

非瞬間イベントとその時間関係の表現が可能な

4 Z C - 1 0

状態遷移図エディタ

宮宗 正樹 ガミニ ウィジャヤラトナ 前川 守

電気通信大学大学院情報システム学研究科

1 はじめに

ソフトウェア開発にオブジェクト指向言語が使用されるにともない、分析・設計段階にもオブジェクト指向が導入されることが多くなってきている。このオブジェクト指向分析・設計で用いられる主な記述法にはOMT[2]の記述法や、UML[3]などが存在している。これらの記述法において動的な振舞いについて記述するにはUMLでは状態図やアクティビティ図、シーケンス図、コラボレーション図が、OMTでは動的モデル図が主に使用されている。

これらの図を含め、UMLなどで使用されている図は開発者と顧客との意志疎通におけるギャップを少なくするために使用される。ソフトウェア開発技術を持たないエンドユーザーはわかりやすさを、開発者は正確で簡潔なものをこの記述方法に求める。

一方、制御系やネットワーク系などのリアルタイムシステムソフトウェアを構築するには細かいイベントの時間の認識が必要である。しかし、オブジェクト、システム、サブシステムなどの状態の遷移を表現する場合において状態図において細かい時間の関係の表現は難しく、その他の表現、つまりイベントトレース図(OMT)、シーケンス図(UML)またはインフォーマルな文章説明を併用しなければならない場合がある。

2 非瞬間イベント

通常の瞬間的なイベントだけでは細かいイベントの時間表現は難しい。そこで、ここでは瞬間的ではない時間幅を持ったイベントを非瞬間イベントと呼ぶ。そして、通常の瞬間的なイベントは瞬間イベントと呼ぶ。

普通の状態遷移図などでは非瞬間イベントに相当するものを記述する必要に迫られた場合、開始イベントと終了イベントの2つに分割することによって記述される。しかし、これは表現を複雑にする。例えば、通常は

ButtonのイベントはButton PressとButton Releaseの2つに分割されて記述されるが、Buttonに対するイベントで細かな時間指定を必要としない場合はこの方法は煩わしいものである。

以下に、非瞬間イベントとして分類すべきものを列挙する。

- 瞬間イベント p , q が存在し、 p が発生した後に必ず q が発生することが保証されていて、かつ p が発生してから q が発生するまでの間を1つのイベントとして扱いたい場合。例えば、あるボタンが押されてから離されるまでを1つのイベントとして扱いたい場合。
- あるオブジェクトの状態を別のオブジェクトに伝える場合。通常は状態変数によって示すが、これを非瞬間イベントと記述した方が理解しやすい場合がある。例えば、交通信号機と自動車のシミュレーションを想定した場合、信号機オブジェクトの発光状態を自動車オブジェクトにイベントとして伝えるように記述したいと言う欲求が出るのは自然なことである。

3 提案する拡張状態遷移図

我々が提案する拡張状態遷移図は、UMLで使用される状態図をもとに、非瞬間イベントとその時間関係などを表現可能なように拡張する。

- 非瞬間イベントの表現 通常の瞬間的なイベントを普通の矢印つき線で表すのに対し、非瞬間イベントは始点に丸印のついた矢印つき線で表す(図1)。

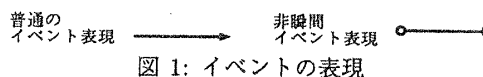


図1: イベントの表現

- 非瞬間イベントの時間関係の表現 非瞬間イベント同士の時間関係をユーザにわかりやすい形で表現する。例えば2つのイベントの関係の場合は図2のように表す。この2つのイベントの時間関係の分類は[1]による。

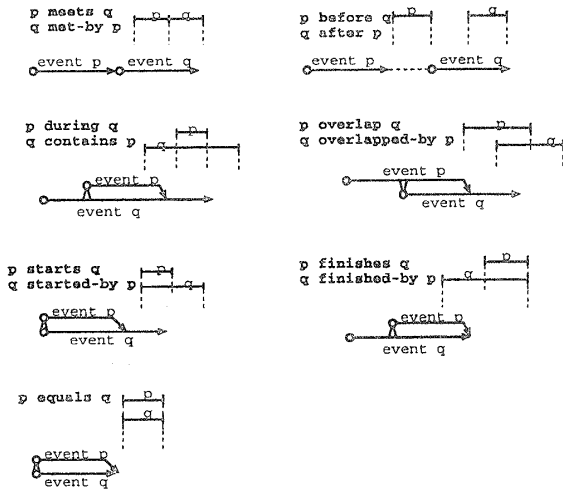


図 2: 非瞬間イベント同士の時間的関係の表現

● その他 状態は基本的にアクティビティ図と同じで、角丸の四角に状態名と状態変数とアクティビティを記述する。ただし、アクティビティ区画の基本イベント entry, do, exit のうち、do イベントのアクション表記に非瞬間イベントの発生を指定できる。それを指定した場合、その状態である間はその非瞬間イベントを (指定した場合は) 指定のオブジェクトへ発信し続けることになる。図 3 は信号機の状態遷移の例である。

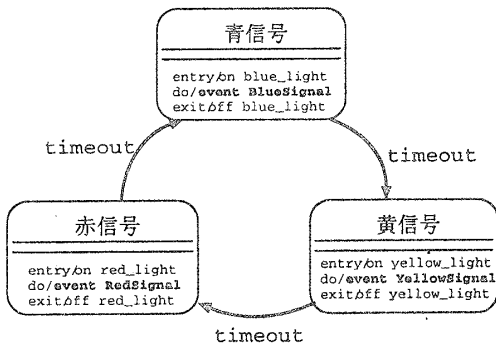


図 3: 信号機の状態遷移

4 状態遷移図エディタ

前述した図法を GUI で記述できるエディタを Tcl/Tk で作成した (図 4)。この画面例では、赤信号になってから白線を超え始めて、白線を超え終わってから青信号になった場合はその車を追跡する。白線を超えている最中に赤信号から青信号になった場合は警告のみするという状態遷移を表している。なお、図中の OverWhiteLine は、自動車白線を越え始めてから越え終るまでを表現する非瞬間イベントを表す。

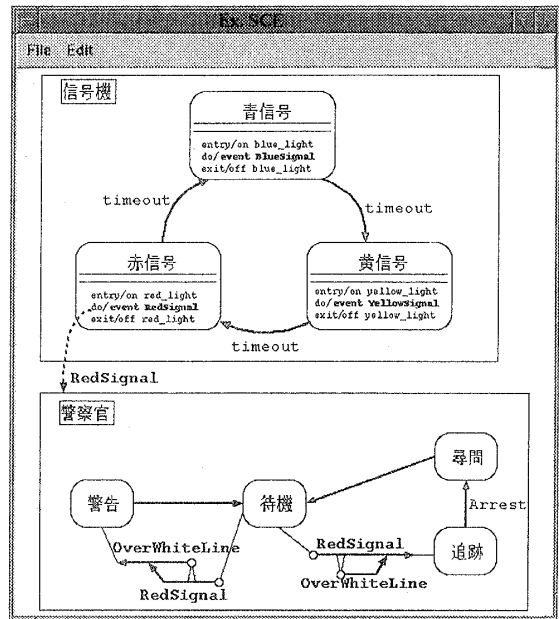


図 4: 画面例

5 今後の課題

- 例えば、ネットワークにおいてクライアントがサーバに対して要求を出した場合、要求は少し遅れてサーバに到達する。このような要求をサーバ側から見た場合、イベントとしてみなせる。これを非瞬間イベントとしてみる方法を現在模索中である。
- 記述した拡張状態遷移図を検証するためのシミュレーション機能の追加。
- 通常の UML への自動変換機能の実装。この場合、アクティビティ図を中心に、それをシーケンス図などによって補完した出力を計画している。

参考文献

- [1] James F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, Vol. 23, pp. 123-154, 1984.
- [2] James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorenzen. *Object-Oriented Modeling And Design*. Prentice Hall, 1991. (日本語訳) 羽生田栄一監訳, 「オブジェクト指向方法論 OMT—モデル化と設計」, トッパン, 1992.
- [3] Jim Rumbaugh, Ivar Jacobson, and Grady Booch. *Unified Modeling Language Reference Manual*. Addison-Wesley Object Technology Series. Addison-Wesley, circa summer 1998.