

ADC Systems

~ Distributed Programs and Cooperation Protocol ~

慶應義塾大学名誉教授
徳田英幸

© H.Tokuda 2017

What is an Autonomous Distributed Cooperative System (ADC System) ?

- No supervisor which can control/manage the entire system
システム内にシステム全体を制御／統治するスペーカイザは存在しない。
- The system consists of autonomous, distributed and cooperative subsystems
各サブシステムは、自律、分散した構成要素からなる。
- The system functions are realized by cooperations among subsystems
全体のシステムの機能は、サブシステム間の協調作業によって遂行される。

© H.Tokuda 2017

自律分散協調システムの目的

Goals of ADC Systems

- 機能拡大 Functional Extension
- コスト性能比の改善 Performance per cost improvements
- 分散処理による効率・サービスの改善 Efficiency and Quality of Service
- オンラインリアルタイム処理の実現
- 局所化による通信量の低減
- 構成要素のモジュール化
- 拡張性の保証
- 集団組織の効率化
- 信頼性・耐故障性の改善 Reliability and Fault Tolerance
- 状況・環境変化への適応 Adaptability
- 生存可能性の増大

© H.Tokuda 2017

情報システムにおける自律分散協調

(ADCS in Information Systems)

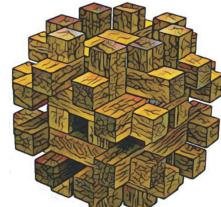
- 情報システムにおける自律分散協調モデル
 - 集中型モデル vs. 自律分散協調型モデル
 - Centralized vs. Decentralized Model, ADC Model
- 自律分散協調コンピューティングのパラダイム (Computing Paradigm)
 - 分散コンピューティング Dist. Computing
 - クライアントとサーバモデル Client/Service
 - 並行オブジェクト指向モデル Concurrent Objects
 - 分散エージェント／マルチエージェント Multi-Agent
- 自律分散協調アルゴリズム
 - 分散アルゴリズム Dist. Alg.
 - 遺伝アルゴリズム Genetic Alg.
 - ニューラルコンピューティング Neural Comp.
- ネットワークアーキテクチャ
 - アドホックネットワーク
 - センサーネットワーク



© H.Tokuda 2017

参考書

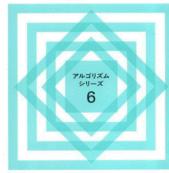
DISTRIBUTED SYSTEMS



Maarten van Steen
Andrew S. Tanenbaum

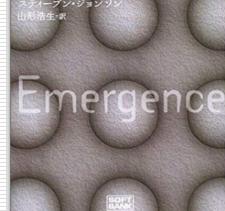
THIRD EDITION - VERSION 01

分散アルゴリズム



アルゴリズム
シリーズ 6

創発



スティーブン・ジョンソン
山形浩生・訳

著者: Steven Johnson
翻訳者: 山形浩生
出版社: SBP Books

© H.Tokuda 2017

参考文献

- 「自律分散システム研究の課題と将来」伊藤正美, 計測と制御, vol.32, no. 10, 1993.
- また、工学的システムおよび情報システムに関する参考文献としては、次のようなものがある。
 - Proceedings of the International Symposium on Autonomous Decentralized Systems, IEEE Computer Society Press, 1993.
 - Distributed Systems, 2nd Ed., S. Mullender (Ed), Addison-Wesley, 1993.
 - 分散アルゴリズム, 亀田/山下共著, 近代科学社, 1992.
 - 協調型計算システム--分散型ソフトウェアの技法と道具立て--, R.E. フィルマン/D.P. フィルマン共著, 雨宮真人/尾内理紀夫/高橋道久共訳, マグロウヒル, 1986.
 - 協調プログラミング例題集, 中島/松原/木本位田編, 共立出版, 1996.
 - エージェント指向コンピューティング--エージェントの基礎と応用--, 木下/菅原共著, ソフトリサーチセンター, 1995.
 - Artificial Intelligence: A Modern Approach, S. Russell and P. Norvig, Prentice Hall, 1995.
 - 知の創発--ロボットは知恵を獲得できるか, 伊藤宏司, NTT出版, 2000.
 - Peer-to-Peer : Harnessing the Power of Disruptive Technologies, Andy Oram (Editor), O'Reilly & Associates, 2001
- 創発
 - 創発－場・脳・都市・ソフトウェアの自己組織化ネットワーク 単行本 - 2004/3 [スティーブン・ジョンソン](#) (著), [Steven Johnson](#) (著者), [山形 浩生](#) (翻訳)
 - 人工知能の創発 知能の進化とシミュレーション伊庭 齊志 | 2017/5/26
 - 進化計算と深層学習 -創発する知能-伊庭 齊志 | 2015/10/21

© H.Tokuda 2017

DARPA Urban Challenge

～ 事例1 ～

© H.Tokuda 2017

Urban Challenge 2007/11/3

Robot
Watch

記事検索

検索

最新ニュース
[2009/04/17]

■ 第15回総合福祉
展「パリアフリー
2009」レポート
～ロボットスース
「HAL」や本田技
研工業の歩行アシ
ストも体験できる
[19:46]

■ 「第12回 ロボッ
トグランプリ」レ
ポート【大道芸コ
ンテスト編】
～自由な発想でつ
くられた、楽しい

Urban Challenge現地レポート

米国の無人口ボット車レース－優勝はカーネギー・メロン大学

～完全自律制御車はここまできた!

米国防総省高等研究計画局(DARPA)の主催による、完全自動制御の無人口ボット車レース「Urban Challenge(アーバン・チャレンジ)」の決勝が11月3日、カリフォルニア州ピクタービルの軍事基地跡で開催された。11台のロボット車が模擬市 街地の中で、定められたチェック・ポイントを順番通りにできるだけ早く回るレースで、このうち規定の6時間以内に完走できたのは4台。カーネギー・メロン大学のロボット車「Boss(ボス)」が優勝し、賞金の200万ドルを獲得した。複数のロボット車が同時に街中を走るのは世界初めてで、互いにどのように振る舞うのかが最大の見ものだったこのレース、現地からその様子をリポートする。

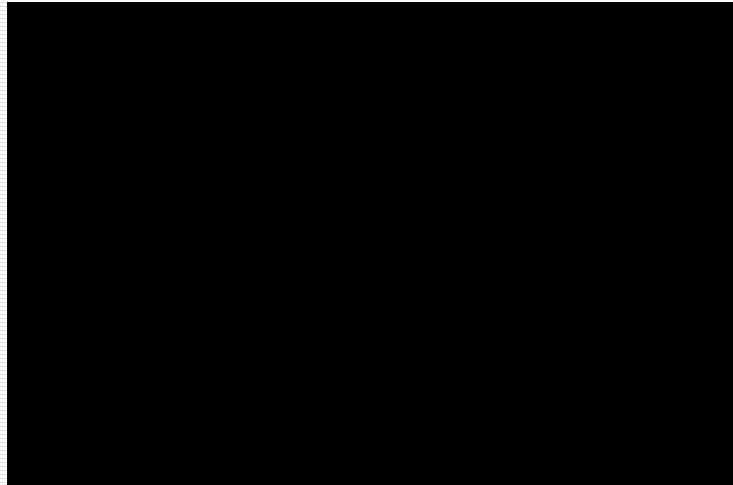


優勝したカーネギー・メロン大学のロボット車
「Boss」のゴールの瞬間

11月3日の米国西海岸時間の午前7時半。アーバン・チャレンジの開会式が始まった。米国防省傘下の研究機関であるDARPAのトニー・テザー局長は次のように語った。「今日のレースでいったい何が起きるのか、全く予想がつかない」。

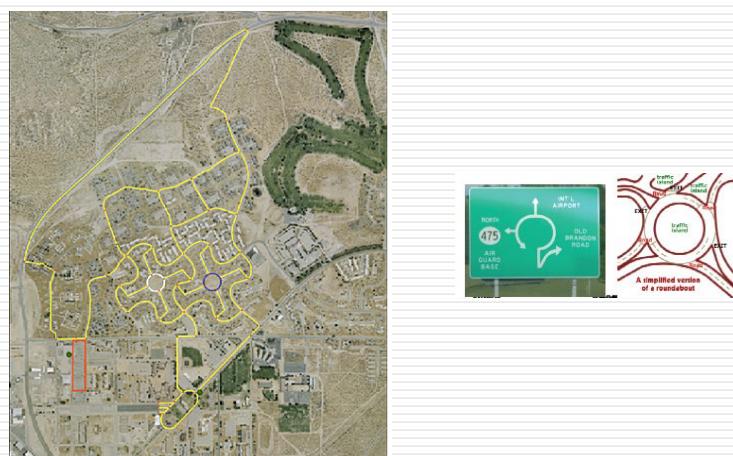
© H.Tokuda 2017

Urban Challenge 2007/11/3



© H.Tokuda 2017

DARPA Urban Challenge (1)



© H.Tokuda 2017

Autonomous Ground Vehicles (AGV)11

- [Car Crash1](#)
- [Car Crash2](#)
- [Cornell and MIT](#)

© H.Tokuda 2017

[Nissan Pro Pilot Test Drive](#) (D2D in 2017)



Autonomous Car as ADS

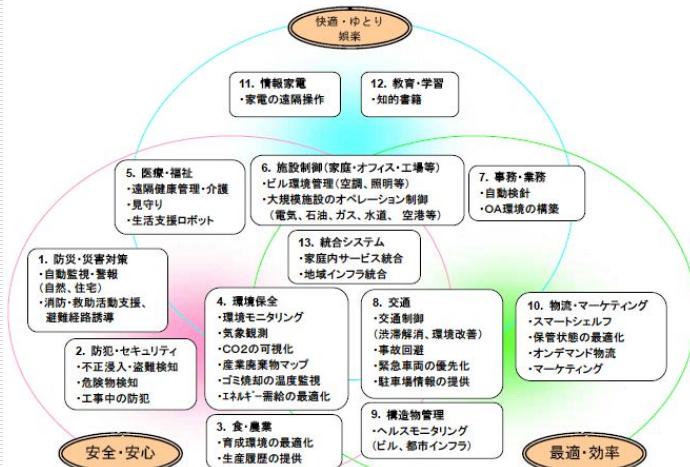
- オープン環境におけるADS
 - 環境における自律走行
 - ADSの検証と誤動作
 - 自動走行車プログラムのテスト、検証
 - 自動走行車のバグ、誤作動の検知
 - ADSの学習機能
 - 自動走行車の学習
 - 人間との協調作業 (制御の移譲 (Control Transfer))
 - Car to Human, Human to Car
 - ADSによる最適化
 - Mobility as a Service (MaaS)/ Autonomous Car as a Service (ACaaS)
-

Wireless Sensor Networks

～ 事例3 ～

Wireless Sensor Networksセンサネットワーク利用分野

(出展: ユビキタスセンサネットワーク技術に関する調査研究会)

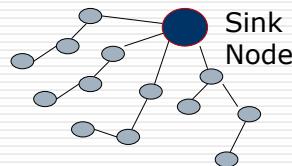


© H.Tokuda 2017

Sensor Network Models

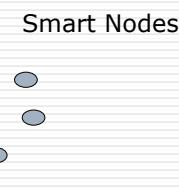
□ Sink-node Model

- Habitat sensing
- Seismic monitoring
- Structuring instrumentation
- Soil conditioning



□ P2P Model

- Smart Kindergarten
- Smart Objects, Smart Toys
- Smart Surroundings
- Lovegety
- Navigety



© H.Tokuda 2017

P2p models: Lovegety & Navigety



The Original p2p goods:
Lovegety

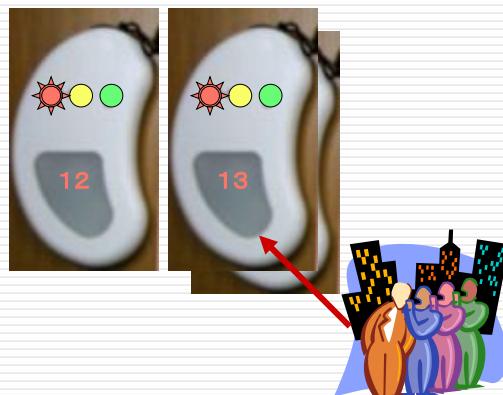
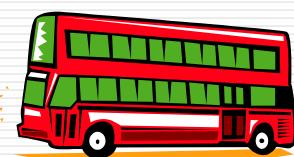


Navigety

© H.Tokuda 2017

A Group Tour Keeper

- Smart Sensor with LED lights + button



© H.Tokuda 2017

生物系システム: Ants ～ 事例4 ～

© H.Tokuda 2017

Ants Nest 蟻の群行動

□ 巨大な蟻の巣！

□ ACO (蟻コロニー最適化)

□ Giant Ant Colony

□ <https://www.youtube.com/watch?v=JyDwzXWVQHg>



© H.Tokuda 2017

Configuration of ADC System

~ A Constructive View ~

© H.Tokuda 2017

Configuration of ADC Systems

- Constructive Approach 構成論的手法
 - Top-down, bottom-up手法
 - Client-server model
 - Self-Autonomous Configuration Approach 自己組織論的手法
 - 自律的にシステムを構成
 - 群知能、集団行動
 - アリの巣
 - 都市のかたち
 - 自律型ロボットシステム

© H.Tokuda 2017

Constructive Approach 構成論的手法

- Creating a system architecture based on well-defined cooperation protocols
 - 全体に対して個々のエージェントの役割が位置づけられる。
 - コミュニケーションプロトコルの設計
 - マスタースレーブ的な協調、機能分割的な協調
 - 位置的、機能的分散処理の重視
 - 工学的な効率、システム動作解析可能性の重視

© H.Tokuda 2017

Self-autonomous Configuration

自己組織論的手法

- Creating system architecture which can emerge cooperation among subsystems 協調的行動を発現するアーキテクチャの構築
 - System functions are outcome of cooperation among autonomous agents 全体は、個々のエージェントのインテラクションの結果として創出される。
- システムダイナミクスの研究
- 柔軟性、頑健性の重視
- 自律性に基づく協調
- 生物、社会現象の模倣

© H.Tokuda 2017

Autonomy 自律性とは？

- Autonomy 自律:
 - 自分で自分の行為を規制すること
 - 外部からの制御から脱して、自身の立てた規範に従って行動すること
- Self-sustained 自立:
 - 他の援助や支配を受けず自分の力で身を立てること
- How to program autonomy? 自律性をプログラムするとは?
 - エージェントを作るとは？

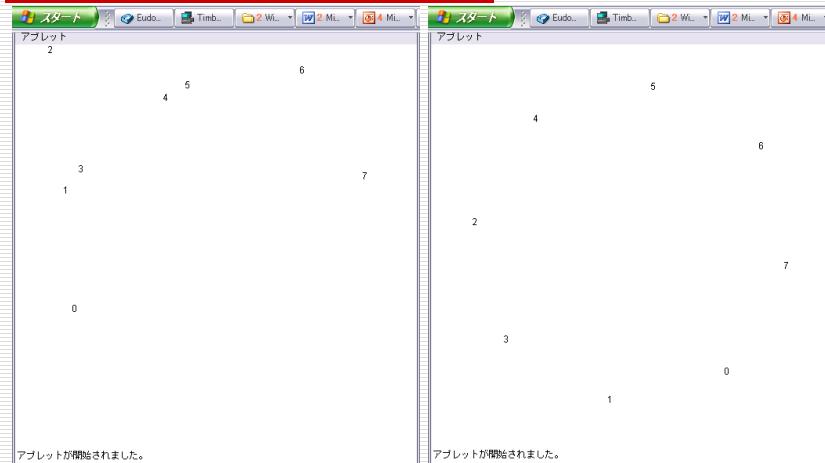
© H.Tokuda 2017

Robot Alignment ロボット整列問題

- Lineup a group of robots around a circle—群のロボットを自律的に円に整列される
 - Algorithm 1: Big Brother Scheme
 - 各位置を明示
 - Algorithm 2: Teacher Robot + Student Robots
 - 先生ロボットが計算し、支持
 - Algorithm 3: Dynamic teacher selection先生ロボットを選出
 - 先生ロボットを動的に選出
 - Algorithm 4: Cooperating Robots 同一ロボット

© H.Tokuda 2017

NR整列問題



© H.Tokuda 2017

演習-1 (4/15/2018)

- 集中型システムをリストせよ
 - 事例-Ca, Cb, Cc
- 分散型システムをリストせよ
 - 事例-Da, Db, Dc
 - Da-Dcの中で、各構成要素が自律性をもっているものに○を付与せよ
 - Da-Dcの中で、各構成要素が自律性とかつ協調性を持っている場合は、◎を付与せよ

© H.Tokuda 2017

Distributed Programs

Concurrent Processes and Concurrent Objects

慶應義塾大学環境情報学部
徳田英幸

© H.Tokuda 2017

ADC Models in Information Systems

- System Architecture システムアーキテクチャ的な見方
- Distributed Program 分散プログラム的な見方
- Basic Concepts
 - What are system components? 構成要素は何か?
 - How to cooperate? それらがどのように協調作業を進めるのか?
 - Control Flow 制御の流れ
 - Data Flow データの流れ
 - Message Flow メッセージの流れ

© H.Tokuda 2017

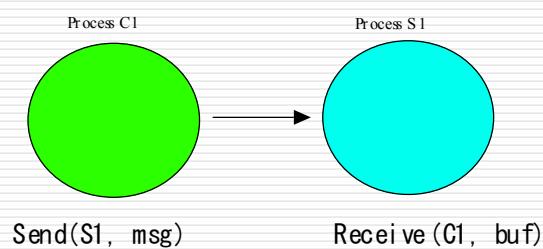
Distributed Program Model 分散プログラムのモデル

- Concurrent Process Model 並行プロセスモデル
 - Program = a set of cooperating (sequential) processes
- Concurrent Object Model 並行オブジェクトモデル
 - Program = a set of cooperating objects
- 対象としているシステムのモデル化に適している。
- いくつかの分散・並行プログラミング言語も開発されている。

© H.Tokuda 2017

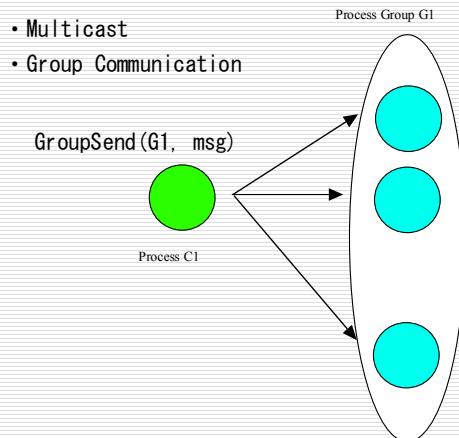
Concurrent Process: Interprocess Comm. (1)

- Uni cast



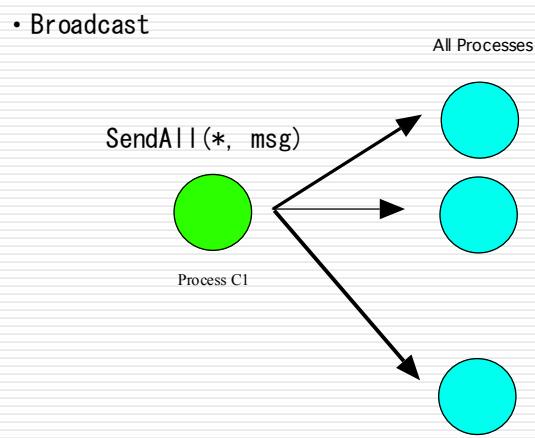
© H.Tokuda 2017

Concurrent Process: Interprocess Comm. (2)



© H.Tokuda 2017

Concurrent Process: Interprocess Comm. (3)



© H.Tokuda 2017