

DOI:10.11918/j.issn.0367-6234.201710120

一种求解控制系统可达集的网格投影法

邵立珍¹, 张扬帆¹, 胡广大²

(1. 北京科技大学 自动化学院, 北京 100083; 2. 上海大学 理学院, 上海 200444)

摘要: 控制系统在某一时刻的前向可达集是指从初始状态出发, 在该时刻能够达到的状态的集合. 为了求解非线性控制系统的前向可达集, 首先通过常微分方程数值方法对连续控制系统进行离散, 转化为离散系统; 其次由于连续控制系统的可达集可用其相应的离散系统的可达集来近似, 针对离散控制系统, 提出了基于最优化技术的网格投影法近似描述可达集, 该方法布置了均匀分布的网格点, 并将网格点向可达集边界投影, 每个投影问题都对应一个最优化问题, 通过求解这些优化问题得到可达集的近似描述; 进一步地, 理论分析证明了该方法布置的网格点间隔越小得到的可达集近似误差越小; 最后通过数值仿真验证了该方法的有效性, 并将其与文献中已有的 DFOG (Distance fields on grids) 方法进行对比. 研究表明: 网格投影法可以有效地处理具有非凸可达集的控制系统; 相较于 DFOG 方法, 该方法能够得到分布均匀的边界点, 且求解的优化问题的数量少, 计算时间短.

关键词: 控制系统; 非凸可达集; 数值方法; 网格投影; 优化

中图分类号: TP273

文献标志码: A

文章编号: 0367-6234(2018)10-0088-07

A grid projection method for reachable sets of control systems

SHAO Lizhen¹, ZHANG Yangfan¹, HU Guangda²

(1. School of Automation, University of Science and Technology Beijing, Beijing 100083, China;

2. School of Science, Shanghai University, Shanghai 200444, China)

Abstract: The forward reachable set of a control system at a certain time is the set of states that can be reached at this time from the initial state. In order to solve the forward reachable set of a nonlinear control system, numerical methods for ordinary differential equations was used to discretize the continuous control system. Since the reachable set of a continuous system can be approximated by its discrete counterpart, a grid projection method based on optimization technology was proposed to approximate the reachable set. In this method, uniformly distributed grid points were placed and projected to the boundary of reachable set for a discrete control; each projection problem corresponded to an optimization problem; by solving these problems the approximated description of the reachable set was obtained. Theoretical analysis proves that the smaller the grid interval, the smaller the approximation error. Finally, numerical results verified the effectiveness of the proposed method, which were compared with those of the DFOG (distance fields on grids) method in literature. The research shows that the grid projection method can effectively handle control systems with non-convex reachable sets. Compared with the DFOG method, the method gets uniform distributed boundary points, solves less optimization problems and uses less computation time.

Keywords: control system; non-convex reachable set; numerical method; grid projection; optimization

控制系统的可达集有前向可达集和后向可达集之分, 其中前向可达集为系统从初始状态出发, 到终止时刻的所有状态的集合; 后向可达集为能够到达终止时刻目标状态的所有状态的集合^[1]. 本文仅研究前向可达集, 文中所提到的可达集均指前向可达集. 确定可达集的边界对于分析系统的动态性能、设

计系统控制器具有重要的实际意义. 对于非线性控制理论和动态系统中不定参数的定量研究而言, 可达集的数值计算是关键的问题之一. 可达集的研究可广泛地应用在众多的学科领域. 例如, 在飞行器控制领域, 可以对飞行器轨道的可达集进行分析, 实现良好的变轨过程^[2-3]、确定飞行器着陆区域^[4]、为驾驶员完成标准机动动作提供决策支持^[5]; 通过对导弹运动轨迹的可达集研究, 可以帮助确定制导导弹的目标命中区域^[6]; 通过研究机器人运动轨迹的可达集, 可以使其选取最短的行动路径并躲避障碍物^[7].

研究表明, 线性控制系统的可达集是凸集^[8],

收稿日期: 2017-10-24

基金项目: 国家自然科学基金 (11371053); 北京市自然科学基金 (4152034); 中央高校基本科研业务费专项资金 (FRF-BD-16-005A)

作者简介: 邵立珍 (1976—), 女, 副教授;
胡广大 (1962—), 男, 教授, 博士生导师

通信作者: 邵立珍, lshao@ustb.edu.cn

而非线性系统的可达集则可能是凸的或非凸的两种. 目前针对线性系统, 已经提出了很多种近似可达集的理论及算法. 文献[9-10]通过构造若干接触可达集边界的外部或内部椭圆, 来得到可达集的近似估计; 文献[11-12]通过将可达集等价描述为有限个最优控制问题, 通过直接求解这些最优控制问题来得到可达集的近似; 文献[13]将可达集描述为一个 Hamilton-Jacobi 偏微分方程, 并利用水平集方法近似可达集. 但对于非线性系统, 尤其是具有非凸可达集的系统, 计算其可达集目前受到了以下几方面的阻碍: 可达集几何复杂度会随着时间间隔增加而增加, 非凸性所带来的计算问题以及非线性系统造成的计算不稳定性. 为了求解非线性控制系统的可达集, 文献[14]分析了非线性离散系统的可达集为凸的充分和必要条件; 文献[15]在二维可达集内部布置了若干点, 并迭代更新, 使这些点的分段线性轮廓线所围成的区域最大, 从而近似可达集; 文献[16]指出连续系统的可达集可用其相应的离散系统的可达集来近似, 近似精度取决于离散化方法和离散化步长; 针对离散非线性系统, 文献[17]提出了 DFOG (distance fields on grids) 算法, 通过在包含可达集的区域内部布置均匀分布的网格点, 除去以外部网格点为圆心、该点到可达集的最短距离为半径的开球集, 剩余集合即为可达集的近似集合. 然而上述方法在可达集近似精度和计算效率方面都有待进一步的提高.

本文在已有的研究中, 提出了基于凸优化技术的外部投影法来求解线性离散控制系统的可达集, 但该方法不适用于具有非凸可达集的非线性控制系统, 且没有给出近似结果的误差分析. 本文将继续研究求解离散控制系统的可达集的方法, 重点针对具有非凸可达集的非线性控制系统. 文中首先对非线性系统及其可达集进行描述, 然后介绍离散化后的系统及其相应的离散可达集, 进一步地提出了近似离散系统可达集的网格投影法, 并给出了近似误差的理论分析, 最后将该方法与文献[17]中的 DFOG 方法进行数值仿真对比, 给出了仿真和分析结果.

1 控制系统及其可达集

考虑非线性动态系统为

$$\begin{cases} \dot{\mathbf{x}}(t) = f(t, \mathbf{x}(t), \mathbf{u}(t)), \\ \mathbf{x}(t_0) = \mathbf{x}_0, \\ \mathbf{u}(t) \in U. \end{cases} \quad (1)$$

式中: $\mathbf{x} \in \mathbf{R}^n$ 为系统的状态向量; $\mathbf{x}_0 \in \mathbf{R}^n$ 为系统的初始状态; $U \subset \mathbf{R}^m$ 是一个非空凸紧集; $t \in [t_0, t_f]$,

其中 t_0 、 t_f 分别为动态系统的初始时刻和终点时刻, 且 $t_0 < t_f$.

定义 1 系统(1)从初始条件 $(t_0, \mathbf{x}_0)$ 出发, 在 t_f 时刻的前向可达集可定义如下

$$\mathfrak{R}(t_f, t_0, \mathbf{x}_0) := \{\mathbf{y} \in \mathbf{R}^n : \exists \mathbf{u}(\cdot) \text{ s.t. } \mathbf{x}(t_f) \in \mathbf{y}\}. \quad (2)$$

对于连续系统, 可以用数值方法如欧拉法、龙格-库塔法、线性多步法等^[18]对其离散化得到离散系统. 以欧拉法为例, 在 $[t_0, t_f]$ 时间内, 取等间隔时间 $t_i = t_0 + ih$, $h = (t_f - t_0)/N$, $i = 0, \dots, N$. 假定控制函数 $\mathbf{u}(t)$ 在 $[t_i, t_{i+1}]$ 间隔内恒定为 $\tilde{\mathbf{u}}(t_i)$, 则采用欧拉法对系统(1)离散化, 可得离散系统为

$$\begin{cases} \tilde{\mathbf{x}}(t_{i+1}) = \tilde{\mathbf{x}}(t_i) + hf(t_i, \tilde{\mathbf{x}}(t_i), \tilde{\mathbf{u}}(t_i)), \\ \tilde{\mathbf{u}}(t_i) \in U, \\ \tilde{\mathbf{x}}(t_0) = \mathbf{x}_0, \end{cases} \quad (3)$$

式中 $i = 0, \dots, N-1$.

定义 2 对于离散系统(3), 从初始条件 $(t_0, \mathbf{x}_0)$ 出发, 在 t_N 时刻的可达集可定义如下

$$\mathfrak{R}(t_N, t_0, \mathbf{x}_0) := \{\mathbf{y} \in \mathbf{R}^n : \exists \tilde{\mathbf{u}}(\cdot) \text{ s.t. } \tilde{\mathbf{x}}(t_N) \in \mathbf{y}\}. \quad (4)$$

用离散系统的可达集(4)对连续系统的可达集(2)进行近似, 其近似精度可以利用豪斯多夫距离 (Hausdorff distance) 来衡量.

定义 3 设集合 $C, D \subset \mathbf{R}^n$ 为两个非空紧集, 则 C 与 D 之间的豪斯多夫距离为

$$d_H(C, D) := \max_{c \in C} \min_{d \in D} \|c - d\|_2,$$

式中 $d(C, D) = \max_{c \in C} \min_{d \in D} \|c - d\|_2$.

文献[16]指出, 连续系统的可达集 $\mathfrak{R}(t_f, t_0, \mathbf{x}_0)$ 可以用其离散系统的可达集 $\mathfrak{R}(t_N, t_0, \mathbf{x}_0)$ 来近似, 近似精度取决于离散步长 h .

定理 1 设系统(1)中的 f 满足利普希茨 (Lipschitz) 条件, 取正整数 N 为离散步数, $h = (t_f - t_0)/N$ 为离散步长. 利用欧拉方法对系统(1)进行离散化处理得到系统(3), 则存在一常数 $M > 0$ 使得连续系统可达集 $\mathfrak{R}(t_f, t_0, \mathbf{x}_0)$ 与离散系统可达集 $\mathfrak{R}(t_N, t_0, \mathbf{x}_0)$ 有如下关系:

$$d_H(\mathfrak{R}(t_f, t_0, \mathbf{x}_0), \mathfrak{R}(t_N, t_0, \mathbf{x}_0)) \leq Mh.$$

本文用离散控制系统可达集近似连续控制系统可达集, 重点研究离散控制系统可达集的数值求解方法. 根据定义 2, 离散控制系统的可达集问题可以归结为如下求集合 Y 的问题:

$$\text{Find } Y := \{\mathbf{y} \in \mathbf{R}^n : \mathbf{y} = \tilde{\mathbf{x}}(t_N)\},$$

$$\begin{aligned}
& \text{s. t. } \tilde{\mathbf{x}}(t_0) = \mathbf{x}_0, \\
& \tilde{\mathbf{x}}(t_{i+1}) = \tilde{\mathbf{x}}(t_i) + hf(t_i, \tilde{\mathbf{x}}(t_i), \tilde{\mathbf{u}}(t_i)), \\
& \tilde{\mathbf{u}}(t_i) \in U, \\
& i = 0, \dots, N-1.
\end{aligned} \quad (5)$$

进一步地, 将式(5)中的每个约束都转化为 $g_i(\mathbf{x}) \leq 0$ 的形式, 则上述问题描述为

$$\begin{aligned}
& \text{Find } Y: = \{\mathbf{P}\mathbf{x}; \mathbf{x} \in X\}, \\
& \text{s. t. } X = \{\mathbf{x} \in \mathbf{R}^n; g_i(\mathbf{x}) \leq 0, i = 1, \dots, l\}.
\end{aligned} \quad (6)$$

式中: $\mathbf{x} = (\tilde{\mathbf{u}}(t_0)^T, \dots, \tilde{\mathbf{u}}(t_{N-1})^T, \tilde{\mathbf{x}}(t_0)^T, \dots, \tilde{\mathbf{x}}(t_{N-1})^T, \tilde{\mathbf{x}}(t_N)^T)^T$ 为变量; $\mathbf{P}$ 为一个最后 n 列为单位阵而其他元素为 0 的矩阵.

2 网格投影法

网格投影法是一种求解式(6)中 Y 的方法. 该方法从集合某个内部点出发, 沿着坐标轴的正负方向, 逐渐构造一系列等间距的网格, 并将这些网格点沿着各个坐标轴方向向集合边界进行投影以得到边界点.

2.1 网格点生成与投影点计算

设集合 $Y \subseteq \mathbf{R}^n$ 为一非空紧集, 用 $\text{int}Y$ 表示集合 Y 内部, 而 $\text{bd}Y$ 表示集合 Y 边界. 取任意一点 $\mathbf{a} \in \text{int}Y$. 将点 $\mathbf{a}$ 沿坐标轴 i 的正、反方向向集合 Y 的边界进行投影, 可以分别通过解决以下最优化问题来求得相应的投影边界点:

$$P_{\max}^{i+}(\mathbf{a}) \max_{d_{i+}} \text{ s. t. } \mathbf{a} + d_{i+}\mathbf{e}_i = \mathbf{y}, \mathbf{y} \in Y;$$

$$P_{\max}^{i-}(\mathbf{a}) \max_{d_{i-}} \text{ s. t. } \mathbf{a} - d_{i-}\mathbf{e}_i = \mathbf{y}, \mathbf{y} \in Y.$$

式中 $\mathbf{e}_i$ 为第 i 个元素值为 1, 其余元素均为 0 的 n 维向量. 设 d_{i+}^* 、 d_{i-}^* 分别为求解 $P_{\max}^{i+}(\mathbf{a})$ 、 $P_{\max}^{i-}(\mathbf{a})$ 得到的最优目标函数值, 则从 $\mathbf{a}$ 点出发沿着坐标轴 i 正、负方向向 Y 投影得到的边界点分别为:

$$\mathbf{p}_{i+} = \mathbf{a} + d_{i+}^*\mathbf{e}_i, \mathbf{p}_{i-} = \mathbf{a} - d_{i-}^*\mathbf{e}_i.$$

以 $\mathbf{a}$ 点为起点, 在 $\mathbf{p}_{i+}$ 、 $\mathbf{p}_{i-}$ 之间, 可沿坐标轴 i 的正、反方向布置间隔为 $d_s > 0$ 的均匀分布的参考点. 进一步将这些参考点沿着其它的坐标轴上投影, 生成新的参考点, 所有的参考点形成了网格点.

图 1 展示了投影及网格点的生成过程. 图中二维平面内的闭合实线围成的区域为所求集合 Y , 内部两条带箭头实线的交点为任意选取的初始点 $\mathbf{a}$, 箭头分别表示 4 个投影方向, 边界线上有 4 个投影边界点 $\mathbf{p}_{1-}$ 、 $\mathbf{p}_{1+}$ 、 $\mathbf{p}_{2-}$ 、 $\mathbf{p}_{2+}$, 线上的每个点表示由 $\mathbf{a}$ 生成的等间隔的网格点.

2.2 算法描述

网格投影法的算法流程如图 2 所示.

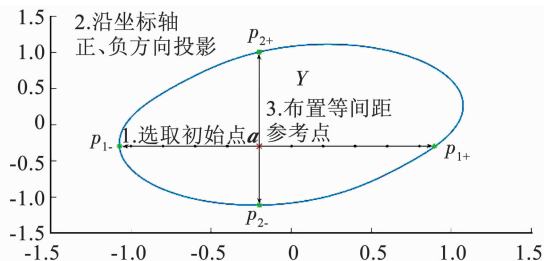


图 1 投影及网格点生成

Fig. 1 Projection and generation of grid points

算法 1 网格投影法

输入: 网格点间距 $d_s > 0$; 初始点 $\mathbf{a} \in Y$.
 输出: Y 的边界点集 B .
 1: $G \leftarrow \{\mathbf{a}\}, B \leftarrow \emptyset, D_i \leftarrow \emptyset, i = 1, 2, \dots, n$.
 % G 为未投影的参考点集合; B 为边界点集; D_i 为已沿第 i 个坐标轴完成投影的点集.
 2: while $G \neq \emptyset$ do
 3: 取一点 $\mathbf{g} \in G$.
 4: for $i = 1$ to n do
 5: if $\mathbf{g} \notin D_i$ then
 6: 分别求解 $P_{\max}^{i+}(\mathbf{g}), P_{\max}^{i-}(\mathbf{g})$, 得投影点 $\mathbf{p}_{i+}, \mathbf{p}_{i-}$.
 7: $P \leftarrow \text{SetRefPts}(\mathbf{g}, i, \mathbf{p}_{i+}, \mathbf{p}_{i-}, d_s)$.
 % 函数 SetRefPts 的功能为: 以 $\mathbf{g}$ 为出发点, 沿坐标轴 i , 在 $\mathbf{p}_{i+}, \mathbf{p}_{i-}$ 之间布置间距为 d_s 的参考点, 并返回参考点集.
 8: $G \leftarrow G \cup P, D_i \leftarrow D_i \cup P$.
 9: $B \leftarrow B \cup \{\mathbf{p}_{i+}, \mathbf{p}_{i-}\}$.
 10: end if
 11: end for
 12: $G \leftarrow G \setminus \{\mathbf{g}\}$.
 13: end while
 14: return B

图 2 网格投影法的算法流程

Fig. 2 Algorithm flow of the grid projection method

算法 1 的输出结果 B 为 Y 的边界点集, 将 B 中的相邻点依次连接, 围成的内部区域记作 $\text{cl}B$, $\text{cl}B$ 即为 Y 的近似集. 算法 1 可以有效地解决具有凸可达集的控制系统问题, 如线性系统的可达集. 但是对于非凸可达集, 如图 3 所示, 二维平面内的实现围成的闭合区域为集合 Y , $\mathbf{a}$ 为初始点, 闭合区域的边界上 $\mathbf{p}_{1-}, \mathbf{p}_{1+}$ 为投影点, 两投影点间的圆点为参考点, 出现了参考点位于 Y 外部的情况. 这是由于求解 $P_{\max}^{i+}(\mathbf{a})$ 、 $P_{\max}^{i-}(\mathbf{a})$, 内部点的投影会跨过可达集非凸部分的边界线, 而得到最远投影距离的投影点 $\mathbf{p}_{1+}$, 致使生成的内部网格点实际落在了集合 Y 的外部, 如图中标注的 6 个外部参考点. 若对外部的参考点继续按照算法 1 的流程生成网格点, 将无法求得正确的边界点, 且导致算法无法停止.

鉴于实际问题中, 大多数可达集 Y 具有如下特点: 对于任意一点 $\mathbf{y} \notin Y$, 将该点沿每个坐标轴的正、负方向 Y 做投影, 至少有一个方向的投影不存在. 因此对于这种类型的非凸可达集, 可通过在网格

投影法中添加额外的判断条件以区分外部投影点. 对某个参考点 a' 沿所有坐标轴求解 $P_{\max}^{i+}(a')$ 、 $P_{\max}^{i-}(a')$, 若出现沿某个方向的解不存在时, 则认为 $a' \notin Y$. 这时, 可计算如下的优化问题得到 a' 沿坐标轴在 Y 上的投影, 进而修正投影边界点.

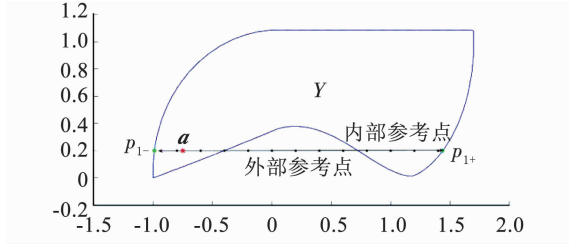


图 3 错误生成外部网格点

Fig. 3 Erroneous generation of external grid points

$$P_{\min}^{i+}(a') \min d_{i+} \text{ s. t. } a' + d_{i+} e_i = y, y \in Y;$$

$$P_{\min}^{i-}(a') \min d_{i-} \text{ s. t. } a' - d_{i-} e_i = y, y \in Y.$$

具体修正过程如下: 当 a' 沿坐标轴 i 投影时, 若 $P_{\max}^{i+}(a')$ 的解存在而 $P_{\max}^{i-}(a')$ 的解不存在, 则求解 $P_{\min}^{i+}(a')$ 得到投影点 $p'_{i+} = a' + d_{i+}^* e_i$, 并在 p_{i+} 、 p'_{i+} 之间布置网格点; 同理, 若 $P_{\max}^{i+}(a')$ 的解不存在而 $P_{\max}^{i-}(a')$ 的解存在, 则求解 $P_{\min}^{i-}(a')$, 得到投影点 $p'_{i-} = a' - d_{i-}^* e_i$, 并在 p_{i-} 、 p'_{i-} 之间布置网格点; 对该点的其他坐标轴方向, 需要求解 $P_{\min}^{i+}(a')$ 、 $P_{\min}^{i-}(a')$ 得到 p'_{i+} 、 p'_{i-} 来补全边界投影点, 且这两个边界投影点之间的网格点均被判断为外部网格点. 修正后的网格投影法算法如图 4 所示.

算法 2 在处理外部参考点时如图 5 所示, 图 5 中二维平面内的实线围成的闭合区域为所求集合 Y , a 为初始点, 图中有 6 个外部网格点位于集合 Y 外部, 它们沿纵坐标轴负方向的投影点不存在, 其中带有箭头实线的交点为计算投影的网格点 g , 闭合区域的边界上 p'_{1+} 、 p'_{1-} 为对 g 求解 $P_{\min}^{1+}(g)$ 、 $P_{\min}^{1-}(g)$ 得到的投影点, p_{2+} 、 p_{2-} 为对 g 求解 $P_{\max}^{2+}(g)$ 、 $P_{\max}^{2-}(g)$ 得到的投影点, 圆点为网格点.

2.3 算法近似误差分析

本文把由算法 2 得到的边界点集 B 相邻点依次连接, 所围成的区域记作 $\text{cl}B$. $\text{cl}B$ 与真实集合 Y 间存在如下关系.

定理 2 对于一个非空闭集 $Y \subseteq \mathbb{R}^n$, 假定选取参考点间隔为 $d_s > 0$, 且 Y 与内部参考点集 $G_{\text{int}} \subset Y$ 满足 $d_H(Y, G_{\text{int}})$, 利用算法 2 近似 Y , 记所得投影边界点集为 B , 则 Y 与 B 满足如下关系:

$$d_H(Y, \text{cl}B) \leq \sqrt{n} d_s.$$

证明 根据定义 3, 有 $d_H(Y, \text{cl}B) = \max\{d(Y, \text{cl}B), d(\text{cl}B, Y)\}$, 其中:

$$d(Y, \text{cl}B) = \max_{y \in Y} d(y, \text{cl}B), \quad (7)$$

$$d(\text{cl}B, Y) = \max_{b \in \text{cl}B} d(b, Y). \quad (8)$$

算法 2 修正的网格投影法

输入: 网格点间距 $d_s > 0$; 起始点 $a \in Y$.

输出: Y 的边界点集 B .

```

1:  $G \leftarrow \{a\}, B \leftarrow \emptyset, D_i \leftarrow \emptyset, i = 1, 2, \dots, n$ .
2:  $G_{\text{ext}} \leftarrow \emptyset$ . %  $G_{\text{ext}}$  表示外部参考点.
3: while  $G \neq \emptyset$  do
4:   取一点  $g \in G$ , 令  $\text{flg} \leftarrow 0$ .
   % 若  $\text{flg} = 0$  表示  $g \in Y$ .
5:   for  $i = 1$  to  $n$  do
6:     if  $g \notin D_i$  then
7:       求解  $P_{\max}^{i+}(g), P_{\max}^{i-}(g)$ , 若解存在, 则分
       别记相应的投影点为  $p_{i+}, p_{i-}$ .
8:       if  $p_{i+}, p_{i-}$  均存在 then
9:          $B \leftarrow B \cup \{p_{i+}, p_{i-}\}$ .
10:         $P \leftarrow \text{SetRefPts}(g, i, p_{i+}, p_{i-}, d_s)$ .
11:      else if  $p_{i+}$  存在而  $p_{i-}$  不存在 then
12:        求解  $P_{\min}^{i+}(g)$  得到  $p'_{i+}$ .
13:         $B \leftarrow B \cup \{p_{i+}, p'_{i+}\}$ .
14:         $P \leftarrow \text{SetRefPts}(g, i, p_{i+}, p'_{i+}, d_s)$ .
15:         $\text{flg} \leftarrow 1$ .
16:      else if  $p_{i-}$  不存在而  $p_{i+}$  存在 then
17:        求解  $P_{\min}^{i-}(g)$  得到  $p'_{i-}$ .
18:         $B \leftarrow B \cup \{p_{i-}, p'_{i-}\}$ .
19:         $P \leftarrow \text{SetRefPts}(g, i, p_{i-}, p'_{i-}, d_s)$ .
20:         $\text{flg} \leftarrow 1$ .
21:      end if
22:       $G \leftarrow G \cup P, D_i \leftarrow D_i \cup P$ .
23:    end if
24:  end for
25:  if  $\text{flg} = 1$  then
26:    for  $i = 1$  to  $n$  do
27:      if  $g \in D_i$  then
28:        求解  $P_{\min}^{i+}(g), P_{\min}^{i-}(g)$  得  $p'_{i+}, p'_{i-}$ .
29:         $B \leftarrow B \cup \{p'_{i+}, p'_{i-}\}$ .
30:        将  $g$  和  $p'_{i+}$  与  $p'_{i-}$  间的参考点并入
         $G_{\text{ext}}$ .
31:      end if
32:    end for
33:  end if
34:   $G \leftarrow G \setminus \{g\}$ .
35: end while
36: 将  $G_{\text{ext}}$  去除重复点.
37: return  $B, G_{\text{ext}}$ 

```

图 4 修正后的网格投影法算法流程

Fig. 4 Algorithm flow of the modified grid projection method

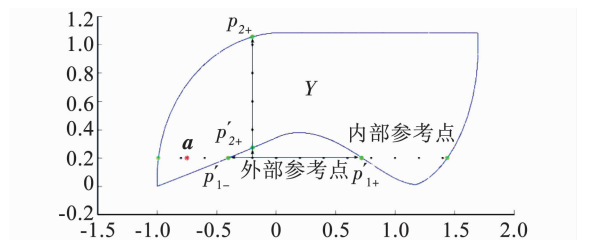


图 5 修正算法处理的外部网格点

Fig. 5 Revised algorithm to deal with external grid points

对于式(7),取任意一点 $y \in Y$,若同时 $y \in \text{cl}B$, 则 $d(y, \text{cl}B) = 0$; 若 $y \notin \text{cl}B$, 由于离散点集 $G_{\text{int}} \subset \text{cl}B$ 和定理 2 的假定条件, 有 $d(y, \text{cl}B) \leq d(y, G_{\text{int}}) = \max_{y \in Y} \min_{g \in G_{\text{int}}} d(y, g) \leq \sqrt{n}d_s$, 式(7)得证.

对于式(8),取任意一点 $b \in \text{cl}B$, 若 $b \in Y$, 则 $d(b, Y) = 0$; 若 $b \notin Y$, 则有 $d(b, Y) \leq d(b, G_{\text{int}}) \leq \max\{d(F^1, G_{\text{int}}), \dots, d(F^k, G_{\text{int}})\}$, 其中 $F^i (i = 1, \dots, k)$ 为 $\text{cl}B$ 的外围边界面, 且有 $\cup_{i=1}^k F^i = \text{bd}\{\text{cl}B\}$. n 维空间内每个 F^k 可由相邻的投影边界点 $b_i^k (i = 1, \dots, n)$ 的凸包来表示. 对于每个 F^k 及其对应的 b_i^k 而言, 假定 b_1^k, b_2^k 为相邻的投影点, 与它们对应的最近的网格点为 g_1^k, g_2^k , 如果生成 b_1^k, b_2^k 的两个投影方向相同, 则 $d(g_1^k, g_2^k) = d_s$, 否则 g_1^k, g_2^k 为同一点. 若对每个 g_i^k 以其自身为起点沿 $g_i^k b_i^k$ 方向生成新的参考点 $g_i^{k'} \notin Y$, 有 $d(g_i^k, g_i^{k'}) = d_s$. 将 $g_i^k, g_i^{k'}$ 作为顶点可以构建一个超立方体, 该立方体的最长对角线距离为 $\sqrt{n}d_s$. 由立方体的构建方式可知 b_i^k 位于该立方体的边界面上, F^k 位于该立方体内. 因此有 $d(F^k, G_{\text{int}}) = \max_{b \in F^k} d(b, g_i^k) \leq \sqrt{n}d_s$, 则式(8)得证, 因此定理 2 成立.

定理 2 表明了网格投影法的近似误差与 d_s 有关, d_s 越小, 布置的网格点密度越大, 得到的投影边界点数量越多, 近似误差越小.

3 数值仿真

本文将网格投影法和文献[17]中的 DFOG 方法进行对比. 实验采用 MATLAB 2016a 仿真了两种算法, 选取了同样的可达集内部网格点, 利用求解器 IPOPT^[19] 对投影时的非线性优化问题进行求解. 将上述程序用于求解以下 3 个控制系统问题. 对于所有的例子, 均采用带有分段恒定控制量的欧拉法来离散化.

首先考虑两个二维系统的例子.

例 1 考虑如下一个二维双线性系统:

$$\begin{cases} \dot{x}_1(t) = \pi x_2(t), \\ \dot{x}_2(t) = -\pi u(t)x_1(t), \\ x_1(0) = -1, \\ x_2(0) = 0, \\ u(t) \in U = [0, 1], \end{cases}$$

式中初始时刻 $t_0 = 0$, 终点时刻 $t_f = 1$, 取欧拉法的步长 $h = 1/30$.

图 6(a)、(b) 分别显示了网格投影法和 DFOG 法的结果. 图中圆点为网格点, 星点为求得的可达集

边界点, 图 6(b) 中的大圆 (不含边界线) 展示了 DFOG 方法计算得到的开球集, DFOG 法得到的可达集为圆圈集围成的内部区域. 可以看出, 两种方法的得到的可达集类似, 但网格投影法求得的边界点疏密更为均匀.

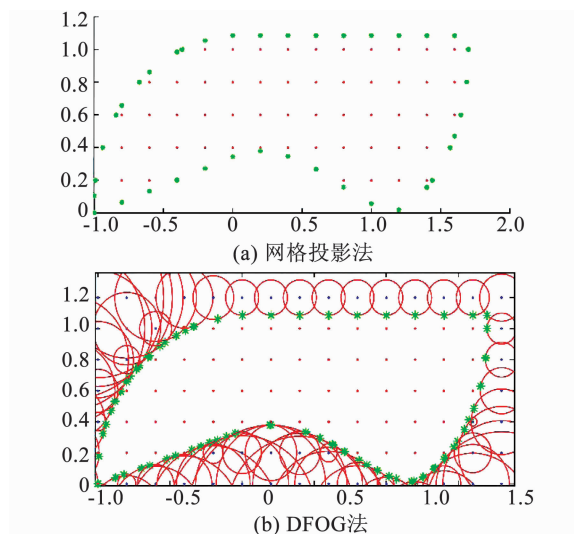


图 6 例 1 的近似可达集

Fig. 6 Approximated reachable set for example 1

例 2 考虑如下一个二维非线性系统:

$$\begin{cases} \dot{x}_1(t) = x_2^2(t) + x_1(t) + u(t), \\ \dot{x}_2(t) = x_2(t) + 5\sin(x_1(t)) + u(t), \\ x_1(0) = 0, \\ x_2(0) = 0, \\ |u(t)| \leq 1, \end{cases}$$

式中: 初始时刻 $t_0 = 0$, 终点时刻 $t_f = 1$, 取欧拉法的步长 $h = 0.01$.

图 7(a)、(b) 分别显示了网格投影法和 DFOG 法的结果. 由于本系统的可达集在坐标轴上跨度较大, DFOG 算法在可达集外部布置的网格点较多; 此外也有部分网格点的优化问题没有求得全局最优值, 使得相应的开球集半径过大, 导致所求可达集比实际可达集小. 而网格投影法并不需要求解如此多的优化问题, 耗时较短.

例 3 考虑如下的一个三维线性时不变系统. 系统矩阵和控制矩阵分别为:

$$A = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -2 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

式中: 控制约束为 $U = \{u \in \mathbf{R}^3 : \|u\|_2 \leq 1\}$, 初始状态 $x_0 = (0, 0, 0)^T$, 初始时刻 $t_0 = 0$, 终点时刻 $t_f = 1$, 取欧拉步长 $h = 0.02$.

图 8(a)、(b) 分别显示了网格投影法和 DFOG

法求得的边界点凸包. 图中可以看出 DFOG 法左侧某些边界点落在了网格投影法得到的可达集内部, 表明了 DFOG 法中某些外部网格点的优化问题没有求得全局最优值, 使计算得到的边界点位于真实可达集的内部.

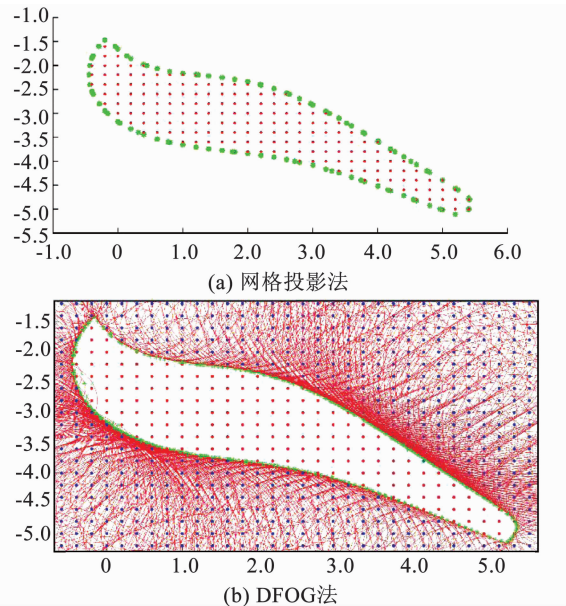


图 7 例 2 的近似可达集

Fig. 7 Approximated reachable set for example 2

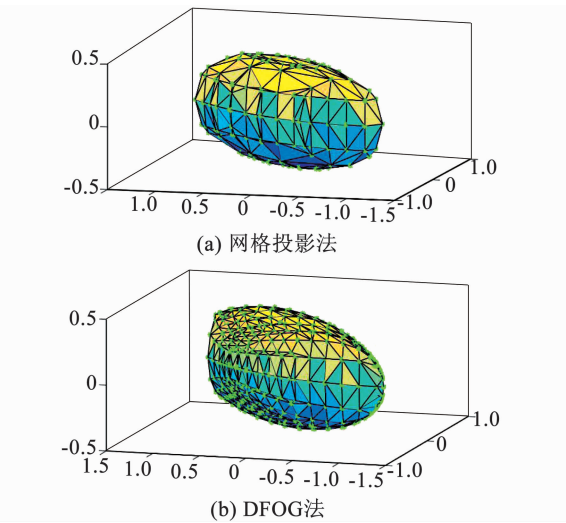


图 8 例 3 的近似可达集

Fig. 8 Approximated reachable set for example 3

表 1 进一步总结了用网格投影法和 DFOG 法计算例 1~3 得到的结果, 表中列出两种方法的计算时间, 计算非线性优化问题的个数 (NPs), 最终得到的边界点的个数 (BPs). 如对于问题例 1, 表 1 记录了网格投影法的耗时/s、NPs、BPs 分别为 22.81、54.00、46.00; 而 DFOG 法的相应数据分别为 28.97、128、72. 这是由于 DFOG 法需要布置大量外部网格点, 且每个网格点都要求解一个优化问题; 而网格投影法中, 仅对于非凸集才需要布置外部网格点, 且

沿每个投影方向只需要计算一次投影. 可以说明, 在同样内部网格点的情况下, 相较于 DFOG 方法, 网格投影法所需的计算时间短, 求解优化问题的个数少. 进一步地, 仿真结果图也显示了网格投影法得到的可达集边界点分布更加均匀. 与 DFOG 法相比, 得到的可达集描述更接近真实的可达集.

表 1 数值仿真结果

Tab. 1 Numerical simulation results

例子	算法	耗时/s	NPs	BPs
1	网格投影法	22.81	54	46
	DFOG 法	28.97	128	72
2	网格投影法	142.79	124	108
	DFOG 法	841.50	936	760
3	网格投影法	256.13	218	194
	DFOG 法	868.70	882	791

4 结 论

1) 网格投影法在可达集内部布置均匀分布的网格点, 将网格点向可达集边界投影, 每个投影点都是可达集的边界点, 这些投影点围城的区域即为可达集的近似描述.

2) 网格投影法的近似误差与布置网格点的间隔相关, 网格点间隔越小, 则近似误差越小, 需要求解优化问题个数越多.

3) 相较于 DFOG 算法, 网格投影法具有近似误差小、边界点分布均匀、速度快的优点; 当 DFOG 方法出现优化问题无法求得全局最优时, 会使求得的可达集比真实的可达集小, 实验结果显示本方法可靠性较高.

4) 提出了网格投影法直接近似离散系统的可达集. 该方法简单, 易于实施, 可以有效地解决可达集为非凸的情况.

参考文献

[1] TRINH H, NAM P T, PATHIRANA P N, et al. On backwards and forwards reachable sets bounding for perturbed time-delay systems [J]. Applied Mathematics and Computation, 2015, 269: 664. DOI: 10.1016/j.amc.2015.07.116

[2] 雪丹, 李俊峰, 宝音贺西. 平面脉冲作用下卫星轨道的可达范围研究[J]. 宇航学报, 2009, 30(1): 88. DOI: 10.3873/j.issn.1000-1328.2009.00.015

XUE Dan, LI Junfeng, BAODYIN Hexi. Study on reachable domain for satellite trajectory with coplanar impulse applied[J]. Journal of Astronautics, 2009, 30(1): 88. DOI: 10.3873/j.issn.1000-1328.2009.00.015

[3] LI Xuehua, HE Xingsuo, ZHONG Qinfang, et al. Reachable domain for satellite with two kinds of thrust[J]. Acta Astronautica, 2011, 68(11/12): 1860. DOI: 10.1016/j.actaastro.2011.01.004

- [4] 解永锋, 唐硕. 亚轨道飞行器再入可达域快速计算方法[J]. 飞行力学, 2011, 29(4): 72
XIE Yongfeng, TANG Shuo. Rapid calculation of entry footprint of suborbital launch vehicles[J]. Flight Dynamics, 2011, 29(4): 72
- [5] 刘琰, 杜光勋, 全权, 等. 基于 Hamilton-Jacobi 方程的飞行器机动动作可达集分析[J]. 自动化学报, 2016, 42(3): 347. DOI: 10.16383/j.aas.2016.c140888
LIU Ying, DU Guangxun, QUAN Quan, et al. Reachability calculation for aircraft maneuver using Hamilton-Jacobi function[J]. Acta Automatica Sinica, 2016, 42(3): 347. DOI: 10.16383/j.aas.2016.c140888
- [6] 寇英信, 陈磊, 李战武, 等. 基于命中概率的制导炸弹可达域定量缩减方法[J]. 电光与控制, 2013, 20(7): 11. DOI: 10.3969/j.issn.1671-637X.2013.07.003
KOU Yingxin, CHEN Lei, LI Zhanwu, et al. Guided bomb accessible region quantitative reduction method based on hit probability[J]. Electronics Optics and Control, 2013, 20(7): 11. DOI: 10.3969/j.issn.1671-637X.2013.07.003
- [7] GERDTS M, HENRION R, HOMBERG D, et al. Path planning and collision avoidance for robots[J]. Numerical Algebra Control and Optimization, 2012, 2(3): 437. DOI: 10.3934/naco.2012.2.437
- [8] 邵立珍, 赵方园, 胡广大. 一种求解线性控制系统可达集的数值方法[J]. 控制与决策, 2017, 32(3): 541. DOI: 10.13195/j.kzyjc.2016.0212.
SHAO Lizhen, ZHAO Fangyuan, HU Guangda. A Numerical method for reachable sets of linear control systems[J]. Control and Decision, 2016, 32(2): 541. DOI: 10.13195/j.kzyjc.2016.0212
- [9] KURZHANSKIY A, VARAIYA P. Ellipsoidal techniques for reachability analysis of discrete-time linear systems[J]. IEEE Transactions on Automatic Control, 2007, 52(1): 26. DOI: 10.1109/TAC.2006.887900
- [10] KOSTOUSOVA E K. External polyhedral estimates for reachable sets of linear discrete-time systems with integral bounds on controls[J]. International Journal of Pure and Applied Mathematics, 2009, 50(2): 187
- [11] VARAIYA P. Reach set computation using optimal control[M]. INAN M K, KURSHAN R P. Verification of digital and hybrid systems. Berlin, Heidelberg: Springer, 2000: 323. DOI: 10.1007/978-3-642-59615-5_15
- [12] BAIER R, BÜSKENS C, CHAHMA I A, et al. Approximation of reachable sets by direct solution methods for optimal control problems[J]. Optimization Methods and Software, 2007, 22(3): 433. DOI:10.1080/10556780600604999
- [13] MITCHELL I M, BAYEN A M, TOMLIN C J. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games[J]. IEEE Transactions on Automatic Control, 2005, 50(7): 947. DOI: 10.1109/TAC.2005.851439
- [14] REISSIG G. Convexity of reachable sets of nonlinear discrete-time systems[C]//Proceedings of the 13th IEEE International Conference on Methods and Models in Automation and Robotics. Szczecin, Poland: IEEE, 2007:199
- [15] GORNOV A Y, FINKEL SHTEIN E A. Algorithm for piecewise-linear approximation of the reachable set boundary[J]. Automation and Remote Control, 2015, 76(3): 385. DOI: 10.1134/s0005117915030030
- [16] DONTCHEV A L, FARKHI E M. Error estimates for discretized differential inclusions[J]. Computing, 1989, 41(4): 349. DOI: 10.1007/bf02241223
- [17] BAIER R, GERDTS M, XAUSA I. Approximation of reachable sets using optimal control algorithms[J]. Numerical Algebra, Control and Optimization, 2013, 3(3): 519. DOI: 10.3934/naco.2013.3.519
- [18] CHAPRA S C, CANALE R P. Numerical methods for engineers[M]. New York: McGraw-Hill, 2010
- [19] WACHTER A, BIEGLER L T. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming[J]. Mathematical Programming, 2006, 106(1): 25. DOI: 10.1007/s10107-004-0559-y

(编辑 张 红)