

文章编号: 1001-0920(2009)05-0653-05

求解动态优化问题的自组织进化算法

武 燕^{1a}, 王宇平^{1b}, 刘小雄², 冶继民^{1a}

(1. 西安电子科技大学 a. 理学院, b. 计算机学院, 西安 710071; 2. 西北工业大学 自动化学院, 西安 710072)

摘 要: 为在环境发生变化后跟踪最优解的变化, 提出一种自组织单变量边缘分布算法 (SOUMDA) 来求解动态优化问题. 自组织策略包含扩散和惯性速度模型, 扩散模型利用当前环境的局部信息使群体向外扩散, 惯性速度模型利用最优解的历史信息进行预测. 将自组织策略与单变量边缘分布算法 (UMDA) 结合, 使得算法在环境变化后自适应地增加种群多样性, 提高算法适应能力, 快速跟踪最优解. 利用动态 sphere 函数对所提出的算法进行测试, 并与 iUMDA 和 MUMDA 算法进行比较, 结果表明所设计的算法能快速适应环境的变化, 跟踪最优解.

关键词: 单变量边缘分布算法; 动态优化; 自组织策略

中图分类号: TP18 **文献标识码:** A

Self-organization evolutionary algorithm for dynamic optimization problems

WU Yan^{1a}, WANG Yurping^{1b}, LIU Xiaoxiong², YE Ji-min^{1a}

(1a. School of Science, 1b. School of Computer Science and Technology, Xidian University, Xi'an 710071, China; 2. College of Automation, Northwestern Polytechnical University, Xi'an 710072, China. Correspondent: WU Yan, E-mail: yanerch@163.com)

Abstract: Dynamic optimization problems require an algorithm to continuously track a changing optimal solution over time. In this paper, a self-organization univariate marginal distribution algorithm (SOUMDA) is proposed to solve dynamic optimization problems. The self-organization scheme is used to increase the diversity of population in a guided fashion. The self-organization scheme consists of two parts. One part uses the history information of optimal solution to predict the change direction, and the other part uses the current information to diverge from the local optimum. Finally, an experiment on dynamic sphere function is carried out to compare the performances of several UMDAs. The experimental results show that the SOUMDA is effective and can adapt to the dynamic environments rapidly.

Key words: Univariate marginal distribution algorithm (UMDA); Dynamic optimization problems; Self-organization scheme

1 引 言

在进化算法研究领域, 大多数研究集中于静态优化问题. 然而, 现实中许多问题都是动态优化问题, 因而利用进化算法求解动态优化问题受到人们越来越多的关注^[1-10]. 动态优化问题是指其目标函数、自变量、约束条件均可能随时间或其他参数发生变化的一类问题. 对于动态优化问题, 进化算法求解的目标已不再是仅仅找到一个最优解, 而是要持续地适应环境的变化, 尽可能地跟踪最优解.

传统进化算法已被证明可以有效地求解各种静态优化问题. 然而, 直接用其求解动态优化问题时

会遇到以下 2 个难题: 1) 一旦环境发生变化, 群体所提供的信息以及一些历史信息过时, 如果直接以这些信息指导种群进化必然会产生错误; 2) 如果种群收敛, 群体中的个体趋于一致, 种群多样性减少, 则环境发生变化后, 群体易陷入局部最优, 因而不能跟踪最优解的变化. 为克服这些问题, 研究者提出了许多方法来改进传统进化算法, 这些方法大致可分为 2 类: 一类是以增加种群多样性为主, 例如重新初始化^[4]、超变异^[5]、随机迁移^[6]和多群体策略^[7,8]等. 利用此类方法虽然可以增加种群多样性, 有利于算法适应环境的变化, 但是增加多样性的方式具有随机

收稿日期: 2008-04-15; 修回日期: 2008-09-13.

基金项目: 国家自然科学基金项目 (60775013, 60873099).

作者简介: 武燕 (1976—), 女, 太原人, 讲师, 博士生, 从事组合优化、进化算法等研究; 王宇平 (1961—), 男, 西安人, 教授, 博士生导师, 从事进化算法、最优化理论等研究.

性,适应环境所需的计算量和计算时间比较大.另一类是以历史信息重复利用为主,例如基于记忆的策略^[9]和时间序列预测^[10]等.基于记忆的策略只适用于循环变化问题;时间序列预测方法目前研究较少,存在的问题是预测复杂而不准确,需要的数据较多.

基于以上分析,为了避免盲目地增加种群多样性,并克服预测复杂和所需数据较多的缺点,本文提出一种自组织策略.将此策略与 UMDA 算法结合,进而提出自组织单变量边缘分布算法(SOUMDA),并使用改进的算法求解动态优化问题.算法充分利用当前群体以及最优解的历史信息,有目的、有方向地增加种群多样性,快速跟踪最优解的变化.通过动态测试函数进行仿真比较,充分验证了本文算法的有效性.

2 连续单变量边缘分布算法

单变量边缘分布算法(UMDA)的连续形式由 Larranaga^[11]给出,是一类基于概率模型的进化算法,结合了竞争学习和进化计算两个领域的知识.

UMDA 首先将群体中每一个个体对应于 1 个 n 维随机变量 $X^t = (X_1^t, X_2^t, \dots, X_n^t)$ 的取值;然后通过从当前代群体的优秀解中提取信息,建立随机变量 X^t 的概率模型,并由概率模型抽样产生下一代群体指导种群进化.UMDA 采用 n 维高斯分布作为概率模型,并假设此联合概率分布可以分解为 n 个一维高斯分布,即

$$X_i^t \sim N(\mu_i^t, \sigma_i^t),$$

$$f(X_i^t = x_i^t) = \frac{1}{\sqrt{2\pi}\sigma_i^t} \exp\left\{-\frac{(x_i^t - \mu_i^t)^2}{2(\sigma_i^t)^2}\right\}, \quad (1)$$

式中 μ_i^t 和 σ_i^t 表示高斯分布的均值和方差.那么,采用精英保留策略的 UMDA 描述如下:

Step1: $t = 0$, 对每一个变量 X_i^t , 随机产生正态分布的各个参数 (μ_i^t, σ_i^t) , $i = 1, \dots, n$, 给定变异概率 P_m , elitist = [].

Step2: 由概率分布抽样产生 N 个个体,组成群体 $\text{Pop}(t)$, N 为种群规模.

Step3: 采用某种选择机制(本文采用截断选择),从群体 $\text{Pop}(t) \setminus \text{elitist}$ 中选择 $M < N$ 个个体组成中间群体 $S(t)$, 并将部分最优个体作为精英保留.

Step4: 依据下式估计新的概率密度函数参数:

$$\mu_i^{t+1} = \frac{\sum_{j=1}^M x_j^{i,t}}{M}, \quad \sigma_i^{t+1} = \sqrt{\frac{\sum_{j=1}^M (x_j^{i,t} - \mu_i^{t+1})^2}{M}},$$

$$i = 1, \dots, n. \quad (2)$$

Step5: 依据变异概率对 X_i^{t+1} 进行位变异, $i = 1, \dots, n$.

Step6: 若终止条件不满足,则转 Step2; 否则,

停止.

算法在计算出 X_i^{t+1} ($i = 1, \dots, n$) 的值后,为了保持抽样的多样性,在 Step5 中对 X_i^{t+1} ($i = 1, \dots, n$) 进行变异操作.由上述算法可以看出,随着迭代次数的增加,概率分布的方差逐渐收敛到 0,均值收敛到一个固定的值,群体逐渐收敛.

与传统进化算法一样,当 UMDA 算法收敛时,群体中个体趋于一致,失去种群多样性.如果这时环境发生变化,算法将陷入局部最优,失去适应环境的能力.因此,在求解动态优化问题时,需改进 UMDA,以适应环境的变化.

3 自组织单变量边缘分布算法

在复杂系统中,自组织机制是适应环境变化的根本机制.基于此,本文提出一种自组织策略,并应用此策略改进传统进化算法来求解动态优化问题.

自组织策略的基本思想是:在环境发生变化后,扩大搜索空间,以增加种群多样性,寻找变化后最优解的潜在区域.然而,盲目增加种群多样性,将会增加计算量,延迟找到最优解的时间,不能快速跟踪最优解的变化.为了克服这些缺点,文中所提出的自组织策略不但自动扩大搜索空间,而且充分挖掘当前代群体所处环境的信息以及最优解的历史信息,从而有目的地增加种群多样性.下面对本文所提出的自组织策略进行详细的讨论.

3.1 扩散模型

在静态优化问题的求解中,当前代群体的最优解 $x^*(t)$ 作为重要信息被利用,通常将其称为吸引子,使得群体在保持一定多样性基础上朝着当前代群体的最优解方向移动.然而,在动态优化问题中,当环境发生变化时,当前代群体的最优解 $x^*(t)$ 提供的信息可能已经过时甚至错误,因此不能作为吸引子来指导进化.但是,此时当前代群体的最优解 $x^*(t)$ 可以提供这样一个相反的信息,即 $x^*(t)$ 不是变化后最优解的位置,而是一个陷阱.基于以上分析,为了快速找到环境变化后最优解的潜在区域,其他个体可以将 $x^*(t)$ 作为排斥子,从而自动朝背离 $x^*(t)$ 的方向移动.

设环境在第 t 代结束发生变化,则扩散模型可表示为

$$x^i(t+1) = x^i(t) + \alpha(x^i(t) - x^*(t)). \quad (3)$$

其中: $x^i(t)$ 为第 t 代的第 i 个个体, $x^*(t)$ 为第 t 代的最优解, α 为参数.由上式可以看出,环境在第 t 代结束发生变化,这时第 t 代最优解 $x^*(t)$ 已不再是吸引子,而是排斥子,其他个体将它作为一个陷阱而纷纷远离它,朝着远离第 t 代的最优解 $x^*(t)$ 的方向进行扩散, α 表示沿此方向移动的步长.因此,扩散模

型的目的是利用当前环境的信息增加种群多样性, 以适应环境的变化.

3.2 惯性速度模型

环境发生变化后, 如果能预测出最优解的位置, 则有助于算法快速并有效地适应环境的变化并找到最优解. 在文献 [10] 中将最优解看作时间序列来预测下一刻最优解的位置, 但是存在模型复杂、所需数据大和计算量大等问题. 为了避免模型过于复杂, 本文采用前一次环境变化时最优解移动的速度作为这一次最优解移动的速度, 并将此速度称为惯性速度, 用来预测变化后最优解的位置.

设环境第 k 次 (第 t 代) 发生变化时, 群体的最优解为 $B(k)$, $B(k) = x^*(t)$, 初始群体的最优解记为 $B(0)$, 那么记 $v(k-1) = B(k) - B(k-1)$, $v(k-1)$ 称为环境第 $k-1$ 次变化最优解移动的速度. 以此来预测第 k 次最优解变化的速度, 其目标是利用以前的信息预测出变化后最优解的位置, 快速找到最优解. 则惯性速度模型可表示为

$$x^i(t+1) = x^i(t) + \alpha v(k-1), \quad (4)$$

其中 α 为参数. 由上式可以看出, 当环境第 k 次发生变化时, 最优解仍惯性地向着第 $k-1$ 次最优解变化的方向移动, α 表示沿此方向移动的步长. 如果预测较准确, 则算法可以很快找到最优解的邻近区域, 并很快找到最优解.

3.3 自组织策略

通过以上讨论可知, 扩散模型充分利用当前环境信息增加种群多样性, 惯性速度模型利用最优解的历史信息预测变化后的最优解位置. 然而, 单独使用扩散模型, 会导致解过于分散, 影响算法的收敛性, 而单独使用惯性速度, 如果预测不准确, 反而会延长算法找到最优解的时间. 因此, 本文结合扩散模型和惯性速度模型形成自组织策略, 从而使得算法有目的、自适应地朝着最优解潜在区域进行扩散, 提高算法的效率. 基于以上讨论, 本文提出的自组织策略如图 1 所示. 图中

$$x^i(t+1) = x^i(t) + \alpha_1(x^i(t) - x^*(t)) + \alpha_2 v(k-1). \quad (5)$$

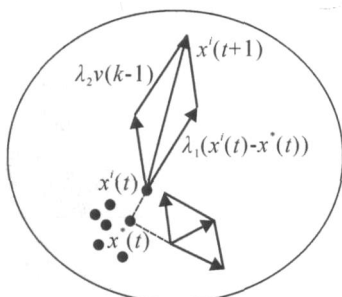


图 1 自组织策略结构图

由式 (5) 可以看出, 如果环境在第 t 代发生变化 (对应环境第 k 次发生变化), 则群体中的个体将自组织地以当前环境的信息以及历史信息为基础进行移动, 并且有目的、有方向地增加种群多样性, 而不是盲目地在整个搜索空间进行搜索, 从而提高了算法适应环境的能力.

3.4 自组织 UMDA

UMDA 是基于概率模型的一种进化算法. 此算法克服了杂交和变异算子对积木块的破坏, 是一种改进的进化算法. 本文将自组织策略与 UMDA 相结合, 提出自组织 UMDA, 称为 SOUMDA. 当环境不发生变化时, 采用 UMDA 进行求解; 当环境发生变化时, 则采用自组织策略进行调整. 环境的检测采用最优解的重复计算. 算法描述如下:

Step1: $t = 0$, 对每一个变量, 随机产生正态分布的各个参数 (μ_i^t, σ_i^t) , $i = 1, \dots, n$, 变异概率 P_m , elitist = [].

Step2: 由概率分布抽样产生 N 个个体, 组成群体 $Pop(t)$, N 为种群规模.

Step3: 检测环境是否变化, 若变化, 则转 Step4; 否则, 转 Step5.

Step4: 采用自组织策略对 $Pop(t)$ 中的个体进行调整, 组成新的群体 $Pop(t)$.

Step5: 采用某种选择机制 (本文采用截断选择) 从群体 $Pop(t)$ 中 elitist 中选择 $M < N$ 个个体组成中间群体 $S(t)$, 并且 elitist = $S(t)$.

Step6: 依据式 (2) 估计新的概率密度函数的参数.

Step7: 依据变异概率对 μ_i^{t+1} 进行位变异, $i = 1, \dots, n$.

Step8: 若终止条件不满足, 则转 Step2; 否则, 停止.

SOUMDA 通过自组织策略将个体向着最优解可能存在的区域迁移, 实现有指导地增加种群多样性. 在使用自组织策略进行迁移的过程中, 对于同一个个体所使用的参数 α_1 和 α_2 是相同的, 而对于不同个体所使用的参数则不同. 参数取值为 1 到最大移动步长之间的随机整数, 最大移动步长事先给定, 本文最大步长取 5.

4 实验分析

4.1 动态测试函数

为了测试所提出算法的性能, 采用动态三 sphere 函数 [12] 进行测试. 函数形式如下:

$$\min f(x, y, z) = \min (x - \hat{x}_1)^2 + (y - \hat{y}_2)^2 + (z - \hat{z}_3)^2.$$

其中: $(\hat{x}_1, \hat{y}_2, \hat{z}_3)$ 为全局最优解, 可以按不同的轨迹

周期地或随机地移动,形成动态函数; k 表示每间隔时间,环境变化一次,即 k 增加1.根据 $(\frac{k}{1}, \frac{k}{2}, \frac{k}{3})$ 移动的形式,可以形成如下3种不同类型的动态优化模型:

1) 线性模型.在线性动态模型中,全局最优点在每一维上的偏移量计算如下:

$$x_m^k = x_m^{k-1} + s, \quad x_m^0 = 0, \quad m = 1, 2, 3.$$

2) 环形模型.在环形动态模型中,全局最优点在每一维上的偏移量计算如下:

$$x_m^k = x_m^{k-1} + s \sin \frac{2\pi k}{T}, \quad m = 1, 3;$$

$$x_m^k = x_m^{k-1} + s \cos \frac{2\pi k}{T}, \quad m = 2, \quad x_m^0 = 0.$$

3) 随机模型.在随机动态模型中,全局最优点在每一维上的偏移量计算如下:

$$x_m^k = x_m^{k-1} + sN(0, 1), \quad m = 1, 2, 3, \quad x_m^0 = 0.$$

其中: s 为正常数,用来控制移动的步长.

动态三维 sphere 函数变化的强度由移动的步长 s 确定,变化的频率由 τ 确定,即每隔 τ 代,环境发生一次变化.

4.2 实验结果及分析

在动态优化问题的求解中,衡量算法好坏的最基本的性能指标是离线误差,它是某一代所得的最优值与实际最优值误差的平均值

$$E(t) = \frac{1}{T} \sum_{i=1}^T e_i(t).$$

其中: $e_i(t)$ 为第 i 次运算第 t 代的误差; T 为算法重复独立运行的次数,对于动态 sphere 函数,为了减小随机性对算法性能的影响, T 的取值为50.

利用4.1节给出的动态优化测试函数,对算法 SOUMDA, MUMDA 和 iUMDA 进行仿真比较.其中算法 MUMDA 是指基于超变异的 UMDA^[5],算法 iUMDA 是指基于随机迁移的 UMDA^[6],即 UMDA 在概率向量抽样产生 N 个个体后,利用随机产生的个体随机替换抽样中的 $r \times N$ 个个体,其中 r 为迁移率.超变异和随机迁移是增加种群多样性的两种非常有效的方法,已被用来求解动态优化问题^[5,6],为此本文将 SOUMDA 的性能与 MUMDA, iUMDA 的性能进行了比较.

实验中,测试函数的搜索空间为 $[-50, 50]^3$,种群规模均为40, $M = 0.5N$, $r = 0.2$,变异概率 $P_m = 0.2$,最大代数500,环境变化的频率分别取为1, 5和10,变化的强度 s 分别取为0.1, 1和5.用 SOUMDA, MUMDA 和 iUMDA 对测试函数每一组实验参数分别独立运行50次,实验比较结果如图2~图4所示.

图2给出了3种算法对于线性模型在不同变化强度、不同变化频率下的实验结果.由图2可以看出,算法 SOUMDA 明显优于算法 MUMDA 和 iUMDA,这是由于环境发生变化后,采用自组织策略增加了种群多样性,使得个体向最优解的可能区

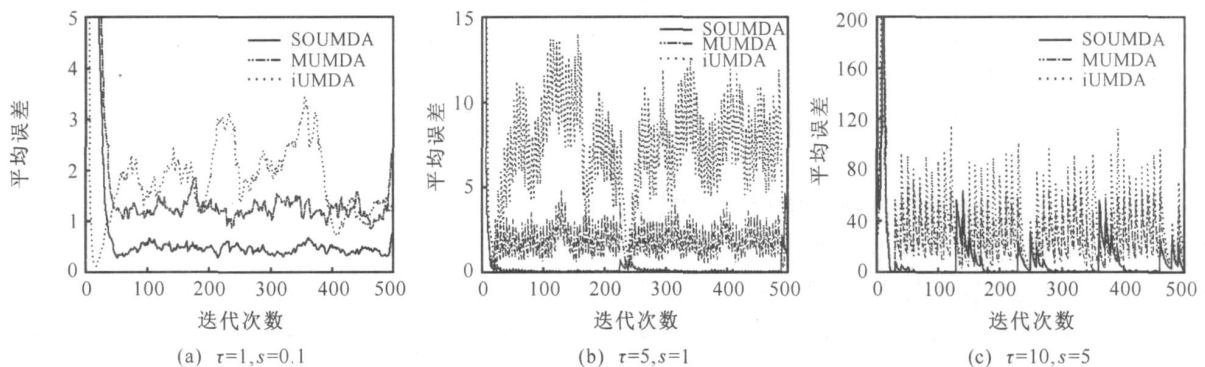


图2 3种算法在线性模型下的实验结果

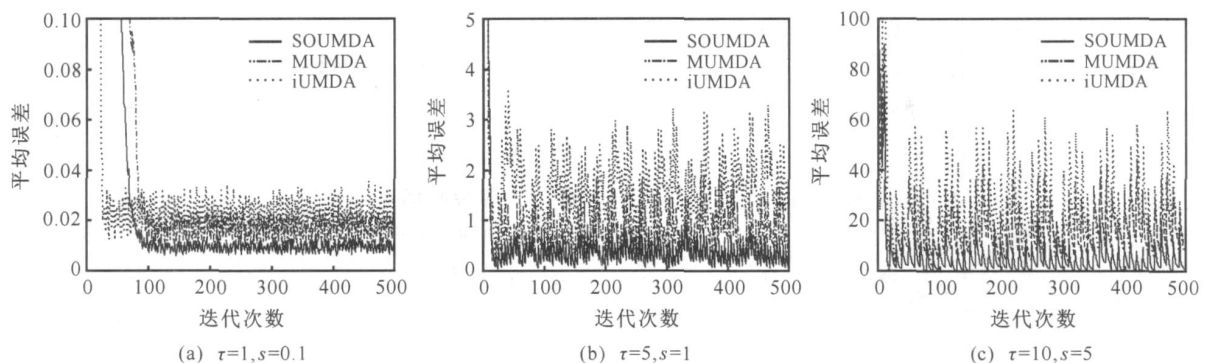


图3 3种算法在环形模型下的实验结果

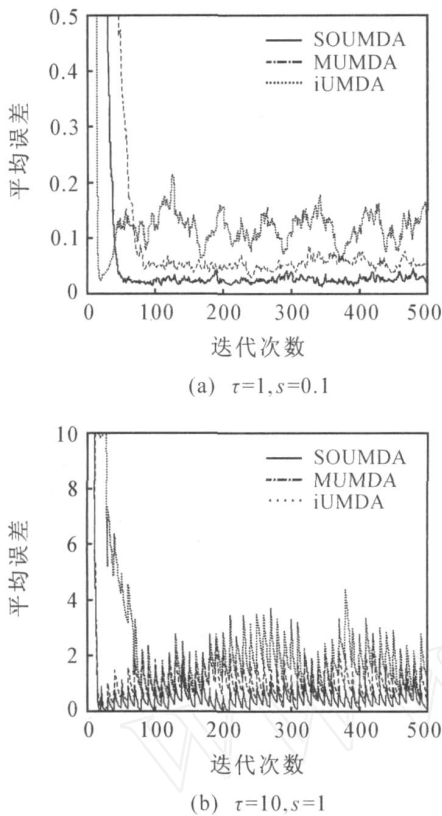


图 4 3 种算法在随机模型下的实验结果

域进行扩散,实现对部分区域的搜索,从而避免在整个搜索空间进行搜索,提高了算法的搜索效率;算法 iUMDA 表现的性能最差,这是由于随机迁移存在盲目性,它是通过对整个空间进行取点来增加种群多样性,因而对于这种线性模型,不能有效发现最优解的潜在区域,影响了算法的收敛速度;因为变异操作只是保留了一部分原群体的信息,没有更好地适应环境的变化,所以算法 MUMDA 的表现介于算法 SOUMDA 与 iUMDA 之间。

图 3 给出了 3 种算法对于环形模型在不同变化强度、不同变化频率下的实验结果。由图 3 可以看出,对于环形模型,算法 SOUMDA 仍然优于算法 MUMDA 和 iUMDA。在环形模型下,最优解移动的方向不断变化,虽然利用惯性速度不能获得潜在区域,但是自组织策略综合考虑了当前群体的信息,即使方向不断变化,仍能快速适应环境的变化。在环形模型下,算法 MUMDA 与 iUMDA 的性能较为接近,算法 MUMDA 稍好于 iUMDA。MUMDA 利用超变异的方式增加种群多样性,变异后所得到的新的群体较 iUMDA 集中,且较接近于潜在区域,所以适应环境的能力较 iUMDA 好。算法 iUMDA 始终在整个空间搜索,因此适应环境的能力最差。

图 4 给出了 3 种算法对于随机模型在不同变化强度、不同变化频率下的实验结果。由图 4 可以看出,对于随机模型,算法 SOUMDA 仍能很快适应环

境的变化。这是由于自组织策略包含了扩散模型,虽然环境的变化是随机的,但采用自组织策略后新群体仍能快速发现或接近最优解的潜在区域。

由实验结果可以看出,对于每一种动态模型,随着变化强度的增加,3 种算法的误差加大,振荡加大,适应能力减弱,说明对于同一算法在变化强度较小时的适应能力强,在变化强度较大时的适应能力弱。算法 SOUMDA 在线性模型下始终保持较好的适应能力,在环形模型和随机模型下,随着变化强度的增加,适应能力减弱,但是与算法 MUMDA 和 iUMDA 相比,仍然表现出良好的性能。

5 结 论

本文提出了一种自组织 UMDA 算法,用于求解动态优化问题。算法的基本思想是在环境发生变化后,采用自组织策略有方向、有指导地增加种群多样性,以快速适应环境的变化。利用文中给出的动态测试函数对 SOUMDA, MUMDA 和 iUMDA 在不同动态环境下分别进行了计算。实验结果表明,SOUMDA 算法能够快速适应环境的变化,跟踪最优解。本文方法简单实用,可应用于其他进化算法,为解决动态优化问题拓宽了研究思路。

参考文献(References)

[1] Yaochu J, Branke J. Evolutionary optimization in uncertain environments — A survey[J]. IEEE Trans on Evolutionary Computations, 2005, 9(3): 1-15.

[2] 王洪峰,汪定伟,杨圣祥. 动态环境中的进化算法[J]. 控制与决策, 2007, 22(2): 127-131.
(Wang H F, Wang D W, Yang S X. Evolutionary algorithms in dynamic environments [J]. Control and Decision, 2007, 22(2): 127-131.)

[3] 刘淳安,王宇平. 动态多目标优化的进化算法及其收敛性分析[J]. 电子学报, 2007, 35(6): 1118-1121.
(Liu C A, Wang Y P. Evolutionary algorithm for dynamic multi-objective optimization problems and its convergences[J]. Acta Electronica Sinica, 2007, 35(6): 1118-1121.)

[4] Cartwright H M, Tuson A L. Genetic algorithms and flowshop scheduling: Towards the development of a real-time process control system[C]. Proc of the AISB Workshop on Evolutionary Computing. San Francisco: Morgan Kaufmann Publishers, 1994: 277-290.

[5] Cobb H G. An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent nonstationary environment [R]. Washington: Naval Research Laboratory, 1990.

(下转第 662 页)

$$K_3 = [2.1410 \quad 17.0880].$$

给定系统初始值 $x(0) = [1.2, -0.5]^T$, 在所求得的控制作用下系统的状态轨迹如图5所示.

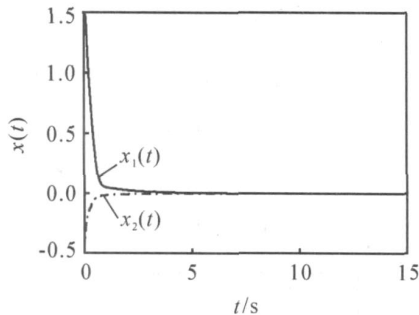


图5 系统状态轨迹

5 结论

本文给出了一种基于连续时间 T-S 模糊模型的二次镇定条件. 在控制器的设计中采用一种更有效的方式考虑隶属函数的有用信息, 进而可以大大减少结果的保守性. 从仿真结果可以看出本文方法是有效的.

参考文献 (References)

- [1] Takagi T, Sugeno M. Fuzzy identification of systems and its application to modeling and control [J]. IEEE Trans on Systems, Man, and Cybernetics, 1985, 15(2): 116-132.
- [2] Feng G, Cao S, Rees W. An approach to H control of a class of nonlinear systems[J]. Automatica, 1996, 32(10): 1469-1474.
- [3] Wang H, Tanaka K, Griffin M. An approach to fuzzy control of nonlinear systems: Stability and design issues [J]. IEEE Trans on Fuzzy Systems, 1996, 4(1): 14-23.
- [4] Tanaka K, Ikeda T, Wang H. Fuzzy regulators and fuzzy observers: Relaxed conditions and LMFbased designs [J]. IEEE Trans on Fuzzy Systems, 1998, 6(2): 250-265.
- [5] Kim E, Lee H. New approaches to relaxed quadratic stability conditions of fuzzy systems [J]. IEEE Trans on Fuzzy Systems, 2000, 8(5): 523-534.
- [6] Liu X, Zhang Q. New approaches to H controller designs based on fuzzy observers for T-S fuzzy systems via LMI [J]. Automatica, 2003, 39(9): 1571-1582.
- [7] Teixeira M, Assuncao E, Avellar R. On relaxed LMF-based designs for fuzzy regulators and fuzzy observers [J]. IEEE Trans on Fuzzy Systems, 2003, 11(5): 613-622.
- [8] Fang C, Liu Y, Kau S, et al. A new LMFbased approach to relaxed quadratic stabilization of T-S fuzzy control systems [J]. IEEE Trans on Fuzzy Systems, 2006, 14(3): 386-397.
- [9] Sala A, Arino C. Asymptotically necessary and sufficient conditions for stability and performance in fuzzy control [J]. Fuzzy Sets and Systems, 2007, 158(24): 2671-2686.
- [10] Sala A, Arino C. Relaxed stability and performance conditions for T-S fuzzy systems with knowledge on membership function overlap [J]. IEEE Trans on Systems, Man, and Cybernetics — Part B, 2007, 37(3): 727-732.
- [6] Grefenstette J J. Genetic algorithms for changing environments [C]. Proc of the Parallel Problem Solving from Nature. Brussels, 1992: 137-144.
- [7] Oppacher F, Wineberg M. The shifting balance genetic algorithm: Improving the GA in a dynamic environment [C]. Proc of Genetic and Evolutionary Computation. San Francisco: Morgan Kaufmann Publishers, 1999: 504-510.
- [8] Branke J, Kaubler T, Schmidt C, et al. A multi-population approach to dynamic optimization problems [C]. Proc of the Adaptive Computing in Design and Manufacturing. Berlin: Springer-Verlag, 2000: 299-308.
- [9] Shengxiang Yang. Memory-enhanced univariate marginal distribution algorithms for dynamic optimization problems [C]. IEEE Congress on Evolutionary Computation. Edinburgh, 2005, 3: 2560-2567.
- [10] Iason Hatzakis, David Wallace. Dynamic multi-objective optimization with evolutionary algorithms: A forward-looking approach [C]. Genetic and Evolutionary Computation Conf. Proc of the 8th Annual Conf. Seattle, Washington, 2006: 1201-1208.
- [11] Larranaga P, Etxeberria R, Lozano L A, et al. Optimization by learning and simulation of Bayesian and Gaussian networks [R]. Euskal Herriko Unibertsitatea, Spain: Department of Computer Science and Artificial Intelligence, University of the Basque Country, 1999: 2254-2265.
- [12] Guanyu Pan, Quansheng Dou, Xiaohua Liu. Performance of two improved Particle swarm optimization in dynamic optimization environments [C]. Proc of the 6th Int Conf on Intelligent Systems Design and Applications. Washington, 2006: 2075-2085.

(上接第 657 页)