

文章编号:1000-8055(2011)07-1659-06

微型姿控固体推力器阵列点火算法

刘旭辉¹, 方蜀州¹, 刘书杰¹, 罗 莉², 权 恩¹

(1. 北京理工大学 宇航学院, 北京 100081;

2. 中国航天科技集团公司 中国空间技术研究院
北京控制工程研究所, 北京 100190)

摘 要: 对用于微型卫星姿态控制的固体推力器阵列的组合点火问题进行了研究, 分析了微型固体推力器阵列的工作性能, 建立了组合点火问题的数学模型, 设计了基值组合算法和阵列规划算法两种点火算法, 并利用 Matlab/Simulink 对设计的算法进行了仿真. 结果表明: 阵列规划算法计算准确, 快速有效, 并且提高了阵列中推力器的利用率.

关 键 词: 点火算法; 姿态控制; 微型固体推力器阵列; 基值组合算法; 阵列规划算法
中图分类号: V448.22 **文献标志码:** A

Ignition algorithm of solid propellant micro-thruster array for attitude control

LIU Xu-hui¹, FANG Shu-zhou¹, LIU Shu-jie¹, LUO Li², QUAN En¹

(1. School of Aerospace Engineering,

Beijing Institute of Technology, Beijing 100081, China;

2. Beijing Institute of Control Engineering, China Academy of Space Technology,
China Aerospace Science and Technology Corporation, Beijing 100190, China)

Abstract: The study on ignition algorithm of solid propellant micro-thruster array for attitude control of micro-satellite was made and described in this paper. The performance capabilities of solid propellant micro-thruster array were analyzed and the theoretical model was set up, then two kinds of ignition algorithms, that were combined base value algorithm and array programming algorithm, were proposed and verified by simulation using Matlab/Simulink. The simulation shows that, array programming algorithm is accurate and fast, also can improve the utilization efficiency of thrusters in the array.

Key words: ignition algorithm; attitude control; solid propellant micro-thruster array; combined base value algorithm; array programming algorithm

自 20 世纪 90 年代末开始研究的微型固体推力器阵列, 由于具有体积小、成本低、功耗低、可靠性高等优点, 受到了广泛的重视^[1-4]. 许多国家和地区对微型固体推力器阵列的结构设计、性能试验和仿真分析进行了大量的研究^[5-8]. 然而, 对于

其中很重要的推力器阵列组合点火算法的研究却较少, 如何构建阵列点火问题的数学模型、如何进行快速的点火寻址以及如何提高微型固体推力器阵列的利用率, 对于微型固体推力器阵列的应用具有重要意义.

收稿日期: 2010-06-28; 修订日期: 2010-08-23

基金项目: “十一五”民用航天预先研究项目(微型固体推力器阵列技术研究)

作者简介: 刘旭辉(1983—), 男, 山东莱州人, 博士生, 主要从事微小型姿轨控动力系统研究. E-mail: xhliu@bit.edu.cn

1 组合点火问题的规划模型

1.1 微型固体推力器工作模型简化

将实际推力等效为推力方波,周期 T_{eq} 与实际推力器工作时间 t 相同,单个推力器的总冲量与实际推力总冲量相同^[9].

$$\int_0^T F(t) dt = F_{eq} T_{eq} \quad (1)$$

式中 F 为单个推力器的推力; F_{eq} 为等效推力.

本算法研究中采用正六面体的形式布置阵列,如图1所示^[10].当微型卫星在进行姿态稳定控制或姿态机动时,将所需力矩矢量在三个主惯量轴上分解.单个轴上的力矩由相对分布的两个推力器阵列(如阵列A和阵列B)提供,可令这一对阵列同一行、列间距为特定值的两个推力器(相对阵列上各点一个)同时点火,控制姿态角、姿态角速度达到期望值^[11].

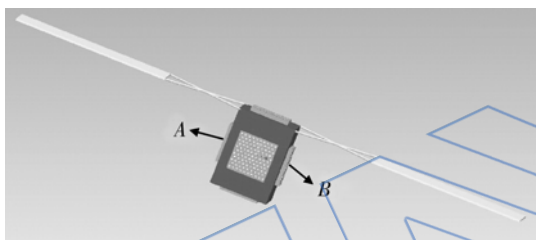


图1 微型固体推力器阵列安装位置示意图

Fig.1 Schematic diagram of array installation

1.2 微型固体推力器的控制精度及快速响应性分析

根据阵列的特点,最小单位力矩决定了姿态的控制精度.当卫星由稳定状态至受到扰动,短时间内偏差变化不大,俯仰与偏航、滚转通道可以认为是解耦的,以滚转通道为研究对象,来分析微型固体推力器的控制精度及快速响应性.

由1.1节已知,周期 T_{eq} 与实际推力器工作时间 t 相同,因此最小冲量矩可表示为 $A_{min} = F_{eq} L_{min} T_{eq}$,其中 L_{min} 为两个推力器之间的最小间距.由于最小冲量矩 A_{min} 的存在,因此存在最小角速度增量 $\Delta \dot{\theta}$,可表示为

$$\Delta \dot{\theta} = A_{min} / I_z \quad (2)$$

式中 I_z 为滚转方向转动惯量.

本文研究的推力器阵列 $F_{eq} = 100 \text{ mN}$, $T_{eq} = 10 \text{ ms}$, $L_{min} = 2 \text{ mm}$.设卫星滚转方向转动惯量 $I_z = 12 \times 10^{-3} \text{ kg} \cdot \text{m}^2$,由计算可知,角速率控制精度达 $\Delta \dot{\theta} = 9.554 \times 10^{-3} (^{\circ})/\text{s}$.

微型固体推力器阵列不但具有较高的控制精度,而且在进行大角度机动方面也有良好的应用前景.

当卫星进行 30° 大角度机动时,对于 100×100 的阵列,控制滚转轴的一对阵列最大可以提供 $5 \text{ N} \cdot \text{m}$ 的正向力矩和 $5 \text{ N} \cdot \text{m}$ 的负向力矩.最小单位力矩为 $M_{min} = 2 \times 10^{-4} \text{ N} \cdot \text{m}$,因此单方向具有 25000 个单位的 M_{min} (因为推力器的使用个数需要经过算法求得,同样的推力器数量产生的力矩也不同,因此计算中采用 M_{min} 的数量作为力矩的计量单位,而不采用推力器个数).

设推力器工作延迟时间低于 10 ms ,由于其小于实际推力器工作时间 10 ms ,因此可以使推力器连续工作,以恒定加速度进行机动.先在 $t_w/2$ 时间里加速,加速机动角度为 $\theta/2$,再在 $t_w/2$ 时间里减速,减速机动角度为 $\theta/2$ ^[12].考虑加速机动和减速机动的总时间,可得如下公式:

$$K_c = I_z \ddot{\theta} = \frac{4 I_z \theta}{t_w^2} \quad (3)$$

式中 K_c 为控制力矩; $\ddot{\theta}$ 为机动角加速度; θ 为机动角度; t_w 为机动时间.

进行本次大角度机动要求控制力矩小于单方向可提供总力矩的 20%,即 $K_c \leq 1 \text{ N} \cdot \text{m}$,最多使用 5000 个单位的 M_{min} .

将正、负方向要求使用的力矩进行 n 次平均,考虑到该阵列点火延迟时间小于工作时间,可以连续控制,加速度恒定,加速时间、减速时间各为 nT_{eq} ,总工作时间为 $2nT_{eq}$.假设总共使用了 m 个单位的 M_{min} ,因此控制力矩 $K_c = mM_{min}/n$,可得

$$\frac{mM_{min}}{n} = \frac{4 I_z \theta}{(2nT_{eq})^2} \quad (4)$$

因为 $m \leq 5000$,经计算可得

$$m \times n \leq 314000, \quad n \geq 62.8$$

如果 $m = 5000$,则 $n = 62.8$,因为 n 为整数,并且需要被 m 整除,因此 n 取 63, m 为 4977,总共使用了 4977 个单位的 M_{min} ,而控制力矩 $K_c = \frac{mM_{min}}{n} = \frac{4977M_{min}}{63} = 0.0158 \text{ N} \cdot \text{m}$,总工作时间为 $2nT_{eq} = 126T_{eq} = 1.26 \text{ s}$,即经过 1.26 s ,机动角度为 29.96° ,存在 0.13% 的误差,但是工作时间远远小于其他执行机构^[12].

由此可知,该推力器阵列具有较高的控制精度,并且进行大角度机动时具有较快的响应速度.

1.3 微型固体推力器阵列设计

在前人的研究中,通常设计的阵列是正方形

阵列(如图 2 所示)^[11,13],本文中设计的阵列以正六边形为基础,依次进行扩展,形成推力器阵列(如图 3 所示),因为相邻两个推力器之间的间距缩短为原来的 $\sqrt{3}/2$,因此这种设计方式可以提高阵列的面积利用率.无论是正方形阵列还是正六边形阵列,推力器位置都可以表示为 (n, m) ,即第 n 行第 m 个,因此该算法具有良好的适用性,既适用于正方形阵列也适用于正六边形阵列.在进行算法研究时为了表示的直观性,采用正方形阵列作为研究对象.

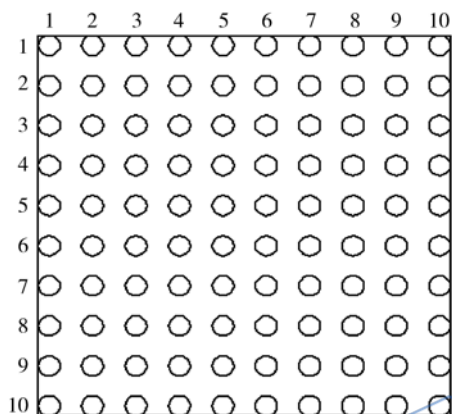


图 2 正方形阵列示意图

Fig. 2 Schematic diagram of normal array

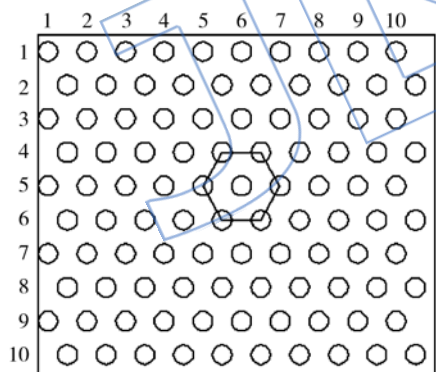


图 3 正六边形阵列示意图

Fig. 3 Schematic diagram of hexagon array

1.4 阵列最优点火寻址的数学模型

基于微型固体推力器阵列的点火算法,可以将其转化为整数规划中的 0-1 规划问题^[14].其数学描述如下:

目标函数

$$\min Z = \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J (u_{(i,j,k)} \times x_{(i,j,k)})$$

$$1 \leq K \leq N, \quad 1 \leq I \leq N, \quad 1 \leq J \leq N \quad (5)$$

式中 k 为所选推力器的行数; i 为所选推力器阵列的列数; j 为相对阵列的列数. N 为阵列规模,

如 100×100 的阵列,则 $N=100$;若 10×10 的阵列,则 $N=10$.

成本函数

$$u_{(i,j,k)} = \left[i - \frac{1+N}{2} \right]^2 + \left[j - \frac{1+N}{2} \right]^2 + \left[\left(1 - \frac{1+N}{2} \right)^2 + \left(N - \frac{1+N}{2} \right)^2 \right]$$

$$1 \leq i \leq N, \quad 1 \leq j \leq N, \quad 1 \leq k \leq N \quad (6)$$

约束条件

$$\begin{cases} K_c = \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J (F_{eq} \times L_{min} \times |i-j|_k \times x_{(i,j,k)}) \\ x_{(i,j,k)} = \begin{cases} 0 \\ 1 \end{cases} \end{cases} \quad (7)$$

当选择了位于阵列第 i 列,其相对阵列第 j 列,第 k 行的推力器时, $x_{(i,j,k)}=1$;否则 $x_{(i,j,k)}=0$.

$L_{min} \times |i-j|_k$ 为分别位于两个相对阵列第 k 行上的第 i 列和第 j 列的两个推力器之间的距离,实际上也就是两个推力器产生的力矩的力臂.阵列能提供的最小力矩为

$$M_{min} = F_{eq} L_{min} \quad (8)$$

最小力矩 M_{min} 可以定义为 1 个单位力矩.

该模型可以约束推力器如何进行组合提供所需的控制力矩,以达到利用率的最大化以及使用成本最低的目的.离阵列中心越近则成本函数 u 越小,优先使用离阵列中心近的推力器,可以保证留下的推力器能够提供更多的产生力矩的推力器组合形式.

以 10×10 的阵列为例,需要产生 18 个单位的力矩,有多种组合.依据建立的数学模型,表 1 中列出其中的几种方案进行比较说明.

表 1 多种组合方案比较

Table 1 Comparison of various combinations program

	行数 k	列数 i	列数 j	$ i-j $	成本	总成本	推力器 个数
方案 1	1	1	10	9	81	162	4
	2	1	10	9	81		
方案 2	1	2	9	7	65	181	6
	1	3	8	5	53		
	1	1	7	6	63		
	1	2	9	7	65		
方案 3	1	1	6	5	61	189	6
	1	4	10	6	63		

由表1可知方案1最优,使用成本最低、使用推力器个数最少。

2 寻址搜索算法

2.1 算法选择

本文研究的点火问题属于0-1规划问题,可以将成本值、 $|i-j|$ 的值建立数据库,运用遗传算法、蚁群算法、模拟退火算法、禁忌搜索算法等,进行搜索求出最优解,然后再根据最优解、成本值及 $|i-j|$ 的值,反求出阵列中点火推力器的位置,将每次点完的推力器成本值设高,继续进行下次寻址,但是这些优化算法需要建立的数据库非常庞大,收敛速度慢,寻找出的推力器组合产生的力矩可能达不到要求的精度^[14]。

在求得精确解的前提下,进行了算法搜索效率和阵列利用率的研究。根据阵列的特点,提出了两种算法:一种是基值组合算法,类似人民币币值能够组合出任意数值大小,将阵列中推力器定义为产生基值力矩的组合,叠加以产生需要大小的力矩;另一种方法是阵列规划算法,融合0-1规划、动态规划及贪心算法的思想进行搜索,进行分阶段寻优求解。

2.2 算法实现及仿真

利用 Matlab 软件编程实现微型固体推力器阵列的点火寻址算法,并且验证算法的正确性以及对阵列推力器的利用率。在算法实现时,根据公式(8)将阵列能够提供的最小力矩 $M_{\min}$ 定义为1个单位的力矩。数值10表示10个单位的力矩,即10个单位的 $M_{\min}$ 。

2.2.1 基值组合算法的实现

首先基于人民币由1,2,5,10这4个面值组成,研究了一种以1,2,5,9为基值的算法。如图1、图2所示,以 10×10 的阵列为例,将阵列A中某行的第1个推力器与相对阵列B中同行的第10个推力器定义为第1组,用来产生9个单位的力矩;将阵列A中某行的第2个推力器和相对阵列B中同行的第3个推力器定义为第2组,用来产生1个单位的力矩;将阵列A中某行的第3个推力器和相对阵列B中同行的第8个推力器定义为第3组,用来产生5个单位的力矩;将阵列A中某行的第4个推力器和相对阵列B中同行的第6个推力器定义为第4组,用来产生2个单位的力矩,将阵列A中某行的第7个推力器和相对阵列B中同行的第9个推力器定义为第5组,也

用来产生2个单位的力矩。如此前后两个阵列位于同行的推力器便可产生1,2,2,5,9个单位的力矩,而这5个数可以组合出1~9中的任意整数。

对于任意需要产生的力矩,都可以将其分解,利用每一行中的5个数进行组合产生。对于已用推力器将其成本值设为40.5(该值为整个阵列中的最高成本值,已用推力器成本值也可以设为大于该值,可以由公式(6)求得)。

基值组合算法的流程如图4所示。

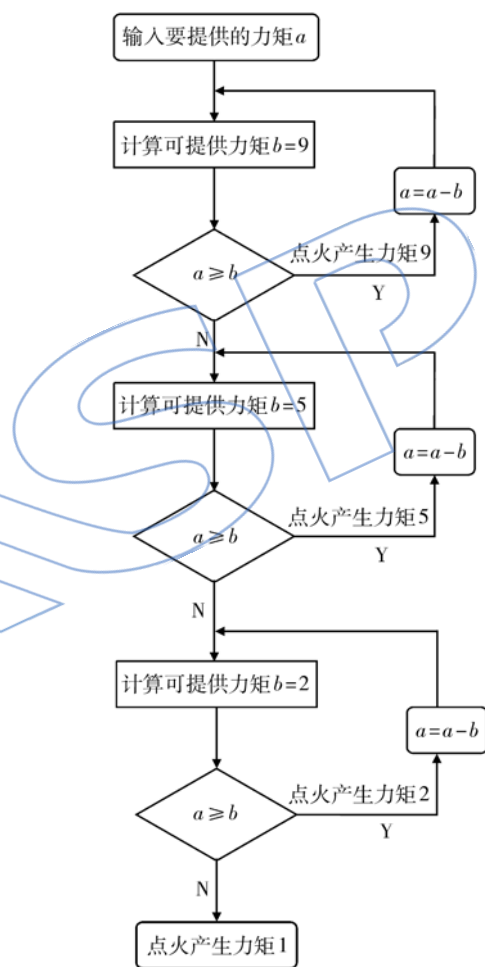


图4 基值组合算法流程图

Fig. 4 Flow diagram of combined base value algorithm

基值组合算法搜索效率非常高,为一维搜索,搜索规模小于阵列行数,可以快速寻址到要求的推力器位置。

经过仿真,发现了基值组合算法的两个弊端。

1) 不能提供大的力矩,如果需要产生大于100个单位的力矩时,算法便会将两个阵列中所有产生9个单位的力矩的组合点完,利用率太低,仅为20%,而阵列最大能够提供250个单位的力矩。

2) 不宜多次提供余数为 1 个单位的力矩,因为每行有 1 个组合可以产生 1 个单位的力矩,如此,10 次就会点完所有能够产生 1 个单位的力矩组合,这样会使得其他很多剩余推力器不能得到利用,整个阵列的利用率变低。

2.2.2 点火位置寻址改进算法的实现

由于基值组合算法的缺陷,本文设计了另一种寻址改进的算法,因为其属于 0-1 规划问题,但又不同于常规 0-1 规划,因此将其命名为阵列规划算法。同样以 10×10 的阵列为例,相对的两个推力器阵列单行最大能够提供 9 个单位的力矩,最小能够提供 1 个单位的力矩。参照图 1 和图 2,以第 1 行为例,阵列 A 中第 1 个推力器能够与阵列 B 中同行的推力器组合产生 1~9 个单位的力矩,而第 5 个推力器由于距离对称中心近,只能组合产生 1~5 个单位的力矩。根据建立的数学模型定义成本函数为

$$u = |i - 5.5|^2 + |j - 5.5|^2 + 40.5 \quad (9)$$

式中 u 为用来产生力矩的一对组合的成本; i 为阵列 A 中点火的推力器的列数; j 为相对阵列 B 中点火的推力器的列数。

对于任意需要产生的力矩,先对阵列进行搜索,找到各行能够产生的最大力矩,然后在所有可能的情况中找出成本最低的组合,如果成本一样,则选择其中的第 1 组,然后用所需力矩减掉已经找出的力矩,继续重复前面的操作,直到最后实现所需力矩。对于已用推力器将其成本值设为 100。

该算法进行寻址时,根据所需力矩以及阵列剩余的推力器所能提供的力矩情况,进行力矩分解,分解成若干执行步,寻找最优的决策序列,每个序列步骤按照成本最低、推力器个数最少的原则进行组合,保证了各个子步所需推力器组合成本最低、推力器个数最少。

阵列规划算法的流程如图 5 所示。

阵列规划算法寻址方法简单,执行效率高,单次最高可产生 250 个单位的力矩,利用率达到 100%,成功避免了基值组合算法的弊端,虽然效率略低,但是较枚举法搜索效率要高很多。阵列规划算法搜索为二维搜索,以 100×100 的阵列为例,搜索规模小于阵列行数乘以列数,即每次搜索规模小于 100×100 ,而枚举法搜索规模为 $100 \times 100 \times 100$,并且随着阵列规模的增大,这种对比将更加明显。

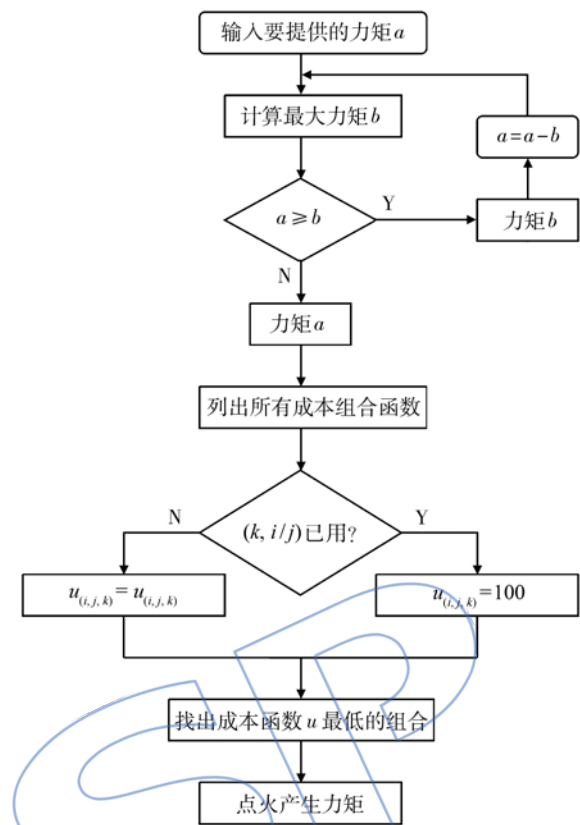


图 5 阵列规划算法流程图

Fig. 5 Flow diagram of array programming algorithm

进行了输出力矩依次为 13, 24, 16, 17, 6, 3 个单位的力矩的动态连续输出仿真,阵列规划算法的图形用户界面(graphical user interface, 简称 GUI)如图 6 所示。

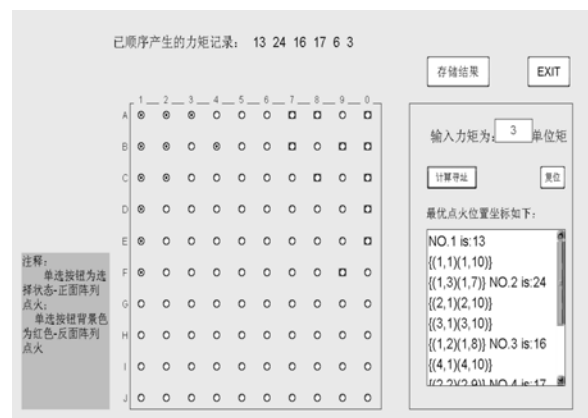


图 6 阵列规划算法的图形用户界面

Fig. 6 GUI of array programming algorithm

经过实例验证,能够准确寻找到成本最低、所需推力器个数最少的组合,表 2 为一系列力矩的动态连续输出仿真结果。

表2 力矩的动态连续输出仿真结果

Table 2 Simulation results of continuous output torque

待产生 力矩	输出 力矩	推力器位置 $\{(k, i)(k, j)\}$	推力器 个数
13	13	$\{(1, 1)(1, 10)\}$ $\{(1, 3)(1, 7)\}$	4
24	24	$\{(2, 1)(2, 10)\}$ $\{(3, 1)(3, 10)\}$ $\{(1, 2)(1, 8)\}$	6
16	16	$\{(4, 1)(4, 10)\}$ $\{(2, 2)(2, 9)\}$	4
17	17	$\{(5, 1)(5, 10)\}$ $\{(6, 1)(6, 9)\}$	4
6	6	$\{(3, 2)(3, 8)\}$	2
3	3	$\{(2, 4)(2, 7)\}$	2

注: $\{(k, i)(k, j)\}$ 表示阵列A中第k行第i列, 相对阵列B中同一行(第k行)第j列。

3 结 论

1) 简化了微型固体推力器工作模型, 设计了推力器阵列的正六边形布局方式, 提高了阵列面积利用率。

2) 分析了微型固体推力器阵列的控制精度及快速响应性, 得出该执行机构不但有较高的控制精度, 而且在进行大角度机动方面也有良好的应用前景。

3) 建立了微型固体推力器阵列点火组合问题的数学模型, 通过比较多种方案验证了该模型的正确性。

4) 实现了基值组合算法和阵列规划算法两种点火组合算法, 经过对比发现, 改进后的阵列规划算法, 提高了阵列中推力器的利用率, 并且能够准确快速地搜索出可用的推力器。

参考文献:

[1] Larangot B, Conédéra V, Dubreuil P, et al. Solid propellant microthruster: an alternative propulsion device for nanosatellite[C]// Proceedings of the Conference on Aerospace

Energetic Equipment. Avignon France: [s. n.], 2002: 10-19.

[2] Chaalane A, Larangôt B, Rossi C, et al. Main directions of solid propellant micro-propulsion activity at LAAS[R]. AIAA Paper 2004-6706, 2004.

[3] Kwon S, Lee J, Hoon L D. Design and performance evaluation of components of micro solid propellant thruster[R]. Fort Lauderdale, USA: The 40th AIAA/ASME/SAE/AS-EE Joint Propulsion Conference and Exhibit, 2004.

[4] Green A A. Demonstration and quantitative characterization of a MEMS fabricated propulsion system for the next generation of microspacecraft [R]. AIAA-2001-0009, 2001.

[5] Zhang K L, Chou S K, Ang S S. Performance prediction of a novel solid-propellant microthruster[J]. Journal of Propulsion and Power, 2006, 22(1): 56-63.

[6] Alexeenko A A, Levin D A, Girmelshin S F, et al. Numerical study of flow structure and thrust performance for 3-D MEMS-based nozzles[R]. AIAA-2002-3194, 2002.

[7] Wu M H, Yetter R A, Yang V. Development and characterization of ceramic micro chemical propulsion and combustion systems[R]. AIAA-2008-0966, 2008.

[8] 尤政, 张高飞, 林杨, 等. MEMS 固体化学推进器设计与建模研究[J]. 光学精密工程, 2005, 13(2): 117-126.

YOU Zheng, ZHANG Gaofei, LIN Yang, et al. Design and modeling of MEMS-based solid propellant propulsion[J]. Optics and Precision Engineering, 2005, 13(2): 117-126. (in Chinese)

[9] 屠善澄. 卫星姿态动力学与控制[M]. 2版. 北京: 中国宇航出版社, 1998: 178-179.

[10] Lewis D H, Jr., Janson S W, Cohen R B, et al. Digital micropropulsion[J]. Sensors and Actuators: Physical, 2000, 80(2): 143-156.

[11] 章仁为. 卫星轨道姿态动力学与控制[M]. 北京: 北京航空航天大学出版社, 1997: 141-155.

[12] 张景瑞. 灵敏航天器快速倾斜机动的 MCMG 参数和力矩估算[J]. 清华大学学报(自然科学版), 2007, 47(11): 2037-2040.

ZHANG Jingrui. Estimation of MCMG parameters and torques for rapid slewing maneuver of agile spacecraft[J]. Journal of Tsinghua University(Science and Technology), 2007, 47(11): 2037-2040. (in Chinese)

[13] Rossi C, Do Conto T, Estève D C, et al. Design, fabrication and modelling of MEMS-based microthrusters for space application[J]. Smart Materials and Structures, 2001, 10(6): 1156-1162.

[14] 李国勇. 最优控制理论及参数优化[M]. 北京: 国防工业出版社, 2006: 118-123.