



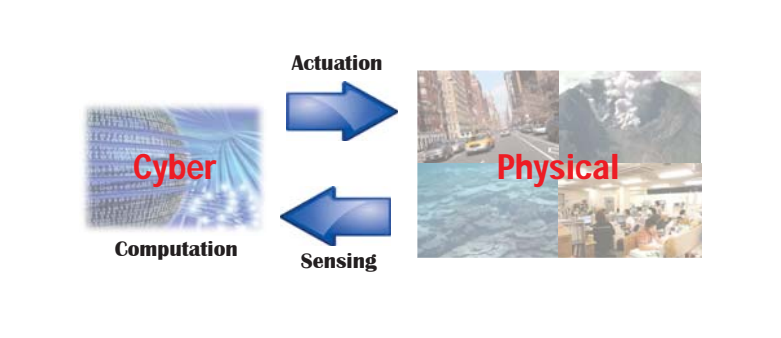
Cyber-Physical Systems

チュートリアル@SWEST13

カリフォルニア大学
加藤真平



情報技術と物理世界の融合



応用例: 自動運転技術



Cyber-Physical Systems (CPS)

Cyber-physical systems represent a new class of **embedded systems** featuring a **tight integration** of **computational** and **physical** elements.

~Wikipediaより~



応用分野



アメリカでは多額の予算投資



自動車・交通の問題



交通事故



排気ガス



交通渋滞

交通事故による死者数一年間**120万人**
(日本では約**5000人**)



自動車の**CO2**排出量一年間 **50億トン**
(地球における**CO2**排出量の**15%**に相当)



9日間続いた**100 km**の渋滞
(**2010年**中国北京)



20年後メガシティの問題

- 全世界の人口の**60%**が都市に移住(今日よりも**50%**増)
- 資産の**80%**が都市に集中



都市電力の問題



医療や環境、災害の問題



介護



医療機器



災害救助

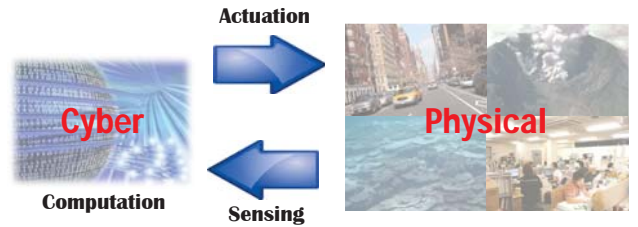


環境汚染



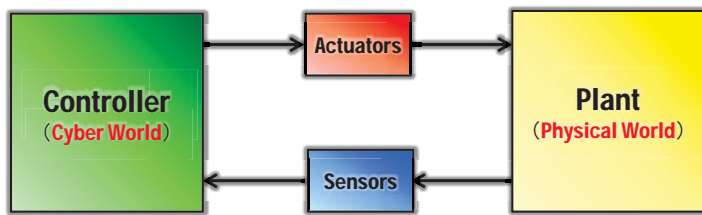
異常気象

CPSの意義



情報技術を駆使して物理世界を正常な状態に保つこと

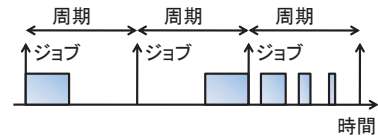
CPSは制御システム



制御対象(Plant)はアプリケーション依存

CPSはリアルタイムシステム

```
task() {
  for (;;) {
    job();
    wait_for_next_period();
  }
}
```



時間制約はアプリケーション依存

CPSは組み込みシステム



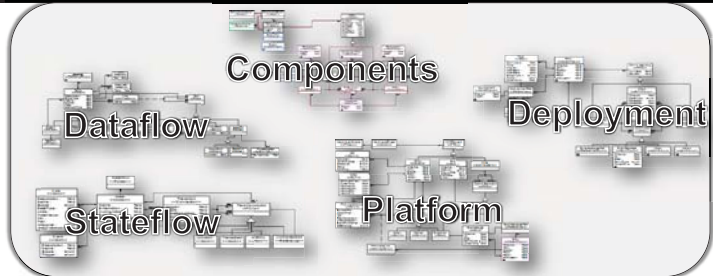
プラットフォームはアプリケーション依存

CPSはセンサーネットワークシステム

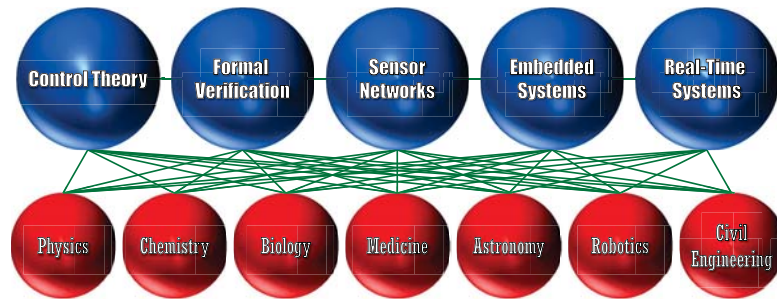


ネットワークはアプリケーション依存

CPSは要検証システム



CPSは異分野集合体



複数技術の融合が最大の挑戦



まとめ

- **Cyber-Physical Systems**の目的は
 - 情報技術と物理世界の融合
- **Cyber-Physical Systems**とは
 - 制御システム
 - リアルタイムシステム
 - 組み込みシステム
 - センサーネットワークシステム
 - 要検証システム
- **Cyber-Physical Systems**には
 - 異分野間の連携が不可欠！

おわり



CPSチュートリアルⅡ 15分で掴む制御理論屋の考え方

CPSでは他分野の研究者との理解を深めることも必要になってきています。CPSパネルの前段として、組込み（ソフト）屋視点で、制御（理論）屋と組込み屋の考え方の違い（隔たり）を解説します。

安積卓也（立命館大学）

注：所々で安積主観（偏見）の箇所があるかもしれませんが、お許しください。

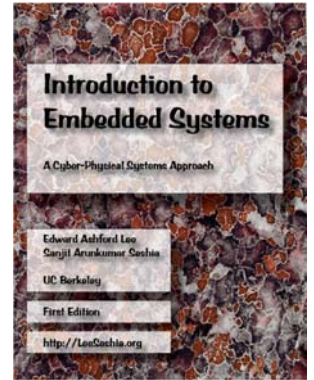
1

CPS参考書

- I Introduction **制御の話**
- 1 Modeling Dynamic Behaviors
- 2 Continuous Dynamics
- 3 Discrete Dynamics
- 4 Hybrid Systems
- 5 Composition of State Machines
- 6 Concurrent Models of Computation

- II Design of Embedded Systems
- 7 Embedded Processors
- 8 Memory Architectures
- 9 Input and Output
- 10 Multitasking
- 11 Scheduling **組込みシステムの話**

- III Analysis and Verification
- 12 Invariants and Temporal Logic
- 13 Equivalence and Refinement
- 14 Reachability Analysis and Model Checking
- 15 Quantitative Analysis



<http://leeseshia.org/>

2

組込み屋と制御屋の交流

●異文化コミュニケーション

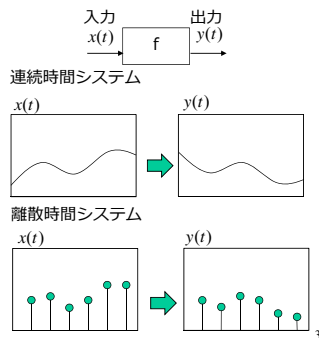
- 言葉が通じない

- モデル
- 環境

- 暗黙の知識が違う

- 考え方が違う

- 制御
- 連続世界 : アナログ
- 組込
- 離散世界 : デジタル



3

隔たりの原因

●バックグラウンドの違い

- 組込み（ソフト屋）
 - コンピュータサイエンス
- 制御屋
 - 制御理論

●興味関心の違い

- 組込みソフト屋が気にしていること
 - 仕様通りにどう動かすのか
 - 一回動いたソフトウェアはあまりさわりたくない
- 制御屋が気にしていること
 - 安定するか
 - 汎用性があるか

4

言葉の違い

●性能

- 組込み屋
 - プロセッサ等の性能
- 制御屋
 - 安定度、ロバスト性（外乱の強さ）

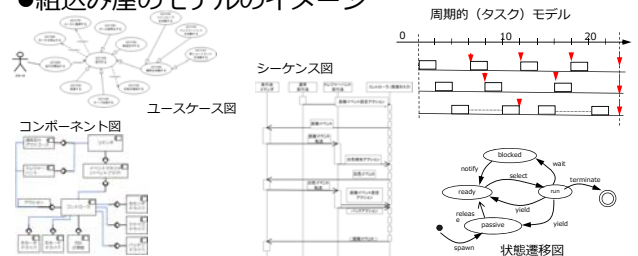
●モデル（例は次ページ）

- 組込み屋
 - 実際のものから抽象度をあげて（ある情報を捨てて）モデルを作成
- 制御屋
 - 数式モデル→汎用性

5

イメージするモデルの違い

●組込み屋のモデルのイメージ



●制御屋のモデルのイメージ

$$\begin{aligned} \dot{x} &= Ax(t) + Bu(t) & \dot{z} &= Fz(t) + Gy(t) \\ y(t) &= Cx(t) & \dot{u} &= Hz(t) \end{aligned}$$

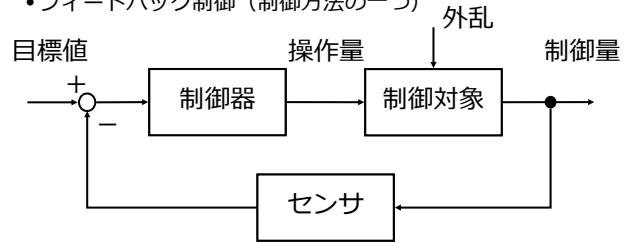
6

制御の考え方

7

制御の考え方

制御とは、簡単に言うと「目標」があり、
 「目標」を達成させようとする事
 →目標を達成するために、
 制御器が適切な操作量を制御対象に入力する
 ・フィードバック制御（制御方法の一つ）



8

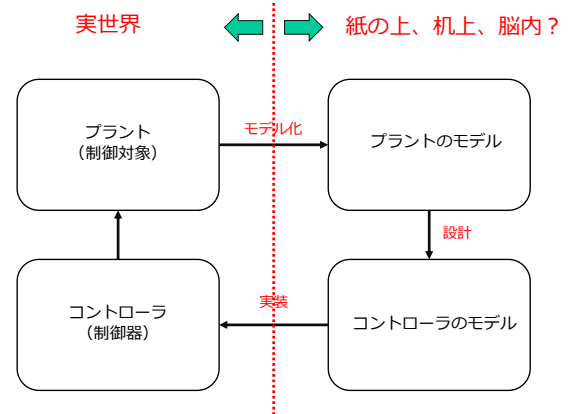
制御に対してのとらえ方の違い

- 組込み屋: コンピュータに制御対象がついたもの
 - コンピュータ: challenging
 - 制御対象: 環境

制御屋さんにとって重要な制御対象が、
 環境と呼ばれることに驚くようです。
- 制御屋: 制御対象にコンピュータがついたもの
 - 制御対象: challenging
 - コンピュータ: 思い通り動く

9

制御の考え方：基本



10

制御の考え方：モデル化

(Lagrangian L) := (kinematic energy) - (potential energy)

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \left(\frac{\partial L}{\partial \theta} \right) = \begin{bmatrix} \tau \\ 0 \end{bmatrix}, \quad \theta := \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$$

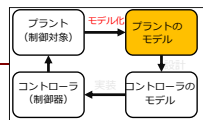
状態空間表現

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t), \quad x(t) := \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \tau(t) \end{bmatrix}$$

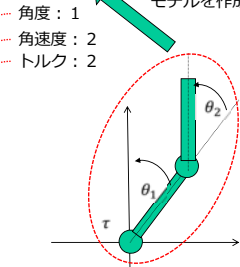
線形化

$$\dot{x}(t) = Ax(t) + Bu(t), \quad A := \left. \frac{\partial f}{\partial x} \right|_{x=0}, \quad B := \left. \frac{\partial g}{\partial x} \right|_{x=0}$$

定数行列



近似して
モデルを作成



11

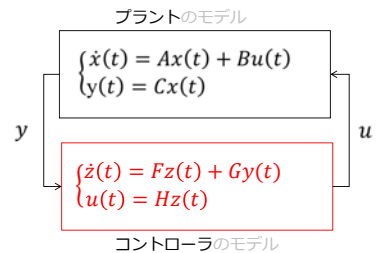
制御の考え方：設計

前ページで求めたプラント (制御対象) のモデル

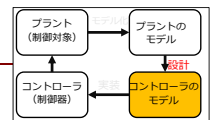
$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$$

コントローラのモデル

$$\begin{cases} \dot{z}(t) = Fz(t) + Gy(t) \\ u(t) = Hz(t) \end{cases}$$



12



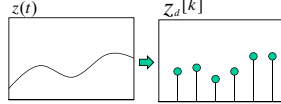
制御の考え方：実装

前ページで求めたコントローラのモデル

$$\begin{cases} \dot{z}(t) = Fz(t) + Gy(t) \\ u(t) = Hz(t) \end{cases}$$

離散化する (デジタルで扱えるようにする)

$$\begin{cases} z_d[k+1] = F_d z_d[k] + G_d y[kh] \\ u_d[k] = H_d z_d[k] \end{cases}$$



制御屋さんは、実装にはあまり関心がないようです。

13

組込み屋と制御屋が
それぞれに対して知りたいこと

- 組込みソフト屋が知りたいこと
 - 数式 → 離散化 → アルゴリズム
→ どうやってプログラムにしているか？
 - システム制御理論がどの辺りで役に立っているのか？
- 制御屋が知りたいこと
 - 実装の方法
 - 数式のパラメータがいつ決まるのか？

14