

# ハイブリッドシステムモデリング言語 HyLa を用いた DAE システムの解析

渋井 隆弘<sup>†</sup> 上田 和紀<sup>†</sup>

早稲田大学 基幹理工学研究科 情報理工・情報通信専攻

## 1. ハイブリッドシステムモデリング言語 HyLa

時間の経過に伴って状態変数が連続変化と離散変化を起こすシステムをハイブリッドシステムと呼ぶ。物体同士の衝突や電気回路中のスイッチの開放などの現象は、ハイブリッドシステムとしてモデル化が可能である [1]。HyLa[2] は微分方程式及び代数方程式で表される制約式を記述することでハイブリッドシステムをモデル化するための宣言型言語である。HyLa 言語の処理系として HyLaGI[3] が開発されている。HyLaGI は C++ で開発されているソフトウェアで、方程式や不等式を記号的に解くためのバックエンドシステムとして Mathematica[4] を使用している。そのため LTspice[5] などの回路シミュレータとは異なり、各変数を時刻  $t$  についての数式として記号的に求めることができる。

## 2. 微分代数方程式

微分代数方程式 (Differential Algebraic Equations, DAE) は微分方程式と代数方程式を連立させたものである。次の形式で表される連立微分方程式を考える。

$$A\mathbf{x}'(t) + B\mathbf{x}(t) = F \quad (1)$$

ただし、 $A$ ,  $B$  は各要素が定数または独立変数  $t$  についての関数である行列、 $F$  は各要素が定数または独立変数  $t$  についての関数であるベクトル、 $\mathbf{x}(t)$  は  $t$  についての未知関数ベクトルである。このとき  $t$  の値に関係なく  $\det A = 0$  ならば DAE である。(そうでなければ常微分方程式である。) 本論文で扱う DAE は (1) の形式で表されるものに限る。インダクタやキャパシタを含むアナログ電気回路は (1) の形式で表すことのできる DAE システムである [6]。Mathematica の組み込み関数 DSolve は DAE を解くことができるが、DAE は適切な数の初期条件を与えることが容易でなく、与えるべき初期条件の数を間違えると解が得られないため利用する側が注意しなければならない。

## 3. HyLaGI における DAE の解析

HyLaGI は初期値問題を解くための工夫として、DSolve に与える初期条件を変数ごとに異なる記号パラメータで与え、得られた解に含まれる対応する記号パラメータを変数の初期値で置き換えるという方法をとっている。HyLaGI は次のような DAE を解こうとすると、解が存在しないと判定されてそこで実行が終了してしまう。

$$\begin{cases} x'(t) + y'(t) = x(t) \\ x'(t) + y'(t) = y(t) \\ 0 < x(0) < 2 \\ 1 < y(0) < 3 \end{cases} \quad (2)$$

(2) の解は存在し、 $x(t) = y(t) = x(0)e^{t/2} \wedge 1 < x(0) < 2$  である。問題が起こる原因は、Mathematica は  $x(0) = a \wedge y(0) = b$  のように異なる文字で与えられた初期条件について  $a$  と  $b$  は無関係なものとして扱い、 $a$  と  $b$  が何らかの関係 ((2) の例では  $a = b$ ) を持つ場合は解が存在しないと判定することになっているからである。

## 4. 提案手法

3章で述べた問題を解決するために、初期値問題を DSolve で直接求める方法と、一般解を DSolve で求めてから代数方程式を解くための Mathematica の組み込み関数 Solve を用いてその結果と初期条件から任意定数を消去する方法を HyLaGI に実装した。また、問題が解消されて実行できるようになった HyLa プログラムについて各方法ごとの実行時間を比較した。

### 4.1 初期値問題の解を直接求める方法

最初に、初期条件の集合の冪集合を求め、これを DSolve に与える初期条件の候補とする。求めた冪集合の要素のうち、要素数が多いものから DSolve に与える初期条件として用いて DAE が解けるか試行してゆき、最初に得られた解を用いて、与えなかった初期条件を充足できるか確認する。DSolve に与えた初期条件についての解が存在しないことは、短い時間で判断できる。初期条件を充足できるならば解であり、充足できなければ解は存在しないということになる。しかし、この方法は 4.2 節に例を示すように、DAE の形によっては効率が悪い。

Analysis of differential-algebraic equation systems using hybrid system modeling language HyLa

<sup>†</sup> Takahiro Shibui and Kazunori Ueda, Waseda University

4.2 一般解を求めてから初期条件を連立させて任意定数を消去する方法

$n$  を 2 以上の整数とすると、次の DAE は任意定数の数は 1 つのみである。

$$\sum_{i=1}^n x'_i(t) = x_j(t), \quad j = 1, \dots, n \quad (3)$$

なお、(3) で  $n = 2$  のときは (2) から初期条件を除いたものと一致する。(3) を 4.1 節の方法で解くと、解けるまでに DSolve を  $O(2^n)$  回試行することになるため効率が悪い。

そこで、一般解を求めてから任意定数を消去することで初期値問題の解を得る方法を考える。最初に、初期条件を与えずに DSolve を実行し一般解を求め、得られた一般解に含まれる任意定数の数 (= 解の自由度) を調べる。次に、解の自由度に等しい数の要素を持つ初期条件の部分集合の集合を求め、得られた集合の各要素と一般解と連立させた代数方程式を Solve を用いて任意定数についての解が得られるかどうか調べてゆき、最初に得られた解を一般解の任意定数に代入して残りの初期条件を充足できるか調べ、充足できたらそれが初期値問題の解である。Solve に与えた初期条件の集合が解けない場合は、DSolve と同様短い時間で解けないことを判断し次の初期条件の集合を試すことができる。この方法で (3) を解く場合、最大試行回数は  $n$  であり 4.1 節の方法よりも効率が良いと言える。

4.3 実装により実行可能となったプログラムおよび各方法ごとの実行時間の比較

4.1 節または 4.2 節の方法を実装することにより、(2) の初期値問題や図 1 のトランジスタ増幅回路をモデル化した HydLa プログラムを正常に実行できるようになった。どちらの方法でもすべての変数について同一の正しい解が得られた。図 1 において入力信号は  $vin = \sin at$ 、出力信号は  $vo$  である。出力信号  $vo$  の記号的な解は図 2 であり、角周波数  $a$  が変化すると出力信号の増幅率や位相の変化にどのような影響を与えるかを調べることができる。HydLa プログラムの文法および意味論については文献 [2] を参照されたい。2 つの方法の実行時間を表 1 に示す。4.2 節の方法は 4.1 節の方法と比べて 1/10 程度の時間で実行を完了することがわかった。差が生まれた理由は 4.1 節と 4.2 節で述べた通り、解が得られるまでの初期値を与える試行回数之差が Mathematica の処理の負荷に影響を及ぼしたためであると考えられる。

5. まとめと今後の課題

HydLaGI に DAE システムを解析するための実装を 2 通り試し、どちらの方法でも (2), (3) の DAE について正しい解を得ることができた。Mathematica を扱う上では初期値問題を直接解くよりも一般解を求めてから任意定数を代

```
R(v,i,r) <=> [(v=r*i).
L(v,i,l) <=> [(v=l*i')].
C(v,i,c) <=> [(c*v'=i).
VIN <=> [(vin=sin(a*t))].
PAR(a,b,c) <=> a<b<c & [(b'=0).
INIT <=> v12'=0 & v2'=0 & v5'=0.
CIRCUIT <=> [(vin+v11+v12=v3 & v3+v6=12
& vo+v2+v7=12 & v4=v5 & v4=v3 & v4+vce=vo+v2
& i1+i6=i3+ibe & ice=i2+i7 & ibe+ice=i4+i5).
TRANSISTOR(v,i,g) <=> [(i=g*v).

R(v11,-i1,5), C(v12,-i1,5), R(v3,i3,20), R(v6,i6,20),
R(vo,-i2,20), C(v2,-i2,5), R(v7,i7,5), R(v4,i4,3),
C(v5,i5,10), VIN, PAR(0.5,a,1.5),
INIT, CIRCUIT, TRANSISTOR(ibe,ice,120).
```

図 1 トランジスタ増幅回路

```
(6000*a*(-46223846*(139129+625*a^2*(1225369+90000
*a^2))*E^(((5531+5*Sqrt[1216417])*t)/3000)+(1+15625
*a^2)*(22500*a^2*(169209686785+153405767*Sqrt[1216417]
+(169209686785-153405767*Sqrt[1216417])*E^((Sqrt[1216417]
*t)/300))+139129*(23111923-11941*Sqrt[1216417]+(23111923
+11941*Sqrt[1216417])*E^((Sqrt[1216417])*t)/300)))
-1150122273500*a*E^(((1111+Sqrt[1216417])*t)/600)
*(30*a*(2107+3448125*a^2)*Cos[a*t]+(-373+125*a^2
*(9979+225000*a^2))*Sin[a*t]))/(2300244547*(1+15625
*a^2)*(139129+625*a^2*(1225369+90000*a^2))
*E^(((1111+Sqrt[1216417])*t)/600))
```

図 2 vo の解

表 1 図 1 の実行時間の比較

HyLaGI に実装した方法	実行時間 [秒]
4.1 節の方法	445.81138
4.2 節の方法	47.744469

数的に解く方が効率よく解を求めることができることが示された。しかし扱った問題で初期値や素子定数を不等式で与えるなど、より複雑な場合は Mathematica の処理能力が及ばずに解を求めることに失敗する。Maple などの記号実行に強い数式ソルバを Mathematica と併用することができれば HyLaGI が扱える DAE の範囲を広げることができると考えられる。

参考文献

- [1] Edward A. Lee : Constructive Models of Discrete and Continuous Physical Phenomena, *IEEE Access*, Vol. 2, pp. 797–821 (2014).
- [2] 上田和紀, 細部博史 : ハイブリッド制約言語 HydLa の宣言的意味論, コンピュータソフトウェア, Vol. 28, No. 1, pp. 306–311 (2011).
- [3] 松本翔太 : Validated Simulation of Parametric Hybrid Systems Based on Constraints, 博士論文, 早稲田大学大学院基幹理工学研究科 (2017).
- [4] Wolfram: Mathematica, (online), available from (<https://www.wolfram.com/mathematica/index.ja.html>) (accessed 2019-01-10).
- [5] Analog Devices: LTspice, (online), available from (<https://www.analog.com/jp/design-center/design-tools-and-calculators/ltspice-simulator.html>) (accessed 2019-01-10).
- [6] 岩田覚, 高松瑞代 : 電気回路の混合解析における微分代数方程式の指数最小化, 数理解析研究所講究録, Vol. 1629, pp. 104–114 (2009).