

一种线性系统可重构控制分析方法

段文杰^{1,2} 王大轶^{1,2} 刘成瑞^{1,2}

摘要 提出了一种线性系统在线或者离线的可重构控制分析方法, 该方法基于功能目标模型, 能够定性分析线性系统的可重构控制问题, 包括发生多个故障时是否具有可重构能力, 采用哪些组件和何种控制方法, 以及重构后系统是否能达到期望的控制目标等. 首先定义了功能、目标、最小重构单元状态、可行集等概念, 并基于这些概念建立系统功能目标模型. 该模型由功能目标关系和各个目标的可行集组成. 总目标的可行集为系统顶层可行集, 可重构控制方案的选择基于顶层可行集. 应用本文方法, 离线建立起控制系统的功能目标模型后, 可以在线或离线分析其多种故障模式下的可重构问题, 还可以用于指导可重构性设计. 最后, 给出一个卫星控制系统可重构控制的例子.

关键词 可重构性, 航天器, 冗余, 功能目标模型

引用格式 段文杰, 王大轶, 刘成瑞. 一种线性系统可重构控制分析方法. 自动化学报, 2014, 40(12): 2726–2736

DOI 10.3724/SP.J.1004.2014.02726

An Analysis Method for Reconfigurable Control of Linear Systems

DUAN Wen-Jie^{1,2} WANG Da-Yi^{1,2} LIU Cheng-Rui^{1,2}

Abstract A function-objective model is proposed as a description of a controlled system. Based on this model, reconfigurable control problems of linear systems are analyzed, including whether the whole system can achieve reconfigurable control, which components should be selected, what control scheme should be adopted, whether the system can reach the desired control objectives, etc. Firstly, some concepts are defined such as function, objective, minimal reconfigurable unit condition, feasible set, etc. Then, based on these definitions, the function-objective model is established with the logical relation between functions and objectives and feasible sets of all objectives as its two main aspects. The reconfigurable control scheme can be determined by selecting a suitable feasible configuration in the top feasible sets, which are the feasible sets of top objectives. After the establishment of all the feasible sets offline, we can analyze the system reconfigurability offline or online. The method proposed can also be used to guide the design stage of controlled systems. Finally, a satellite model is given to illustrate the method.

Key words Reconfigurability, spacecraft, redundancy, function-objective model

Citation Duan Wen-Jie, Wang Da-Yi, Liu Cheng-Rui. An analysis method for reconfigurable control of linear systems. *Acta Automatica Sinica*, 2014, 40(12): 2726–2736

当前, 对飞机^[1–2]、卫星^[3–4]等系统和相关设备的可靠性、自主性和稳定性要求越来越高, 因此, 对故障的检测隔离和系统的可重构性要求也越来越高.

对于主动容错控制, 目前研究主要集中在故障的检测、诊断 (Fault detection and isolation, FDI) 和系统的可重构控制算法 (Reconfigurable control, RC) 两个方面^[5]. 研究的目的是发生故障时能正确检测故障、隔离故障, 并在有需要时对故障进行进一步的诊断; 然后在 FDI 的基础上设计可重构控制方

法, 使得系统性能维持在可接受的范围内. 此时, 要求系统具有充分的冗余, 包括硬件冗余、软件冗余和解析冗余. 当故障发生后, 经过诊断和隔离, 重新分配, 仍然可以实现预定的控制目标或者保证系统安全运行. 故障后安全的控制系统可以分为以下三种情况^[6]: 1) 系统继续运行, 性能不受影响; 2) 系统继续运行, 此时主要功能保持正常; 3) 系统功能受限, 但是系统安全. 系统是否可重构以及重构后是否能达到预定的性能是本文可重构性分析的重点.

系统可重构性描述的是系统对故障的处理能力, 指在资源配置和运行条件一定的情况下, 在维持系统安全的时间内, 系统从故障恢复到既定功能的能力. 其中, 既定功能为系统的全部功能或者可接受的部分功能. 如果故障系统仍然能控能观, 此时认为系统是可重构的. 文献 [7] 根据线性系统的能控性和能观性给出多种目标下的可重构性条件.

文献 [6] 采用了一种功能分析的定性模型研究系

收稿日期 2014-02-25 录用日期 2014-07-16
Manuscript received February 25, 2014; accepted July 16, 2014
国家自然科学基金 (61203093, 11202011) 资助
Supported by National Natural Science Foundation of China (61203093, 11202011)
本文责任编辑 贾英民
Recommended by Associate Editor JIA Ying-Min
1. 北京控制工程研究所 北京 100190 2. 空间智能控制技术重点实验室 北京 100190
1. Beijing Institute of Space Technology, Beijing 100190 2. Science and Technology on Space Intelligent Control Laboratory, Beijing 100190

统的容错控制过程, 该过程包括了故障诊断和容错控制方案的生成. 定义 $\lambda_1 > \lambda_2 > \dots > \lambda_n$ 是系统格拉姆矩阵的特征值, 文献 [8] 采用系统最小二阶模态的大小描述系统可重构性程度, 将系统的最小二阶模态看成执行机构安装角的函数, 通过仿真确定该变量的取值对系统可重构性的影响程度. 实际系统是否可重构与系统组件运行状态和系统重构目标有关. 文献 [9] 通过定义系统服务 (Service)、版本 (Edition) 与运行模式 (User-selected operating mode, USOM) 等构建了分析系统是否可重构的一般组件模型. 该模型是一种金字塔形式的功能逻辑关系树 (AND/OR tree). 系统发生故障时, 通过分析系统是否有可行“版本”, 分析故障系统是否可重构. 文献 [10–11] 介绍了结构分析的一些基本定义. 文献 [12] 采用有向图方法分析了系统的容错控制问题.

系统故障导致系统丧失某个功能; 某个故障是否具有可重构性, 取决于相应功能是否具有冗余配置. 当系统某些功能发生故障时, 利用冗余配置承担故障组件的工作, 可以减少系统的故障时间, 保证系统更加可靠、安全的工作. 冗余技术通常包括硬件冗余、软件冗余和解析冗余. 系统受限于成本、重量、性能等因素, 不能保证对每个部件都实现充分冗余. 当系统关键部件发生故障时, 系统关键功能可能因此而丧失, 此时可重构控制算法无能为力. 因此, 研究组件和整个系统的可重构性是至关重要的问题.

现有的研究主要集中在故障发生后的诊断和控制算法设计, 而我们希望设计一种定性方法不仅能在故障后分析系统可重构性, 还能在故障前甚至系统设计阶段分析系统的可重构控制问题, 包括多故障时是否可重构, 采用哪些组件和什么重构控制方法, 以及重构后能否达到预定的控制目标等.

为了解决以上的实际问题, 我们首先定义了功能、目标、最小重构单元状态、可行集等概念, 并基于这些概念建立系统功能目标模型. 应用该模型可以在线或离线分析控制系统的可重构控制问题, 还可以用于指导可重构性设计.

本文的结构如下: 第 1 节介绍本文涉及到的基本概念和方法; 第 2 节介绍功能目标模型的建模方法; 第 3 节基于功能目标模型分析控制系统可重构控制问题; 第 4 节以一个卫星系统为对象, 给出本方法的应用实例; 第 5 节为全文总结.

1 方法及概念介绍

包含控制分配的主动容错控制原理如图 1^[13] 所示, 控制器设计出的虚拟控制律通过控制分配发送到各个执行机构, 作用于系统. 控制分配则需要故障诊断和辨识得到的执行机构健康状况信息. 控制器,

量测, 故障诊断等各个部分发生故障都可能导致系统丧失预定的功能.

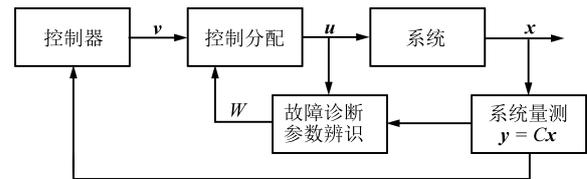


图 1 容错控制流程图

Fig. 1 Fault-tolerant control flowchart

本文分析线性系统可重构性, 标称对象的状态空间描述如下:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \quad (1)$$

其中, $\mathbf{A} \in \mathbf{R}^{n_A \times n_A}$, $\mathbf{B} \in \mathbf{R}^{n_A \times n_u}$, $\mathbf{C} \in \mathbf{R}^{n_C \times n_A}$, $\mathbf{x} \in \mathbf{R}^{n_A}$, $\mathbf{y} \in \mathbf{R}^{n_C}$, 当系统某些组件故障时, 故障的系统表示为

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}_f\mathbf{u} \\ \mathbf{y} = \mathbf{C}_f\mathbf{x} \end{cases} \quad (2)$$

本文中故障系统 (2) 是否可重构的判断依据是系统的能控性、能观性、组件间逻辑关系和目标约束. 此时能控能观性的判断可以根据能控能观格拉姆矩阵的乘积^[8], 即 $W_C W_O$ 奇异时, 系统不能重构.

系统配置、重构构型/模式、控制策略、重构目标、组件状态等都是影响系统可重构性的因素. 为了分析故障系统的可重构能力及可行的可重构方案, 本文采用可重构性模型分析系统的特性. 建立本文提出的功能目标模型, 需要对系统进行结构分解和功能分解, 下面定义了最小重构单元状态、功能、目标、可行集等概念.

1.1 结构分解相关概念

结构分解是工程上常用的方法, 可以按照系统 \rightarrow 分系统 $\rightarrow \dots \rightarrow$ 重构单元 \rightarrow 最小重构单元的从顶向下的顺序将系统分解.

1) 重构单元 (Reconfigurable unit, RU): 当系统中多个单元组合实现某一功能时, 该组合即构成一个重构单元. 如敏感机构中的所有陀螺组合, 其在功能上可以实现卫星三轴方向上的姿态角速度的测量, 故可将所有陀螺视为一个重构单元. 当该重构单元发生某一故障时, 可以通过对陀螺组合的内部重构实现对该故障的屏蔽, 使其不影响系统姿态角速度的正确输出.

2) 最小重构单元 (Minimal reconfigurable unit, MRU): 不能利用自身资源进行重构, 而只能通过其他单元替换来恢复系统功能的单元.

此时含 n_p 个最小重构单元的重构单元 j 为

$$RU_j = \sum MRU_i, \quad i = 1, \dots, n_p \quad (3)$$

包含 n 个最小重构单元的系统为

$$S = \sum MRU_i, \quad i = 1, \dots, n \quad (4)$$

3) 最小重构单元状态 (Minimal reconfigurable unit condition, MRUC): 每个最小重构单元可以有几种不同的状态, 比如未启动、正常、完全故障、比例故障^[13-14] 等, 系统在线运行时, 需要实时诊断出各个最小重构单元的状态. 将最小重构单元 t 时的状态定义为最小重构单元状态, 表示为 $MRUC(t)$.

对于包含多个最小功能的最小重构单元, 其故障形式可能是某个或者某几个功能受限, 此时, 为了更清晰地表明系统的可行配置, 需要分析所有的最小重构单元状态, 多种状态之间的逻辑关系为异或. 异或门如图 2 所示, Y 的实现要求有且只有一个 x 实现, x_i 与 x_j 不能同时实现, 数学表达式为 $Y = x_p$ ($1 \leq p \leq N$).

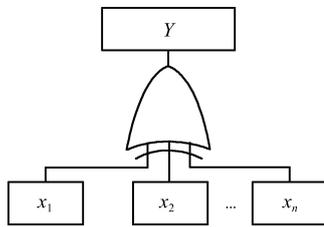


图 2 异或门
Fig. 2 XOR gate

以一个动量轮 (MRU_j) 为例, 预设以下四种状态: 关机 ($c1$)、正常 ($c2$)、完全故障 ($c3$)、比例故障 ($c4$). 将四种状态划分为四个虚单元, 其逻辑关系为异或, 如图 3 所示. 在线工作的动量轮状态为四种虚单元中的一个. 如果 $c1 \sim c4$ 都不能描述此动量轮的状态, 则该状态为非预期状态, 可以根据在线诊断的情况按照 $c3$ 或者 $c4$ 处理.

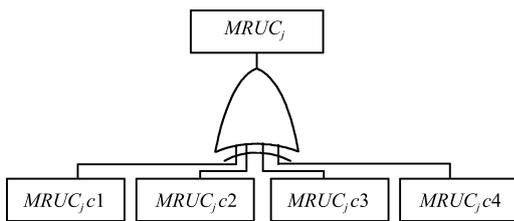


图 3 动量轮最小重构单元状态图
Fig. 3 MRUC of a reaction wheel

对于一个包含 n 个最小重构单元的在轨卫星, 在 t 时刻, 在线诊断得到的系统可以表示为

$$S_t = \sum MRUC_i(t), \quad i = 1, \dots, n \quad (5)$$

4) 故障模式集 (Fault set, FauS): 系统所有的最小重构单元故障模式组成集合, 称为故障模式集. 假设第 j 个最小重构单元 MRU_j 的故障模式数为 $p_{f,j}$, 则

$$FauS = \{fau_{j,i} | i = 1, \dots, p_{f,j}; j = 1, \dots, n\} \quad (6)$$

1.2 功能分解相关概念

本文的功能结构模型首先在功能层下定义了目标层, 将功能按照精度、性能或可靠性等指标划分为不同目标; 然后以最小重构单元状态为元素建立各个目标的可行集. 应用该模型的优点是可以清晰表明不同的目标下的可行配置和多种故障后系统能实现的目标, 进而选择各种故障后的可重构控制方案. 搭建功能目标模型之前, 先定义以下概念.

1.2.1 功能中的逻辑关系

不同的功能模块之间的逻辑关系主要有与门、或门和异或门. 与门如图 4 (a) 所示, 所有子功能 x_i 实现时, 上层功能 Y 才实现. 串联模型是一种典型的与门关系. 与门的数学表达式为 $Y = x_1 \cap x_2 \cap \dots \cap x_n$. 或门如图 4 (b) 所示, 任一子功能 x_i 实现时, 上层功能 Y 就能够实现, 代表模型是并联模型, 数学表达式为 $Y = x_1 \cup x_2 \cup \dots \cup x_n$. 异或门如图 2 所示.

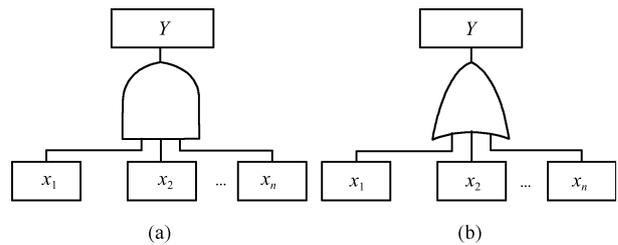


图 4 与门和或门
Fig. 4 AND gate and OR gate

1.2.2 功能与目标

1) 功能 (Function): 功能是部件或者系统实现的作用效果. 最小重构单元实现的功能称为最小功能 (Minimal function, MF). 一个最小重构单元可以有多个最小功能, 最小重构单元 MRU_j 的第 k 个最小功能 $F_{j,k}$ 可以由输入 ($inpt_{j,k}$)、输出 ($outp_{j,k}$) 和部件 (MRU_j) 表示如下:

$$F_{j,k} : outp_{j,k} = f_{j,k}(inpt_{j,k}, MRU_j) \quad (7)$$

其中, $f_{j,k}$ 有时是逻辑关系, 没有表达式. 本文中所有功能 (包括系统总功能和重构单元功能) 都包含输

入、输出和部件三个参数。

2) 功能域 (F): 对系统所有的最小重构单元, 依次分析其功能, 得到最小功能集合称为功能域, 如式 (8) 所示。

$$F = \{F_i | i = 1, 2, \dots, p_F\} \quad (8)$$

其中的每个元素被称为一个最小功能, F_i 为系统中的第 i 个最小功能, p_F 为系统功能包含的最小功能数。为了便于分析, 本文的功能域中, 当两个最小功能的作用效果相同时, 即输出相同时, 两个最小功能视为相同功能。

3) 目标 (O): 最小功能 F_j 按照用户划分的性能、精度、时间、能耗、可靠性等指标定义为 p_j 个目标 $\{O_{F_j,1}, O_{F_j,2}, \dots, O_{F_j,p_j}\}$ 。 F_j 的第 i 个目标可以表示为

$$O_{F_j,i} : g_{F_j,i} \left(inpt_{F_j}, outp_{F_j}, \sum MRU_k \right) \geq l_{F_j,i} \quad (9)$$

其中, $g_{F_j,i}$ 为用户和系统要求的指标, $l_{F_j,i}$ 为下界。每个目标的实现都能保证该层功能的实现, 同一层的目标可以为并列关系, 也可以为包含关系, 可以分为不同的级别。如式 (10) 所示, 将功能 F_j 划分为如下目标:

$$\left\{ \left\{ O_{F_j,1}, \dots, O_{F_j,p_1} \right\}, \left\{ O_{F_j,p_1+1}, \dots, O_{F_j,p_1+p_2} \right\} \right\} \quad (10)$$

4) 高级目标: $\{O_{F_j,1}, \dots, O_{F_j,p_1}\}$ 中任何一个目标的实现都能保证 $\{O_{F_j,p_1+1}, \dots, O_{F_j,p_1+p_2}\}$ 的实现, 则称 $\{O_{F_j,1}, \dots, O_{F_j,p_1}\}$ 为 $\{O_{F_j,p_1+1}, \dots, O_{F_j,p_1+p_2}\}$ 的高级目标。可以用包含关系表示为

$$\{O_{F_j,1}, \dots, O_{F_j,p_1}\} \subset \{O_{F_j,p_1+1}, \dots, O_{F_j,p_1+p_2}\} \quad (11)$$

1.2.3 可行集与可行配置

1) 可行配置 (Feasible configuration, FC): $O_{F_j,i}$ 的第 k 个可行配置是使 $F_{j,i}$ 实现的第 k 个最小重构单元状态组合, 表示为

$$FC_{O_{F_j,i},k} = \sum MRES_p \quad (12)$$

2) 可行集 (Feasible set, FS): $O_{F_j,i}$ 的可行集是所有使得 $O_{F_j,i}$ 成立的 $p_{F_j,i}$ 个可行配置的集合, 记为

$$FS_{F_j,i} = \{FC_{O_{F_j,i},k} | k = 1, \dots, p_{F_j,i}\} \quad (13)$$

3) 继承 (Inheriting): 当 $O_{F_j,i} \subset O_{F_j,k}$ 时, 可行集之间存在关系:

$$FS_{F_j,i} \subset FS_{F_j,k} \quad (14)$$

此时, 称 $FS_{F_j,k}$ 继承了 $FS_{F_j,i}$ 的全部可行配置。

对于图 2 和图 4 所示的逻辑关系, 假设 $n = 2$, 每个 x_j 有成立和不成立两种状态, 表示为 $x_{j,1}$ 和 $x_{j,0}$ 。对于与门、或门和异或门, Y 的可行集分别为式 (15)~(17):

$$FS_Y = \{\{x_{1,1}, x_{2,1}\}\} \quad (15)$$

$$FS_Y = \{\{x_{1,1}, x_{2,1}\}, \{x_{1,1}, x_{2,0}\}, \{x_{1,0}, x_{2,1}\}\} \quad (16)$$

$$FS_Y = \{\{x_{1,1}, x_{2,0}\}, \{x_{1,0}, x_{2,1}\}\} \quad (17)$$

2 建立功能目标模型

前面定义了最小重构单元状态、功能、目标等基本概念, 这一节介绍如何建立系统的功能目标模型 (Function objective model, FOM), 主要分为四步, 下面将详细介绍。

2.1 结构分解

前面已经介绍过结构分解的相关概念。结构分解是建立功能目标模型的第一步: 1) 首先将系统结构分解, 得到最小重构单元; 2) 依次分析每个最小重构单元的故障模式, 建立故障模式集; 3) 依次分析每个最小重构单元, 列出所有可能的最小重构单元状态。

2.2 建立功能逻辑关系

最小功能是功能逻辑关系图 (Function map, FM) 的最底层功能。第 1.2 节中已经定义了最小重构单元功能、最小功能、功能域等。为了建立功能逻辑关系, 还需要定义系统功能及子功能:

1) 总功能 (M_0): 功能逻辑关系的最顶层功能, 表示为

$$M_0 : outp_0 = f_0(inpt_0, S) \quad (18)$$

2) k 级子功能 ($M_{k,j}$): 功能逻辑关系中第 k 层的第 j 个功能, 如式 (2) 所示, 其中 $\sum MRU_i$ 为该功能相关的所有最小重构单元。

$$M_{k,j} : outp_{M_{k,j}} = f_{M_{k,j}} \left(inpt_{M_{k,j}}, \sum MRU_i \right) \quad (19)$$

本文的功能逻辑关系中, 功能的命名规则为: 顶层为第 0 层, 往下分别命名为 1, 2, \dots , n 层。第 k 层的功能从左到右分别命名为 $M_{k,1}, M_{k,2}, \dots, M_{k,p}$ 。

对于组件少的系统, 很容易建立功能逻辑关系图。对于一般系统, 建立功能逻辑关系可以通过三步实现: 1) 建立功能域 F ; 2) 建立最小功能有向图; 3) 建立系统的功能逻辑关系图。下面具体介绍这三个步骤。

2.2.1 建立功能域

依次分析每个最小重构单元的功能, 确定最小功能集合, 可得到系统功能域 F . 最小重构单元功能和功能域的定义见式 (7) 和式 (8).

2.2.2 建立输入输出有向图

本文中, 将有向图^[12] 用于描述输入输出关系, 并给出如下操作步骤:

- 1) 列出所有最小功能的输入输出关系;
- 2) 根据输入输出关系, 用箭头连接所有最小功能. 连线表示有信息在最小功能间传递, 箭头表示信息传递的方向;
- 3) 将输入输出连线相同的最小功能画到一个框内.

下面结合实例说明有向图的建立步骤. 例如一个控制系统的功能域为 $\{F_1, F_2, F_3, F_4, F_5\}$, 首先列出输入输出关系, 得到表 1, 其中 CB 代表被控对象, “&” 表示与门, “OR” 表示或门.

表 1 输入输出表
Table 1 Input-output list

对象	inpt	outp
F_1	$x_1 \& x_2$	y_1
F_2	$x_1 \& x_2$	y_2
F_3	$y_1 \text{ OR } y_2 \text{ OR } y_5$	y_3
F_4	$y_1 \& y_2 \& y_3$	y_4
F_5	$x_1 \& x_2$	y_5
CB	y_4	$x_1 \& x_2$

然后根据输入输出关系, 用箭头连接各个最小功能, 得到最小功能有向图, 如图 5 所示. 其中, F_1 和 F_2 的输入输出关系相同, 所以画在一个框内. 一个框内的最小功能之间的关系, 由最小功能的定义给出.

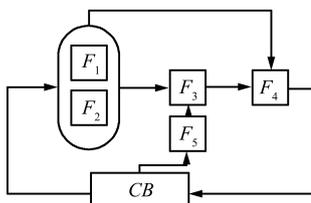


图 5 最小功能有向图
Fig. 5 Directed map of MRU

2.2.3 建立系统功能逻辑关系图

分析最小功能的有向图, 合并功能, 就可以搭建出系统的功能逻辑关系. 可以按照如下的步骤进行:

- 1) 建立系统功能逻辑关系图

将有向图一个框内的功能合并, 定义为 M_j , $M_j = F_{p_j} \cup \dots \cup F_{q_j}$:

$$outp_j = f_{M_j} \left(inpt_{M_j}, \sum MRU_i \right) \quad (20)$$

$F_{p_j} \sim F_{q_j}$ 为功能 M_j 的下层功能, 并画出它们之间的逻辑关系图. 通过如此合并, 可得 M_1, \dots, M_k 及它们各自的下层功能关系图. 例如, 由表 1 可知, $(F_1, F_2) \rightarrow F_3$ 中, F_1 和 F_2 为或门关系, 定义 $M_1 = F_1 \cup F_2$, 并绘制出逻辑关系图, 如图 6 所示.

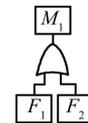


图 6 功能合并图

Fig. 6 Function incorporation

- 2) 进一步分析

进一步分析初步合并之后的控制系统功能 $M_1, \dots, M_k, \dots, F_m, \dots, F_q$ 之间的逻辑关系, 其中 F_m, \dots, F_q 为未合并的最小功能.

进一步合并功能, 最终得到的控制系统由 M_1, \dots, M_p 和最小功能 F_{m_1}, \dots, F_{q_1} 组成, 它们之间为与门或为或门关系. 功能之间的相互关系, 如图 5 中 $(F_1, F_2) \rightarrow F_4$, 可转化为目标中的约束条件, 这点将在下文中说明.

每合并一次, 功能逻辑关系增加一层, 功能合并有以下原则: a) 符合实际意义: 例如卫星系统可以把执行机构或者敏感机构的最小重构单元的功能合并. b) 化简原则: 合并过程中尽量减少功能层次. 例如在 1) 中, $(F_1, F_2) \rightarrow F_3$, 合并得到 $M_1 = F_1 \cup F_2$, M_1 和 F_5 也为或门关系, 可定义 $M_2 = M_1 \cup F_5$. 我们还可重定义 M_1 为 $M_1 = F_1 \cup F_2 \cup F_5$ (如图 7 所示). 基于化简原则, 我们应该选择后者, 以减少功能层次. c) 合并终止: 最终得到的控制系统功能 $M_1, \dots, M_p, \dots, F_{m_1}, \dots, F_{q_1}$ 之间为与门或为或门关系时, 继续合并就可得到系统总功能 M_0 , 为合并终止的必要条件.

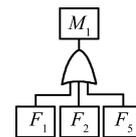


图 7 重定义的 M_1

Fig. 7 Redefined M_1

- 3) 搭建功能逻辑关系图

首先画出 M_0 和 $M_1, \dots, M_p, \dots, F_{m_1}, \dots, F_{q_1}$ 之间的功能逻辑关系图; 然后加入前面得到的各个

功能的下层功能逻辑关系; 最后按照命名规则, 重新命名各个合并得到的功能为 $M_{k,j}$ ($M_{k,j}$ 表示第 k 层第 j 个功能), 就得到了功能逻辑关系图. 图 5 中控制系统经过初步合并和进一步分析合并后, 由 M_1 、 F_3 和 F_4 组成, 为与门关系. 绘制功能逻辑关系, 然后将 M_1 按照命名规则重新命名为 $M_{1,1}$, 得到功能逻辑关系图, 如图 8 所示.

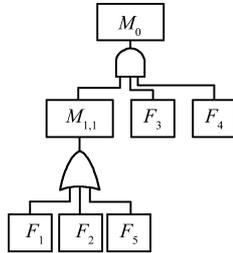


图 8 功能逻辑关系图
Fig. 8 Logic relation map

功能逻辑关系的层数由功能间的逻辑关系决定, 逻辑关系的增加将导致功能逻辑关系层数的增加. 总功能反映了控制系统的作用效果, 总功能的实现依赖于底层功能的逐层实现. 对于复杂的控制系统, 功能逻辑关系的划分不唯一, 便于分析即可.

2.3 建立功能目标关系

式 (9)~(11) 定义了最小功能的目标以及高级目标的概念. 类似地, 我们可以定义功能 $M_{k,j}$ 的第 i 个目标为 $O_{M_{k,j},i}$, 表示为

$$g_{M_{k,j},i} \left(inpt_{M_{k,j}}, outp_{M_{k,j}}, \sum MRU_p \right) \geq l_{M_{k,j},i} \quad (21)$$

根据具体的指标要求, 每个功能至少建立一个目标; 将制定的目标加入到 FM 中, 就得到了功能目标关系图, 如图 9 所示. M 的目标称为系统顶层目标. 一个目标可能是一个约束方程, 也可能是一个约束方程组. 通过式 (9) 和式 (21) 可知, 本文目标是功能的约束条件. 目标的制定没有一般性的方法, 根据具体的系统、功能和用户需求明细而定.

有时功能目标模型的同一层功能之间有逻辑关系, 需要将其转化为目标中的约束条件, 加入到这些功能的目标中. 例如, 图 5 中的 M_1 和 F_4 之间有连线, 此时顶层目标建立时, 可能要将其转化为逻辑约束加入到目标中, 而约束则需依据具体情况分析后定义.

目标制定的要求为: 能满足该层功能, 且能覆盖所有可能的情况, 以避免非预期目标; 高级目标的实现能保证低级别目标的实现, 如图 9 中的 $O_{M_0,1}$ 是 $O_{M_0,2}$ 的高级目标. 图 9 是一个简单的功能目标关

系图, 当一个系统很复杂时, 画出的关系图可能很大, 此时可以在图上画出主干, 然后在其他图中画出分支.

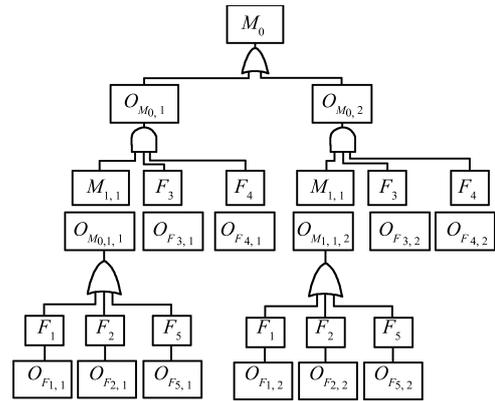


图 9 功能目标关系图
Fig. 9 Function-objective relation map

2.4 建立可行集

获得最底层目标的可行集: 首先遍历最小重构单元, 找到包含该最小功能的最小重构单元, 所有可行的最小重构单元状态组合构成的集合为底层可行集. 高层的可行集是通过底层可行集的逻辑合并运算得到, 如式 (15)~(17) 所示. 顶层目标的可行集称为顶层可行集.

顶层可行集中的可行配置的判断准则为:

1) 满足逻辑关系. 此步骤是为防止人为操作失误或者编程失误的再检验过程. 操作时从上而下带入到功能目标模型中依次检验即可. 对于多次检验无失误情况, 此步骤就可以省略.

2) 对每一个配置, 需要分析和建立系统方程 (2), 系统顶层可行集中的每个可行配置能使系统 (2) 满足能控性条件 (22) 和能观性条件 (23).

$$\text{rank} \left([B_f, AB_f, \dots, A^{n_A-1} B_f] \right) = n_A \quad (22)$$

$$\text{rank} \left(\left[C_f^T, A^T C_f^T, \dots, (A^T)^{n_A-1} C_f^T \right]^T \right) = n_A \quad (23)$$

3) 满足目标中的所有约束条件.

当控制系统的组件特别多时, 多故障分析是相当复杂的. 本文的功能结构模型方法能有效分析此种问题, 虽然组件多时计算量比较大, 但是可以通过计算机辅助建立功能结构关系中的可行集, 从而大大简化分析过程.

3 基于功能目标模型的可重构控制分析

图 1 所示系统的执行机构、系统量测和故障诊断等都可能运行中出现故障. 本文的最小重构单

元不仅包含硬件,还包含软件,如控制方法和故障诊断等,用于分析软硬件故障时系统的重构问题.

上一节建立了系统的功能目标模型,该模型通过目标和可行集,将功能、目标、系统配置、控制策略和组件状态结合在一起.本文的方法可以有效在线或者离线分析系统发生多种故障时是否具有可重构能力,采用何种重构策略,采用哪些组件以及重构后系统是否能达到预定的控制目标.

应用本文功能目标模型方法,可以通过顶层可行集离线分析系统可重构控制问题.分析多故障是否可重构时,只需在顶层可行集中找到包含当前系统所有最小重构单元状态可行配置.如果故障之前系统处于目标 $O_{M_0,i}$ 中,且有 $O_{M_0,i} \subset O_{M_0,j}$; 发生故障后,原目标 $O_{M_0,i}$ 的可行集 $FS_{M_0,i}$ 中没有含当前状态的可行配置,而 $FS_{M_0,j}$ 中有可行配置.此时可重构控制策略只能选择 $FS_{M_0,j}$ 中的可行配置,为降级使用.如果当前 $FS_{M_0,i}$ 中有包含当前状态的可行配置,此时系统可重构且重构后的系统仍然能满足预定的控制目标.

如果将本文方法用于控制系统在线可重构控制分析,需要实时在线诊断出系统所有最小重构单元状态,得到 S_t . 选择可行配置时,在离线建立的顶层可行集中寻找包含当前状态的配置,并且根据系统当前状态和能控能观条件重新判断该配置是否可行.

比例故障是本文涉及到的一种故障形式.比如动量轮作为执行机构时,如果电压供给、转轴摩擦等发生异常,都会导致动量轮的输出与实际值发生偏离.文中没有考虑动量轮其他故障形式,比如产生偏置力矩或者动量轮卡死等.应用处于比例故障状态的 $MRUC_j$,则要求 FDI 不仅能检测出故障,还能实时在线辨识出最小重构单元的健康因子 $w_{j,k}$. 此时,有

$$F_{j,k}: \text{outp}_{j,k} = w_{j,k} f_{j,k}(\text{inpt}_{j,k}, MRU_j) \quad (24)$$

对于健康因子已知的最小重构单元,由式(24)知其函数关系,可以按照健康的最小重构单元处理,但需要考虑到实际的输出能力变化.

4 卫星控制系统实例

卫星姿态控制系统由 4 个动量轮、3 个陀螺、2 个星敏感器、控制器和诊断部分组成.为了分析更简便,动量轮 1 和动量轮 2 考虑比例故障和完全故障,其他组件则只考虑完全故障.

4.1 数学模型

航天器系统的动力学模型为

$$I_x \ddot{\phi} + [(I_y - I_z) \omega_0^2 - \omega_0 h_y(t)] \phi +$$

$$[(I_y - I_z - I_x) \omega_0 - h_y(t)] \dot{\phi} - \dot{h}_x(t) + \omega_0 h_z(t)$$

$$I_y \ddot{\theta} + h_x(t) (\dot{\phi} + \omega_0 \phi) - h_z(t) (\dot{\phi} - \omega_0 \phi) = -\dot{h}_y(t)$$

$$I_z \ddot{\psi} + [(I_y - I_x) \omega_0^2 - \omega_0 h_y(t)] \varphi - [(I_y - I_z - I_x) \omega_0 - h_y(t)] \dot{\phi} - \dot{h}_z(t) - \omega_0 h_x(t) \quad (25)$$

其中, $\mathbf{h}(t) = \mathbf{h}_0 + \int_{t_0}^t \dot{\mathbf{h}}(t) \text{d}\tau$.

式(25)写成状态方程形式为

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A_1^{-1} (A_2 + \Delta A) \mathbf{x}(t) + \\ &A_1^{-1} B_1 \mathbf{v}_1(t) + A_1^{-1} \boldsymbol{\xi}_1(t) = \\ &A \mathbf{x}(t) + B \mathbf{v}_1(t) + \boldsymbol{\xi}_0(t) \end{aligned} \quad (26)$$

其中,

$$A_1 = \text{diag}\{1, I_x, 1, I_y, 1, I_z\}$$

$$A = A_1^{-1} (A_2 + \Delta A)$$

$$B = A_1^{-1} B_1$$

$$\boldsymbol{\xi}_0(t) = A_1^{-1} \boldsymbol{\xi}_1(t)$$

$$A_2 = \begin{bmatrix} 0, 1, 0, 0, 0, 0 \\ -(I_y - I_z) \omega_0^2, 0, 0, 0, 0, -(I_y - I_z - I_x) \omega_0 \\ 0, 0, 0, 1, 0, 0 \\ 0, 0, 0, 0, 0, 0 \\ 0, 0, 0, 0, 0, 1 \\ 0, (I_y - I_z - I_x) \omega_0, 0, 0, -(I_y - I_x) \omega_0^2, 0 \end{bmatrix}$$

$$\Delta A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \omega_0 h_y & 0 & 0 & 0 & 0 & h_y \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\omega_0 h_x & h_z & 0 & 0 & -h_z \omega_0 & -h_x \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -h_y & 0 & 0 & \omega_0 h_y & 0 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$$

$$\mathbf{v}(t) = -[\dot{h}_x(t), \dot{h}_y(t), \dot{h}_z(t)]^T = -B_0 \dot{\mathbf{h}}(t)$$

$$\boldsymbol{\xi}_1(t) = [0, \omega_0, h_z(t), 0, 0, 0, -\omega_0 h_x(t)]^T$$

$$\mathbf{x} = \begin{bmatrix} \phi & \dot{\phi} & \theta & \dot{\theta} & \psi & \dot{\psi} \end{bmatrix}^T$$

其中, 参数为 $I_x = 200 \text{ kg} \cdot \text{m}^2$, $I_y = 100 \text{ kg} \cdot \text{m}^2$, $I_z = 180 \text{ kg} \cdot \text{m}^2$, $T_{\text{orbit}} = 10 \text{ h}$, $\omega_0 = (2\pi/3600) T_{\text{orbit}}$.

$t = t_0$ 时, $\mathbf{h} = [0, 0, 0, 0]^T$.

4.2 建立功能目标模型

4.2.1 结构分解

将图 1 所示的卫星姿态控制系统分解为四个重构单元, 分别为执行机构、控制器、诊断、敏感机构, 然后再分解到最小重构单元, 如图 10 所示. 接下来依次分析每个最小重构单元的功能和状态, 如表 2 所示.

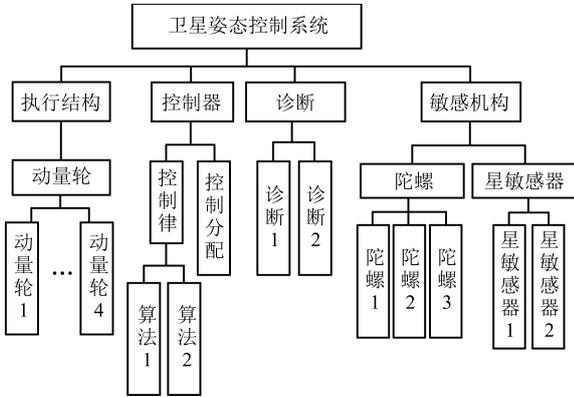


图 10 卫星姿态控制系统结构分解图

Fig. 10 Structural decomposition for the satellite control system

本例没有给出所有最小重构单元功能的函数表达式, 最小功能的相关说明将在下一小节中给出. 通过表 2, 可以得系统故障模式集为 $FauS = \{ss1_fau, ss2_fau, gx_fau, gy_fau, gz_fau, w1_faus, w1_faup, w2_faus, w2_faup, w3_faus, w4_faus\}$.

4.2.2 建立功能逻辑关系

1) 建立功能域

由表 2 可得系统功能域为: $F = \{phim, varphim, thetam, wxm, wym, wzm, u1, u2, u3, u4, ca, fdi1, fdi2, c, uc\}$, $p_F = 15$.

2) 建立输入输出有向图

各个最小功能的输入输出关系由表 3 给出. 有向图关系如图 11 所示, 其中 (phim, varphim, thetam, wxm, wym, wzm) 之间为与门关系, fdi1 和 fdi2 之间为异或门关系, c 和 uc 之间为或门关系, (u1, u2, u3, u4) 之间为与门关系.

3) 建立功能逻辑关系图

通过初步合并和进一步分析, 得到功能逻辑关系图, 如图 12 所示.

4.2.3 建立功能目标关系

逐层分析图 12 各层的功能, 划分目标, 制定各层的目标, 如表 4 所示. 进一步可以建立系统功能目标关系图, 如图 13 所示.

4.2.4 建立可行集

依次得到各个目标的可行集, 如表 5 所示.

表 2 卫星姿态控制系统最小重构单元表

Table 2 MRUs of the satellite control system

序号	最小重构单元	功能	状态
1	星敏感器 1 (ss1)	ϕ, φ 和 θ 测量 (phim, varphim, thetam)	正常 (nor); 完全故障 (fau)
2	星敏感器 2 (ss2)	ϕ, φ 和 θ 测量 (phim, varphim, thetam)	正常 (nor); 完全故障 (fau)
3	陀螺 x (gx)	ω_x 测量 (wxm)	正常 (nor); 完全故障 (fau)
4	陀螺 y (gy)	ω_y 测量 (wym)	正常 (nor); 完全故障 (fau)
5	陀螺 z (gz)	ω_z 测量 (wzm)	正常 (nor); 完全故障 (fau)
6	动量轮 1 (w1)	提供力矩 (u1)	完全故障 (faus); 比例故障 (faup); 正常 (nor);
7	动量轮 2 (w2)	提供力矩 (u2)	完全故障 (faus); 比例故障 (faup); 正常 (nor);
8	动量轮 3 (w3)	提供力矩 (u3)	完全故障 (faus); 正常 (nor);
9	动量轮 4 (w4)	提供力矩 (u4)	完全故障 (faus); 正常 (nor);
10	控制分配模块 (ca)	将控制律分配到各个动量轮 (ca)	正常 (nor)
11	诊断 1 (fdi1)	诊断故障 (fdi1)	正常 (nor)
12	诊断 2 (fdi2)	诊断故障 (fdi) + 健康因子辨识 (pi)	正常 (nor)
13	控制算法 1	控制算法 (c)	正常 (nor)
14	控制算法 2	欠驱动算法 (uc)	正常 (nor)

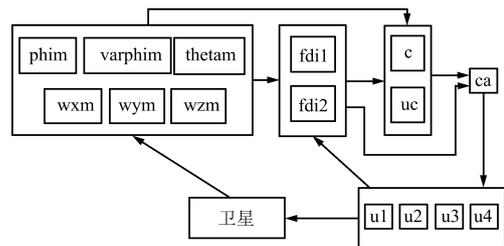


图 11 卫星姿态控制系统最小功能有向图

Fig. 11 Directed map for the MFs of the satellite control system

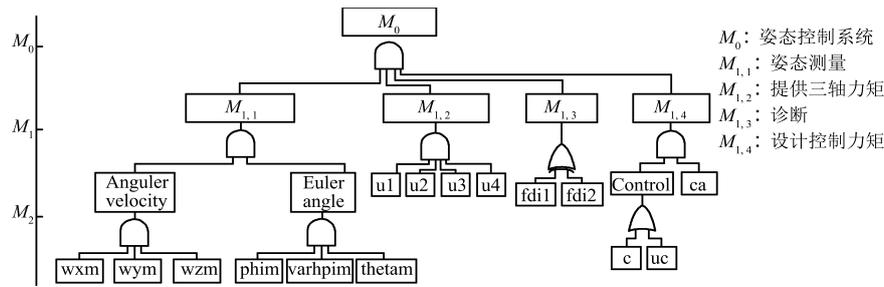


图 12 系统逻辑关系图

Fig. 12 Logic relation map

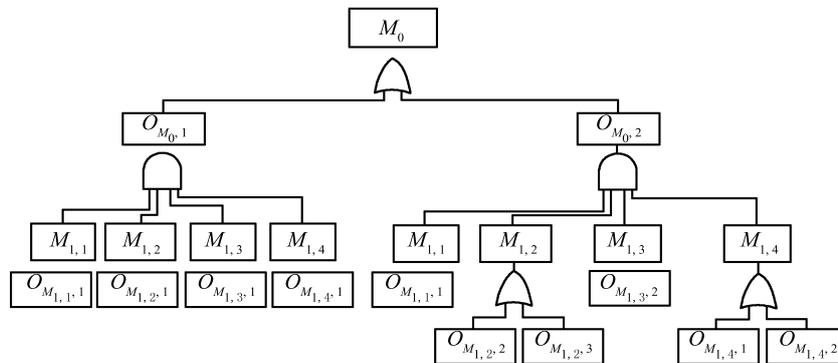


图 13 功能目标关系图

Fig. 13 Function-objective map

表 3 最小功能输入输出表
Table 3 Input-output list for the MF

MF	inpt	outp
phim	星图 & 卫星姿态	ϕ
varphim	星图 & 卫星姿态	φ
thetam	星图 & 卫星姿态	θ
wxm	卫星姿态角速度	ω_x
wym	卫星姿态角速度	ω_y
wzm	卫星姿态角速度	ω_z
u1	u1d (w1 输入指令)	u1 (w1 输出力矩)
u2	u2d (w2 输入指令)	u2 (w2 输出力矩)
u3	u3d (w3 输入指令)	u3 (w3 输出力矩)
u4	u4d (w4 输入指令)	u4 (w4 输出力矩)

通过可行集的合并 (表 5), 可得包含所有可能的可行配置的顶层可行集 (表 6). 表 6 中 $FC_{M_0,2}$ 合并运算中, 因为存在逻辑关系 $O_{M_{1,2,2}} \subset O_{M_{1,2,3}}$, 所以 $FC_{M_{1,2,2}} \subset FC_{M_{1,2,3}}$, 即 $FC_{M_{1,2,3}}$ 继承 $FC_{M_{1,2,2}}$ 的全部可行配置. 此时 $FC_{M_{1,2,2}}$ 的相应控制算法可以选择 c 或者 uc. 为了便于分析, 选择 uc 时以健康因子高且满足可控性条件的两个飞轮作为执行机构, 视为一种可行配置.

对于卫星系统 (26), 系统矩阵随着动量轮的角动量变化而变化. 表 6 得到的可行集由逻辑运算得到, 需要进一步判断是否满足能控能观性条件 (22) 和 (23). 比例故障是本文涉及到的故障类型之一, 这里分析动量轮 1 和动量轮 2 的比例故障时, 健康因子分别取为 0.5 和 0.6. 经过计算, 动量轮状态 $\{w1_faup, w2_faus, w3, w4_faus\}$ 不满足能控性条件, 所以可行配置数量为 $3 \times 9 \times 1 \times 2 + 3 \times 7 \times 1 \times 1 = 75$.

4.3 可重构控制分析

对于一个处于设计阶段的卫星控制系统, 应用本文方法, 可以分析系统各种故障模式下的可重构控制问题. 在线运行的卫星控制系统, 发生故障且正确诊断故障以后, 根据系统状态和能控能观性条件, 再次判断离线获得的可行集中是否有含系统当前状态的可行配置, 即可分析卫星控制系统可重构性.

例如, 当动量轮 1 完全故障, 其他最小重构单元健康时, 目标 1 中仍有可行配置, 此时系统重构后的控制方案仍然满足目标 1; 当动量轮 1 和 2 同时发生故障, 目标 1 中没有可行配置, 目标 2 中有多个可行配置, 此时为降级使用, 系统仍然安全; 当动量轮 1、动量轮 2 和动量轮 3 都发生完全故障时, 系统无法重构.

表 4 功能目标明细表

Table 4 Function-objective specifications

层	用户要求明细	目标
M_0 层	按控制能力划分为较强控制能力及安全两个级别.	$O_{M_0,1}$: 至少有三个健康动量轮, 有姿态角和姿态角速度信息;
		$O_{M_0,2}$: 至少有两个可用动量轮, 有姿态角和姿态角速度信息.
M_1 层	姿态测量: 有姿态角和姿态角速度信息; 驱动力矩: $O_{M_0,1}$ 时至少有三个健康动量轮; $O_{M_0,2}$ 时至少有两个可用动量轮; 诊断: 用比例故障动量轮作为动力时采用 fdi2, 其他情况采用 fdi1; 控制算法: $O_{M_0,1}$ 时采用控制算法 1; $O_{M_0,2}$ 时根据具体的情况采用控制算法 2 或者控制算法 1.	$O_{M_1,1,1}$: 有姿态角和姿态角速度信息;
		$O_{M_1,2,1}$: 至少有三个健康动量轮;
		$O_{M_1,3,1}$: 用故障诊断 1;
		$O_{M_1,4,1}$: 用控制算法 1 和控制分配;
		$O_{M_1,2,2}$: 有 3 个可用动量轮, 用控制算法 1;
		$O_{M_1,2,3}$: 有 2 个可用动量轮, 用控制算法 2;
		$O_{M_1,3,2}$: 有比例故障动量轮作为执行器时用 fdi2, 其他情况用 fdi1;
		$O_{M_1,4,2}$: 用控制算法 2 和控制分配.

注: 为了简便, 正常则不注明状态.

本文方法可以在系统设计阶段分析系统的可重构性. 假设该卫星系统要求单故障可重构, 应用本文方法, 依次分析所有故障模式集中的单故障情况, 可得子集 $\{gx_fau, gy_fau, gz_fau\}$ 无法重构. 因此, 陀螺设计不能满足要求明细, 改进设计时应该考虑增加陀螺冗余.

5 总结

本文建立的功能目标模型通过功能、目标、可行集、最小重构单元状态等定义, 将功能、组件和系统状态等结合在一起, 该模型适用于离线/在线分析线性系统可重构控制问题, 包括发生多个故障时是否具有可重构能力, 采用哪些组件和何种控制方法, 以及重构后系统是否能达到预定的控制目标等.

本文首先详细介绍了功能目标模型的相关概念和建立步骤. 功能目标关系和可行集是本文功能目标模型的两个重要组成部分. 系统的功能目标模型可

表 5 可行集表

Table 5 FS list

可行集	可行配置
$FS_{M_1,1,1}$	$\{ss1, ss2, gx, gy, gz\}, \{ss1_fau, ss2, gx, gy, gz\}, \{ss1, ss2_fau, gx, gy, gz\},$ $p_{M_1,1,1} = 3.$
$FS_{M_1,2,1}$	$\{w1, w2, w3, w4\}, \{(w1_fau, w1_fau), w2, w3, w4\}, \{w1, (w2_fau, w2_fau), w3, w4\}, \{w1, w2, w3_fau, w4\}, \{w1, w2, w3, w4_fau\},$ $p_{M_1,2,1} = 7.$
$FS_{M_1,3,1}$	$\{fdi1\}, p_{M_1,3,1} = 1$
$FS_{M_1,4,1}$	$\{c, ca\}, p_{M_1,4,1} = 1$
$FC_{M_1,2,2}$	$\{(w1_fau, w2_fau), \{w1_fau, w2_fau\}\} \cup \{w3, w4\}; \{(w1, w2_fau), \{w1_fau, w2\}, \{w1_fau, w2_fau\}\} \cup \{(w3, w4_fau), \{w3_fau, w4\}\}; p_{M_1,2,2} = 9.$
$FC_{M_1,2,3}$	$\{(w1_fau, w2_fau), \{w1, w2_fau\}, \{w1_fau, w2_fau\}, \{w1, w2\}\} \cup \{w3_fau, w4_fau\}; \{(w1_fau, w2_fau), \{w1_fau, w2_fau\}\} \cup \{w3, w4_fau\}; \{(w1_fau, w2_fau), \{w1_fau, w2_fau\}\} \cup \{w3, w4\}; \{(w1_fau, w2_fau), \{w1_fau, w2_fau\}\} \cup \{w3_fau, w4\};$ $p_{M_1,2,3} = 8.$
$FC_{M_1,3,2}$	$\{fdi1\}$ or $\{fdi2\}, p_{M_1,3,2} = 1.$
$FC_{M_1,4,2}$	$\{uc, ca\}, p_{M_1,4,2} = 1.$

表 6 顶层可行集表

Table 6 Top FS list

目标	可行集
$O_{M_0,1}$	$FC_{M_0,1} = FS_{M_1,1,1} \cup \dots \cup FS_{M_1,4,1}$ $p_{M_0,1} = 3 \times 7 \times 1 \times 1 = 21$
$O_{M_0,2}$	$FC_{M_0,2} = FS_{M_1,1,1} \cup \dots \cup FC_{M_1,4,3}$ $p_{M_0,2} = 3 \times 9 \times 1 \times 2 + 3 \times 8 \times 1 \times 1 = 78$

以通过四步建立, 第 4 步建立的顶层目标的可行集涵盖了所有的故障模式和可行的最小重构单元组件状态组合 (可行配置). 用于可重构控制分析时, 对某一个多故障情况, 可以通过顶层可行集中是否有包含该种状态组合的可行配置判断系统能否可重构. 用该方法在线分析时, 先要诊断出系统当前状态, 还

要再次判断顶层可行集中含当前状态的可行配置. 最后本文以一个简化了的卫星控制系统为例, 建立功能目标模型, 说明了本文的可重构控制分析过程, 并解释了降级使用、继承等概念.

对于一个组件多、故障模式多、性能明细要求多的系统(如卫星控制系统), 建立可行集计算量较大, 可以通过计算机辅助. 基于本文方法, 还可以进一步研究控制系统可重构性评价方法和可重构性设计方法.

References

- 1 Lombaerts T, Van Oort E, Chu Q P, Mulder J A, Joosten D. Online aerodynamic model structure selection and parameter estimation for fault tolerant control. *Journal of Guidance, Control, and Dynamics*, 2010, **33**(3): 707–723
- 2 Alwi H, Edwards C, Stroosma O, Mulder J A. Evaluation of a sliding mode fault-tolerant controller for the El Al incident. *Journal of Guidance, Control, and Dynamics*, 2010, **33**(3): 677–694
- 3 Cao L, Chen X Q, Sheng T. Fault tolerant small satellite attitude control using adaptive non-singular terminal sliding mode. *Advances in Space Research*, 2013, **51**(12): 2374–2393
- 4 Cai W C, Liao X H, Song D Y. Indirect robust adaptive fault-tolerant control for attitude tracking of spacecraft. *Journal of Guidance, Control, and Dynamics*, 2008, **31**(5): 1456–1463
- 5 Zhang Y M, Jiang J. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual Reviews in Control*, 2008, **32**(2): 229–252
- 6 Gehin A L, Hu H, Bayart M. A self-updating model for analyzing system reconfigurability. *Engineering Applications of Artificial Intelligence*, 2012, **25**(1): 20–30
- 7 Guan Shou-Ping, Yang Fei-Sheng. Reconfiguration-goal-oriented control system reconfigurability. *Information and Control*, 2010, **39**(4): 391–396
(关守平, 杨飞生. 面向重构目标的控制系统可重构性. *信息与控制*, 2010, **39**(4): 391–396)
- 8 Eva Wu N, Zhou K M, Salomon G. Control reconfigurability of linear time-invariant systems. *Automatica*, 2000, **36**(11): 1767–1771
- 9 Gehin A L, Staroswiecki M. Reconfiguration analysis using generic component models. *IEEE Transactions on Systems, Man, and Cybernetics*, 2008, **38**(3): 575–583
- 10 Düstegör D, Frisk E, Cocquempot V, Krysander M, Staroswiecki M. Structural analysis of fault isolability in the DAMADICS benchmark. *Control Engineering Practice*, 2006, **14**(6): 597–608
- 11 Izadi-Zamanabadi R, Staroswiecki M. A structural analysis method formulation for fault-tolerant control system design. In: *Proceedings of the 39th Conference on Decision and Control*. Sydney, Australia: IEEE, 2000. 4901–4902
- 12 Staroswiecki M. Observability and the design of fault tolerant estimation using structural analysis. *Lecture Notes in Control and Information Sciences*, 2007, **353**(1): 257–278
- 13 Alwi H, Edwards C. Fault tolerant control using sliding modes with on-line control allocation. *Automatica*, 2008, **44**(7): 1859–1866
- 14 Xiao B, Hu Q L, Singhose W, Huo X. Reaction wheel fault compensation and disturbance rejection for spacecraft attitude tracking. *Journal of Guidance, Control, and Dynamics*, 2013, **36**(6): 1565–1575



段文杰 北京控制工程研究所博士研究生. 2011 年获得北京航空航天大学硕士学位. 主要研究方向为容错控制, 可重构性分析与设计.

E-mail: wenjie_duan@163.com

(**DUAN Wen-Jie** Ph.D. candidate at Beijing Institute of Control Engineering. He received his master degree

from Beijing University of Aerospace and Aerospace in 2011. His research interest covers fault-tolerant control, reconfigurability analysis and design.)



王大轶 北京控制工程研究所研究员. 主要研究方向为深空探测航天器自主控制和卫星控制系统故障诊断. 本文通信作者. E-mail: dayiwang@163.com

(**WANG Da-Yi** Professor at Beijing Institute of Control Engineering. His research interest covers autonomous

control of deep space exploration and fault diagnosis for satellite control systems. Corresponding author of this paper.)



刘成瑞 北京控制工程研究所高级工程师. 主要研究方向为卫星控制系统故障诊断及容错控制.

E-mail: liuchengrui@gmail.com

(**LIU Cheng-Rui** Senior engineer at Beijing Institute of Control Engineering. His research interest covers fault diagnosis and fault-tolerant control of

spacecraft.)