

メカ設計における状態遷移表



新 誠一

電気通信大学
情報理工学研究科 知能機械工学専攻
教授

1980年 東京大学工学部計数工学科助手
1988年 筑波大学電子・情報工学系助教授
1992年 東京大学工学部計数工学科助教授
2006年 電気通信大学電気通信学部 システム工学科教授

■はじめに

私自身は制御工学を大学院時代から研究している。DDC(Direct Digital Control)と呼ばれ、大型の計算機を圧延や抄紙の装置とつないだ制御から始まり、マイコンの普及とともに自動車や家電の制御まで研究領域が拡大した。後者は油圧系やモータなどを電子制御するものであり、組込み系と呼ばれる最近注目を浴びている分野でもある [1]。

組込み系では制御対象としてメカ系を扱うことが多い。これは物理系であり、エネルギー保存則や運動量保存則などの自然法則で表わされる。具体的には、連続時間連続値の微分方程式である。一方、計算機のソフトはC言語などで記述される離散時間、有限語長の世界である。いわば、プログラム言語で記述される言語である。

この「法則と論理」という対比が象徴する「連続時間と離散時間」、そして「連続値と有限値」という二つの表現手段の対立が組込み系の開発を阻害してきた。メカ、すなわち法則に立つ旗頭が制御工学であり、ソフト、すなわち論理に立つ旗頭が情報工学である。

メカ設計に於いては両者統合される必要がある。ここでは、学問の立場から両者の融合の道を探りたい。キーワードは「状態」である。



1. ソフト

計算機はビットが基本単位である。つまり、真または偽、1または0という論理単位であるビットを組み合わせた世界である。演算の基本はAND、OR、NOTであり、ご承知のようにANDとNOT、またはORとNOTの組み合わせでスイッチング論理は全て表現できることが知られている。もっとも、スイッチング論理とは一階述語論理の世界であり、記憶は伴わない。つまり、「if A then B」であり、「AならばBを行う」というものである。熱いものに触れたら手を引っ込めるという反射行動である。これは組込み系では表引きという形で多用されている。もっとも、これを論理と呼ぶのは寂しい。それが「一階」と呼ばれる所以でもある。

実はBを行ったことでAに影響を及ぼすことがある。これを二階述語論理という。たとえば、if $X > 0$ then $X = -X$, if $X < 0$ then $X = X + 1$ という論理があり、 $X = 1$ を初期値として、この論理を一定周期ごとに評価すると変数 X は、1, -1, 0, 1, -1, …、と同じパターンを繰り返すことになる。これは反射よりレベルが高い。

X が-1であるとき真、つまり1となる変数を x_1 とす

る。もちろん、偽のときは0をとるとする。同様に x_0, x_1 を定義し、それらをまとめてベクトル $x(k)$ とする。

$$x(k) = (x_1 \ x_0 \ x_1)^T$$

ただし、 k は論理を評価する周期を表わす。つまり、 $x_1 \ x_0 \ x_1$ は周期 k における値である。この $x(k)$ を用いると、先に上げた二階述語論理は以下のように表わされる。

$$x(k+1) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} x(k)$$

これは状態遷移方程式と呼ばれるものであり、 $x(k)$ は状態と呼ばれる。ここでは簡単のため、入力も出力も考えていない。入出力まで考慮すると、一般に、

$$x(k+1) = Fx(k) + gu(k)$$

$$y(k) = Hx(k)$$

と表わされる。もっとも、これは線形系と呼ばれる基本的なものである。これを非線形系まで拡張すると、

$$x(k+1) = f_d(x(k), u(k))$$

$$y(k) = g_d(x(k))$$

と表わすことができる。ここで、 $f_d(\bullet)$ 、 $g_d(\bullet)$ は適当な

関数である。これが非線形の状態方程式表現と呼ばれるものである。

この表現をソフトウェアと考えると、 $u(k)$ はソフトウェアに渡す引数、 $y(k)$ は帰りの値となる。また、 $x(k)$ は内部変数である。つまり、論理は全て、このような状態方程式表現で表わすことが可能である [2]。

2. メカ

メカ系の基本は運動方程式である。つまり、質量×加速度=力である。加速度は位置を二階微分したものである。微分方程式となる。制御工学では微分方程式を以下の状態方程式表現で表わす。

$$\dot{x}(t) = f_c(x(t), u(t))$$

$$y(t) = g_c(x(t))$$

ただし、 t は時間であり、連続量である。そして、 $\dot{\bullet}$ は時間微分を表わす。これを線形化した、

$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t)$$

が連続時間形式線形状態方程式表現と呼ばれるものであり、制御工学の基礎的な表現となっている。

非線形でも線形でも、このような表現がモデルと

呼ばれており、自動車業界で中心的に使われているMBD(Model Based Development)の出発点である [3]。メカの状態方程式表現を求めることをモデリングと呼んでいる。

このモデルに基づいて、制御アルゴリズムを設計する。これをデザインと呼んでいる。たとえば、線形状態方程式表現されたモデルに対して、

$$u(t)=Fx_m(t)+gr(t)$$

$$\dot{x}_m(t)=Ax_m(t)+Bu(t)+K(y(t)-Cx_m(t))$$

のようなオブザーバー・状態フィードバック系制御則を構築する [4]。ここで、 $r(t)$ は指令信号である。また、設計パラメータは、 F, g, K である。

もちろん、このままではマイコンに実装できない。そこで、微分を差分で近似する。具体的には上の制御アルゴリズムの状態遷移方程式部分を以下のようにする。

$$u(k)=Fx_m(k)+gr(k)$$

$$x_m(k+1)=x_m(k)+T(Ax_m(k)+Bu(k)+K(y(k)-Cx_m(k)))$$

ここで、 T は制御周期であり、第1章で論じた一定周期である。また、 $x_m(k)$ などは、時刻 $t=kT$ における $x_m(t)$ の値などである。

これが時間の離散化である。さらにマイコンでは有限語長しか扱えないので値の量子化を行う。具体的には、16ビットの固定小数点化または32ビットの浮動小数点化を行う。さらに、起動・停止の論理や異常時のハンドリングの論理を加えて制御アルゴリズムとして実装する。このモデリング、デザイン、実装にテストまでを含めてMBDという言い方をしている。



3. メカと論理の融合

以上見てきたように、状態方程式表現という視点から見るとソフトもメカも近いところまで来ていることが分かる。第2章で論じた時間の離散化についてはリフティングという考え方をを使うことで、離散化によってどの程度の誤差が生じるか理論的に見積もることが可能となっている。それは、制御可能性だけでなく制御周期の設定も理論的に計算できるといことである。また、非線形と線形の関係についても長い時間をかけた研究から、両者の関係、線形化によって生じる誤差の見積もりなどができている。

この中で、制御工学的観点から足りないものは量子化による影響である。ここも、量子化誤差を外乱ととらえて安定論や情報理論の立場から最適な量子化幅を求める理論が盛んに研究されている。もっとも、これは量子化研究の入り口である。ここで紹介した状態方程式表現が量子化以前と量子化後にどのような関係にあるかの研究が次に期待される。

一方、ソフトウェア工学の観点から見ると、ソフトウェアが状態方程式だという視点は一般的ではない。現在、形式検証が注目されているが、それは人工知能的、いいかえればif then的な論理を基本にした検証である。このため、検証に膨大な時間がか

かることが問題である。我々は、半導体の検査において、人工知能的な手法を状態方程式の観点から見直すことでテストパターン生成が効率化できることを示している。この生成法をソフトウェアに展開することで、テストパターン生成を効率化できる可能性がある。

連続値と離散値が混在するシステムをハイブリッドシステムと呼んでいる。この扱いは、
a)両者を統合した理論の構築。
b)離散値システムとして全体を扱う。
c)連続値システムとして全体を扱う。
の三つの手法しかない [5]。状態遷移表は b) の立場であり、MBDは c) の立場を原点にしている。この両者を統合化していく組込みシステム理論の構築は a)に通じるものである。

4. まとめ

以上、組込み系を理論的な立場から考察した。理論の進展は産業界から見れば遅々たるものである。今、現実の課題を解決してくれるツールをユーザー諸氏は望んでいると思う。ユーザーの立場に立って使いやすいツールを開発していくとともに、理論の進展、理論の目指す方向を視野に入れて革新的なツールへとつなげていただくことを期待している。

参考文献

- [1] 新誠一:制御理論の変遷と自動車への適用拡大、Engine Technology, vol. 6, no. 6, pp. 104-108 (2004)
- [2] 新誠一:ソフトウェアの標準化活動とシステム理論、システム/制御/情報, vol. 49, no. 7, pp. 254-259 (2005)
- [3] 新、生産と設計をつなぐMBD、カーエレクトロニクス・テクノロジー(新誠一監修)、工業調査会, pp. 12-22 (2009)
- [4] 木村、美多、新、葛谷、制御系設計理論とCADツール、コロナ社、1998
- [5] 新、池田、湯浅、藤田、自律分散システム、朝倉書店(1995)