计算微分代数系统的实时仿真算法*1)

罗新龙

(中国科学院计算数学与科学工程计算研究所)

刘德贵

(航天工业总公司第二研究院 204 所)

REAL–TIME SIMULATION ALGORITHMS FOR COMPUTING DIFFERENTIAL-ALGEBRAIC EQUATION

Luo Xinlong

(*Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing,* 100080, *P.O.*)

Liu Degui

(*Beijing Institute of Computer Application and Simulation Technology,* 100854)

Abstract

Differential-algebraic equations (DAE's) arise naturally in many applied fields, but numerical and analytical difficulties that have not appeared in ordinary differential equations (ODE's) occur in DAE's because it includes algebraic constrained equations. Some efficient numerical methods for ODE's can not work well for DAE's. So many eminent numerical analysis scholars are interested in this field recently. But few numerical methods are able to solve all DAE's because of its essential difficulties.

This paper discusses the simulation algorithm character of DAE's. And we construct an efficient constrained-algebraic algorithm based on the Runge-Kutta methods of order two for the semi-explicit differential-algebraic equations with index two and give the computational experiment results for specific examples. The experiment results indicate that the constrained-algebraic algorithm is high efficient for semi-explicit differential-algebraic equations with index two.

Key words: Runge-Kutta Methods, Simulation Algorithms, Differential-Algebraic Equations

* 1999 年 6 月 30 日收到.

1) 国家自然科学基金资助 (课题号 19731010) 和国防科技预研基金资助项目.

§1. 引 言

对于由微分代数方程所表示的动力系统的数值算法，针对微分代数方程的一些特殊形式已经构造了一些有效算法如文献 [1]–[4]. 这些数值算法大部分都是基于常微分方程的一些隐式公式如隐式 Runge-Kutta 方法，向后微分公式 (BDF) 等，因此这些算法都是非实时仿真算法. 如果我们直接用求解常微分方程的显式公式如显式 Runge-Kutta 方法，显式线性多步法等，虽然满足了实时仿真算法的一些特点，但是这些数值公式对微分代数方程的求解不甚理想. 由于一个实时仿真算法具有实时性、周期性、可靠性等特性要求，因此我们设计的实时仿真算法为了满足实时响应的特点，一般要求计算量不能太大. 这样所采用的算法一般是显式方法而且算法的收敛阶一般不高，如具有二阶或三阶收敛速度的算法. 我们对在航天控制领域中出现的指标为二的半显式微分代数方程构造了基于二阶显式 Runge-Kutta 方法的实时仿真算法，这个算法对在微分代数系统中出现的代数方程组具有三阶收敛速度. 我们分析了该算法的数值稳定性，并对潜地式弹道约束实时控制问题进行了实际仿真计算，理论分析以及数值结果表明所构造的算法对指标 (index) 为 2 的半显式微分代数系统的实时控制计算是非常有效的.

许多空间飞行器的轨道控制问题的抽象数学模型可以表示为显式的微分代数方程组. 飞行器的运动由常微分方程组

$$y' = f(t, y, u), \tag{1a}$$

$$y(t_0) = y_0 \tag{1b}$$

给出，飞行轨道还必须满足如下的代数约束方程组：

$$g(y) = 0, \tag{2}$$

其中 y 与 f 的维数为 l_1, u 与 g 的维数为 l_2, $l_2 \le l_1$, y 为状态变量或微分变量，u 称为控制变量或代数变量. 微分变量可以是飞行器的位置、速度、重量等. 控制变量 u, 它可以是攻角 α. 侧滑角 β, 控制姿态角 φ 等. 轨道控制问题为求控制变量 u, 使轨道 $y(t)$ 满足代数方程 (2). 即求解微分代数问题 (1)–(2). 如果函数 $f(t, y, u)$ 和 $g(y)$ 满足可微性条件而且矩阵 $g_y(y)f_u(t, y, u)$ 是非奇异的，y_0 满足 (2), 那么微分代数问题 (1)–(2) 的微分指标为 2 且存在唯一的连续可微解.

§2. 实时控制仿真算法的构造

在实时控制计算时，状态变量 y 的值由系统的实际运动状态采样得到. 设采样间隔为 h, 则采样时间序列为 $t_n = t_0 + nh$, $n = 0, 1, 2, \cdots$. 设在时间区间 $[t_n, t_{n+1}]$ 上的控制值为 $u(t) = u_n$ 在 t_n 处的采样值为 y_n, 则在 t_{n+1} 处的采样值 y_{n+1} 为初值问题

$$\frac{dy}{dt} = f(t, y, u_n), \tag{3a}$$

$$y(t_n) = y_n \tag{3b}$$

的解 $y(t)$ 在 t_{n+1} 处的值，即 $y_{n+1} = y(t_{n+1})$. 这里的所谓实时控制算法，是指在时刻 t_{n+1} 到来之前，完成系统在区间 $[t_{n+1}, t_{n+2}]$ 上的控制值 u_{n+1} 的全部计算的算法. 该算法在计算过程中可利用的信息只能是 t_{n+1} 以前采样得到的值. 在文 [5]–[6] 中分析了直接用 BDF

方法求解模型方程时所遇到的困难，并给出了一个有效的实时仿真算法．我们对在文 [5]–[6] 中提出的方法进行了推广．

我们先给出基于二阶显式 Runge-Kutta 公式且对代数方程 $g(y)=0$ 具有 3 阶收敛速度的实时仿真算法．假设 t_n 点的值 y_n 由系统采样得到，控制值 u_n 已经由计算机算出，需要计算 t_{n+1} 点的 u_{n+1} 的值以便在 t_{n+1} 时刻输入控制器，使得 y_{n+2} 尽量地满足代数约束方程 (2)．

算法 1．已知 t_n 处的采样值 y_n，控制值 u_n，按如下步骤计算 u_{n+1}：

(1) 由 y_n、u_n、计算 y_{n+2} 的预估值 y^p_{n+2}．

$$K_1 = f(t_n + 2c_1h, y_n, u_n), \tag{4}$$

$$K_2 = f(t_n + 2c_2h, y_n + 2a_{21}K_1h, u_n), \tag{5}$$

$$y^p_{n+2} = y_n + 2(b_1K_1 + b_2K_2)h, \tag{6}$$

其中 h 为时间步长．参数 c_i，$b_i(i=1,2)$，a_{21} 满足如下关系式：

$$b_1 + b_2 = 1, \tag{7a}$$

$$b_1c_1 + b_2c_2 = \frac{1}{2}, \tag{7b}$$

$$a_{21}b_2 = \frac{1}{2}. \tag{7c}$$

(2) 对未知量 Δu 解线性方程组

$$g(y^p_{n+2}) + hg_y(y^p_{n+2})f_u(t_n, y_n, u_n)\Delta u = 0, \tag{8}$$

记其解为 Δu_{n+2}．取

$$u_{n+1} = u_n + \Delta u_{n+2}. \tag{9}$$

由这个算法看出，只要在时刻 t_{n+1} 到来之前完成上述计算，那么它是实时的，因为它只用到 t_{n+1} 点之前的信息．

§ 3. 实时仿真算法的误差估计

由于 y_{n+1} 是实际采样值，因此满足如下的初值问题：

$$\frac{dy}{dt} = f(t, y, u_n), \tag{10a}$$

$$y(t_n) = y_n, \tag{10b}$$

其中 u_n 是实际输入的控制量．记 (10) 的解为 $y(t, t_n, y_n, u_n)$，并简写为 $y_{n+1}(t)$．在以下算法的构造及证明中假设函数 f, g 都具有所需的各阶偏导数，且矩阵 $g_y(y)f_u(t,y,u)$ 在解邻域里存在有界逆矩阵．

定理 3.1. 如果 y_0、u_0 满足

$$\|g(y_0)\| = O(h^2), \tag{11a}$$

$$\|g(y_1)\| = O(h^2), \tag{11b}$$

且函数 f, g 满足以上假定，y_n 由 (4)–(9) 给出的实时仿真算法计算，那么

$$\|g(y_n)\| = O(h^3),\ n = 2, 3, \cdots. \tag{12}$$

证明. 在以下的算法构造及误差估计中都是针对标量 y,u 来进行的，即 $l_1=l_2=1$. 为了叙述的方便，我们针对如下的自治系统:

$$y'=f(y,u), \tag{13}$$

$$g(y)=0 \tag{14}$$

给出证明，但结果对非自治系统也成立. 我们采用归纳法来证明，先证明 $\|\Delta u_2\|=O(h)$. 由于在 (4)–(7) 式中对 y_{n+2} 的预估值 y^p_{n+2} 的计算是利用二阶显式 Runge-Kutta 公式，因此 (4)–(7) 式等价于下式

$$y^p_{n+2}=y_n+2hf(y_n,u_n)+2h^2f_y(y_n,u_n)f(y_n,u_n)+O(h^3). \tag{15}$$

记 $g_y(y^p_{n+2})f_u(y_n,u_n)=A_{n+2}$, 由 (8) 式得

$$\Delta u_2=-A_2^{-1}g(y^p_2)/h. \tag{16}$$

由 (15) 式有

$$\begin{aligned} g(y^p_2) &= g(y_0+2hf(y_0,u_0)+2h^2f_y(y_0,u_0)f(y_0,u_0)+O(h^3)) \\ &= g(y_0)+2hg_y(y_0)f(y_0,u_0)+2h^2g_y(y_0)f_y(y_0,u_0)f(y_0,u_0) \\ &\quad +2h^2g_{yy}(y_0,u_0)f^2(y_0,u_0)+O(h^3). \end{aligned} \tag{17}$$

$y_1(t)$ 为以下方程

$$y'=f(y,u_0), \tag{18a}$$

$$y(t_0)=y_0(t_0)=y_0 \tag{18b}$$

的解. 所以

$$\begin{aligned} y_1(t_1)-y_0(t_0) &= y_1(t_0+h)-y_0(t_0) \\ &= y_1(t_0)+y_1'(t_0)h+\frac{1}{2}y_1''(t_0)h^2-y_0(t_0)+O(h^3) \\ &= f(y_1(t_0),u_0)h+\frac{h^2}{2}f_y(y_1(t_0),u_0)y_1'(t_0)+O(h^3) \\ &= f(y_0,u_0)h+\frac{h^2}{2}f_y(y_0,u_0)f(y_0,u_0)+O(h^3). \end{aligned} \tag{19}$$

以上利用了 $y_1'(t_0)=f(y_1(t_0),u_0)=f(y_0(t_0),u_0)=f(y_0,u_0)$. 因此

$$\begin{aligned} g(y_1)=g(y_1(t_1)) &= g(y_0(t_0))+hg_y(y_0)f(y_0,u_0)+\frac{h^2}{2}g_y(y_0)f_y(y_0,u_0)f(y_0,u_0) \\ &\quad +\frac{h^2}{2}g_{yy}(y_0,u_0)f^2(y_0,u_0)+O(h^3). \end{aligned} \tag{20}$$

由 (20) 式解出 $hg_y(y_0)f(y_0,u_0)$ 代入 (17) 式得

$$g(y^p_2)=2g(y_1)-g(y_0)+O(h^2). \tag{21}$$

因此，由 (16)–(21) 式以及对初值 $\|g(y_0)\|=O(h^2)$, $\|g(y_1)\|=O(h^2)$ 的假定得

$$\|\Delta u_2\|=O(h). \tag{22}$$

现在我们来证明 $\|g(y_2(t_2))\| = O(h^3)$，$y_2(t)$ 为以下初值问题的解

$$y' = f(y, u_1), \tag{23a}$$

$$y(t_1) = y_1(t_1), \tag{23b}$$

则

$$\begin{aligned} y_2(t_2) &= y_2(t_1 + h) \\ &= y_2(t_1) + hy_2'(t_1) + \frac{h^2}{2}y_2''(t_1) + O(h^3) \\ &= y_1(t_1) + hf(y_2(t_1), u_1) + \frac{h^2}{2}f_y(y_2(t_1), u_1)f(y_2(t_1), u_1) + O(h^3). \end{aligned} \tag{24}$$

由 (19) 式，那么 (24) 式可化为

$$\begin{aligned} y_2(t_2) = y_0(t_0) &+ f(y_0, u_0)h + \frac{h^2}{2}f_y(y_0, u_0)f(y_0, u_0) + O(h^3) \\ &+ hf(y_0, u_1) + h^2 f_y(y_0, u_1)f(y_0, u_0) + O(h^3) \\ &+ \frac{h^2}{2}f_y(y_0, u_1)f(y_0, u_1) + O(h^3). \end{aligned} \tag{25}$$

由于

$$\begin{aligned} f_y(y_0, u_1)f(y_0, u_1) &= f_y(y_0, u_0 + \Delta u_2)f(y_0, u_0 + \Delta u_2) \\ &= f_y(y_0, u_0)f(y_0, u_0) + O(\|\Delta u_2\|) \\ &= f_y(y_0, u_0)f(y_0, u_0) + O(h), \end{aligned} \tag{26}$$

因此 (25) 式可以化为

$$\begin{aligned} y_2(t_2) = y_0(t_0) &+ 2f(y_0, u_0)h + f_u(y_0, u_0)\Delta u_2 h \\ &+ 2h^2 f_y(y_0, u_0)f(y_0, u_0) + O(h^3). \end{aligned} \tag{27}$$

又根据 (15) 式可以把 (25) 式进一步化为

$$y_2(t_2) = y_2^p + hf_u(y_0, u_0)\Delta u_2 + O(h^3), \tag{28}$$

所以

$$g(y_2(t_2)) = g(y_2^p) + hg_y(y_2^p)f_u(y_0, u_0)\Delta u_2 + O(h^3). \tag{29}$$

由 (8) 式得

$$\|g(y_2(t_2))\| = O(h^3). \tag{30}$$

当 $n = 3, 4\cdots$ 时，只要重复 $n = 2$ 时的证明过程即可．定理得证.

§ 4. 实时仿真算法的稳定性分析

本节，我们将对实时仿真算法给出稳定性分析．模型方程选为

$$y' = Ay + Bu, \tag{31a}$$

$$y + C = 0, \tag{31b}$$

其中 A,B,C 为常数. 由于飞行器轨道约束实时控制实际问题的特点, y_{n+1} 是由以下方程

$$y' = f(t,y,u_n) = Ay + Bu_n, \tag{32a}$$

$$y(t_n) = y_n \tag{32b}$$

确定出的精确解 $y_{n+1}(t)$ 在 $t = t_{n+1}$ 的值, 所以 y_{n+1} 与 u_n, y_n 的递推式必须从 (32) 式精确解出. 这也是实时算法与非实时算法的区别之一. 由 (32) 式解出 $y_{n+1}(t)$ 得

$$y_{n+1}(t) = A^{-1}[(Ay_n + Bu_n)e^{-At_n}e^{At} - Bu_n], \tag{33}$$

所以

$$y_{n+1} = y_{n+1}(t_{n+1}) = e^{Ah}y_n + A^{-1}B(e^{Ah} - 1)u_n. \tag{34}$$

由算法 (4)-(9) 得到 u_{n+1} 与 u_n, y_n 的递推关系式

$$Bhu_{n+1} = -(1 + 2hA + 2h^2A^2)y_n - (2h^2AB + hB)u_n - C. \tag{35}$$

令 $x = \begin{bmatrix} y \\ u \end{bmatrix}$, 记 $\mu = hA$ 则得

$$x_{n+1} = \begin{bmatrix} e^{\mu} & A^{-1}B(e^{\mu} - 1) \\ -\dfrac{1 + 2\mu + 2\mu^2}{Bh} & -2\mu - 1 \end{bmatrix} x_n - \frac{C}{Bh} \tag{36}$$

因此 (36) 稳定的充分必要条件是其系数矩阵的特征方程

$$\lambda^2 + (2\mu + 1 - e^{\mu})\lambda + (1 + 2\mu + 2\mu^2)(e^{\mu} - 1)/\mu - e^{\mu}(2\mu + 1) = 0 \tag{37}$$

的根的模都小于 1. 当 $\mu \to 0$ 时 (2.37) 退化为

$$\lambda^2 = 0. \tag{38}$$

因而特征根为 0, 由方程 (37) 根与系数的连续依赖性, 当 h 较小时, 由算法 (2.7)-(2.9) 得到的控制过程是稳定的. 图 1 给出了代数约束算法的稳定区域图, 其中横坐标为 $\mathrm{Re}(\lambda)$, 纵坐标为 $\mathrm{Im}(\lambda)$. 从图中可以看出, 本文构造的实时仿真算法具有较大的稳定区域.

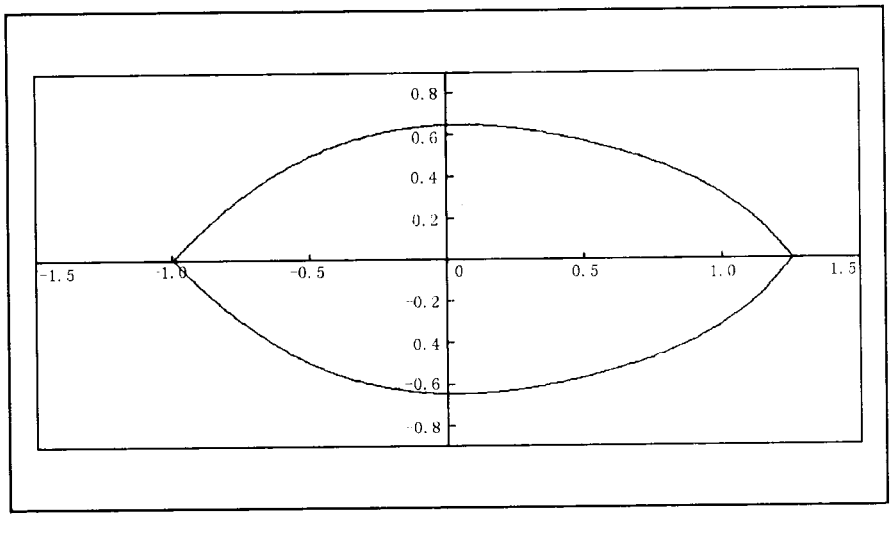

图 1

§ 5. 潜地式弹道约束控制问题的实时仿真计算

远程潜地弹道式弹道由于长度受到潜艇的限制，末级发动机与仪器舱并联配置，也就无法用打开反向喷管的方法来终止动力. 这样在有各种干扰作用的情况下，当导弹捕获到预定射程装订所对应的弹道诸元 (v,θ,x,y) 时，在发动机推进剂尚未耗尽又无法关机的情况下，我们如果还必须让导弹准确射中目标，就必须引入一个控制姿态角 φ 以及一个弹道代数约束方程 $g(v,\theta,x,y,\varphi)=0$ 使得导弹多余的能量消耗掉，又使得导弹能到达装订射程 L. 这就出现了实时控制的微分代数系统问题. 刘捷在文 [8] 中提出了“等射程段”的概念，即导弹在该段程序飞行时的任意瞬间耗尽推进剂，经被动段飞行后所得的射程都将是预定的射程. 我们对潜地式弹道导弹约束控制问题进行了实时仿真计算.

因为由 (4)–(9) 式给出的实时仿真算法对指标为 2 的半显式微分代数系统的收敛阶都为 3 阶收敛，而且这类算法对线性模型的稳定区域都一样大. 因此，在算法 (4)–(9) 式中我们以如下方式选取参数时，所得到的数值仿真算法的计算量很小.

$$b_1=0, \tag{39a}$$

$$b_2=1, \tag{39b}$$

$$c_1=0, \tag{39c}$$

$$c_2=\frac{1}{2}, \tag{39d}$$

$$a_{21}=\frac{1}{2}. \tag{39e}$$

考虑到微分代数问题的特点，$g_y(y)f_u(t,y,u)$ 的性态有可能不好，因此为了保证数值求解的稳定性，我们在解线性代数方程组 (8) 式时，采用 QR 分解算法求解. 表 1 是计算的数值结果，对数据的分析可以看出误差是 3 阶敛速的. 假设当 $t=151$ 秒时导弹进入等射程段飞行，可以发现表 1 的计算结果有些没有达到理论分析的误差阶. 这是因为实际数学模型方程的性态不是很好，而且在求解方程组 (8) 时，$g_y(y)$ 与 $f_u(t,y,u)$ 的值是用差商代替导数而得到. 如果方程的性态比较好则计算结果能够达到理论分析的误差阶，我们只要分析 5.1 节中对单摆模型的仿真计算结果表 2 就可以发现这一点.

表 1　潜地式弹道约束控制问题实时仿真计算的数值结果

	$G(Y)$	$G(Y)$	$G(Y)$	$G(Y)$
$t(s)$	$h=0.1$	$h=0.01$	$h=0.001$	$h=0.0001$
152	1.70230	0.00162933	6.06757E-07	−7.91624E-09
153	2.04166	0.00192814	4.10713E-07	−1.58325E-08
154	2.48826	0.00232502	1.92784E-07	−2.28174E-08
155	3.08548	0.00286395	−1.30385E-08	−2.84053E-08
156	3.90015	0.00361273	−1.57394E-07	−3.58559E-08
157	5.03661	0.00467919	−1.44355E-07	−4.79631E-08
158	6.66496	0.00624207	2.10013E-07	−5.9139E-08
159	9.07930	0.00861505	1.24425E-06	−7.21775E-08
160	12.82800	0.01239110	3.61353E-06	−8.42847E-08

5.1. 刚体运动的单摆例子

测试一个微分代数系统算法的好坏，有一个较流行的标准就是用这个算法去解指标为 2 的单摆模型 [1,9]. 指标为 2 的单摆模型是：

$$y_1' = y_3, \tag{40a}$$

$$y_2' = y_4, \tag{40b}$$

$$y_3' = -y_1 u, \tag{40c}$$

$$y_4' = -y_2 u - g_0, \tag{40d}$$

$$0 = y_1 y_3 + y_2 y_4, \tag{40e}$$

其中 $y = y(t)$, $u = u(t)$, $g_0 = 1$ 为地球重力加速度, $t = 0$ 时, $y(0) = (1, 0, 0, 1)^T$, $u(0) = 1$, 满足初始相容条件. 表 2 给出了这个问题的数值结果. 由于构造的是实时控制算法，因此评价一个算法是否有效主要是看输入 u_n 后，采样值 y_{n+1} 是否满足代数约束方程 $g(y_{n+1}) = 0$. 我们从表 2 中可以看出 $g(y)$ 的收敛阶至少是 3 阶的.

表 2 单摆的数值例子

	$G(Y)$	$G(Y)$	$G(Y)$	$G(Y)$
$t(s)$	$h = 0.1(s)$	$h = 0.01(s)$	$h = 0.001(s)$	$h = 0.0001(s)$
0.1	−0.014438800	5.81272E-08	5.54089E-12	5.41234E-16
0.2	0.001143380	3.78430E-08	3.57570E-12	3.33067E-16
0.3	0.000114977	2.26235E-08	2.11042E-12	2.22045E-16
0.4	0.000272824	1.18301E-08	1.07855E-12	1.11022E-16
0.5	0.000136180	4.67187E-09	4.00541E-13	2.77556E-17
0.6	6.06327E-05	3.55793E-10	−2.22045E-15	−2.77556E-17
0.7	1.18161E-05	−1.83245E-09	−1.99896E-13	−2.77556E-17
0.8	−1.38983E-05	−2.49724E-09	−2.51590E-13	0.000000000
0.9	−2.31399E-05	−2.13327E-09	−2.05780E-13	0.000000000
1	−2.11434E-05	−1.14422E-09	−1.01724E-13	−1.04083E-17

§ 6. 结　论

从以上的理论分析及数值结果可以看出，我们给出的实时仿真算法对指标为 2 的半显式微分代数系统的实时控制计算是非常有效的，而且这个算法具有很好的数值稳定性. 特别突出地是，这种算法由于只需解一个低维的代数方程组以及采用 (39) 式给出的参数求解微分方程组时，这个算法的计算量很小. 因此，我们所给出的实时仿真算法对航天领域中出现的应用背景问题 (1)–(2) 式是很有效的.

参 考 文 献

[1] K.E. Brenan, S.L. Campbell, L.R. Petzold, Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations, North-Holland, 1989.

[2] E. Hairer, C. Lubich, M. Roche, The Numerical Solution of Differential-Algebraic Systems by Runge-Kutta Methods Springer-Verlag, Lecture Notes in Mathematics, 1989.

[3] 费景高，微分代数问题的一类数值算法，计算数学， **16** :1 (1994), 47–58.
[4] W.C. Rheinboldt, Solving Algebraically Explicit DAEs with the MANPAK-Manifold-Algorithms. Computers Math. Applic. **33** :3 (1997), 31–43.
[5] 罗新龙，宋晓秋，刘德贵，实时控制计算微分代数系统的代数约束算法，系统工程与电子技术， 12 (1998). 66–71.
[6] 罗新龙，数值微分代数系统及飞行轨道约束实时控制计算方法，航天部二院 204 所硕士论文， 1998.
[7] 罗新龙，宋晓秋， BDF 方法解常系数微分代数方程的稳定分析，系统工程与电子技术， 5 (1998). 52–54.
[8] 刘捷，潜地弹道式导弹射程控制问题的研究，系统工程与电子技术， 5 (1987), 21–31.
[9] Timothy Maly, Linda R. Petzold, Numerical methods and software for sensitivity analysis of differential-algebraic systems, Applied Numerical Mathematics, 20 (1996), 57–79.
[10] S.L. Campbell, Singular Systems of Differential Equations , Pitman, 1982.
[11] Hisaoshi Shintani, Semi-explicit Methods for Differential-Algebraic Systems of Index 1 and Index 2, Numerical Analysis of Ordinary Differential Equations and its Applications, World Scientific. 1995.
[12] W.L. Feehery, J.E. Tolsma, P.I. Barton, Efficient sensitivity analysis of large-scale differential-algebraic systems, Applied Numerical Mathematics, 25 (1997), 41–54.
[13] S.L. Cambell, C.W. Gear, The index of general nonlinear DAEs, Numerische Mathematik, 72 (1995), 173–196.
[14] 刘德贵，费景高，韩天敏，刚性大系统数值仿真方法，河南教育出版社， 1996.
[15] 费景高，微分代数控制问题的实时计算，自动化学报， **19** :4 (1993), 385–392.
[16] 费景高，微分代数系统的实时控制计算，计算机工程与设计， 12 (1996), 3–10.
[17] 陈丽容，刚性大系统数值仿真并行算法，博士论文，北京计算机应用与仿真技术研究所， 1996.
[18] 远程火箭弹道学，航天二院图书馆内部资料.
[19] 袁兆鼎，费景高，防空导弹设计中的数值方法，宇航出版社， 1991.