自律分散型ライントレーシングロボットの安全性検証

岡野浩三† 関澤俊弦††

アブストラクト
サイバーフィジカルシステム(CPS)への関心の高まりにともない，組み込みシステムの設計，安全性検証に関する期待も高まってきている．分散型自律システムの安全性保証はその中でも大きな関心が寄せられている．本ポジションペーパーではこれまで筆者らが取り組んできたライントレーシングロボットの自律分散への拡張とそれに伴う，形式的検証の課題，とりわけ外乱のモデル化と耐故障性の検証の課題についてまとめる．

Verification on Safety Properties for Distributed Self-Adaptive Line Tracing Robots

Kozo OKANO† Toshifusa SEKIZAWA††

Abstract
Increasing expectations for Cyber-Physical Systems also rises expectations for design and safety verification on embedded systems. This position paper describes an extension of line tracing robots in which multiple self-adaptive line tracing robots interact. Also issues to resolve on formal verification of safety properties for the robots, especially models for disturbance and self-adapt and those verification.

1. はじめに

サイバーフィジカルシステム(CPS)への関心の高まりにともない，組み込みシステムの設計，安全性検証に関する期待も高まってきている．分散型自律システムの安全性保証はその中でも大きな関心が寄せられている．本ポジションペーパーではこれまで筆者らが取り組んできたライントレーシングロボットの自律分散への拡張とそれに伴う，形式的検証の課題についてまとめる．

2. これまでの取り組みの概要

モデル検証を行うことを前提にライトレーシングロボットの制御部をモデル化すると図 1が 1 つの解である．制御部は Controller Model が相当する．Environment Model ではロボットの位置の計算を行う．制御部の入出力は Color sensor, motor を介して行われる．これらの信号は誤差，外乱を含むと想定する

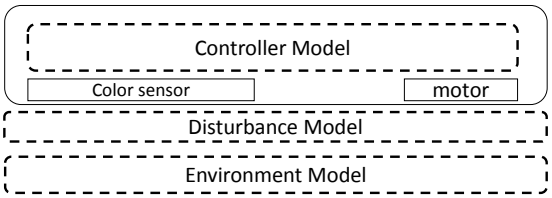

図 1 Architecture models for a line tracing robot

ことが適切である場合がある．この場合は Disturbance Model を適切に与えることによってこれをモデル化する．筆者らのこれまでの取り組み[1][2]では，モデル検査ツールとして時間オートマトンのためのモデル検査ツール UPPAAAL を用いてモデル検査を行って来た．状態変数は（時間以外は）原則として離散値しか取り得ないため Environment Model では状態変数をサンプリング，量子化などの技法を用いて近似的に扱っている．またラインの線形として円弧も扱っているが，Disturbance Model は与えていない（外乱，誤差は生じないものとして扱っている）．

最近の研究として文献[1]ではこれらのロボットの複数化の扱いについて検討している．複数化の前提としてロボットがトレースするラインの交差を許している．このため，そのような交差点では複数のロボットが衝突回避の行動をとる必要がある．また交差しないライン上で

†大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University
††日本大学工学部情報工学科
Department of Computer Science, College of Engineering, Nihon University

も，同一ライン上での速度差による衝突や対向の衝突の回避が必要である．文献[1]では同一ライン上での衝突はないことを仮定として，交差点での衝突回避の問題を扱った．衝突回避と，ロボット間の通信のためにProximity Sensor と Wireless Communication device が追加されている．また，モデルとしてCommunication Medium Model が追加され，Controller Model が Upper Controller Model と Lower Controller Model に分割された．後者は従前のController Model に対応し，後者は衝突回避や相互通信のための制御をになう分担としている．この分担により，本質的に検証したい部分に集中できる利点がある．

3. 外乱モデル

外乱の与え方は2つの方法が考えられる．一つはエラーモデルを与えて，エラーを与える方法である[4]．他方は一部のサブモデルで，処理の優先順度を意図的に変更することである[3]．後者は処理プロセスが負荷増加など何らかの理由で遅れることを意図している．処理優先順の指定（変更）は FSP や UPPAAL でサポートされている．負荷増加に対する耐故障性の検証に一定の有効性はあると思われる．ただ，モデル検査的なアプローチでは（微少な）可能性の探査に主眼があるため，気まぐれな変動をする外乱に対する検査には限界があると思われる．前者のアプローチを確率的なモデルと組み合わせる手法について今後考察していきたい．また文献[5]のような協調解析の枠組みの活用も考えたい．図 2に時間オートマトンを用いて記述した外乱モデルを表す．UPPAAL の選択機能を用いてランダムに値を与え，それを用いて各種の状態変数の値に外乱を与える

(関数 update(x,y,dist,lsensor,rsensor)).

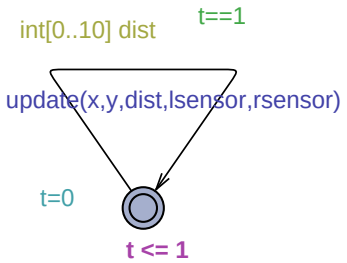

図 2 Disturbance Model Automaton

4. 軌道修正

自律システムは適切な自己安定の機構を持っていることが望ましい．よって，自己安定の形式的検証法を考案することも重要である．個々のロボットの自己安定も考慮する必要があるし，複数のロボット全体としての（準）自己安定の状態を検証をする必要もあるかもしれない．前者の課題として個々のロボットが軌道を逸したときの起動修正について考える．軌道を逸したときの修正手法はいくつか考えられる．

手法の1つは，軌道を逸したとロボットが判断した地点を中心として螺旋状に外側に旋回しながら，ラインを捉える方法である．このような軌道修正アルゴリズムに対して，軌道修正に関する諸性質をモデル検査的に調べることは課題として考えられる．

一方，ロボット集団としての振る舞い（準自己安定等）の検証をする場合は，モデル化の抽象度を上げた上で，確率モデル検査など適切な手法の適用を考慮すべきと考える．

5. まとめ

これまで筆者らが取り組んできたライントレーシングロボットの自律分散への拡張とそれに伴う，形式的検証の課題について述べた．

今後の課題は3，4で述べた手法を実際に実例に適用することによる有効性の評価である．

参考文献

[1] Kozo Okano, and Toshifusa Sekizawa: “Safety Verification of Multiple Autonomous Systems by Formal Approach,” LNCS 8696, pp.11-18, 2014.
[2] Toshifusa Sekizawa, Kozo Okano, Ayako Ogawa, and Shinji Kusumoto: “Verification of a Control Program for a Line Tracing Robot using UPPAAL Considering General Aspects,” In Proceedings of International Workshop on Informatics 2013, pp.153-162, 2013.
[3] Jeff Magee and Jeff Kramer: “Concurrency: State Models & Java Programs 2nd Edition,” Welly, 2006.
[4] Fubito Tokairin, Hideyuki Kobayashi, Kaoru Takahashi and Keishi Okamoto: “Verifying Fault-tolerance of Distributed Systems using Model Checking,” 3rd International Symposium on Technology for Sustainability (ISTS) 2013.
[5]渡辺翠，上田賀一，中島震:“制御状態の切り替えを考慮した組込みシステムモデルの協調解析” 日本ソフトウェア科学会「ソフトウェア工学の基礎」研究会第21 回ワークショップFOSE2014,ソフトウェア工学の基礎XXI, pp.159-164, 2014