

組み込み制御ソフトウェア開発のための Simulink モデルから UML モデルへの変換ツール

神山 達哉 横山 孝典 志田 晃一郎 兪 明連

東京都市大学

1. はじめに

組み込み制御ソフトウェア開発においてモデルベース開発が急速に普及してきている。モデルベース開発では、システムをモデルで表現し、それを検証しながら開発プロセスを進めていく。

一般的に、組み込み制御ソフトウェア開発は制御設計とソフトウェア設計の二段階で行われる。制御設計では、制御系 CAE/CAD ツールである MATLAB/Simulink[1]を用いて、制御ロジックをブロック線図モデル(以下:Simulinkモデル)で設計することが多い。ソフトウェア設計では、制御ロジックに基づいて制御プログラムをUML(Unified Modeling Language)[2]を用いて設計する。

しかし、制御設計の成果物であるSimulinkモデルとソフトウェア設計で使用されるUMLモデルの間には明確な関連性がないという問題がある。そこで、Simulinkモデルを用いた制御設計からUMLを用いたソフトウェア設計への自然な移行方法として、SimulinkモデルからUMLモデルへの変換方法の確立が求められている。

変換の方法として、Simulinkモデルの要素をそれぞれクラスとしてUMLモデルに変換する手法が提案されている[3]。この手法では、Simulinkモデルをそのままオブジェクトとして表現することになり、Simulinkモデルの構造がそのままUMLモデルに反映される。そのため、必ずしもソフトウェア設計において適したオブジェクト構造にはならず、UMLモデルへの新たな機能追加や部品化再利用は容易ではない。

そこで本研究では、Simulinkモデルから機能追加や部品化再利用に適したソフトウェア構造になるUMLモデルへの変換方法を提案するとともに、その変換方法に対応した変換ツールを開発する。

2. UML モデル

本研究で対象とする UML モデルはクラス図とオブジェクト図とする。クラス図は UML の最も基本となる図で、ソフトウェア設計において最も多用される重要な静的構造図である。クラス図ではクラス、属性、クラス間の関連を記述することが可

A Simulink-UML Translator for Embedded Control Software Development
Tatsuya Kamiyama, Takanori Yokoyama,
Kouitiro Sida and Myungryun Yoo
TOKYO CITY UNIVERSITY

能である。

オブジェクト図は、システムが動作しているある時点でのオブジェクトとオブジェクト間の関係を表現するための図である。ところが、多くの組み込み制御システムでは、オブジェクトを動的に生成する必要はなく、固定的なオブジェクト構成になることが多い[4]。したがって、オブジェクト図を用いてシステムの定常的なオブジェクトとオブジェクト間の関連を表すことが可能である。

3. 変換方法

自動車車間距離システムを例にモデルの変換方法を説明する。図1にスロットル開度を算出する部分のSimulinkモデルを示す。

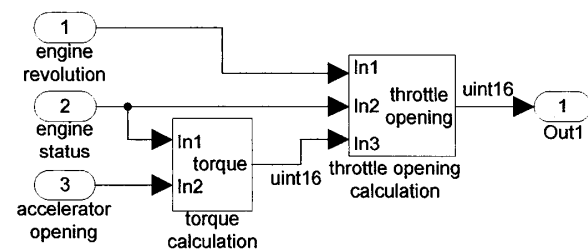


図1 階層化した Simulink モデル

Simulinkモデルの設計段階で、制御的に意味のあるデータを抽出し階層化する[5]。Simulinkが提供しているSubsystemブロックを用いることで階層化することが可能である。上位階層では、Inportブロック、Outportブロック、Subsystemブロックの3種類のブロックでシステムをモデル化する。そうすることで、Simulinkモデルを整理された再利用性の高い構造にすることができる。

図1のモデルではエンジン回転数(engine revolution)、エンジン状態(engine status)、アクセル開度(accelerator opening)、トルク(torque)、スロットル開度(throttle opening)の5つを制御的に意味のあるデータとして階層化している。5つのデータのみが上位階層に現れ、トルクとスロットル開度を求める具体的な処理はそれぞれのSubsystemブロックの下位階層で行う。

次にクラス図への変換方法を説明する。図1のSimulinkモデルを変換したクラス図を図2に示す。階層化したSimulinkモデルで上位階層に現れる

データに対応させてクラスを作成する。Simulink モデル内で値を算出しているデータのクラスには属性とメソッドを記述する。Simulink モデルでデータの型が明示されている場合には、属性に型情報を付けて記述する。メソッドとして、値を渡すための get メソッドと値を更新するための update メソッドを設ける。また、Simulink モデルでのデータの流れに対応させて、関連を記述する。以上により、Simulink での処理をクラス図で表現することができる。

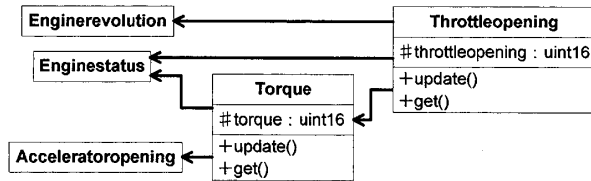


図 2 クラス図

図 2 のクラス図では、トルクはエンジン状態とアクセル開度、スロットル開度はエンジン回転数とエンジン状態とトルクを参照する。トルククラスの update メソッドでは、参照先の値を用いてトルクの値を算出し属性値に記憶する。スロットル開度クラスでも同様に値を算出し記憶する。これにより、Simulink モデルに対応したクラス図を実現できる。

次にオブジェクト図への変換方法を説明する。図 1 の Simulink モデルから変換したオブジェクト図を図 3 に示す。

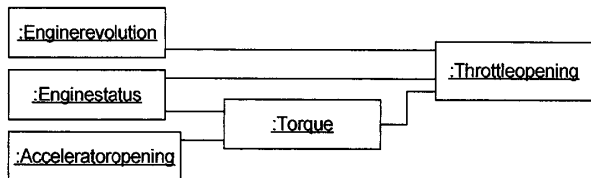


図 3 オブジェクト図

Simulink モデルのデータに対応させてオブジェクトを作成し記述する。また、Simulink モデルのデータの流れに対応させて、オブジェクト同士の関連を示すリンクを記述する。

図 3 のオブジェクト図では、Simulink の 5 つのデータに対応させてオブジェクトを作成し、それらオブジェクト間はリンクで繋いでいる。リンクの繋がりには図 2 のクラス図とも対応している。このオブジェクト図により、クラス図では表わせない実際のオブジェクト間のつながりを表現できる。

4. 変換ツール

以上提案した変換処理を自動化するための変換

ツールを開発した。開発した変換ツールは、Simulink モデルの mdl ファイルを入力し、変換後の UML モデルの XMI (XML Metadata Interchange) ファイルを出力する。図 1 の Simulink モデルを変換ツールに読み込ませることで、図 2 のクラス図と図 3 のオブジェクト図を得ることができる。

5. 考察

本研究で提案した手法は、制御設計では制御機能にのみ特化して設計し、ソフトウェア設計に移行した後で、タイミング設計やスケジューリング設計などソフトウェア固有の設計を行うことを前提としている。このため、Simulink モデルにおいてスケジューリング設計などがされていて、ブロック線図が一貫して繋がっていないモデルに対しては、うまく階層化できないことがある。しかし、部品化再利用に適した組み込み制御システムを実現するには、前述の順序で開発を行うべきであると考えられる。

開発した変換ツールを、いくつかの自動車制御モデルに適用し、組み込み制御ソフトウェア開発に使用できることを確認した。

6. おわりに

本研究では、組み込み制御ソフトウェアを対象に、Simulink モデルから UML モデルへの変換方法を提案し、その変換方法に対応した変換ツールを開発した。今後の課題は、Simulink モデルを階層化する作業もツールで行えるようにすることである。

謝辞

本研究は研究費 (20500037) の助成を受けたものである。

参考文献

- [1] <http://www.mathworks.com>
- [2] Object Management Group Unified Modeling Language Specification, ver. 2.0, 2006.
- [3] Klaus D. Müller-Glaser, Gerd Frick, Eric Sax and Markus Kühl, Multiparadigm Modeling in Embedded Systems Design, IEEE Transactions on Control Systems Technology, Vol.12, No.2, pp.279-292, 2004.
- [4] 成沢 文雄, 納谷 英光, 横山 孝典, 組み込み制御システムのためのオブジェクト指向コード生成ツール, 情報処理学会論文誌, Vol.46, No.5, pp.1306-1317, 2005.
- [5] 吉村 健太郎, 宮崎 泰三, 横山 孝典, オブジェクト指向組み込み制御システムのモデルベース開発法, 情報処理学会論文誌, Vol.46, No.6, pp.1436-1446, 2005.