

文章编号: 1001-0920(2009)06-0911-05

基于动态罚函数法的协同优化算法

李海燕^a, 井元伟^b, 马明旭^a, 张嗣瀛^b

(东北大学 a. 流程工业综合自动化教育部重点实验室, b. 信息科学与工程学院, 沈阳 110004)

摘要: 为保证协同优化的系统级优化存在可行域, 采用动态罚函数法, 将学科间一致性约束条件下的系统级优化问题转化为无约束优化问题. 提出了应用更为普遍的学科间不一致信息的定义形式, 对各种定义形式进行了分析比较, 并利用该值构造动态罚因子的表达式. 从增强算法可靠性的角度, 使用遗传算法来取代系统级优化问题中基于梯度的优化算法, 同时减少了对优化函数的连续性要求. 利用减速器典型算例对该方法进行了验证, 结果表明该方法具有良好的优化性能.

关键词: 协同优化; 遗传算法; 动态罚因子; 学科间不一致信息

中图分类号: TP301.6 **文献标识码:** A

Collaborative optimization algorithm based on dynamic penalty function method

LI Hai-yan^a, JING Yuan-wei^b, MA Ming-xu^a, ZHANG Si-ying^b

(1. Key Laboratory for Process Industry Automation of Ministry of Education, b. College of Information Science and Engineering, Northeastern University, Shenyang 110004, China. Correspondent: LI Hai-yan, E-mail: lihaiyan0313@163.com)

Abstract: In order to ensure the solvability of system level optimization in collaborative optimization, the dynamic penalty function approach is applied to turn the system-level compatibility constrained optimization problem to a no-constrained optimization problem. The applicable definitions of interdisciplinary discrepancy information are presented and the expression of dynamic penalty factor is determined by the interdisciplinary discrepancy information. In order to improve the reliability of collaborative optimization, the genetic algorithm is used to search the design space in system level optimization problem instead of the gradient-based optimization algorithms, which has no restriction on the continuity property of optimization function. The speed reducer example is adopted to test this optimization algorithm and the results show that the presented approach is efficient.

Key words: Collaborative optimization; Genetic algorithm; Dynamic penalty factor; Interdisciplinary discrepancy information

1 引言

复杂工程系统的设计问题, 往往涉及多个相互交叉的学科领域, 由于多学科设计优化能有效处理其复杂的设计过程, 受到了国内外学者的广泛关注. 斯坦福大学 Kroo 等^[1]提出的协同优化(CO)算法, 被认为是一种较有前途的多学科设计优化方法, 并已在飞行器设计等研究工作中得到了应用. CO 算法的分解形式与工程设计分工的组织形式相一致, 各学科保持了各自的分析设计自由. 同时, 具有软件易于集成、可并行处理等优点, 能极大地缩短设计周

期, 非常适合于大规模工程系统的多学科设计优化问题^[2].

作为一门新兴的学科, CO 在很多方面还不够完善. 系统级优化中采用的一致性等式约束是一种理想状态, 而在一般情况下, 系统级优化问题的可行域很可能不存在^[3]. 同时, 由于最优解处学科一致性约束的 Jacobian 矩阵不连续, 导致 CO 算法的系统级优化问题无法满足 Kuhn-Tucker 条件^[4,5]. 另外, 部分研究结果表明, CO 的优化结果对初始点的选取较为敏感, 而且优化结果可能不收敛^[6,7]. 目前, 出现了一些协同优化算法的改进策略^[3,8]. 如: 基于

收稿日期: 2008-04-02; 修回日期: 2008-09-22.

基金项目: 解放军总装备部武器装备预研基金项目(9140A18010106LN0101).

作者简介: 李海燕(1979—), 女, 河北承德人, 助教, 博士生, 从事多学科优化、稳健优化的研究; 井元伟(1956—), 男, 辽宁西丰人, 教授, 博士生导师, 从事复杂系统、远程通讯网络及通信控制等研究.

响应面的协同优化算法^[8],利用响应面来近似系统级一致性约束函数,响应面的构造依赖于设计点的选择方法和近似技术;松弛因子法^[3]对系统级一致性等式约束进行松弛,将等式约束变为不等式形式,但松弛因子的选取比较困难.

本文针对系统级优化可行域可能不存在以及对初始点选取敏感的问题,采用动态罚函数法将系统级优化转化为无约束优化问题;给出了学科间不一致信息的两种定义形式,并利用其中一种定义形式构造动态罚因子的表达式;同时,采用遗传算法来增强系统级优化的稳定性.该算法不但可以保证子学科间的一致性,还可有效改善优化结果对初始点选取的敏感性.

2 协同优化算法

协同优化将复杂的工程系统设计问题,根据现有的工程分工形式,分解成系统级和学科级两级优化结构.首先,系统级向学科级提供设计变量的期望值,各学科在满足自身约束条件下,使其优化结果与系统级提供给该学科的目标值之间的差异最小,并将优化结果传递给系统级.系统级负责规划协调,通过学科间一致性约束来协调各学科的优化结果.系统级优化目标函数为原问题的目标函数,约束为学科间一致性约束,优化结束后,优化结果再次传给学科级.经过系统级优化和学科级优化之间的多次迭代,最终得到一个最优的系统设计方案.CO算法结构比较简单,容易实现学科自治,各子学科间的连接通过系统级优化问题的等式约束得到加强.

假设系统设计问题可以分解为 n 个子学科,则协同优化的数学表述形式如下:

系统级优化

$$\begin{aligned} \min F(z), \\ \text{s. t. } J_i^*(z) = \sum_{j=1}^{s_i} (z_j - x_{ij}^*)^2 = 0, \\ i = 1, 2, \dots, n. \end{aligned} \quad (1)$$

其中: z 为系统级优化设计向量, z_j 为第 j 个系统级设计变量, s_i 为学科 i 的设计变量数, x_{ij}^* 为学科 i 的第 j 个设计变量的最优解, n 为子学科数.

学科级 i 优化

$$\begin{aligned} \min J_i(x_i) = \sum_{j=1}^{s_i} (x_{ij} - z_j^*)^2, \\ \text{s. t. } c_i(x_i) = 0. \end{aligned} \quad (2)$$

其中: x_i 为学科级 i 的设计向量, s_i 为学科 i 的设计变量数, x_{ij} 为学科 i 的第 j 个设计变量, z_j^* 为系统级分配给学科级的第 j 个设计变量的期望值, $c_i(x_i)$ 为学科级约束.

3 基于动态罚函数法的 CO 算法

在 CO 系统级优化中,学科一致性等式约束的引入,会导致系统级优化出现可行域不存在的现象.系统级优化在优化设计点附近无法满足 Kuhn-Tucker 条件,使得系统级优化不能采用基于梯度的优化算法求解.基于动态罚函数法的 CO 算法,是针对 CO 系统级优化过程进行的改善.

3.1 优化算法的选取

在实际工程应用中,通常存在不连续等多种形式的优化目标函数.当目标函数连续且可导时,约束函数 $J_i^*(z)$ 也可能不可导,甚至是不连续的.基于梯度的优化算法要求目标函数和约束都是可导的,但这一条件在系统级优化问题中并非总满足.文献[9]中的系统级优化采用序列二次规划法,但应用结果表明无法收敛.

遗传算法对优化函数不要求连续,更不要求可导,既可以是数学解析所表示的显函数,也可以是其他方式的隐函数,对优化函数几乎没有限制.遗传算法对搜索空间中的多个解同时进行评估,减少了陷入局部最优解的风险,具有较好的全局搜索性能.因此,在 CO 系统级优化过程中,采用遗传算法进行寻优计算.

3.2 系统级优化形式

为了保证系统级优化可行,将各等式约束作为惩罚项加入系统级目标函数中.在系统级优化过程中,通过惩罚项迫使目标函数优化结果向学科间不一致信息减少的方向靠近,从而使再次传递给学科级优化的系统级期望值更能满足各学科级约束的要求.经两级优化的多次迭代,最终得到既满足各学科间的一致性要求,又能使目标函数取得极值的最优解.此时系统级优化形式为

$$\min F(z) = F(z) + \sum_{i=1}^n J_i^*(z). \quad (3)$$

其中: $F(z)$ 为原系统级目标函数, μ 为惩罚因子, $J_i^*(z)$ 为学科间一致性等式约束.

利用罚函数方法处理约束问题时,其性能很大程度上取决于罚因子的选择,而罚因子的取值难以把握.在 CO 系统级优化过程中,可利用学科间不一致信息较大时,给学科一致性约束赋予较大的权值,以保证学科间的一致性;当学科间不一致信息较小时,给学科间一致性约束赋予较小的权值,使目标函数向极值收敛,并保持学科间的一致性.按照这种规则,构造出的动态罚因子表达式为

$$= b + mk. \quad (4)$$

其中: b , m 和 k 是常数; k 为学科间不一致信息.当 k

非常小时, 的值主要由常数项 b 来决定. 该值使得此时目标函数的优化过程仍受学科间一致性约束的限制, 防止学科间不一致信息 k 再次变大, 维持了学科间的一致性, 增强了 CO 算法的稳定性. 当初始点在可行域内取值时, 学科间不一致信息 k 为零, 由 b 的值将一致性约束加入增广目标函数中, 从而使其初始点在可行域内外均可取值, 减少优化结果对初始点选取的敏感性.

3.3 参数的取值

3.3.1 b, m 和 α 的取值

b 的初值可与最终的系统目标函数值在同一数量级上, 然后根据优化情况进行调整, 以系统目标函数值变化曲线和学科间不一致信息变化曲线不出现震荡现象为原则. m 和 α 的作用是控制学科间一致性约束的权重, 与学科间一致性的精度要求有关.

取正整数即可, 通常取 $\alpha = 1$. 当 α 的值固定时, 优化结果对学科间一致性精度的要求越高时, m 的值越大. 当学科间一致性满足精度要求时, 保证 mk 的值大于 b 的值即可. 因此, 当 α 的值固定时, 若要增加优化结果的学科一致性精度, 则应增加 m 的数量级.

3.3.2 学科间不一致信息 k

文献[10]利用学科级优化过程中所获得的子学科的优化结果, 对学科间不一致信息进行了定义, 其形式为

$$k_1 = x_1^* - x_2^* \quad (5)$$

其中 x_1^* 和 x_2^* 分别为优化过程中, 经子学科 1 和子学科 2 优化得到的优化结果.

由式(5)可知, 该定义只适于各子学科设计向量相同的情况, 而对于大多数情况, 各子学科的设计向量是不完全一致的. 当系统级分配的设计向量期望值越靠近各子学科的可行域时, 各子学科的优化目标函数值越小, 各子学科返回的优化结果越能满足系统级的一致性等式约束. 因而, 系统级优化中的一致性等式约束的不满足程度与子学科的目标函数值也体现了学科间的一致性情况. 故本文给出了学科间不一致信息的另外两种定义形式, 即

$$k_2 = J_1^*(z_1^*, x_1^*) + J_2^*(z_2^*, x_2^*) + \dots + J_n^*(z_n^*, x_n^*) \quad (6)$$

其中: $z_1^*, z_2^*, \dots, z_n^*$ 为本次系统级分配给各子学科的设计向量期望值; $x_1^*, x_2^*, \dots, x_n^*$ 为上一次学科级返回的优化结果; $J_1^*, J_2^*, \dots, J_n^*$ 为系统级一致性等式约束.

$$k_3 = J_1(x_1^*, z_1^*) + J_2(x_2^*, z_2^*) + \dots + J_n(x_n^*, z_n^*) \quad (7)$$

其中: $x_1^*, x_2^*, \dots, x_n^*$ 为本次各子学科的优化结果;

$z_1^*, z_2^*, \dots, z_n^*$ 为系统级传递给各子学科的设计向量期望值; J_1, J_2, \dots, J_n 为各子学科的优化目标函数值.

由式(5) ~ (7)可知, k_1 的值由学科级优化结果计算得到, 且适用于各子学科的设计向量完全一致的情况. k_2 的值由本次系统级设计向量期望值与上次学科级返回的优化结果计算得到, 其计算过程增加了求取一致性等式约束的计算量. k_3 的值由各子学科的优化目标函数值计算得到, 它们的值由各子学科级返回给系统级, k_3 的计算过程增加了信息的传输量. k_2 与 k_3 计算过程中所用的系统级分配的设计向量相同, 所用的学科级优化结果相差一次. 与 k_2 相比, k_3 的计算过程减少了计算量而增加了信息的传输量.

4 实例验证

减速器多学科设计优化问题是 NASA (National aeronautics and space administration) 评估多学科设计优化算法性能的 10 个标准算例之一, 常用来测试 CO 算法的性能^[10,11]. 其目标是在满足转轴和齿轮大量约束的同时, 使得减速器的体积最小(质量最轻).

4.1 数学模型

该优化问题共有 7 个设计变量, 其数学模型为

$$\begin{aligned} \min f(x) = & 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.5079x_1(x_6^2 + x_7^2) + \\ & 7.477(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2); \\ \text{s. t. } & g_1 = 27/(x_1x_2^2x_3) - 1.0 \leq 0, \\ & g_2 = 397.5/(x_1x_2^2x_3^2) - 1.0 \leq 0, \\ & g_3 = 1.93x_3^3/(x_2x_3x_6^4) - 1.0 \leq 0, \\ & g_4 = 1.93x_3^3/(x_2x_3x_7^4) - 1.0 \leq 0, \\ & g_5 = \frac{\sqrt{745x_4}^2 + 16.9 \times 10^6}{110x_6^3} - 1.0 \leq 0, \\ & g_6 = \frac{\sqrt{745x_5}^2 + 157.5 \times 10^6}{85x_7^3} - 1.0 \leq 0, \\ & g_7 = x_2x_3 - 40.0 \leq 0, \\ & g_8 = 5.0 - x_1/x_2 \leq 0, \\ & g_9 = x_1/x_2 - 12.0 \leq 0, \\ & g_{10} = (1.5x_6 + 1.9)/x_4 - 1.0 \leq 0, \\ & g_{11} = (1.1x_7 + 1.9)/x_5 - 1.0 \leq 0, \\ & 2.6 \leq x_1 \leq 3.6, 0.7 \leq x_2 \leq 0.8, \\ & 17 \leq x_3 \leq 28, 7.3 \leq x_4 \leq 8.3, \\ & 7.3 \leq x_5 \leq 8.3, 2.9 \leq x_6 \leq 3.9, \\ & 5.0 \leq x_7 \leq 5.5. \end{aligned} \quad (8)$$

其中： x_1 为齿面宽度， x_2 为齿轮模数， x_3 为小齿轮齿数， x_4 和 x_5 为轴承间距， x_6 和 x_7 为小大齿轮的直径， g_1 为轮齿的最大弯曲应力约束， g_2 为轮齿最大接触应力约束， g_3 和 g_4 为轴的横向最大扰度约束， g_5 和 g_6 为轴内最大应力约束， $g_7 \sim g_9$ 为尺寸和空间限制， g_{10} 和 g_{11} 为轴尺寸计算的经验公式。

4.2 优化方案设计

系统级优化采用遗传算法，编码方式采用实数编码，种群大小为 250。采用目标函数作为适应度函数，采用轮盘赌选择机制，代沟设为 0.9。采用单点交叉方式，交叉概率设为 0.7，采用均匀变异算子，变异概率设为 0.004。该优化问题可分解为 3 个子学科级优化和 1 个系统级优化。子学科 1 由 $g_1, g_2, g_7 \sim g_9$ 组成；子学科 2 由 $g_1, g_2, g_4, g_6, g_7 \sim g_9$ 以及 g_{11} 组成；子学科 3 由 $g_1 \sim g_3, g_5, g_7 \sim g_{10}$ 组成^[11]。因 3 个子学科的设计向量不同，无法利用 k_1 计算学科间不一致信息，故采用 k_3 构造系统级优化中的动态罚因子，罚因子的取值为 $= 1000 + 10^8 k_3$ 。此时系统级和学科级的数学表达式为

系统级

$$\begin{aligned} \min f(z) = & 0.7854z_1z_2^2(3.3333z_3^2 + 14.9334z_3 - \\ & 43.0934) - 1.5079z_1(z_6^2 + z_7^2) + \\ & 7.477(z_6^3 + z_7^3) + 0.7854(z_4z_6^2 + z_5z_7^2) + \\ & (1000 + 10^8 k_3)[(z_1 - x_{11}^*)^2 + \\ & (z_2 - x_{12}^*)^2 + (z_3 - x_{13}^*)^2 + \\ & (z_1 - x_{21}^*)^2 + (z_2 - x_{22}^*)^2 + \\ & (z_3 - x_{23}^*)^2 + (z_5 - x_{25}^*)^2 + \\ & (z_7 - x_{27}^*)^2 + (z_1 - x_{31}^*)^2 + \\ & (z_2 - x_{32}^*)^2 + (z_3 - x_{33}^*)^2 + \\ & (z_4 - x_{34}^*)^2 + (z_6 - x_{36}^*)^2] \end{aligned} \quad (9)$$

其中： $z_1 \sim z_7$ 为系统级设计变量； $x_{11}^* \sim x_{13}^*, x_{21}^* \sim$

$x_{23}^*, x_{25}^*, x_{27}^*, x_{31}^* \sim x_{34}^*$ 和 x_{36}^* 为学科级返回给系统级的优化结果。

子学科 1

$$\begin{aligned} \min J_1(x_1) = & (x_{11} - z_1^*)^2 + (x_{12} - z_2^*)^2 + \\ & (x_{13} - z_3^*)^2, \\ \text{s.t. } & g_j = 0, j = 1, 2, 7, 8, 9. \end{aligned} \quad (10)$$

其中： $x_{11} \sim x_{13}$ 为子学科 1 的设计变量， $z_1^* \sim z_3^*$ 为系统级分配给该学科的设计变量期望值。

子学科 2

$$\begin{aligned} \min J_2(x_2) = & (x_{21} - z_1^*)^2 + (x_{22} - z_2^*)^2 + \\ & (x_{23} - z_3^*)^2 + (x_{25} - z_5^*)^2 + \\ & (x_{27} - z_7^*)^2, \\ \text{s.t. } & g_j = 0, j = 1, 2, 4, 6, 7, 8, 9, 11. \end{aligned} \quad (11)$$

其中： $x_{21} \sim x_{23}, x_{25}$ 和 x_{27} 为子学科 2 的设计变量； $z_1^* \sim z_3^*, z_5^*$ 和 z_7^* 为系统级分配给该学科的设计变量期望值。

子学科 3:

$$\begin{aligned} \min J_3(x_3) = & (x_{31} - z_1^*)^2 + (x_{32} - z_2^*)^2 + \\ & (x_{33} - z_3^*)^2 + (x_{34} - z_4^*)^2 + \\ & (x_{36} - z_6^*)^2, \\ \text{s.t. } & g_j = 0, j = 1, 2, 3, 5, 7, 8, 9, 10. \end{aligned} \quad (12)$$

其中： $x_{31} \sim x_{34}$ 和 x_{36} 为子学科 3 的设计变量， $z_1^* \sim z_4^*$ 和 z_6^* 为系统级分配给该学科的设计变量期望值。

优化结果对约束条件的满足程度，可以作为其可行性能的评价标准。表 1 对本文优化结果与文献[10]和[12]中给出的优化结果进行了约束条件满足程度的比较。其中： x_1, x_2, x_3 分别为表 2 中 3 个初始点的优化结果； x_4 为文献[10]给出的优化结果， x_5 为文献[12]给出的优化结果。

4.3 优化结果及分析

从 3 个不同起始点进行两级优化计算，其计算

表 1 约束条件值比较

优化结果	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}	g_{11}
x_1	< 0	< 0	< 0	< 0	= 0	< 0	< 0	< 0	< 0	< 0	= 0
x_2	< 0	< 0	< 0	< 0	< 0	= 0	< 0	< 0	< 0	< 0	= 0
x_3	< 0	< 0	< 0	< 0	< 0	= 0	< 0	< 0	< 0	< 0	= 0
x_4	< 0	< 0	< 0	< 0	0.0019	0.0030	< 0	< 0	< 0	= 0	< 0
x_5	< 0	< 0	< 0	< 0	0.0002	< 0	< 0	0.0012	< 0	= 0	< 0

表 2 优化结果

起始点	设计变量值	优化结果	目标函数值	迭代次数
1	0, 1, 12, 7, 1, 3, 3	3.5010, 0.7001, 17.0000, 7.3000, 7.7154, 3.3502, 5.2868	29954	16
2	2.0, 0.4, 9.0, 4.0, 4.5, 1.5, 3.0	3.5017, 0.7001, 17.0000, 7.3001, 7.7156, 3.3505, 5.2867	2995.7	15
3	3.5, 0.7, 17, 7.3, 7.715, 3.35, 5.287	3.5002, 0.7000, 17.0000, 7.3001, 7.7154, 3.3505, 5.2867	2994.7	5

结果如表 2 所示. 该问题^[12]的标准优化解为 $x^* = [3.5, 0.7, 17, 7.3, 7.71, 3.35, 5.29]$, $f(x^*) = 2.996$. 文献^[10]利用动态松弛算法给出的优化结果为 $x^* = [3.5, 0.7, 17, 7.3, 7.7151, 3.3481, 5.2814]$, $f(x^*) = 2.990.5$. 与表 2 中的优化结果相比, 动态松弛算法给出的目标函数值更小些. 但动态松弛算法的起始点不宜在可行域内取值, 因其起始点在可行域内时, 子学科间不一致信息为零, 系统级一致性约束又变为等式约束, 松弛算法失效. 基于动态罚函数法的 CO 的起始点在可行域内外均可取值, 动态罚因子中 b 的取值在一定程度上增强了该算法的稳定性.

从表 1 中的分析数据可以看出, 本文的优化结果满足原问题的全部约束条件. x_4 引起的 g_5 和 g_6 的误差在 10^{-3} 数量级上, x_5 引起的 g_5 的误差在 10^{-4} 数量级上, g_8 的误差在 10^{-3} 数量级上. 通过约束条件满足程度的比较可以看出, 与其他两种优化算法相比, 基于动态罚函数法的 CO 的运算结果的可行度最高, 更能满足工程应用中的精度要求. 目标函数和学科间不一致信息变化曲线如图 1 和图 2 所示, 从图中可以看出, 该算例具有良好的收敛性.

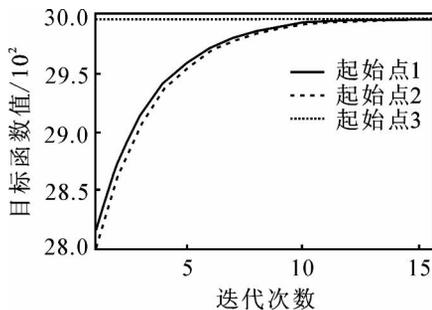


图 1 目标函数变化曲线

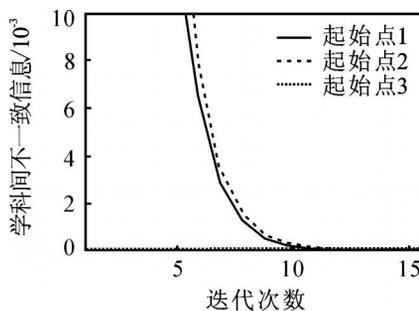


图 2 学科间不一致信息变化曲线

5 结 论

本文从保证 CO 的系统级优化可行的角度, 采用动态罚函数法, 将学科间一致性约束条件下的系统级优化问题转变为无约束优化问题. 对已有学科间不一致信息的定义形式进行了分析, 提出了另外两种应用更为普遍的定义形式, 并对几种定义形式

进行了分析比较. 利用学科间不一致信息, 构造了动态罚因子表达式, 式中的常数项 b 可增强算法的稳定性, 减少对初始点选取的敏感性. 采用遗传算法作为系统级的优化算法, 不但增强了算法的可靠性, 而且对其目标函数与学科一致性约束的连续性无要求. 在减速器优化算例中, 采用学科间不一致信息的定义 k_3 构造罚因子, 数据结果表明, 该算法的计算精度较高, 收敛性较好, 同时具有较好的稳定性.

参考文献(References)

- [1] Kroo I, Altus S, Braun R, et al. Multidisciplinary optimization methods for aircraft preliminary design[C]. The 5th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization. Panama, 1994: 697-707
- [2] Olivier D W, Jeremy A, Jaroslaw Sobieski, et al. State-of-the-art and future trends in multidisciplinary design optimization[C]. The 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conf. Waikiki, 2007: 2467-2487.
- [3] Alexandrov N M, Lewis R M. Analytical and computational aspects of collaborative optimization[R]. Hampton: National Aeronautics and Space Administration, 2000.
- [4] Balling R J, Wilkision C A. Execution of multidisciplinary design optimization approaches on common test problems[J]. AIAA Journal, 1997, 35(1): 178-186.
- [5] 郭红霞, 吴捷, 黄飞龙. 基于蚁群算法的 MAS 多目标协调优化[J]. 控制与决策, 2007, 22(8): 946-950. (Guo H X, Wu J, Huang F L. Multi-objective coordinated optimal of MAS based on ant system[J]. Control and Decision, 2007, 22(8): 946-950.)
- [6] Alexandrov N, Lewis R M. Analytical computational aspects of collaborative optimization for multidisciplinary design[J]. AIAA Journal, 2002, 40(2): 301-309.
- [7] De Miguel, Angel Victor. Two decomposition algorithms for nonconvex optimization problems with global variables[D]. California: Stanford University, 2001.
- [8] Sobieski L P, Kroo L. Collaborative optimization using response surface estimation[R]. Reno: Aerospace Sciences Meeting and Exhibit, 1998.
- [9] Cormier T A, Scott A, Ledsinger L A, et al. Comparison of collaborative optimization to conventional design techniques for conceptual RL V[C]. Proc of 8th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. California, 2000.

(下转第 920 页)

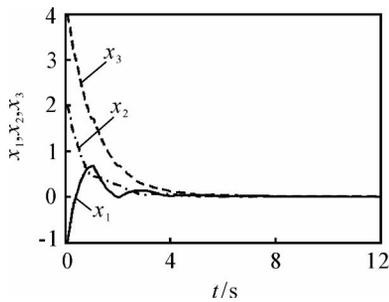


图 4 第 4 组故障矩阵下切换系统的状态响应曲线
其余参数同上. 仿真结果如图 3 和图 4 所示.

5 结 论

本文研究了一类非线性不确定切换系统的鲁棒容错控制问题. 当执行器完全失效或部分失效时, 利用 Lyapunov 函数法给出了混杂状态反馈容错控制器的设计, 使得闭环系统对所有可能的不确定性和所有允许的执行器失效渐近稳定. 在工程实际中, 子系统的各部分故障是否会对整体系统造成同样严重的影响以及如何量化等问题, 值得进一步研究.

参考文献 (References)

- [1] 洪晓峰, 孙洪飞. 切换系统容错控制的研究[J]. 厦门大学学报, 2007, 46(2): 183-186.
(Hong X F, Sun H F. Study on fault-tolerant control of switched systems[J]. J of Xiamen University, 2007, 46(2): 183-186.)
- [2] 赵军, 金刚. 一类不确定非线性切换系统的鲁棒容错控制[J]. 东北大学学报, 2004, 25(3): 209-211.
(Zhao J, Jin G. Robust fault-tolerant control for a class of uncertain switched nonlinear systems [J]. J of Northeastern University, 2004, 25(3): 209-211.)
- [3] Xiaofeng Hong, Housheng Xu, Hongfei Sun. Fault-tolerant control of discrete-time switched systems [C]. 2007 IEEE Int Conf on Control and Automation. Piscataway: IEEE, 2007: 3082-3086.
- [4] Rui Wang, Gang Jin, Jun Zhao. Robust fault-tolerant control for a class of switched nonlinear systems in lower triangular form[J]. Asian J of Control, 2007, 9(1): 68-72.
- [5] Rodrigues M, Theilliol D, Sauter D. Fault-tolerant control design for switched systems[C]. ADHS '06 2nd IFAC Conf on Analysis and Design of Hybrid Systems. Italy: IFAC, 2006: 223-228.
- [6] Wang Z, Huang B, Unbehauen H. Robust reliable control for a class of uncertain nonlinear state-delayed systems[J]. Automatica, 1999, 35(5): 955-963.
- [7] 王福忠, 姚波, 张嗣瀛. 线性系统区域稳定的可靠控制[J]. 控制理论与应用, 2004, 21(5): 835-839.
(Wang F Z, Yao B, Zhang S Y. Reliable control of regional stabilizability for linear systems [J]. Control Theory & Applications, 2004, 21(5): 835-839.)
- [8] Yang Y, Yang G H, Soh Y C. Reliable control of discrete-time systems with actuator failure[J]. IEE Proc Control Theory Applications, 2000, 147(4): 428-432.
- [9] Xie S, Xie L, Wang Y, et al. Decentralised control of multi-machine power systems with guaranteed performance [J]. IEE Proc Control Theory Applications, 2000: 147(3): 355-365.
- [10] Petersen I R, Hollot C V. A Riccati equation to the stabilization of uncertain linear systems [J]. Automatica, 1986, 22(4): 397-411.
- [11] 俞立. 鲁棒控制——线性矩阵不等式处理方法[M]. 北京: 清华大学出版社, 2002.
(Yu L. Robust control — Linear matrix inequality methods [M]. Beijing: Tsinghua University Press, 2002.)

(上接第 915 页)

- [10] 李响, 李为吉. 利用协同优化方法实现复杂系统分解并行设计优化[J]. 宇航学报, 2004, 25(3): 300-304.
(Li X, Li W J. A new collaborative optimization algorithm and its applications to complex system parallel design [J]. Acta Aeronautica et Astronautica Sinica, 2004, 25(3): 300-304.)
- [11] Srinivas Kodiyalam. Evaluation of methods for multidisciplinary design optimization (MDO) [R]. Washington: National Aeronautics and Space Administration, 1998.
- [12] Azam S, Li W C. Multi-level design optimization using global monotonicity analysis [J]. ASME J of Mechanisms and Automation in Design, 1989, 111(2): 259-263.