ ， U 代表一组对象的非空且有限的集合， K 代表有限个属性非空有限集合， f_q 代表信息函数， V_q 代表属性 q 值域， C 代表条件属性值；

输出： $I^R = (U, K, V_q^R, f_q^R)_{q \in K}$ ，其中 $K = C \cup D$ 且 $C \cap D = \phi$ ；

- 1) 对于每个 $q \in K$ ，对其等价类集合 $P\{q\}$ 进行计算；
- 2) 对于每个 $q \in C$ ，将会执行下列步骤：

- (1) 对上述数据按照其属性值排序；
- (2) 对任意的两个相邻属性值 q_i 与 q_j 进行分析，并假设它们所对应的对象是 u_i, u_j ；

假设 $[u_i]_q, [u_j]_q$ 是 D 确定的，且有 $Y_{i0} \in P(D)$ 存在 $[u_i]_q \subseteq Y_{i0}, [u_j]_q \subseteq Y_{i0}$ ，那么：

(a) 把属性值 q_i, q_j 进行合并，且记为 q_{ij} ，代表取值为 q_i 或者 q_j ；

(b) $f_q^R(u_i) = q_{ij}, f_q^R(u_j) = q_{ij}$ ；

(3) 经上述处理之后的数据系统上，假设存在不同属性值 q_i 或者 q_j ，并假设两者对应对象为 $u_i, u_j, [u_j]_q$ 和 $[u_i]_q$ 为 D 确定的，且有 $Y_{i0} \in P(D)$ 存在 $[u_i]_q \subseteq Y_{i0}, [u_j]_q \subseteq Y_{i0}$ ，那么：

(a) 把属性值 q_i, q_j 进行合并，并记为 q_{ij} ，代表 q_i 或者 q_j ；

(b) $f_q^R(u_i) = q_{ij}, f_q^R(u_j) = q_{ij}$ ；

3) 结束过滤。

3 特定网络数据检测模型

通过 2 中的数据粒度粗糙度过滤操作，使基于多约束信息融合的特定网络检测变得更加快速，之后文章基于多约束信息融合的网络检测则利用构建 SVM 融合检测的模型。特定网络检测模型如图 2 所示。

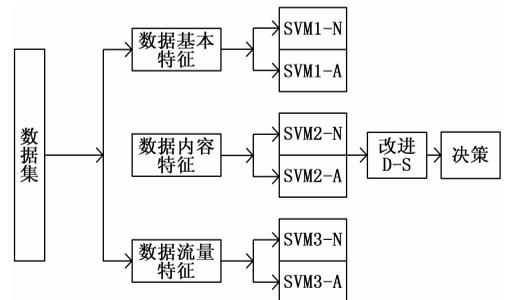


图 2 SVM 融合模型

根据上图显示，模型实现的步骤如下。

1) 训练 SVM 分类器：

把训练数据集按照其特征划分为 3 类，实现 SVM 的训练。则按照图中显示先把训练数据集和预测数据集内的特定网络数据，按照数据的特征属性将数据集中的数据划分成 3 个类别：基本特征、内容特征和流量特征。将分别代

表异常类型数据、和正常类型数据的标签, 加在各类数据集属性的后面, 设置正常数据的标签是 1, 而异常数据的标签是 -1。

依据同类数据间距离小于异类数据间距离理论, 那么在特定网络数据的检测中, 正常数据之间的距离要小于正常数据至异常数据间距离。在融合模型中通过一对 SVM-N 与 SVM-A 分类器, 实现此类数据特征的回归, 则在模型的模块中, 共用到了 6 个基本分类器, 将分类器划分为两类: A 分类器、N 分类器。其中, N 分类器提供了该特征下, 数据正常行为可信程度, A 分类器则提供了于该特征下异常数据行为可行程度。由此先分别对正常数据行为聚类的中心 DN 以及异常数据行为聚类中心 DA 进行计算。训练 N 分类器过程中, 对正常数据行为至正常数据行为的聚类中心 DN 间的距离 $DN \sim N$ 进行计算, 并取其正值, 其次对异常数据行为至正常数据行为的聚类中心 DN 间的距离进行计算 $DN \sim A$ 并取负值, 均存储于 $N \sim L$, 并在 $[0, 1]$ 间进行归一化, 当作分类器 N 训练的标签。

相同的道理, 在训练分类器 A 的过程中对异常数据行为至异常数据行为聚类中心 DA 间的距离 $DA \sim A$ 进行计算并取负值, 再对正常数据行为至异常数据行为的聚类中心 DA 间距离进行计算, 取正值, 均存储于 $A \sim L$ 内, 并在 $[0, 1]$ 间进行归一化, 当作分类器 A 训练标签。

综上所述, 对分类器 N 和 A 相应的标签进行了回归训练, 并将已经划分的 3 类特征进行训练, 分别为 SVM1-N 和 SVM1-A, SVM3-N 和 SVM3-A, SVM2-N 和 SVM2-A, 由此 SVM 各模块训练结束, 得到各类数据特征中的 N 类分类器以及 A 类分类器训练标签。

以上述为基础, 利用 D-S 完成融合决策。把获取的各个数据特征属性训练标签, 当作各类数据特征属性预测时所需要的标签, 并加上已经分好类别的预测数据集完成预测, 这样就能够获得, 测试数据集中的各类测试数据的标签类型^[9]。

2) 关于 SVM 的融合实现:

SVM 的融合实现中, 要完成基本置信分配函数的确定: 通过 D-S 理论, 设置出识别的框架 Θ 为 $\{N, A\}$ 。其中 N 是正常数据行为, A 是异常数据行为, $N \cap A = \phi$ 。将基本置信分配函数定义为:

$$m: F(\{N, A\}) \rightarrow [0, 1], m(\phi) = 0 \quad (1)$$

其中, $m(\{N, A\}) + m(N) + m(A) = 1$, 式中 $m(N)$ 代表当前情况下特征支持正常数据行为的可信度, $m(A)$ 代表当前情况下特征支持异常数据行为的可信度, $m(\{N, A\})$ 代表无法确定当前数据行为, 是否为正常的可信度。

于特定网络的检测中, 设定当一条正常的特定网络数据, 分别通过已经训练好的 SVM1-N、SVM1-A 时, SVM1-N 所计算的结果是 $m1(N)$, SVM1-A 所计算的结果是 $m1(A)$, 其中 $m1(N)$ 大于 $m1(A)$, 这样经过 D-S 实现融合时, 才可以得到良好的融合结果。

把分类器训练好, 然后对分类器的回归功能, 进行评

估。而基本置信值的预测是利用 SVM1-N 和 SVM1-A, SVM3-N 和 SVM3-A, SVM2-N 和 SVM2-A 对测试数据集的预测得到的。把结果存储至 $m1(N)$ 、 $m1(A)$ 、 $m2(A)$ 、 $m2(N)$ 、 $m3(N)$ 、 $m3(A)$, 然后将 $m1(N)$ 、 $m2(N)$ 、 $m3(N)$ 组成个新的正常行为置信分配值, 再将 $m1(A)$ 、 $m2(A)$ 、 $m3(A)$ 组成个新异常行为置信分配值。利用计算的形式获取数据无法确定的行为分配值, 设某个数据 x , 对于正常行为的置信分配值是 $m_x(N)$, 异常行为置信分配值是 $m_x(A)$, 假设 $m_x(N) + m_x(A) < 1$, 那么该数据对行为无法确定的置信函数值是 $m_x(NA) = 1 - m_x(N) - m_x(A)$ 。反之, 该数据对行为无法确定的置信函数值是 $m_x(NA) = 0$, 且把 $m_x(A)$ 和 $m_x(N)$ 进行归一化处理。综上所述就能够获得整体的 N 、 A 、 $\{N, A\}$ 的置信函数分配值。也能够获得各 SVM 分类器对 N 和 A 、 $\{N, A\}$ 可信度, 同样确定 m 函数内的置信函数值以及权重, 接着代入 D-S 的合成公式, 就能够获得检测结果。

最终的检测判定结果如下所示:

$$\begin{aligned} & \text{Data}(g) \\ & \left\{ \begin{array}{l} \text{正常} (if m_g(N) = \max\{m_g(N), m_g(A), m_g(\{N, A\})\}) \\ \text{异常} (if m_g(A) = \max\{m_g(N), m_g(A), m_g(\{N, A\})\}) \\ \text{不确定} (if m_g(\{N, A\}) = \max\{m_g(N), m_g(A), m_g(\{N, A\})\}) \end{array} \right\} \quad (2) \end{aligned}$$

4 特定网络数据过滤

当前的特定网络数据检测方法中仅进行一次检测, 内部异常数据可能存在剩余^[10], 为此在本节将对第 3 节检测结果利用群体信任法再次进行过滤, 实现网络异常数据的彻底检测和清除。具体过程如下所示:

1) 将检测结果收集起来, 经过一段时间之后, 所汇聚的节点构建基准数据特征云 BC ;

2) 确定可疑数据门限和异常数据门限, 其中可疑数据门限 ST 和异常数据门限 AT 的确定取决于特定网络, 汇聚节点把 BC 、 ST 和 AT 广播到全部簇头上;

3) 簇头对异常数据的门限 AT 进行初始化, 转换为接收到的 AT 值。且将 BC 和 ST 广播到簇内节点上;

4) 普通的数据节点按预设的频率进行存储, 记为 SV , 比较 SV 和基本数据特征集内 ex 的数值之间的差值, 将该差值和可疑的数据门限 ST 进行比较, 假设小于 ST , 则将可疑的数据标记为 0, 假设大于 ST , 则将可疑数据标记为 1, 并发送数据到簇头。

5) 当簇头接收了正常数据, 正常将数据上传, 当收到可疑的数据, 通过群体信任法计算该数据群体信任度 CT ;

6) 如果 CT 比门限 AT 低, 在判断数据为不可信时, 将该数据清除, 如果 CT 比门限 AT 高, 则判断数据为可信, 把数据发送到汇聚节点, 或者上级簇头节点进行利用。

综上所述, 对检测后的结果进一步过滤, 正常数据会被正常传输利用, 该步骤提高了网络检测的准确性, 降低了误报率, 同时特定网络节点的正常并不会被影响。

5 实验结果与分析

实验环境为：实验平台在 matlab2017a 上建立，实验的硬件环境为 acer 双核笔记本，实验数据取自于 DARPA 数据集 kdd99。

检测准确性是验证检测方法的重要指标之一，下图为不同方法检测准确性，其中图中的圆圈代表特定网络异常数据点，被黑色圆点覆盖的就是已经检测出的异常点。

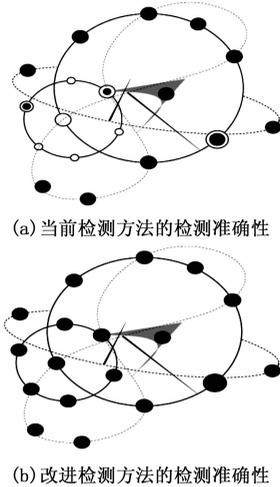


图 3 不同检测方法的检测准确性

检测的误报率同样是验证检测方法的关键因素，表 1 为不同方法检测的误报率。

表 1 不同方法误报率

攻击类型	攻击数量/个	当前方法误报率/%	改进方法误报率/%
Probing	2563	0.98	0.05
DOS	2130	0.89	0.02
R2L	2150	0.74	0.04
U2R	136	0.69	0.01

分析图 3 和表 1 可知，改进方法的检测准确性比当前方法的检测准确性要高。改进方法基于多约束信息融合的网络检测通过构建 SVM 融合检测模型实现，其中利用 D-S 进行融合决策，设置异常数据识别框架，并设置基本置信分配函数，确定函数权重，代入 D-S 理论中，获得检测结果。又利用群体信任法实现检测结果的进一步过滤，实现了异常数据的彻底清除，提高了特定网络检测的准确性，降低了检测的误报率。

下图为在一定时间和一定的数据量下，数据运行时的波形，波形中带有星星的点即为异常点，经统计异常点共有 8 个，下面利用下图对不同方法的检测效率进行验证。

分析表 2，改进特定网络检测效率优于当前网络检测方法。数据粒度的粗糙程度在一定程度上决定了网络检测效率，改进方法在特定网络数据的预处理中，以降低数据粒度粗糙程度为目标，利用 Rough 集理论对特定网络数据进行过滤，并以此来提高网络检测效率。

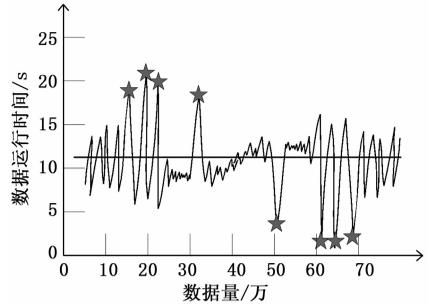


图 4 数据运行波形

表 2 不同方法网络检测效率对比

异常点数量/个	当前方法检测时间/s	改进方法检测时间/s
1	3	1
2	5	2
3	8	3
4	12	5
5	15	6
6	19	7
7	23	8
8	28	8

6 结束语

网络的异常检测是网络安全防御中不可缺少的构成部分，不过单一的方法很难得到比较好的检测结果，这限制了异常检测方法在大规模网络中的应用。分析了特定网络异常的检测，提出基于多约束信息融合的特定网络检测方法。利用支持向量机以及 D-S 证据理论多项方法的结合，以及信息融合等相关技术，将结合的结果应用在网络入侵的检测中，并利用实验证明了所提方法具有一定的可行性。

参考文献：

- [1] 李向军, 张华薇, 郑思维, 等. 基于相对邻域熵的直推式网络异常检测算法 [J]. 计算机工程, 2015, 41 (8): 132-139.
- [2] 杨宏波. 物联网环境下的差异网络数据库异常数据检测 [J]. 计算机测量与控制, 2015, 23 (3): 1008-1010.
- [3] 李艳红, 李德玉, 崔梦天, 等. 基于数据流的网络入侵实时检测框架 [J]. 计算机应用, 2015, 35 (2): 416-419.
- [4] 闫永胜, 王海燕, 张秀, 等. 多传感器网络目标检测方法综述 [J]. 系统工程与电子技术, 2015, 37 (3): 473-484.
- [5] 张晓瑜, 吴志军, 岳猛, 等. 基于网络流量奇异性特征的 LDoS 攻击检测方法 [J]. 计算机工程与设计, 2016, 37 (1): 50-54.
- [6] 赵新杰, 刘渊, 孙剑. 基于迁移学习和 D-S 理论的网络异常检测 [J]. 计算机应用研究, 2016, 33 (4): 1137-1140.
- [7] 岳军红, 王涛, 任英桥, 等. 基于神经网络算法的油田油气管道泄漏检测 [J]. 电子设计工程, 2017, 25 (10): 10-13.
- [8] 张勇, 宁蒙, 牛常勇. 基于多类别特征信息融合的车牌检测 [J]. 计算机工程与设计, 2015, 36 (1): 250-253.
- [9] 高东伟. 在线社交网络中用户伪装攻击检测方法研究 [J]. 科学技术与工程, 2017, 17 (7): 194-198.
- [10] 王曙霞. 大数据环境下的网络主动入侵检测方法研究 [J]. 科技通报, 2015, 31 (8): 225-227.

机械手三维操作空间建立与路径规划

姚舜, 付永忠, 周航

(江苏大学 机械工程学院, 江苏 镇江 212013)

摘要: 针对传统路径规划算法在机械手路径规划中未考虑机械臂干涉、搜索效率低、路径不合理等问题, 在建立了机械手空间模型, 采用二次投影法分析和计算机手操作空间的基础上, 将障碍模型和操作空间栅格化, 并提出了改进的遗传算法; 在遗传算法的设计中, 使用实数编码和三维坐标编码相结合的编码方式将栅格进行编码, 改进交叉算子, 定义最小基因交叉片段数量比, 提高了搜索效率, 简化了计算过程; 通过 Matlab 建立了机械手仿真模型, 验证了算法的有效性, 且在实验条件和实验对象等各项参数相同的情况下, 相对于传统路径规划算法, 搜索效率, 最优路径比等得到明显提升。

关键词: 机械手; 空间模型; 栅格化; 遗传算法; 交叉算子

Three-dimensional Operation Space Establishment and Path Planning of Manipulator

Yao Shun, Fu Yongzhong, Zhou Hang

(School of Mechanical Engineering, Jiangsu University, Zhenjiang 212013, China)

Abstract: According to the traditional path planning algorithm in robot path planning without considering the mechanical arm interference, low searching efficiency and unreasonable path, an improved genetic algorithm is proposed based on the establishment of manipulator space model and the analysis and calculation of manipulator operation space by using two projection method after dividing barrier model and operation space into cubic mesh. In the design of genetic algorithm, methods of using combination of real number coding and 3D coordinate coding, improving crossover operator and defining the minimum gene crossover ratio improve the search efficiency and simplify the calculation process. The simulation model of manipulator is established by Matlab, and the validity of the algorithm is verified. Compared with the traditional path planning algorithm, the search efficiency and the optimal path ratio have been significantly improved under the same experimental conditions and experimental objects.

Keywords: manipulator; space model; cubic mesh; genetic algorithm; crossover operator

0 引言

在以机械加工代替手工劳作的自动化发展进程中, 机械手在上下料、安装、搬运、喷涂、焊接、采摘等领域充当着举足轻重地位。机械手根据接收到的指令信息, 按照一定的评判标准(如时间最短, 距离最短, 能量最低等)规划出一条符合标准的最优路径, 从空间一点运动到空间另一点, 从而完成整个指令操作。

机器人路径规划的研究可以分为二维平面移动机器人的路径规划和三维机械手的操作路径规划^[1-2]。对于二维平面移动机器人的路径规划的技术研究已经较为成熟。机械手空间三维路径规划需要解决的问题包括: 1 重构三维空间环境模型。2 建立机械手空间障碍干涉操作模型。3 搜索满足避障要求的路径。4 对路径合理性进行探讨实现路径优化。5 满足动力学约束以及其它约束问题。

在国外, Khatib 提出的人工势场法^[3]虽然在机械手末端执行器的局部路径规划算法方面已经取得了应用, 但在全局避障路径规划上还需要深入研究。贾庆轩等采用 A*

算法^[4]对机器人路径进行了规划, 虽然满足了避障要求, 但对路径的合理性没有进行深入探讨。伊华伟等采用蚁群算法^[5]对机械手三维操作空间路径完成了优化, 但没有考虑机械手臂连杆与障碍物的干涉情况。

本文在采用栅格法建立机械手的环境模型以及垂直平面两次投影法^[6]建立机械手操作空间栅格模型的基础上, 提出了基于改进遗传算法的机械手三维操作空间路径规划方法。最后采用 Matlab 工具建立了机械手仿真系统, 验证了算法的有效性, 保证了机械臂连杆与障碍物不发生干涉, 且相对于传统路径规划算法, 搜索效率, 最优路径比等得到明显提升。

1 问题描述及空间模型的建立

1.1 问题描述

因为机械手路径规划研究的是机械手末端的位置变化, 不考虑机械手末端的姿态, 因此可以将机械手的末端作为质点处理。本文以 Scara 机械手为研究对象, 机械手避障问题描述如图 1 所示。图中 A 为障碍物, B 为机械手末端。末端 B 从起始点 q_0 运动到目标点 q_1 , 要求保证机械臂本体与障碍物不发生碰撞且使机械手末端点 B 从 q_0 到 q_1 运动路径最短。与行走机器人路径规划相比, 机械手路径规划必

收稿日期:2018-03-04; 修回日期:2018-04-03。

作者简介:姚舜(1992-), 男, 江苏泰兴人, 硕士研究生, 主要从事机电控制方向的研究。

须避开机械臂与障碍物的干涉。

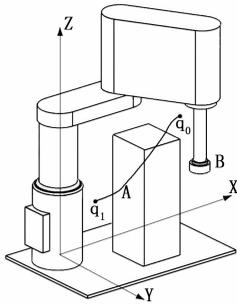


图 1 Scara 机械手避障问题描述模型

机械手三维操作空间的路径规划流程如图 2 所示。机械手获取到外部环境信息，内部建立起障碍物模型，并根据自身尺寸、关节角等信息建立起关节空间障碍模型，通过路径搜索算法获得空间点位，最后运用机械手运动学逆解求解各关节角，机械手根据获取指令从而完成相应的动作。本文主要研究的是机械手的三维操作空间路径规划，因此空间环境信息以一个标准的立方体块作为障碍物。

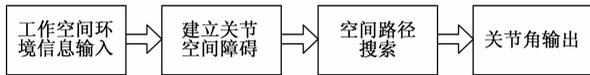


图 2 机械手三维操作空间的路径规划流程图

1.2 空间模型及操作模型的建立

本文运用两次投影的思想研究 Scara 机械手操作空间模型。Scara 机械手操作空间模型 V 平面投影即为障碍物空间的 V 平面投影， H 平面的投影如图 3 所示。上临界碰撞角 θ_{2UC} 和下临界碰撞角 θ_{2LC} 之间的变化关系如式 (1) 和式 (2) 所示：

$$\theta_{2UC} = \theta_1 - \arctan\left(\frac{y - a_1 \sin\theta_1}{x - a_1 \sin\theta_1}\right) - \arctan\left(\frac{w_2}{2a_2}\right) \quad (1)$$

$$\theta_{2LC} = \theta_1 + \arctan\left(\frac{y - a_1 \sin\theta_1}{x - a_1 \sin\theta_1}\right) + \arctan\left(\frac{w_2}{2a_2}\right) \quad (2)$$

点 $A(x, y)$ 和点 $B(x, y_1)$ 为临界碰撞点坐标， a_1 、 a_2 分别为两杆的长度， w_1 、 w_2 分别为两杆的宽度，障碍物高度为 z 。关节坐标 $(\theta_{2UC}, \theta_{2LC})$ 到空间坐标 (X, Y) 的转换公式如式 (3) 和式 (4) 所示：

$$X = a_1 \cos\theta_1 + a_2 \cos(\theta_1 - \theta_{2UC}) \quad (3)$$

$$Y = a_1 \sin\theta_1 + a_2 \sin(\theta_1 - \theta_{2LC}) \quad (4)$$

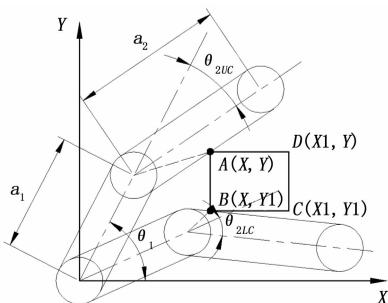


图 3 Scara 机械手临界避障点 H 平面投影

垂直于 H 平面的障碍物空间体集合表示为 D_H ，垂直于 V 平面的障碍物空间体集合表示为 D_V ， D_H 和 D_V 的集合表示如下：

$$D_H = \left\{ (X, Y, Z) \mid \begin{aligned} &x > a_1 \cos\theta_1 + a_2 \cos(\theta_1 - \theta_{2UC}) \\ &y > a_1 \sin\theta_1 + a_2 \sin(\theta_1 - \theta_{2LC}), \\ &Z \in (-\infty, +\infty) \end{aligned} \right\} \quad (5)$$

$$D_V = \{ (X, Y, Z) \mid x < X < x_1, 0 < Z < z, Y \in (-\infty, +\infty) \} \quad (6)$$

则最终的障碍物操作空间即为 H 平面障碍物空间体构成的集合与 V 平面障碍物空间体构成的集合的交叉部分，集合 D_B 可以表示为：

$$D_B = D_H \cap D_V \quad (7)$$

2 路径规划算法设计

针对在解决二维路径规划的问题上，遗传算法在众多的文章中提出，众多学者尝试将其不断改进更好得解决了路径规划的问题，被证明是一种有效的路径规划算法^[7-9]。将这种算法应用到三维空间，本文提出了栅格化操作空间模型，采用改进遗传算法对机械手三维操作空间进行路径规划。三维路径规划的遗传算法主要改进方案有下面几点：采用三维编码和实数编码相结合的编码方式，适应度函数包括了路径最短和避障的要求。

2.1 遗传算法基本原理

1) 遗传算法的原理。

遗传算法是模拟生物界自然选择学说提出的，完全符合了“物竞天择，适者生存”的观点。自然选择学说主要包括生物种群的遗传、变异、生存斗争和适者生存。遗传算法将生物种群特征串联起来并进行相应的编码，将大自然的选择作为适应度函数，使用适应度函数评估现有的种群个体，适应度值大的个体将被保留，按照适应度值提供的方向进行交叉变异等操作从而产生新的后代个体，新的后代个体组成的种群再次被评估执行下一步操作，周而复始，得出问题的最优解。遗传算法以初始种群为问题的解，以适应度函数为导向，确定解搜索方向，避免了传统优化算法求导和不可行解的问题；同时它是一种全局搜索方法，能够有效解决局部最优的问题；概率选择的方法增加了遗传算法使用的灵活性，针对不同问题都可以灵活的建立相应问题的解决模型。

2) 遗传算法的步骤。

遗传算法的流程图如图 4 所示。观察流程图可以知道遗传算法的主要步骤包括如下几步^[10]。

Step1: 将种群特征进行编码，使实际问题的解 X 转化为编码解 D 的问题。

Step2: 确定个体编码特征参数 x, y, z, \dots ，确定适应度函数 $fit(x, y, z, \dots)$ 。

Step3: 确定遗传策略，包括确定种群数量 M ，选择、交叉、变异方法，交叉概率 p_c ，遗传概率 p_m 等。

Step4: 随即产生由初始个体编码构成的初始种群 p_0 。

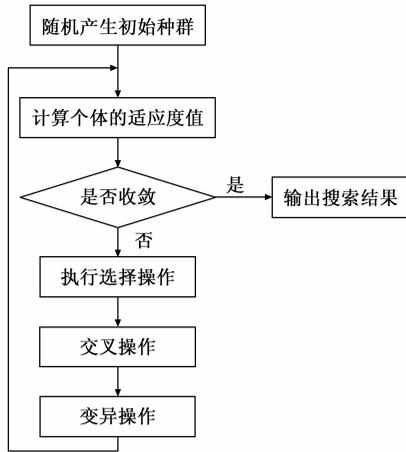


图 4 遗传算法流程图

Step5: 将个体编码特征参数代入适应度函数 $fit(x, y, z, \dots)$, 计算适应度函数值。

Step6: 按照遗传策略, 进行选择、交叉和变异等操作, 进化出下一代新的个体组成的种群。

Step7: 判断是否所需求解问题的最终解或者中间解是否趋达到终止条件, 若达到条件则结束当前循环, 输出搜索结果, 若没有达到终止条件则跳到 Step5, 计算新一代个体的适应度值, 直到满足终止条件。

2.2 环境模型及遗传算法路径编码

机器人路径规划的研究都要基于环境模型和操作模型, 对二维平面移动机器人路径规划的研究一般都是以栅格化的方法建立栅格模型。将这种方法应用到三维空间建立三维栅格模型如图 5 所示。三维栅格模型由空间 $N \times N$ 个立方体素构成。体素分为两个状态, 障碍物体素 (如体素 A) 和自由体素 (如体素 B)。对于环境模型, 障碍物体素表达的是空间障碍物模型, 对于操作空间而言障碍物体素除了表达空间障碍物模型外还表达机械臂与障碍物的干涉模型, 即为上述障碍物操作空间集合的栅格化模型。每一个体素的位置由笛卡尔空间坐标系来表达。对于 Scara 机械手而言, 将机械手末端作为质点处理, 笛卡尔空间坐标系中每一个体素的位置与机械手关节空间一一对应。

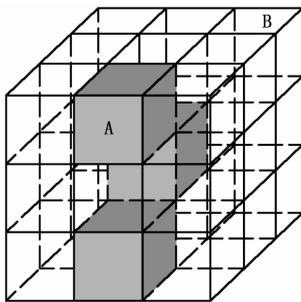


图 5 三维栅格模型

图 5 中, 将环境栅格模型的左下角作为原点, 建立如图所示坐标系, 则每一个栅格的位置可以用坐标表示, 如左下角第一个栅格的坐标即为, 沿着轴方向的第二个栅格

坐标为, 依次将栅格进行编号。路径编码策略主要有实数编码, 浮点数编码, 二进制编码等, 实数编码能够方便遗传算子的交叉和变异等操作。同时, 若将三维坐标进行编码则能简单直观表示出各编码实际路径, 方便计算出路径的长度, 评估路径的可行性。综合考虑以上两点, 针对空间三维路径规划的问题, 本文提出实数编码与三维坐标编码相结合的编码策略。实数编码栅格编号 i 与三维坐标编码栅格坐标 (x_i, y_i, z_i) 的映射关系如下:

$$N = \text{int}(\sqrt[3]{i}) + 1$$

$$x_i = N$$

if

$$0 < i - (N - 1)^3 \leq N^2$$

then

$$y_i = \text{mod}([i - (N - 1)^3] / N)$$

$$z_i = \text{int}([i - (N - 1)^3] / N)$$

else

$$y_i = \text{mod}([i - (N - 1)^3 - N^2] / N)$$

$$z_i = \text{int}([i - (N - 1)^3 - N^2] / N)$$

上述运算关系中, mod 表示取余运算, int 表示取整运算, N 为实数编号栅格 i 的 x 轴坐标, 空间中表示了该栅格在第 N 个阵列层, 求栅格 i 的 y 轴和 z 轴坐标即为求该栅格所在阵列层的相应位置坐标。该映射关系的主要思想为正方体素的空间阵列, 将正方体素作为基本单位, 以正方体素构成的正方体的一个边角顶点为起始点, 不断以正方体素向空间三维的同一个方向延伸构成由 i 个正方体素构成的新正方体。

2.3 适应度函数

由于机械手臂与障碍的干涉已经在操作空间模型中考虑到了, 所以在正确建立了机械手操作空间模型之后, 机械手的三维操作空间的路径规划的适应度函数与普通的二维平面行驶机器人路径规划适应度函数所满足的条件类似, 即在保证机械手不与障碍物发生碰撞的基础上使得总规划路径最短, 对应到栅格操作空间即要求各自由编码栅格之间的连线不经过障碍物编码栅格的基础上, 使整条编码栅格的总长度最短。

1) 任意一条路径内的编码栅格 (用 p_i 来表示) 必须为自由编码栅格即不能落在障碍物编码栅格内。将障碍物栅格编码的集合表示为 W , 则路径点编码栅格不落在障碍物编码栅格内的适应度函数可以表示为:

$$Fit_{11} = \begin{cases} 1, p_i \notin W (i = 0, 1, 2, \dots, n) \\ 0, p_i \in W (i = 0, 1, 2, \dots, n) \end{cases} \quad (8)$$

式中, p_i 表示操作路径上第 i 个栅格点, n 表示操作路径中栅格点个数。公式表明, 路径中的栅格点落在障碍物栅格区域的适应度函数为 0, 落在自由栅格区域的适应度函数为 1。

2) 用 p_i 和 p_{i+1} 来表示路径中相邻的两点, $p_i p_{i+1}$ 表示相邻两点的连线, 则连线不经过障碍物栅格的适应度函数可以表示为:

$$Fit_{12} = \begin{cases} 1, p_i p_{i+1} \cap W \neq 0 (i = 0, 1, 2, \dots, n) \\ 0, p_i p_{i+1} \cap W = 0 (i = 0, 1, 2, \dots, n) \end{cases} \quad (9)$$

综合以上两点，可以得出机械手路径规划中，机械臂能够有效避免碰撞的适应度函数为：

$$Fit_1 = Fit_{11} \times Fit_{12}$$

对适应度函数的要求除了避免碰撞之外还要求运动的路径最短，即要求计算出每条路径的长度，相邻两点路径 $p_i p_{i+1}$ 的长度计算公式如下：

$$L_{(i)} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \quad (10)$$

设整条操作路径栅格点数为 n ，则整条路径的总长度为：

$$Length = \sum_{i=1}^{n-1} L_{(i)} \quad (11)$$

适应度函数值越大基因编码被保留下来的概率越大，因此可以得到最短路径的适应度函数为：

$$Fit_2 = \frac{1}{Length} \quad (12)$$

综合考虑免碰撞以及路径最短适应度函数，可以得到用于机械手三维操作空间路径规划的最优适应度函数可以表示为：

$$Fit = C_{max} Fit_1 + a \times Fit_2$$

公式中 C_{max} 的取值与操作空间的复杂程度有关， a 为不可行操作路径的惩罚系数。

2.4 遗传算子

2.4.1 选择算子

选择操作是建立在适应度值的基础上的，适应度值越大的个体被保留，作为父代进行交叉变异等操作，从而产生下一代个体。选择算子的种类较多，比较常用的选择算子有轮盘赌选择、随机竞争选择、最佳保留选择、无放回随机选择等。本文采用轮盘赌和精英保留策略相结合的方法进行遗传算法的选择操作。轮盘赌和精英保留策略的思想是每个个体进入下一代的概率等于它的适应度值与整个种群中个体适应度值和的比例，然后将当前群体中适应度最高的个体结构完整地复制到下一代群体中。

2.4.2 交叉算子

遗传算法中，交叉操作应用在选择操作之后，通过染色体之间基因的互换从而产生新的基因序列，每一条新的基因序列组成的染色体即一种新的路径方案。交叉操作是产生新的路径方案的必要手段。基因的交叉互换是随机的，交叉操作主要包括单点交叉、两点交叉、多点交叉、均匀交叉和算术交叉。交叉方法的选择应针对不同的编码策略进行选择，针对前面提出的实数编码策略，本文采用的是单基因片段的交叉即两点交叉。各编码路径是按照实数编码从小到大顺序排列，通过预先设定基因片段交叉最低长度，比较基因片段实数编码的大小，来选择可交叉的基因片段。基因片段长度设定值越小，可交叉的基因片段种类越多，搜索情况越复杂，搜索效率越低。相反设定值越大基因种类越少，不能保证种群的多样性，容易陷入局部最优。本文根据与总路径长度的比值取值为 0.1。

假设两个待交叉的父代个体 (27, 98, 103, 371, 483, 978, 1 239, 1 755, 3 982)，父代个体 (81, 99, 271, 372, 771, 1 560, 2 040, 3 961, 4 092)，如图 5 所示，基因交叉片段长度为 6，按照顺序排序进行实数编码，通过比较各实数编码的大小，父代个体在 103 和 1 755 之间的基因片段与父代个体在 99 和 2 040 之间的基因片段存在可互换性。从三维坐标编码看，每个基因片段表示路径的子路径编码，且这两段子路径可以相互替代，交换得到的染色体仍然是优良的后代，与传统遗传算法的无序随机交叉相比，采用实数顺序编码的编码方式大大加快了算法的收敛速度。



图 6 两父代个体基因片段交叉

2.4.3 变异算子

遗传算法的变异过程是指在原始染色体上的基因位通过一些等位基因来替代，产生新的基因片段的过程。变异方法主要有基本位变异、均匀变异、边界变异、非均匀变异和高斯近似变异等。变异算子是保证种群多样性的必要手段，本文对实数编码进行非均匀变异操作，通过对染色体原有基因位做随机的微小扰动，每个基因位以相同的概率进行变异操作后从而产生新的染色体，在空间路径解中即产生了新的解决方案。变异操作的概率通常取值范围为 0.001~0.4，本文取值为 0.01。

3 实验结果及分析

3.1 实验结果

本文基于酷睿 i7-6700 处理器，内存 6 GB，Windows7 旗舰版 X64 位操作系统，采用 Visual Studio2013 和 Matlab2014a 64 bit 混合编程的方式验证了算法的有效性。

实验中各项参数设定如下：种群规模为 $M=80$ ，交叉操作概率 $P_c=0.75$ ，变异操作的概率为 $P_m=0.01$ ，最大的迭代次数为 $T=500$ ，基于实数编码的交叉基因片段长度比例 $l_c=0.1$ 。Scara 机械手的 Matlab 仿真模型及仿真结果如图 6 所示，为起始点，为目标点，机械手末端从沿着仿真结果的连续路径点运动到，此路径即为最优路径，既避免了机械臂与障碍物的碰撞，同时使得从起始点到目标点路径最优。观察图 7 可以看到，本文提出的改进遗传算法在迭代到第 240 次左右趋向于收敛，即找到了最优路径，总长度为 11.2，单位为栅格。

3.2 算法对比

将本文算法与其它三维路径规划算法进行性能参数比较，从表 1 中可以看出本文所提出的改进遗传算法在最优路径的选择上比传统的一些算法获取的最优路径更短，在相同的实验条件的基础上搜索效率得到提高。在重复性实验过程中所得到的最优路径比例明显高于其它算法。由于目

基于 STL 文件的柱状支撑结构 自动生成算法

宋廷强, 刘亚林, 张敏

(青岛科技大学 信息科学技术学院, 山东 青岛 266061)

摘要: 熔融挤压三维打印以热塑性材料为原料逐层打印完成, 由于熔丝只能沉积在已存在的物体上层, 模型悬空部位需要添加支撑结构; 为解决该问题, 提出一种基于 STL (stereo lithography) 文件的稀疏柱状支撑结构自动生成算法; 算法通过对比三角面片倾斜角度与模型临界角的大小, 获取独立的待支撑区域, 然后基于边长自适应法和射线与面片相交法得到待支撑点集, 自动生成支撑结构; 此外为了保证模型表面质量和支撑结构的稳定性, 支撑结构不在模型表面生成; 调整支撑结构形状, 进一步提高支撑结构的稳定性; 通过实验证明本文算法比 Cura 软件节约 15% 的材料消耗, 支撑结构更容易剥离, 模型表面质量更好。

关键词: 三维打印; STL 模型; 支撑结构; 快速成型; 自动生成算法

Automatic Generation Algorithm of Cylindrical Support Structure Based on STL File

Song Tingqiang, Liu Yalin, Zhang Min

(1. Qingdao University of Science and Technology, Qingdao 266061, China)

Abstract: Fused deposition modeling 3D printing thermoplastic material as a raw material layer by layer print completed, because the fuse can only be deposited on the existing objects in the upper part of the model need to add support structure hanging parts. To solve this problem, an automatic support structure generation algorithm based on STL (Stereo Lithography). Algorithm to obtain an independent area to be supported, based on the edge-length adaptive method and ray and patch intersection method to be supported point set to automatically generate the support structure. In addition, the support structure is prevented from growing on the surface of the model and the shape of the support structure is adjusted. Experiments show that the proposed algorithm saves 15% of the material consumption compared with Cura software, the support structure is more easily peeled off and the surface quality of the model is better.

Keywords: 3D printing; STL model; support structure; rapid prototyping; automatic generation algorithm

0 引言

随着全球工业的快速发展, 快速成型技术 (rapid prototyping, RP) 已经成为发展最快的技术之一。目前存在多种打印技术, 但是其本质的不同在于打印的材料和打印的方式。有一些利用融化的可塑性材料来打印, 比如: 选择性激光烧结 (selective laser sintering, SLS) 和熔融挤压 (fused deposition modeling, FDM), 还有一些技术是用液体材料打印, 例如: 立体平板印刷 (stereolithography, SLA) 和分层实体制造 (laminated object manufacturing, LOM)。每一种技术都有各自的优缺点, 一般根据不同的需求、打印机成本、打印速度和材料等方面考虑, 选择合适的打印技术^[1]。

熔融挤压 (FDM) 型三维打印机一般以热塑性材料, 如 PLA (Polylactic Acid) 塑料、ABS (acrylonitrile - butadiene - styrene copolymer) 塑料、尼龙等为打印原料,

打印时先将材料融化, 然后通过喷头沿着一定路径将材料挤出到打印平台, 材料迅速冷却凝固, 然后在打印第二层, 直到模型打印完毕。由于 FDM 这种堆叠式的加工方式, 遇到模型中悬空部分是, 打印材料无法在半空中凝固成型, 因此需要用户为悬空部分模型添加支撑结构^[2-3]。支撑结构会随着模型一起被打印, 但是由于其不属于模型本身, 因此在完成打印后需要将支撑结构拆除。支撑结构拆除会对模型表面造成损伤, 因此在保证模型顺利成型的前提下应该尽可能的减少支撑结构与模型表面的接触面积和减少支撑结构本身的体积。

目前在支撑研究方面已经有人做了大量的工作, Wang^[4]等提出了一种蒙皮——框架轻量打印模型, 并在此基础上拓展出一种圆柱支撑结构的生成方法。陈岩等^[5]在 wang 的基础上提出一种改进方法, 这两种方法的支撑结构均为独立圆柱体, 虽然可以节省时间, 但是打印时稳定性不够。Cura 软件通过在待支撑面添加稀疏密度的网格结构作为支撑体, 这样虽然支撑结构稳定, 但是会浪费材料, 增加打印时间。Zhang^[6]等受自然骨架的启发, 设计在模型内部的六边形支撑结构, 该结构能够使模型承受外部各个

收稿日期: 2018-03-06; 修回日期: 2018-03-30。

作者简介: 宋廷强 (1971-), 男, 博士, 副教授, 硕导, 主要从事嵌入式系统、计算机控制、计算机应用方向的研究。

方向的负荷, 稳定性好, 但是支撑结构在模型内部, 无法去除。MeshMixer 公司提出一种树状支撑结构生成方法, 但是该方法某些时候仍需要人工干预, 有一定的学习成本, 而且有时会出现打印失败的现象。宋国华^[7]等基于 L-系统原理设计一套仿树状支撑结构的生成方法, 该树状支撑结构对称生长, 稳定性好, 但是支撑结构与模型接触点坐标计算复杂, 且不易去除。

本文设计实现一种自动生成的柱状支撑结构, 首先对 STL 文件建立拓扑结构, 划分出独立的待支撑区域, 然后通过边长自适应法获得平面中的采样点, 进而通过射线与面相交法获得三维模型表面的待支撑点集。为了提高模型整体稳定性, 本文的支撑结构尽量从 XOY 平面生成。最后对支撑结构进行调整, 方便从模型表面剥离。

1 STL 文件拓扑结构建立

STL 文件由美国 3D System 公司提出的一种 CAD 系统与快速成型系统之间的数据交换格式, 由于它格式简单, 对三维模型建模方法没有特定的要求, 因此得到广泛的应用, 成为快速成型系统中事实上的标准文件输入格式^[8]。STL 文件只存放 CAD 模型表面的离散三角形面片信息, 并且三角面片信息是无序存储的。

由于 STL 文件中三角面片信息都是无序存放的, 且存在大量的冗余数据。由于缺乏各面片间的邻接关系, 如果寻找某个三角面片的相邻面片时需要进行大量计算, 非常不利于后续待支撑面片的合并。因此需要建立 STL 文件的拓扑序列, 本文设计的拓扑结构如图 1 所示。

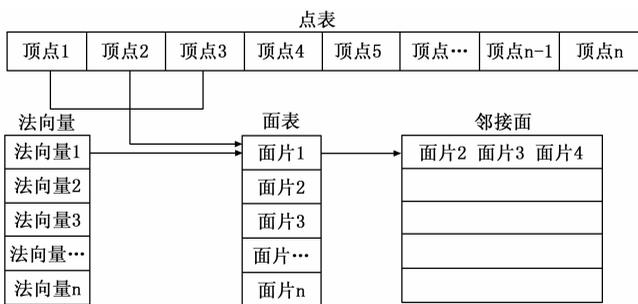


图 1 STL 文件拓扑结构示意图

首先需要建立一个存储无重复顶点的点表, 由于在 STL 文件中相邻三角形面片共用两个顶点, 因此 STL 文件中存储大量的冗余顶点坐标。建立点表的目的是为了消除冗余信息, 每个顶点坐标只存储一次, 减少计算量。传统方法通过先判断两个顶点坐标是否完全一致, 不一致则存储到点表的方法效率太低, 会浪费大量时间。由于用哈希表存储顶点的坐标检索运算平均可能达到常数的时间, 又能通过哈希函数的运算快速找到冗余的顶点坐标^[9], 因此本文把顶点的坐标作为关键字, 通过哈希函数快速计算出顶点对应的哈希地址。哈希地址冲突时说明两顶点坐标相同, 不需要存储到点表中。

然后建立存储所有三角面片点信息的面表, 面表中不直接存储点坐标信息, 而是存储该点坐标在点表中的索引位置。同时建立与面表对应的法向量表。正确的 STL 数据模型必须满足如下一致性规则: 1) 相邻两个三角形之间只有一条公共边, 即相邻三角形必须共享两个顶点; 2) 每一条组成三角形的边有且只有两个三角形面片与之相连。因此可以通过两个面片存储的点坐标索引值是否有两个相同而判断两个三角面片是否相邻, 可以得到面片的邻接面片索引值。这样当有需要时可以通过面表可以直接找到对应顶点的索引, 也可以快速找到相邻的面表索引, 计算机处理速度更快, 可以更快捷的得到需要的信息。

2 待支撑面片提取与快速合并

2.1 临界角定义

在 FDM 技术中由于材料自身的粘性作用, 在一定的倾角范围内, 成型表面不需要添加支撑。计算所有三角面片的法向量 n 与 Z 轴正方向之间的夹角 γ 如图 2 所示, 本文测得打印机的临界角度为 α , 如果 $\gamma > \alpha$, 则该三角面片为待支撑面片。由于不同打印机, 不同打印材料, 以及不同的打印参数的影响, 临界角并不是一个固定的值。本文以 SmartMaker 打印机使用 PLA 材料进行实验。通过实验结果比较, 临界角 $\alpha = 40^\circ$ 比较合适。

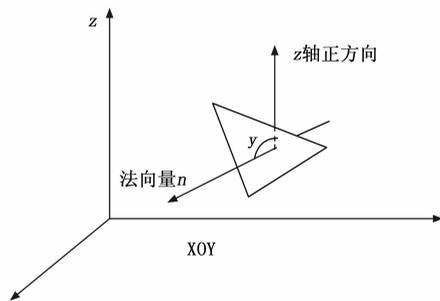


图 2 临界角示意图

2.2 待支撑面片快速合并

通过计算临界角, 得到的待支撑三角面片都是互相独立的, 为了合理添加支撑结构和减少后期计算量, 需要将独立的三角形面片按相邻关系连接起来, 得到独立的待支撑区域。面片合并采用的方法为种子扩散法^[10]如图 3 所示。取待支撑面片集合中的第一个待支撑面片 a 作为种子面片, 存入种子表, 然后使用上一小节 STL 文件拓扑结构建立算法得到的面片邻接表, 从中快速找到 a 面片的 3 个相邻面片 b, c, d , 分别判断这 3 个相邻面片是否在待支撑面片集合中, 如果存在, 则将该相邻面片存入种子表, 并将 a 面片从待支撑面片集合中删除, 如果 3 个相邻面片都不在待支撑面片集合中, 则代表 a 面片是一个孤立的待支撑面片。然后继续寻找种子表中的其他种子的相邻三角面片, 直到种子表中所有种子都查找过为止, 一个独立的待支撑区域查找完成, 重复上述方法, 直到待支撑面片集合为空时,

所有待支撑区域全部查找完毕。

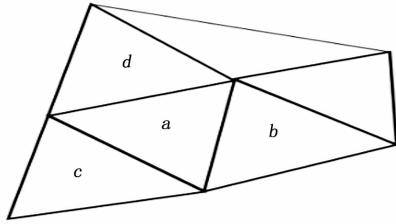


图 3 种子扩散法示意图

3 支撑点采样

通过上节的计算可以得到若干个独立的待支撑区域, 若对每个待支撑区域的所有待支撑面都添加支撑, 则会导致支撑过难以去除的问题。因此对悬空区域进行采样, 获得少量待支撑点是非常必要的。但是对于某一处悬空区域, 如果采样点过少, 则不能保证模型正常成型。若是采样点过多, 虽然可以很好的保证模型成型, 但是支撑难以去除而且浪费大量材料。因此采样点的数目应该在一个合理的范围之内, 既能保证模型正常成型, 又能节约材料。由于在三维空间中进行重采样需要进行大量的计算, 会严重降低算法的效率, 因此本文采样算法在悬空区域的投影区进行, 然后通过计算得到对应的悬空面上的点。

3.1 边长自适应采样法

栅格采样是最常用的方法, 将待支撑区域的三角面片坐标像平面进行投影, 取得投影区域的 X, Y 轴的最大最小值, 继而获得待支撑投影区域的最小矩形包围盒。平面投影区域内通过给定采样间距 d 将投影区域栅格化, 然后在每个栅格内选取对应的点。但是该方法由于间距 d 为给定的固定值, 矩形包围盒的长宽可能不是 d 的整数倍, 因此采样时会存在误差。并且当投影矩形一条边小于间距 d , 而另一条边大于间距 d 时, 短边会被忽略, 影响待支撑点的选取。

边长自适应采样是在栅格采样的基础上改进而来, 栅格采样基于固定的采样间距, 而边长自适应采样的间距根据边长不同是动态变化的。对于一个投影区域, 需要找到最合理的采样点数 N , 既可以使模型正常成型, 又能尽可能的减少支撑数量。假设材料在没有支撑下保持不变形的最长打印长度为 d_{max} , 因此为了避免误差, 我们认为在投影矩形的 X, Y 边上的最大采样数目为:

$$\begin{cases} N_x = \frac{(X_{max} - X_{min})}{N_x} + 1 \\ N_y = \frac{(Y_{max} - Y_{min})}{N_y} + 1 \end{cases} \quad (1)$$

其中, $\lceil \cdot \rceil$ 表示向上取整, 因此 X, Y 边的采样距离分为:

$$\begin{cases} d_x = \frac{(X_{max} - X_{min})}{N_x} \\ d_y = \frac{(Y_{max} - Y_{min})}{N_y} \end{cases} \quad (2)$$

此时可以在投影矩形中得到 N_x, N_y 个采样点 (x_i, y_i) :

$$\begin{cases} x_i = X_{min} + i * d_x \\ y_i = Y_{min} + i * d_y \end{cases} \quad (3)$$

对于一条边小于 d_{max} , 而另一条边远大于 d_{max} 的狭长投影矩形的情况, 我们认为不应该直接将短边忽略掉。因为如果切片时沿长边方向切片时, 打印头沿长边运动, 那这样会因为支撑点不足影响模型成型。因此对于这种情况我们规定长边采样距离正常取值, 短边采样距离应为短边长度的一半。

3.2 局部最低点采样法

上一节边长自适应采样法得到一组位于栅格交点上的采样点, 这种采样点分布的优点是排列整齐, 支撑结构之间距离相同。但是在实际打印过程中, 在某一区域内切片软件一般从该区域的最低点开始切片, 支撑结构在该区域的最低点下方生成相对比较合理。而上一小节的算法得到的采样点中可能会丢失局部最低点, 局部最低点采样法在边长自适应采样法的基础上改进而来。首先根据边长自适应采样法获得投影矩形各边的采样距离, 得到采样栅格, 然后将投影下来的各三角面片的顶点按着栅格边距进行划分, 选择一个栅格中高度最低的三角面片作为采样点。当栅格内最低点采样完成后, 需要对网格最低点进行筛选优化, 因为可能出现两个最低点距离非常近的情况。这时需要删除掉一个点, 选择栅格中心点作为新的采样点。该算法的具体流程如下。

Step1: 将最大采样距离设为 $d_{max} \setminus 2$, 根据边长自适应采样法得到各边的采样距离, 创建存储采样点的数组 supportP;

Step2: 根据各边采样距离, 划分网格;

Step3: 将待支撑三角面片的顶点坐标按着栅格边距划分到各个栅格中;

Step4: 从坐标 (x_{min}, y_{min}) 的栅格开始遍历, 寻找每个栅格中高度最低的面片顶点作为采样点, 加入到数组 supportP 中。继续寻找其他栅格中的最低点, 如果某个栅格中的最低点与 supportP 数组中的某一点距离非常近, 小于 d_{min} , 则放弃该点, 选择该网格的中心点作为采样点加入 supportP 数组; 否则将栅格中最低点加入 supportP 数组中。

采用上述方法得到的采样点可以保证选到该待支撑区域内的最低点, 虽然该方法的采样点不如边长自适应采样法得到采样点分布的均匀, 但是该方法有效的提升了支撑结构生成位置的合理性。

3.3 检测点与三角形位置

在矩形区域内取得采样点坐标后, 需要判断采样点是否位于悬空区域投影的内部。由于计算平面中某个点与不规则图形的位置比较麻烦, 本文提出一种简化算法, 将其转化为判断采样点与三角形的位置关系。通过上文已经获得所有待支撑三角面片坐标值, 以一个待支撑三角面片为

例,如图 4 所示,其坐标值 (A, B, C), 并且待支撑面片垂直向 XOY 平面进行投影, 因此计算可得三角面片在 XOY 投影坐标 (A', B', C')。采样点是否在投影区域的判断, 就可简化为判断每个采样点是否在相应的三角面片的投影三角形内 (A', B', C') 即可。

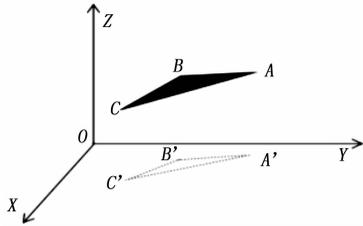


图 4 三角面片投影示意图

本文采用重心法^[11]判断点是否在三角形内部。如图 5 所示, 三角形 3 个点 ABC 在同一水平面上, 若选择 A 为起点, 则 B 点相当于在 AB 方向移动一段距离得到, 同样 C 点相当于在 AC 方向移动一段距离得到。因此, 对于该三角形内任意一点都可以有公式得到:

$$P = A + U * (C - A) + V * (B - A) \quad (4)$$

因此, 如果点 p 在三角形内部需要满足以下条件:

- (1) $U \geq 0$
- (2) $V \geq 0$
- (3) $U + V \leq 1$

通过计算得到 U, V 后, 比较是否满足以上条件, 若满足则点在三角形内部, 若不满足, 则点不在该三角形内部。

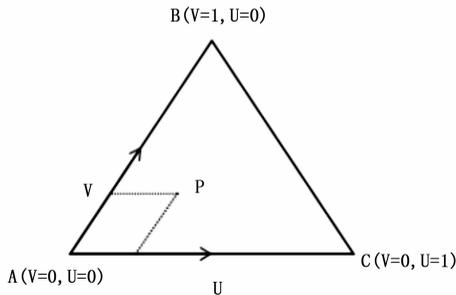


图 5 三角形向量表示图

3.4 射线与三角面片交点

通过上节算法证明采样点 p 在投影三角形内部, 但是由于支撑需要添加到模型表面, 因此需要计算出采样点在模型表面的坐标值。由图可知, 三角形 A' B' C' 内的采样点一定会在空间三角形 ABC 内部, 而且 X, Y 坐标不变, 只是 Z 轴坐标发生变化。因此, 求采样点在三维模型表面坐标简化为求过采样点 p 沿 Z 轴正方向的射线与三角形 ABC 的交点坐标, 射线与三角形的交点公式如下。

$$O + Dt = (1 - u - v)V_0 + uV_1 + vV_2 \quad (5)$$

通过化简求出 t, u, v 然后带入公式即可求出射线与三角形的交点, 该点即为待支撑点。

4 支撑结构生成

通过对每个投影区域的计算, 可以得到待支撑点集 P, 支撑结构的生成就是建立在待支撑点集 P 的基础上。陈岩等^[5]实现的柱状支撑结构将部分支撑生长在模型表面, 这样虽然具有减少支撑结构、缩短打印时间的优点, 但是由于模型表面本身不平坦, 并且与支撑结构接触面较小等原因会导致支撑结构不稳定。打印模型是一个连续的过程, 某一部分支撑的不稳定会极大增加模型打印失败的概率。若是模型打印失败, 那么前期为了节省材料和时间而做的工作就是无意义的, 因此为了提高模型打印成功率, 本文算法增加少量的支撑结构和打印时间从而提高模型稳定性。所以本文算法尽可能避免支撑结构在模型表面生成。对于每一个待支撑点 $p \in P$, 寻找一个支撑基点 q, 生成一个以 p, q 为顶点的柱状支撑结构。基点的选取, 本文算法首选在 XOY 平面上处于 p 点正下方的 q 点作为支撑基点。但是这样选择的 pq 支撑体, 可能会与模型相交, 还是会出现支撑结构生长在模型表面的问题。基于这个问题本文提出一个算法, 保证支撑体不与模型相交的前提下, 在 XOY 平面重新选择一个支撑基点 q', 使支撑体 pq' 既不与模型相交, 而且 pq' 长度是在所有选择中最短的。

4.1 支撑结构与模型相交检测

假设 pq 支撑结构与模型相交, q 是在 XOY 平面上处于 p 正下方的一个支撑基点。为了避免支撑结构与模型相交, 需要在 XOY 平面重新选择一个支撑基点 q', 由上文可知模型不需要支撑能够成型的临界角 $\alpha = 40^\circ$ 。因此对于重新选择的支撑基点 q', pq' 的倾斜角度也必须小于 α , 考虑到 pq' 为支撑结构, 还需要为模型提供有效支撑, 本文将 pq' 的最大倾斜角度定义为 $\beta = 30^\circ$ 。所以 q' 点的选择范围是以 p 为顶点, 以 γ 为半顶角, 高度为 p 点 z 轴高度的圆锥的底面圆内部。如图 6 所示。

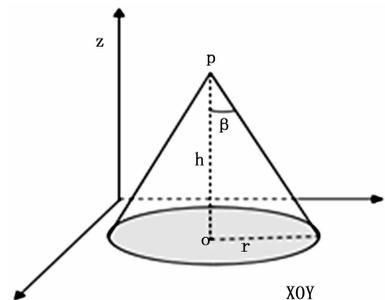


图 6 支撑结构有效生成区域

通过上图可知, 可选择的支撑基点 q' 包含在以 q 为圆心, 以 r 为半径的圆内。设 p 点的坐标为 (x_0, y_0, z_0) , pq' 的距离公式为:

$$l = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (0 - z_0)^2} \quad (6)$$

本文要寻找既不与模型相交, 同时支撑结构长度 pq' 要尽可能小, 即:

$$\min l = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (0-z_0)^2}$$

为了取值方便, x 、 y 需要满足的范围分别为:

$$x_0 - r \leq x \leq x_0 + r$$

$$y_0 - r \leq y \leq y_0 + r$$

$$d(x, y) = \sqrt{(x-x_0)^2 + (y-y_0)^2} \leq r$$

$$f(x, y) = 0$$

其中: $d(x, y)$ 函数表示重新选择的支撑基点 q' 的坐标位置, 当该点圆心距离小于等于圆的半径 r 时, 说明该点坐标位于圆内, 符合要求, $f(x, y)$ 函数表示由 q' 点发出的沿 pq' 方向的射线是否与模型存在交点, 如果存在交点函数值为 1, 不存在交点函数值为 0。

所以针对该算法的数学模型如式 (7) 所示:

$$\min l = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (0-z_0)^2}$$

$$s. t. \begin{cases} x_0 - r \leq x \leq x_0 + r \\ y_0 - r \leq y \leq y_0 + r \\ d(x, y) \leq r \\ f(x, y) = 0 \end{cases} \quad (7)$$

最后针对该数学模型求出相应的解, 即是新的基点 q' 的坐标。

4.2 支撑结构优化

由于支撑结构并不属于模型中的一部分, 是为了保证模型能够成功打印人为添加的部分。因此在模型打印完成后, 支撑结构需要去除。为了提高模型表面质量, 同时保证支撑部分比较容易去除, 最有效的方法就是减少支撑与模型的接触面积。因此本文对与模型直接接触的支撑结构顶部做削尖处理。其次, 支撑结构本身的稳定性与其长度成反比。陈岩^[5]等总结出一个支撑结构宽度与长度的关系公式。

$$w = 1 + l/30 \quad (8)$$

其中: w 为支撑结构宽度 (单位: mm), l 为支撑结构长度 (单位: mm)。

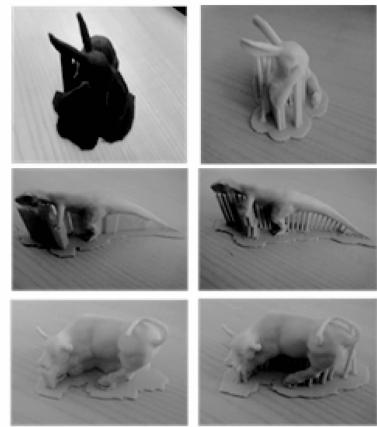
最后增大支撑结构与平台的接触面积, 形成一个底座, 因为接触面积越大越稳定, 不容易发生翘边等现象。

5 实验结果

本文对不同模型通过本算法生成支撑结构, 并与平常使用的 Cura 软件生成的支撑结构进行对比。所有模型均是在 SmartMaker 打印机上采用 PLA 塑料打印, 打印机参数设置完全一样情况下进行打印。我们将本文算法生成的支撑结构与 Cura 软件自带的支撑算法进行对比, 图 7 展示了本文算法生成的支撑结构与 Cura 软件生成的支撑结构实际打印结果。

图中 (a) 列为 Cura 软件生成的支撑情况, (b) 列为本文算法生成的支撑情况。本文主要材料消耗、稳定性、易剥离性 3 个方面比较两个算法的效果。

通过上表实验结果可以看出本文算法在消耗材料方面相比 Cura 软件平均减少 15%, 而在稳定性方面, 两者都很



(a) (b)

图 7 模型打印效果图

表 1 本文算法与 Cura 算法对比表

模型	Cura 软件			本文算法		
	消耗材料	稳定性	易剥离性	消耗材料	稳定性	易剥离性
兔子	16 g	高	差	14 g	高	好
恐龙	12 g	高	差	9 g	高	好
水牛	13 g	高	差	11 g	高	好

稳定, 可以有效的形成对模型的支撑。在易剥离性方面, 由于本文算法对支撑结构进行了削尖处理, 有效的减少了支撑结构与模型的接触面积, 并且支撑结构尽量避免在模型表面生成, 因此支撑结构只有削尖的一端与模型接触, 所以本文算法在较容易去除支撑的同时, 提高了模型表面的质量。Cura 软件生成的支撑结构与模型接触面积非常大, 很难去除, 并且对模型表面质量损害比较严重。综合对比, 本文算法相比 Cura 软件生成的支撑结构, 有用料少, 稳定性高, 易剥离, 对模型质量损害小的优点。

6 结束语

本文通过哈希函数建立了 STL 文件的拓扑结构, 通过三角面片法向量与打印正方向夹角与临界角大小比较获得待支撑三角面片, 提出基于边长自适应法和射线与三角面片相交法得到待支撑点集, 并改进支撑结构, 使支撑结构与模型接触面积减小、不与模型相交, 提高了模型表面质量。

本文算法生成的支撑结构虽然具有稳定性好、易去除、用料少等优点, 但是也存在一些待改进的地方, 比如当支撑结构比较多时, 会增加打印头空移时间, 会相应的增加打印时间, 下一步将针对以上问题进行改进优化。

参考文献:

[1] 李 峰. FDM 支撑原理及支撑工艺研究 [D]. 成都: 西南交通大学, 2017.
 [2] 龚志海. 熔丝沉积成型工艺支撑自动生成技术研究 [D]. 武

汉：华中科技大学，2006.

[3] 王彦云, 陈 鸿, 谢明师, 等. FDM 快速成型支撑结构自动生成算法的研究 [J]. 计算机技术与应用, 2015, 41 (8): 146 - 148.

[4] Wang W, Wang T Y, Yang Z, et al. Cost - effective printing of 3D objects with skin - frame structures [J]. ACM Trans Graph, 2013, 32; 2504 - 2507.

[5] 陈 岩, 王士玮, 杨周旺, 等. FDM 三维打印的支撑结构的设计算法 [J]. 中国科学: 信息科学, 2015, 45 (2): 259 - 269.

[6] Zhang X, Xia Y, Wang J, et al. Media axis tree—an internal supporting structure for 3D Printing [J]. Computer Aided Geometric Design, 2015, 35/36; 148 - 162.

[7] 宋国华, 敬石开, 许文婷, 等. 面向熔融沉积成型的树状支撑结构生成设计方法 [J]. 计算机集成制造系统, 2016, 22 (3): 583 - 588.

[8] 王彦云, 陈 鸿, 谢明师, 等. 基于哈希表的 STL 格式文件拓扑重建的算法 [J]. 现代制造工程, 2015, 12; 62 - 64.

[9] 王增波. STL 格式文件的快速拓扑重建算法 [J]. 计算机应用, 2014, 34 (9): 2720 - 2724.

[10] 魏 群. 基于 STL 模型的 RP 支撑自动生成系统研究 [D]. 西安: 西安交通大学, 2003.

[11] Tomas Moller, Ben Trumbore. Fast, Minimum Storage Ray - Triangle Intersection [J]. Journal of Graphics Tools, 1997, 2 (1): 21 - 28.

(上接第 276 页)

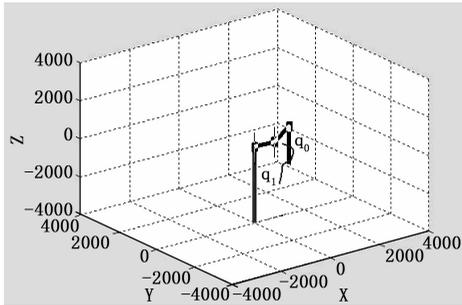


图 7 Scara 机械手 Matlab 模型及路径规划仿真

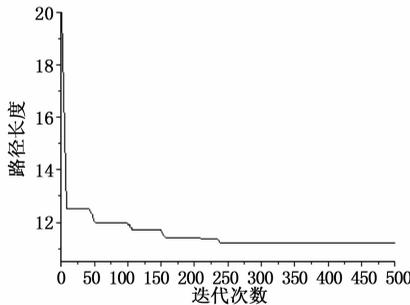


图 8 本文遗传算法迭代次数与路径规划长度的变化关系

前取得较大进展的人工势场算法只适用于局部路径最优解, 本文没有与其进行比较。

表 1 本文遗传算法与其它三维路径规划算法的性能比较

	最短路径长度	搜索时间/ms	最优路径比例/%
本文遗传算法	11.2	328	95
传统遗传算法	15.7	672	65
A * 算法	20.9	341	48

4 结束语

本文在分析和建立了机械手的三维操作空间模型的基础上, 提出了基于三维操作空间路径规划的改进遗传算法, 保证了机械手臂在避开障碍物的基础上, 从起始点到目标点运动路径最短, 并与传统路径规划算法进行了性能比较,

在最优路径比例及搜索时间等性能上占据这相对优势。弥补了人工势场局部最优解, A * 算法避障路径规划不合理, 传统遗传算法搜索效率低, 机械臂与障碍物发生干涉等问题。同时本文提出的将实数编码与三维坐标编码相结合的编码方式提高了搜索效率, 简化了计算过程, 同时使得路径编码更加直观。

参考文献:

[1] Wu X J, Tang J, Li Q, et al. Development of a configuration space motion planner for robot in dynamic environment [J]. Robotics and Computer Integrated Manufacturing, 2007, 25 (1).

[2] Carsten J, Ferguson D, Stentz A. 3D field D * : Improved path planning and replanning in three dimensions [A]. IEEE / RSJ International Conference on Intelligent Robots and Systems [C]. Piscataway, NJ, USA: IEEE, 2006; 3381 - 3386.

[3] Khatib O. Real time obstacle avoidance for manipulators and mobile robots [J]. International Journal of Robotics Research, 1986, 5 (1): 90 - 98.

[4] 贾庆轩, 陈 钢, 孙汉旭, 等. 基于 A * 算法的空间机械臂避障路径规划 [J]. 机械工程学报, 2010, 46 (13): 109 - 115.

[5] 伊华伟. 基于改进蚁群算法的机械手三维操作路径规划 [J]. 计算机应用与软件, 2014, 31 (4): 302 - 304.

[6] 彭礼辉. 机械臂运动学与路径规划研究 [D]. 株洲: 湖南工业大学, 2012.

[7] 徐美清, 孙晨亮. 基于栅格地图的遗传算法路径规划 [J]. 科技信息, 2011 (31): 76 - 77.

[8] 田 欣, 刘广瑞, 周文博, 郭珂甫. 基于改进自适应遗传算法的机器人路径规划研究 [J]. 机床与液压, 2016, 44 (17): 24 - 28.

[9] 张 毅, 代恩灿, 罗 元. 基于改进遗传算法的移动机器人路径规划 [J]. 计算机测量与控制, 2016, 24 (1): 313 - 316.

[10] 杜宗宗. 基于遗传算法的移动机器人路径规划研究 [D]. 无锡: 江南大学, 2009.