

## 時間オートマトンの振舞いに基づくスケジューリング解析

結 縁 祥 治<sup>†1</sup>

### Scheduling analysis based on behavior of timed automata

SHOJI YUEN<sup>†1</sup>

#### 1. はじめに

スケジューリング解析は、実時間処理において長く研究されている分野の一つである。解析の対象となる表現としてタスク実行に対するタイミングチャートが用いられ、スケジューリング手法の評価はテストセットによって示されることが多い。これに対して、時間オートマトン<sup>1)</sup>の概念をスケジューリング解析に適用し、スケジューリング可能性を解析する手法が提案されている<sup>2),3)</sup>。ここでは、本質的に連続値である時間の振舞いを有界に離散化 (discretization) することで、スケジューリング可能性をオートマトンによる記号実行の到達可能性に帰着する。このような手法は、有界に離散化可能であれば柔軟にスケジューリングの問題を定式化することが可能になる。時間領域 (region) を生成することで、一定の時間間隔のクロックによる離散化の近似手法とは異なり、稠密な値を正確に扱うことができる。さらに、時間経過に加えて時間に応じて線形に変化する値を導入し、決定可能でより詳細な振舞いの解析手法が提案されている<sup>2),4),6)</sup>。このようなアプローチを紹介するとともにコストを導入したスケジューリング解析手法について提案する。

#### 2. タスクオートマトン

タスクオートマトン<sup>2)</sup>では、時間オートマトンのロケーションに、実行時間とデッドラインを指定したタスクを割り当てることによって、タスクの生成機構を記述する。生成されたタスクはスケジューリングポリシーに従ってタスクキューに格納され、キューの先頭

から実行される。すべてのタスクがデッドラインを越えずに実行されることが保証される場合、スケジュール可能であることを表す。タスクオートマトンによって、スケジューリングの一般的な表現能力を示すことができる。タスクキューに格納する際に追い越しが可能であるかどうか (プリエンブション)、実行の終了がタスク生成に影響するか (フィードバック)、実行時間が固定か可変かによって、スケジューリングのクラスを定める。時間オートマトンの到達可能性に帰着する。結果として、タスクが固定実行時間をもつ場合、または、プリエンブションがない場合、または、フィードバックがない場合にはスケジューリングは決定可能である。プリエンブション、フィードバック、可変実行時間のすべての条件を満たす場合には、スケジューリング可能性は一般には決定不能であることが示されている。また、複数のタスクキューが存在する場合には、実行時間が可変である場合にはスケジューリングは一般に決定不能となる<sup>3)</sup>。

#### 3. タスクの実行遅延による時間あたりコストの軽減

タスクオートマトンのスケジュール可能なタスクセットの実行においてタスクの実行に対してコストの概念を導入する。タスク実行において、実行時間3でデッドライン6のタスク  $P$  と、実行時間1でデッドライン7のタスク  $Q$  があるとし、優先度を  $P < Q$  とする。図1のタスクオートマトンでは、 $P$  は周期10で生成され、実行コスト18、周期8の  $Q$  では実行コスト7が発生するとする。このことを  $P(3, 6, 18)$ ,  $Q(1, 7, 7)$  のように書くとする。デッドラインを越えない範囲での実行には時間あたりのコストを最小にすることによって時間あたりの実行資源を最適化するこ

<sup>†1</sup> 名古屋大学大学院情報科学研究科  
Graduate School of Information Science, Nagoya University

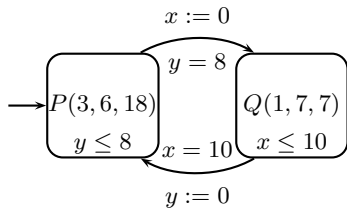


図1 コストつきタスクの生成

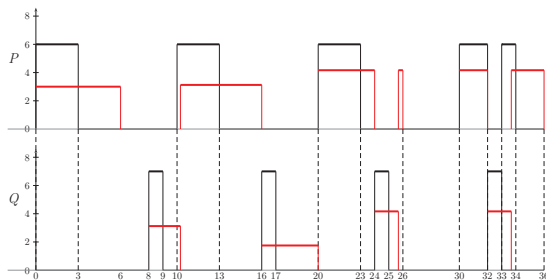


図2 時間あたりコストを意識したタスクスケジューリング

とを目的とする。

最初の  $P$  の起動の際は、デッドラインまで実行を遅らせることが可能である。 $Q$  の実行を遅らせると  $P$  の2回目の実行が保留されるので、 $Q$  の実行を遅らせると  $P$  のデッドラインが満たされなくなる。このため、 $Q$  の実行はデッドラインよりも短かくしなければならない。図2は、この様子を示す。

図2では、黒のように固定優先度で実行すると、時間あたりのコストは7必要であるが、赤のように遅延して実行してもデッドラインは守りながら、時間あたり  $4\frac{1}{6}$  で実行が可能である。

#### 4. 時間あたりコストの最適化

時間あたりのコストを最適化する手法について検討する。このために、多値時間オートマトンにおける最適化手法を導入する。ここでタスクオートマトンが対象とするタスクスケジューリングでは停止しないことが前提となる。二値時間オートマトン (DPTA) の無限パスにおける最適コストの計算可能性<sup>7)</sup> から、最適なコストわりあて計算が可能である。ここでのキーアイデアは、実行コストが時間に対して線形であることを用いると、最適なコストが必ず領域の角で発生することが示されることである。このことを利用して、最適なコストを有限的に計算することが可能となる。タスクオートマトンの技法と同様に、コストを二値時間オートマトン (DPTA)<sup>7)</sup> のエッジ値に、またリワードとして時間の積算値をマッピングすることによって、

時間あたりの最適なコストを計算する。

#### 5. おわりに

時間あたりコストの最適化は、タスクのロードに応じて動的にクロック周波数を変化させる手法のモデル化である。ここでは、タスクオートマトンによってタスクの振舞いが有限的に予測可能であるため、コストの最適化が可能であると考えられる。二値時間オートマトンの最適化計算の計算量が大きいため、実現のためには近似的手法やデータ構造についての考慮が必要である。

#### 参考文献

- 1) R. Alur and D.L. Dill, “A theory of timed automata,” Theoretical Computer Science, vol.126, no.2, pp.183–235, 1994.
- 2) E. Fersman, P. Krcal, P. Patterson, and W. Yi, “Task automata: Schedulability, decidability and undecidability,” Information and Computation, vol.205, pp.1149–1172, 2007.
- 3) P. Krcal, M. Stigge, and W. Yi, “Multi-processor schedulability analysis of preemptive real-time tasks with variable execution times,” FORMATS, pp.274–289, 2007.
- 4) K. Larsen, G. Behrmann, E. Brinksma, A. Fehnker, T. Hune, P. Pettersson, and J. Romijn, “As cheap as possible: Efficient cost-optimal reachability for priced timed automata,” proceedings of Computer Aided Verification, Lecture Notes in Computer Science, vol.2101, pp.493–505, 2001.
- 5) G. Behrmann, K.G. Larsen, and J.I. Rasmussen, “Priced timed automata: Algorithms and applications,” Formal Methods for Components and Objects, Lecture Notes in Computer Science 4111, Springer, pp.162–182 2005.
- 6) P. Bouyer, U. Fahrenberg, K.G. Larsen, and N. Markey, “Quantitative analysis of real-time systems using priced timed automata,” Commun. ACM, vol.54, no.9, pp.78–87, Sept. 2011. <http://doi.acm.org/10.1145/1995376.1995396>
- 7) P. Bouyer, E. Brinksma, and K.G. Larsen, “Optimal infinite scheduling for multi-priced timed automata,” Formal Methods in System Design, vol.32, pp.3–23, 2008.
- 8) T.A. Henzinger, P.W. Kopke, A. Puri, and P. Varaiya, “What’s decidable about hybrid automata?,” J. Comput. Syst. Sci., vol.57, no.1, pp.94–124, 1998.