多柔体系统动力学建模程序源代码的软件实现

杨东武,　段宝岩
(西安电子科技大学 机电工程学院,陕西 西安　710071)

摘要: 采用将较长表达式分割并用变量替换的递归循环方法,编制了相应的符号处理程序模块,将由Mathematica得到的公式化简并转化成FORTRAN格式的源代码. 通过算例说明该程序是正确、有效且可靠的.

关键词: 多柔体系统;动力学建模;符号推导
中图分类号:TB121; O313　　文献标识码:A　　文章编号: 1001-2400(2005)02-0193-04

Software implementation of the dynamic modeling code of flexible multibody systems

YANG Dong-wu, DUAN Bao-yan
(School of Electromechanical Engineering, Xidian Univ., Xi′an　710071, China)

Abstract: The formula deduced with Mathematica for dynamic modeling of flexible multibody systems is not simple enough and well formatted for programming. In order to code these formula into our system program of dynamic analysis for flexible multibody systems, a program module is developed to simplify and recode the formula using a recursion method. A test has been conducted which shows that the symbolic processing module is correct, effective and robust.

Key Words: flexible multibody system; dynamic modeling; symbolic deducing

柔性多体系统动力学建模的基本思想是对柔性体引入浮动坐标系, 柔性体的基本位形表现为浮动坐标系的大范围运动与相对浮动坐标系的弹性变形[1, 2]. 根据描述浮动坐标系(对于刚体即为连体基)运动所采用的广义坐标的选取不同, 相应的建模方法可分为: 最小数目坐标法, 较相对坐标法和绝对坐标法. 其中, 以绝对坐标方法得到的系统动力学方程形式较为统一, 不受系统拓扑结构的影响, 且形式简单, 约束方程可通过标准的约束处理方法产生, 是目前程式化建模最好的选择. 根据对不同系统的适应性来分, 多柔体系统动力学模型可分为传统的零次近似模型与一次近似模型. 实验证明[3], 在系统整体大范围运动速度较低的情况下, 两种模型得到的数值仿真结果基本一致, 传统零次近似模型是适用的.

笔者仅讨论以绝对坐标建立传统柔性多体系统动力学模型的建模方法. 由于国内在该方面的研究较少[4~9], 尚未有相应的高级语言建模程序, 文中简要地说明了由符号演算给出的公式推导结果到高级语言(FORTRAN)源代码生成的软件实现过程, 从而实现了运用高级语言进行柔性多体系统动力学建模的程序编制.

1　多柔体系统动力学建模

利用拉格朗日第二方程与有限元相结合的方法, 列写的多柔体系统动力学控制方程[8]为:

$$\begin{cases} \boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{K}\boldsymbol{q} + \boldsymbol{C}_q^{\mathrm{T}}\lambda = \boldsymbol{Q}_F + \boldsymbol{Q}_v , \\ \boldsymbol{C}(\boldsymbol{q}, t) = 0 , \end{cases} \tag{1}$$

收稿日期: 2004-04-29
基金项目: 国家部委"十五"预研重点项目(41321070301)
作者简介: 杨东武(1978-), 男, 西安电子科技大学硕士研究生.

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

其中, $\boldsymbol{M}, \boldsymbol{K}, \boldsymbol{C}, \boldsymbol{q}, \boldsymbol{C}_q$ 分别为系统的广义质量矩阵、刚度矩阵、约束方程(组)、广义坐标向量以及约束方程的雅克比矩阵; $\boldsymbol{Q}_F$, $\boldsymbol{Q}_v$ 分别为系统的广义力向量及科氏惯性力向量. 系统的建模工作中, 以推导系统单元质量矩阵 $\boldsymbol{M}$ 以及 $\boldsymbol{Q}_v$ 为建模的主要内容. 以平面系统为例, 系统中物体 i 上单元 j 的质量矩阵 $\boldsymbol{M}^{ij}$ 及 $\boldsymbol{Q}_v^{ij}$ 的具体形式为

$$\boldsymbol{M}^{ij} = \int_{V^{ij}} \rho^{ij} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{A}_\theta^i \boldsymbol{u}^{ij} & \boldsymbol{A}^i \boldsymbol{N}^{ij} \\ \text{对} & \boldsymbol{u}^{ij\mathrm{T}} \boldsymbol{u}^{ij} & \boldsymbol{u}^{ij\mathrm{T}} \boldsymbol{A}_\theta^{i\mathrm{T}} \boldsymbol{A}_\theta^i \boldsymbol{N}^{ij} \\ & \text{称} & \boldsymbol{N}^{ij\mathrm{T}} \boldsymbol{N}^{ij} \end{bmatrix}_{9\times 9} \mathrm{d}V^{ij} \quad , \tag{2}$$

$$\boldsymbol{Q}_v^{ij} = \frac{\partial T^{ij}}{\partial \boldsymbol{q}^i} - \dot{\boldsymbol{M}}^{ij} \dot{\boldsymbol{q}}^i \quad , \tag{3}$$

其中 ρ^{ij} 和 V^{ij} 分别为该单元的质量密度和体积, $\boldsymbol{u}^{ij}$ 为该单元上的点在物体 i 坐标系 $O^iX^iY^i$ 下的位置向量, T^{ij} 为该单元上的动能, $\boldsymbol{M}^{ij}$ 为该单元对物体 i 的整体质量矩阵的贡献, $\dot{\boldsymbol{M}}^{ij}$ 为矩阵 $\boldsymbol{M}^{ij}$ 对时间的导数, $\boldsymbol{q}^i$ 及 $\dot{\boldsymbol{q}}^i$ 为描述物体 i 位形的广义坐标及广义速度, $\boldsymbol{A}^i$ 为物体 i 的连体基到整体坐标系 $OXYZ$ 的旋转变换矩阵.

$$\boldsymbol{u}^{ij} = \boldsymbol{u}_0^{ij} + \boldsymbol{u}_f^{ij} \quad , \tag{4}$$

$$\boldsymbol{u}_0^{ij} = \boldsymbol{u}_{e0}^{ij} + \boldsymbol{u}_e^{ij} = \boldsymbol{u}_{e0}^{ij} + \boldsymbol{T}_1^{ij} \overline{\boldsymbol{u}}_e^{ij} \quad , \tag{5}$$

$$\boldsymbol{u}_f^{ij} = \boldsymbol{T}_1^{ij} \boldsymbol{S}^{ij} \boldsymbol{T}_2^{ij} \boldsymbol{L}^{ij} \boldsymbol{q}_f^i \quad , \tag{6}$$

$$\boldsymbol{N}^{ij} = \boldsymbol{T}_1^{ij} \boldsymbol{S}^{ij} \boldsymbol{T}_2^{ij} \boldsymbol{L}^{ij} \quad , \tag{7}$$

式中 $\boldsymbol{u}_0^{ij}$ 表示物体未变形时点 P 在物体坐标系下的位置向量, $\boldsymbol{u}_f^{ij}$ 表示物体变形后点 P 在物体坐标系下的变形向量, $\boldsymbol{u}_{e0}^{ij}$ 表示单元坐标系 $O^{ij}X^{ij}Y^{ij}$ 的原点在物体坐标系中的坐标向量, $\overline{\boldsymbol{u}}_e^{ij}$ 表示物体未变形时点 P 在单元坐标系下的坐标向量, $\boldsymbol{u}_e^{ij}$ 表示向量 $\overline{\boldsymbol{u}}_e^{ij}$ 在物体坐标系下所对应的向量, $\boldsymbol{S}^{ij}$ 为单元坐标系下的单元形函数, $\boldsymbol{T}_1^{ij}$ 为向量从单元坐标系到物体坐标系的变换矩阵, $\boldsymbol{T}_2^{ij}$ 为从物体坐标系到单元坐标系的变换矩阵, $\boldsymbol{L}^{ij}$ 为布尔指示矩阵. 图1为物体变形后点 P 所对应的点 $\widehat{P}$ 的位置向量 $\boldsymbol{r}_P^{ij}$ 的描述.

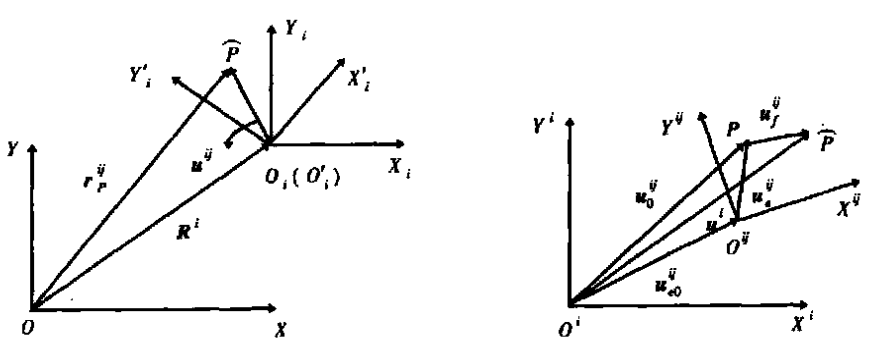

图1 平面系统中单元上一点的位置描述

由上述过程可看出, 多柔体系统动力学建模除需要描述刚体位形信息外, 由于有限元的引入, 使得系统自由度数大大增加, 其动力学方程系数和动力学系统参量的求解规模变得更为庞大. 有限单元系数矩阵的推导因需要进行复杂的体积分运算而变得相当繁杂. 譬如, 要表示出平面梁质量矩阵 $\boldsymbol{M}$ 及 $\boldsymbol{Q}_v$ 的一般性公式, 就必须引入23个变量符号; 如果考虑空间情况, 所引入的符号量还会翻倍(空间梁单元为48个). 如此多的符号参与式(2)及式(3)的混合代数及积分运算, 要推导出 $\boldsymbol{M}^{ij}$ 及 $\boldsymbol{Q}_v^{ij}$ 的公式, 手工演算实在难以胜任: 一方面, 公式推导过程的正确性无法保证; 另一方面, 手工推导所花费的时间及精力非人力所及. 文献[6]在Mathematica环境下实现了简单多柔体系统动力学的建模, 但由于它对相应的软件环境的依赖性太强, 不利于大型项目中的软件平台各个模块的集成,因此,必须研究出一种能够实现高级语言建模的方法.

2 用 Mathematica 推导公式存在不足

通过Mathematica或MATLAB软件, 可以很方便地完成所需的公式推导过程. 但是, 由该类软件所给出的

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

结果中函数及乘方的表达一般都不满足高级语言编程的需要; 同时, 对多柔体系统建模而言, 由于公式中所用到的符号数量规模较大, 且运算较为复杂, 因此, 符号演算结果的规模是非常庞大的. 以空间梁单元为例, 其质量矩阵的文本文件(由 Mathematica 给出, 其中只包含了质量矩阵中的下三角元素) 大小为 2. 364MB, 存储相应的 $\boldsymbol{Q}_v$ 的文件大小为 22. 71MB; 考虑空间板单元节点及自由度数更多, 所以结果更是惊人. 对如此庞大的文件, 如何将其编制成 FORTRAN(或其他语言) 程序代码, 将是一大难题.

3　符号处理程序模块功能及设计思路

为运用高级语言实现多柔体系统动力学建模过程, 笔者研究和开发了一个用于符号处理的软件模块, 该模块以 VC+ + 为开发工具, 基于对话框编制而成, 具有较良好的用户界面; 程序的编制中较好地考虑了内存空间的节省, 所能处理的文件大小直接与程序内部所设数组维数相关, 当文件过大时程序会自动调整内部数组维数并以对话框的形式给出相应的修改信息. 其主要功能为: 对 Mathematic 的公式推导结果进一步简化, 并将其直接转化为 FORTRAN 语言的源代码. 程序设计中着重考虑对公式的尽可能化简, 以减少系统程序的计算量及缩小系统程序的规模. 化简的步骤如下:

(1) 为缩减整体表达式的规模, Mathematica 演算过程中的数值多以浮点数给出, 浮点数取为双精度类型, 将结果中的浮点数以简单的变量取代将大大简化表达式的规模;

(2) 表达式中许多子表达式多次重复出现, 搜索子表达式并将重复次数较高的子表达式以变量取代, 一方面可以提高系统程序的执行效率, 而另一方面无疑也是对整体表达式的进一步大规模简化.

(3) 通过以上两个步骤, 对于多数的表达式, 其长度已经足够的小, 可以直接以 FORTRAN 语言格式输出. 但是, 由于还存在一些比较复杂的表达式, 其长度还是过长, 不满足 FORTRAN 语言格式要求(最大续行数为 19 行) , 需要通过重复第二步骤或采用拆分表达式的方法对其单独地进一步化简.

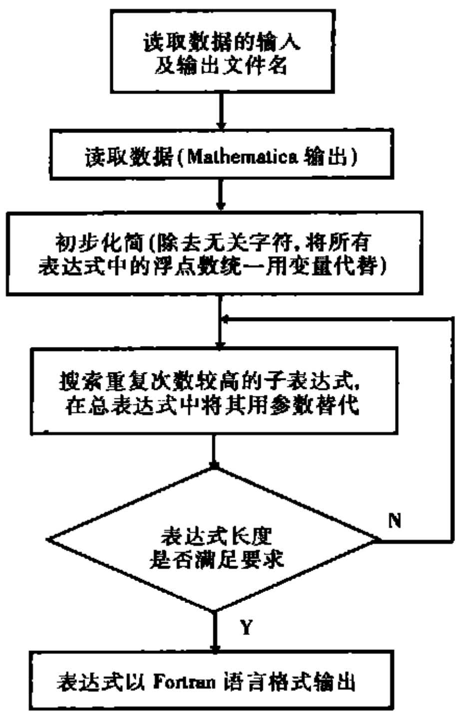

图 2　程序大致流程图

4　算　　例

上述程序模块是一个字符串处理程序, 必须对其进行大量的考题验证. 对多个模型及公式, 可将公式中一组任意变量值所对应的 Mathematica 数值结果与直接转化而生成的 FORTRAN 程序的运行结果进行比较. 考题中, 最长文件达 50MB; 同时, 为排除浮点数对数值精度的影响, 公式中都不包含浮点数. 结果表明, 文件中的表达式大大简化了(比如, 空间梁单元的相应公式中存储向量 $\boldsymbol{Q}_v$ 的文件由 22. 71MB缩减为 722kB) , 而且由以上两种方法得到的数值结果完全一致.

为了考查程序对浮点数的处理是否正确, 也进行了相关的考题验证, 下面仅给出其中的一个算例来说明该程序对浮点数的处理也是正确的. 任取一个由 Mathematica 演算所得到的向量的表达式如下:

```
{{gamai* (- (sin[agf]* sin[ags]* (b* (- 0.5* a* c1* qf2+ 0.08333333333333331* a* c1∧2*
qf5) * cos[ags]+ a* (0.5* b* c1* qf3- 0.08333333333333331* b* c1∧2* qf4) *
cos[agf]* sin[ags])) - cos[agf] * sin[ags] * ((a* b* c1* (c1+ qf1) * cos[ags]) / 2+
a* (- 0.5* b* c1* qf3+ 0.08333333333333331* b* c1∧2* qf4) * sin[agf]* sin[ags]) -
cos[ags] * (- (a* b* c1(c1+ qf1) * cos[agf] * sin[ags]) / 2+
b* (0.5* a* c1* qf2- 0.08333333333333331* a* c1∧2* qf5) * sin[agf]* sin[ags]))}  ,
{{gamai* (- (cos[agf] * (b* (- 0.5* a* c1* qf2+ 0.08333333333333331* a* c1∧2* qf5) *
```

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

cos[ags]+ a* (0.5* b* c1* qf3- 0.083333333333333331* b* c1∧2* qf4) * cos[agf] *
sin[ags])) + sin[agf] * ((a* b* c1* (c1+ qf1) * cos[ags])/2+
a* (- 0.5* b* c1* qf3+ 0.083333333333333331* b* c1∧2* qf4) * sin[agf] * sin[ags]))}} .

由符号处理软件模块得到的相应的FORTRAN格式程序代码为:

```
TMP(1)= SIN(AGF)
TMP(2)= SIN(AGS)
TMP(3)= COS(AGS)
TMP(4)= COS(AGF)
TMP(5)= 0.083333333333333331
TEST(1)= GAMAI* (- (TMP(1) * TMP(2) * (B* (- 0.5* A* CL* QF2+ TMP(
 $5) * A* CL* * 2* QF5) * TMP(3) + A* (0.5* B* CL* QF3- TMP(5) * B* CL* * 2* QF4) *
 $TMP(4) * TMP(2))) - TMP(4) * TMP(2) * ((A* B* CL* (CL+ QF1) * TMP(3))/2+ A
 $* (- 0.5* B* CL* QF3+ TMP(5) * B* CL* * 2* QF4) * TMP(1) * TMP(2)) - TMP(3) * (
 $- (A* B* CL* (CL+ QF1) * TMP(4) * TMP(2))/2+ B* (0.5* A* CL* QF2- TMP(5) * A
 $* CL* * 2* QF5) * TMP(1) * TMP(2)))
TEST(2)= GAMAI* (- TMP(4) * (B* (- 0.5* A* CL* QF2+ TMP(5) * A* CL
 $* * 2* QF5) * TMP(3) + A* (0.5* B* CL* QF3- TMP(5) * B* CL* * 2* QF4) * TMP(4) *
 $TMP(2))) + TMP(1) * ((A* B* CL* (CL+ QF1) * TMP(3))/2+ A* (- 0.5* B* CL* QF
 $3+ TMP(5) * B* CL* * 2* QF4) * TMP(1) * TMP(2)))    .
```

很显然,程序对表达式的处理中,对浮点数进行了有效的提取,将一些特殊的表达格式(如乘方,三角函数)进行了有效的转换,表达式最终也以向量的形式给出(该项在程序界面中由用户选择,也可以选为矩阵形式,但此时必须给出矩阵的列数).

5 结 论

与符号运算语言Mathematica相结合,实现了由多柔体动力学建模公式的推导直接到FORTRAN语言格式建模程序源代码生成的整个过程的"自动化",该部分工作仅仅是多柔体动力学软件开发的一个开始,有关柔性多体系统建模理论的研究以及相应的动力学控制方程求解方法的研究将成为后期的主要工作.

参考文献:

[1] Pradhan S, Modi V J, Misra A K. Order N Formulation for Flexible Multibody Systems in Tree Topology: Lagrangian Approach[J]. Journal of Guidance, Control, and Dynamics, 1997, 20(4): 665-672.

[2] Torby B J, Kimura I. Dynamic Modeling of a Flexible Manipulator with Prismatic Links[J]. Journal of Dynamic Systems, Measurement, and Control, 1999, 121(4): 691-696.

[3] 杨 辉,洪嘉振,余征跃. 刚-柔耦合多体系统动力学建模与数值仿真[J]. 计算力学学报, 2003, 20(4): 402-408.

[4] 杨 辉,洪嘉振,余征跃. 刚柔耦合建模理论的实验验证[J]. 力学学报, 2003, 35(2): 253-256.

[5] 孙怀安,杨广平. 基于虚拟现实语言的机器人三维仿真系统软件[J]. 西安电子科技大学学报, 2001, 28(3): 364-369.

[6] 冯 力,叶尚辉,刘明治. 多柔体系统动力学符号演算的研究[J]. 数学研究与评论, 2000, 20(1): 143-148.

[7] 盛 英,仇原鹰. 6腿支撑液压式平台自动调平算法[J]. 西安电子科技大学学报, 2002, 29(5): 593-597.

[8] 陆佑方. 柔性多体系统动力学[M]. 北京:高等教育出版社, 1996.

[9] 刘明治,刘春霞. 柔性机械臂系统动力学建模及控制研究[J]. 力学进展, 2001, 31(1): 1-8.

(编辑: 齐淑娟)

© 1994-2011 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net