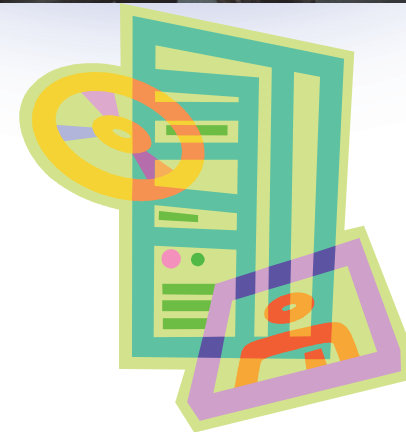


# 組込みデザイン研究室



2019/11/1

2020研究室紹介



# 組込みデザイン研究室(情報棟8F)

近年では、様々な機器の中にコンピュータが組み込まれ、連携するようになってきています。本研究室では、そのような機器を動作・制御するソフトウェアをいかに作るかについて研究しています。

## スタッフ

教授	中田明夫
准教授	島和之
	双紙正和
	村田佳洋
助教	佐藤康臣

組込みシステム  
組合せ最適化

ソフトウェア設計技術  
クラウドコンピューティング

情報セキュリティ  
暗号

# 研究室の特徴

## ■ 教員5名体制

### □ 幅広い分野をカバー

- 研究自体は各教員が個人で実施
- 輪講やゼミなどは共同実施

□ 効率よく勉強できる、分野外のいろいろな意見が聞ける、など

### □ 質問しやすい(おそらく)

## ■ 計算機環境が充実

### □ 一人1台、専有PCを支給

- 基本的には研究対象はソフトウェア

### □ 組込みソフトウェアの研究では、マイコンボードやロボットなどの機器も利用

# 研究以外の活動にも参加

- ・ 広島県警主催のサイバー防犯ボランティアに参加
  - ・ インターネット上で犯罪に巻き込まれないための啓蒙活動を実施
    - ・ コンピュータ犯罪被害のデモプログラム作成など
- ・ ESSロボットチャレンジに参加
  - ・ 自動掃除ロボット(iRobot-Create)を用いた組込みソフトウェア開発コンテスト
    - ・ 情報処理学会 組込みシステム研究会主催
  - ・ 2017～2018年度は参加希望者数が少なかったため不参加



プログラムの開発中



# 学生の就職先(修士)

- NEC(3名)
- 中国電力
- パナソニックアドバンステクノロジー(2名)
- NTTデータ中国
- 野村総研
- NTTソフト
- JR東海情報システム
- RICOH
- SRA
- 日本IBM
- 熊平製作所
- など

進学希望者は  
その先も考えておこう

# こんな学生が欲しい！

- 研究に意欲を持って打ち込める学生
  - 文献調査・ゼミ・実験など
- 積極的に研究室に出てくる学生
  - 日常的に研究室を盛り上げて欲しい
- プログラミングが好きな学生

配属の選考では、オープンラボなどにおいて、研究室を訪問、または、教員にコンタクトしてくれた学生を優先

# 組込みデザイン研究室

## 連絡事項

- 本日の資料は以下のWebページに掲載予定
  - <http://www.sos.info.hiroshima-cu.ac.jp/~nakata/esdlab2020-jibun.pdf>

# 各教員の研究紹介



## 教授 中田明夫(なかたあきお)

- 専門: 組込みシステム, リアルタイム並行分散システム, 形式的手法
- 組込みシステムの設計情報からの性能検証
  - 組込みシステム: 機器に組み込まれたコンピュータシステム
  - 製品出荷後に不具合 → 事故、リコールなどの損害
  - 特に性能が出ないことによる不具合は発見が困難
    - → 設計段階において性能を満たすか否かを検証
- リアルタイムソフトウェアの再利用
  - 以前作ったソフトウェア部品の再利用 → 開発期間の短縮に有効
  - 同じソフトウェアでも、動作するハードウェアが変わると実行時間が変化
    - 同じ速度で動いてくれない → 制御ソフトウェアやゲームなどのリアルタイムシステムでは問題
  - → ハードウェアに合わせて実行時間の調整を行えるようにする手法を開発



# 組み込みシステムとは？

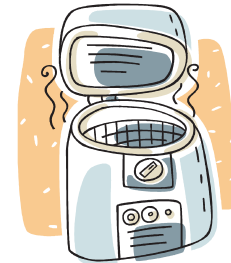
- ▶ 各種の機械や機器に組み込まれて、その制御を行うコンピュータシステム
- ▶ 組み込みシステムのソフトウェア＝組み込みソフトウェア



無線キーロック  
エアコン





メール  
カメラ



炊飯予約  
早炊き

# 組込みシステムの特徴

- 機械の制御を行う場合、**リアルタイム制御**が重要
  - 決められた時刻(**デッドライン**)までに制御が完了することを目的とする制御
  - 例)車のエアバック制御システム
- 大量生産される場合、コストが重要
  - 限られたリソース(必要最低限のCPU, 安価なメモリなど)で   **性能要求**を満たす必要がある

# 性能要求

## ■ スループット要求

□ スループット・・・単位時間当たりの処理能力

✓ 例) 1/30秒以内に内部処理を終わらせ、画面に出力する

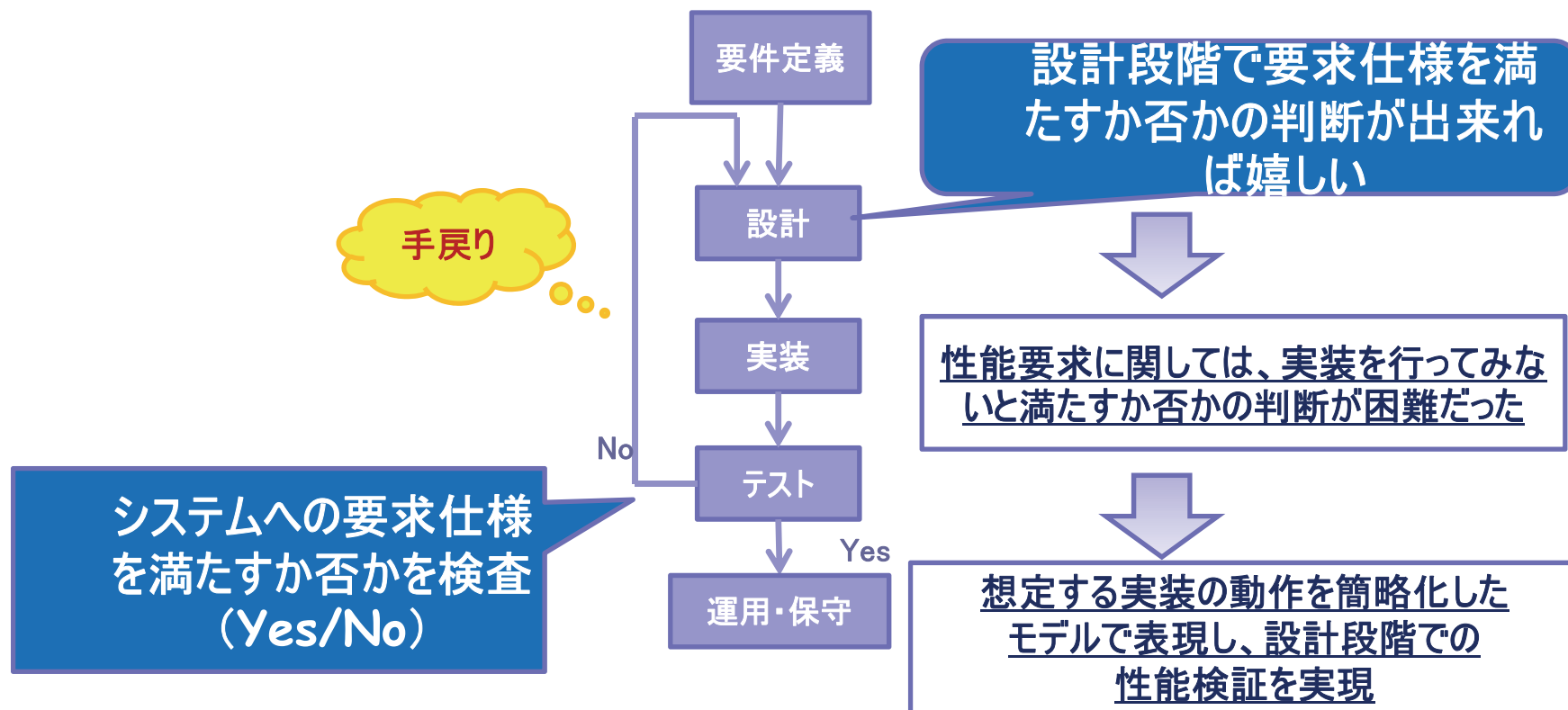


スループット要求を満たす  
→1/30秒ごとに画面が更新される

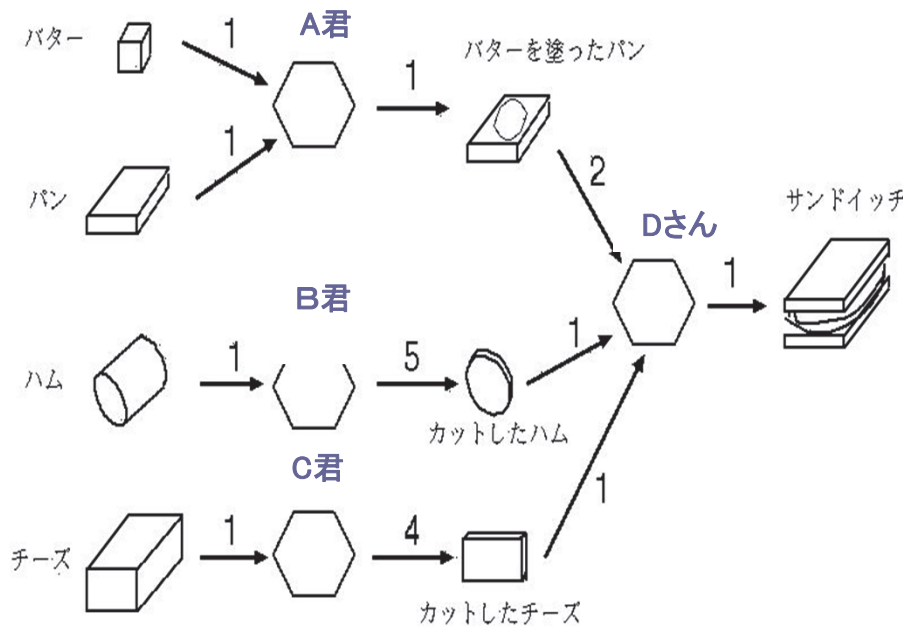


処理が1/30秒以内に終わらず、  
スループット要求を満たさない  
→処理落ち

# ソフトウェアの開発プロセス



# ソフトウェア仕様からの性能検証 (例え話)



リソース: 人(A君、B君、C君、Dさん)、包丁

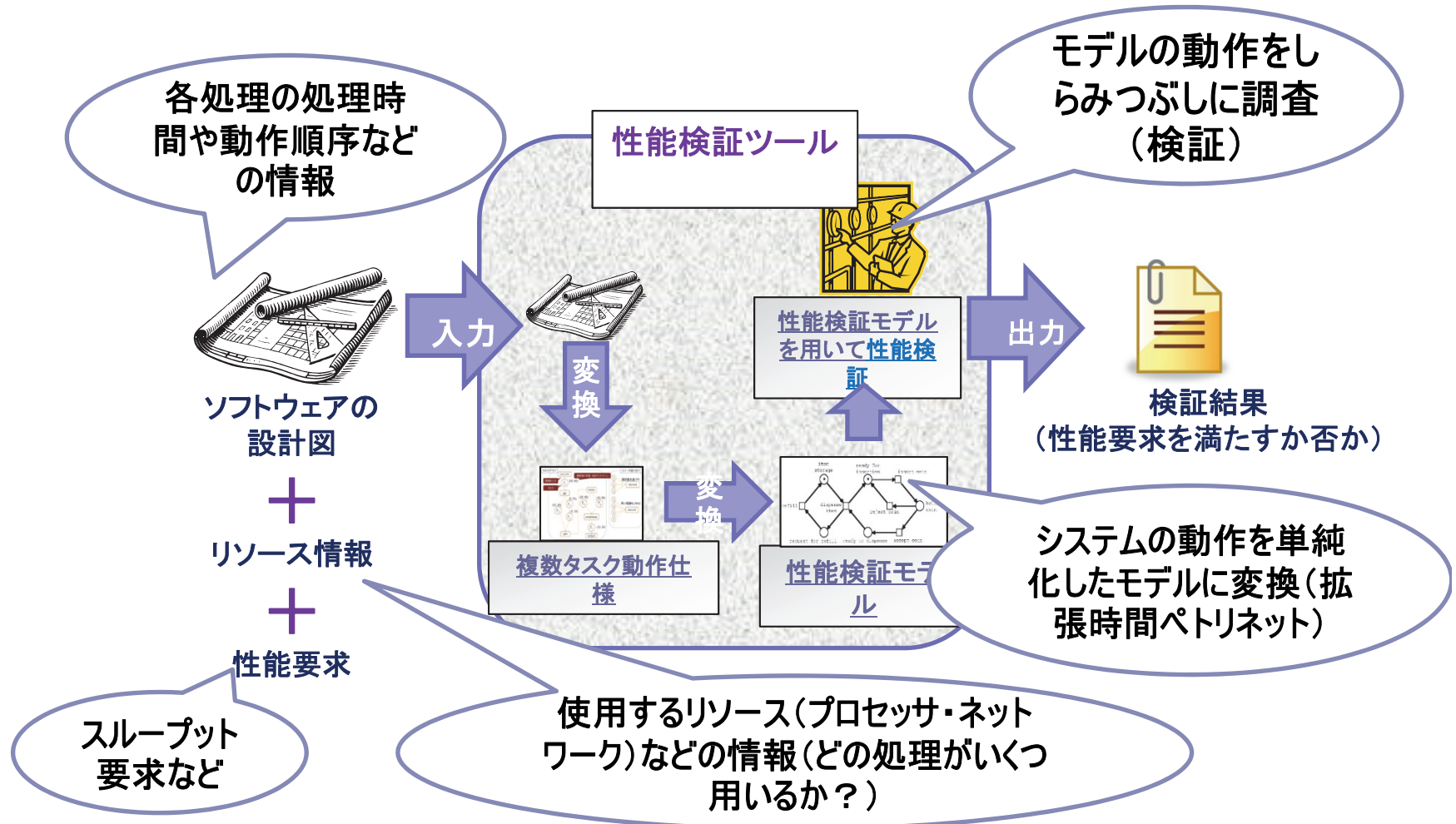
## サンドイッチ製造システム

- ・A君は6分間に5枚のパンにバターを塗る
- ・B君は大きなハム1つを7分間で10枚にカットする
- ・C君は大きなチーズ1つを6分間で8枚にカットする
- ・Dさんは4分間に1枚のハムと1枚のチーズを2枚のパンの間に挟む作業を10回行う

### 性能検証

- ・性能要求  
1分間に2個以上サンドイッチを作れるか?
- ・リソース制約(その1)  
包丁が1本しかないとき、B君とC君は仲良く1本の包丁を使ってサンドイッチが作れるか? 包丁は早い者勝ちとするか? B君を優先させるか?
- ・リソース制約(その2)  
もしもA君とB君とC君の3人しかいないとき、Dさん抜きでサンドイッチを1分間に2個以上作れるか? どんなシフトを組めばDさんの穴埋めができるか?

# ソフトウェア仕様からの性能検証の流れ



# 本研究の課題・成果

- コンピュータシステムでは、リソースはオペレーティングシステム(OS)が管理
- → 実際のOSのリソース管理手法（特にリソースのスケジューリング手法）をモデル化する必要
  - あまり複雑すぎると、しらみつぶしの調査（検証）が困難に → うまく簡単化する必要
- マルチプロセッサ環境に適したLLF(Least Laxity First)スケジューリング手法をモデル化
  - → 電子情報通信学会ソフトウェアサイエンス研究会にて研究奨励賞を受賞(H28年7月)
- 他のスケジューリング手法への対応や検証の効率化などが課題



# その他の研究テーマ

- リアルタイム性を持つソフトウェアの再利用
  - ソフトウェアを別の実行環境で再利用する場合に、ループ回数やCPU動作周波数などのパラメータを変えることで実行時間を調整
  - プログラムの解析により実行時間をパラメータに関する式で導出 → 数理最適化により最適なパラメータを求める
- マルチタスク組込みソフトウェアのメモリ消費量削減
  - 同時には使用しないメモリを共有化し、メモリ消費量を節約
    - 全体のメモリ消費が増えないようにタスクをスケジューリング
  - マルチプロセッサ環境のリアルタイムシステムへ対応
    - 研究成果を2018年8月に国際会議TASE2018で発表
- 入力の混雑制御によるシステムの応答性向上
  - ネットワークルータの流量制御技術を応用 → ユーザインタフェースのスムーズな操作性の実現を目指す

准教授 島和之(しま かずゆき)

専門: 分散システム、ソフトウェア工学

分散システム: 複数のコンピュータが共通目的のため、通信しながら協調して処理するシステム

例: 電話、メール、Web, SNS, ATM、オンラインゲーム、ネットショップ

■ 分散システム用プログラミング言語

災害用分散システムの設計

- 平常時とは異なる状況で機能
- 災害時を想定した設計が必要

- 無人航空機の制御可能範囲の拡大
- 土砂災害の前兆現象検知システム
- 衛星通信を用いた安否確認システム

# 分散システム用プログラミング言語

- コールバック
  - 分散システムの非同期処理において必要
- コールバック地獄
  - コールバックの連鎖によってプログラムが複雑化
- コールバックを簡潔に表現できるプログラミング言語の提案

```
get('a.txt', function(err, a){
  console.log(a);
});
```

コールバックの例  
JavaScript

# コールバック

- 他の関数の引数として渡され、必要なタイミングで呼び出される関数

コールバックの  
連鎖の例

```
get('a.txt', function(err, a){
  get('b.txt', function(err, b){
    get('c.txt', function(err, c){
      console.log(a + b + c);
    });
  });
});
```

- コールバック地獄

□ コールバックの連鎖 ⇒ プログラムの複雑化

- コールバックが別のコールバックを呼び出し、それがさらに別のコールバックを呼び出す

# 提案プログラミング言語

- 継続渡しスタイルのラムダ計算に基づく

- 例

```
get 'a.txt' ^ (a)
print(a)
```

- 関数 `get` に引数として `'a.txt'` と `^(a) print(a)` を渡す
- 関数 `get` は `'a.txt'` の内容をサーバに要求し、応答が返ってきた時点で `^(a) print(a)` を呼び出す

# 従来言語と提案言語との比較

## 従来言語

```
get('a.txt', function(err, a){
  get('b.txt', function(err, b){
    get('c.txt', function(err, c){
      console.log(a + b + c);
    });
  });
});
```

## 提案言語

```
get "a.txt" ^(a)
get "b.txt" ^(b)
get "c.txt" ^(c)
print(a b c)
```

- コールバックの連鎖が増えたとき
  - 従来言語はネストが増える。
  - 提案言語はネストが増えない。

# UAVによる橋梁点検

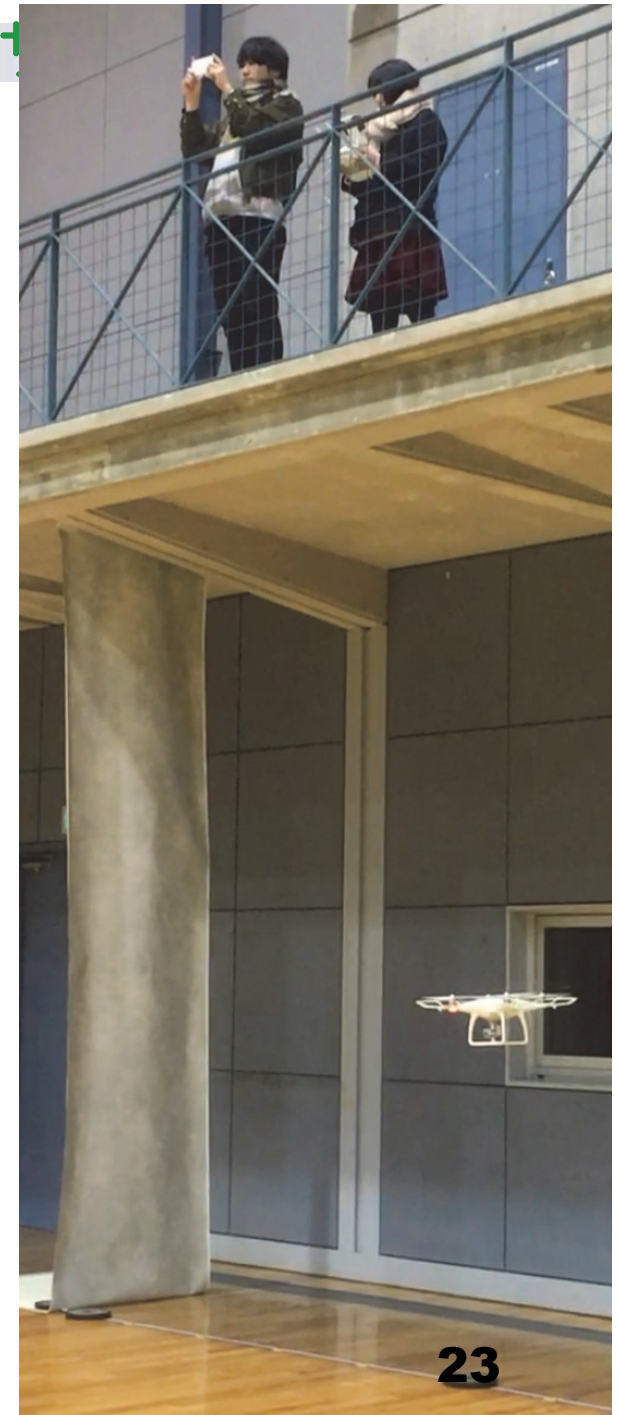
## ■ 問題点：操縦が非常に困難

- 橋桁でGPSが
- 橋桁や橋脚な
- 障害物でUA

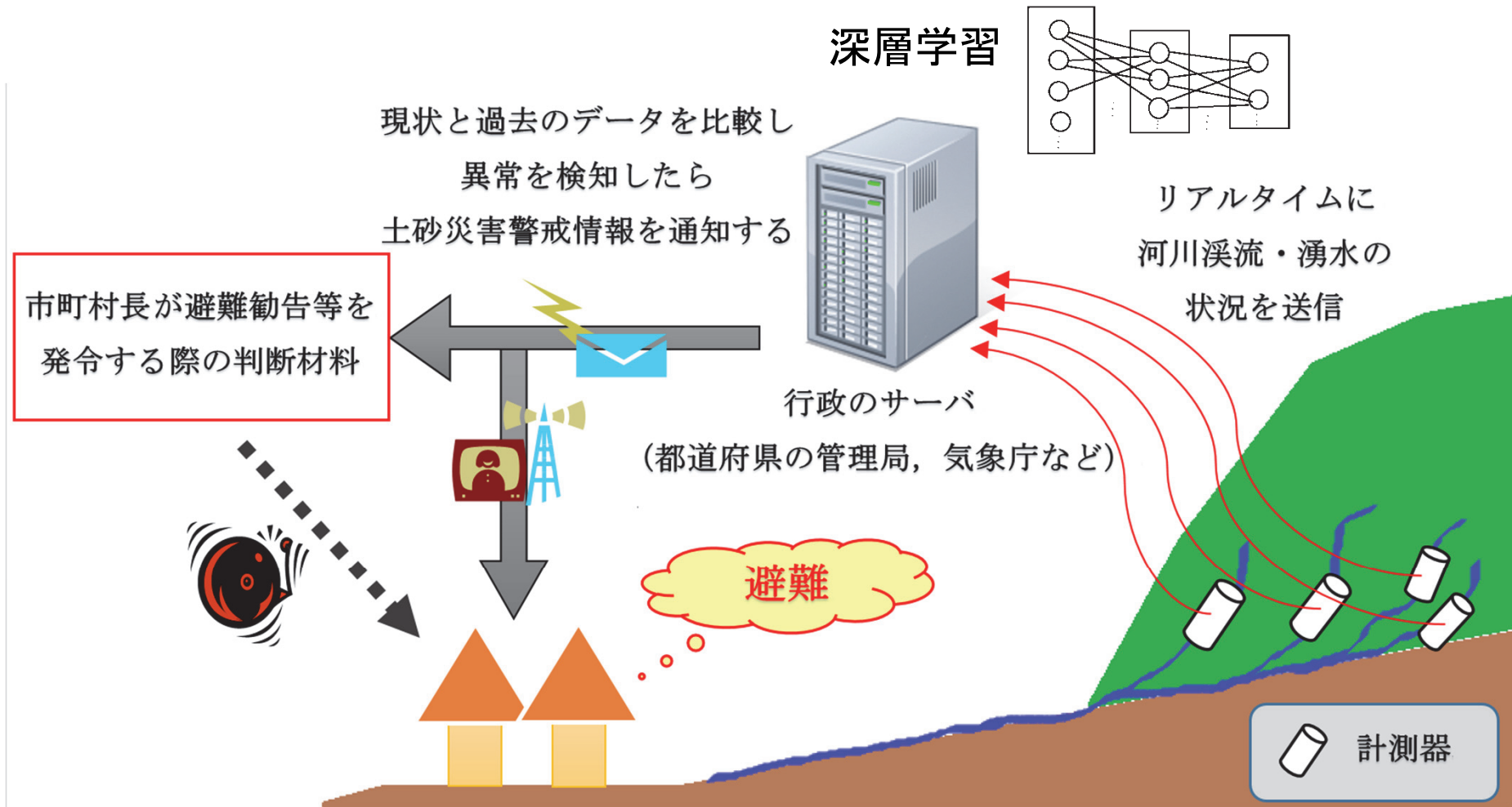


## ■ アプローチ

- 全天球カメラをUAVに搭載し、無線で映像を端末に表示する。
- 全天球映像でUAVの位置や障害物を把握しながら操縦する。

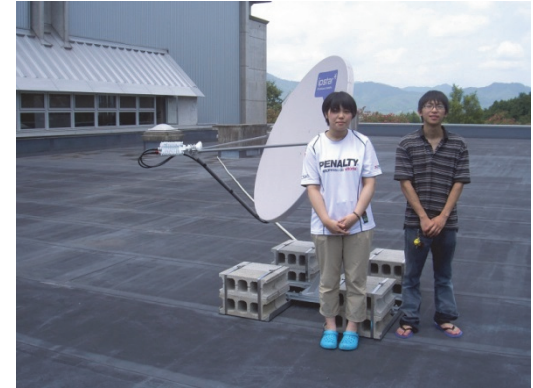
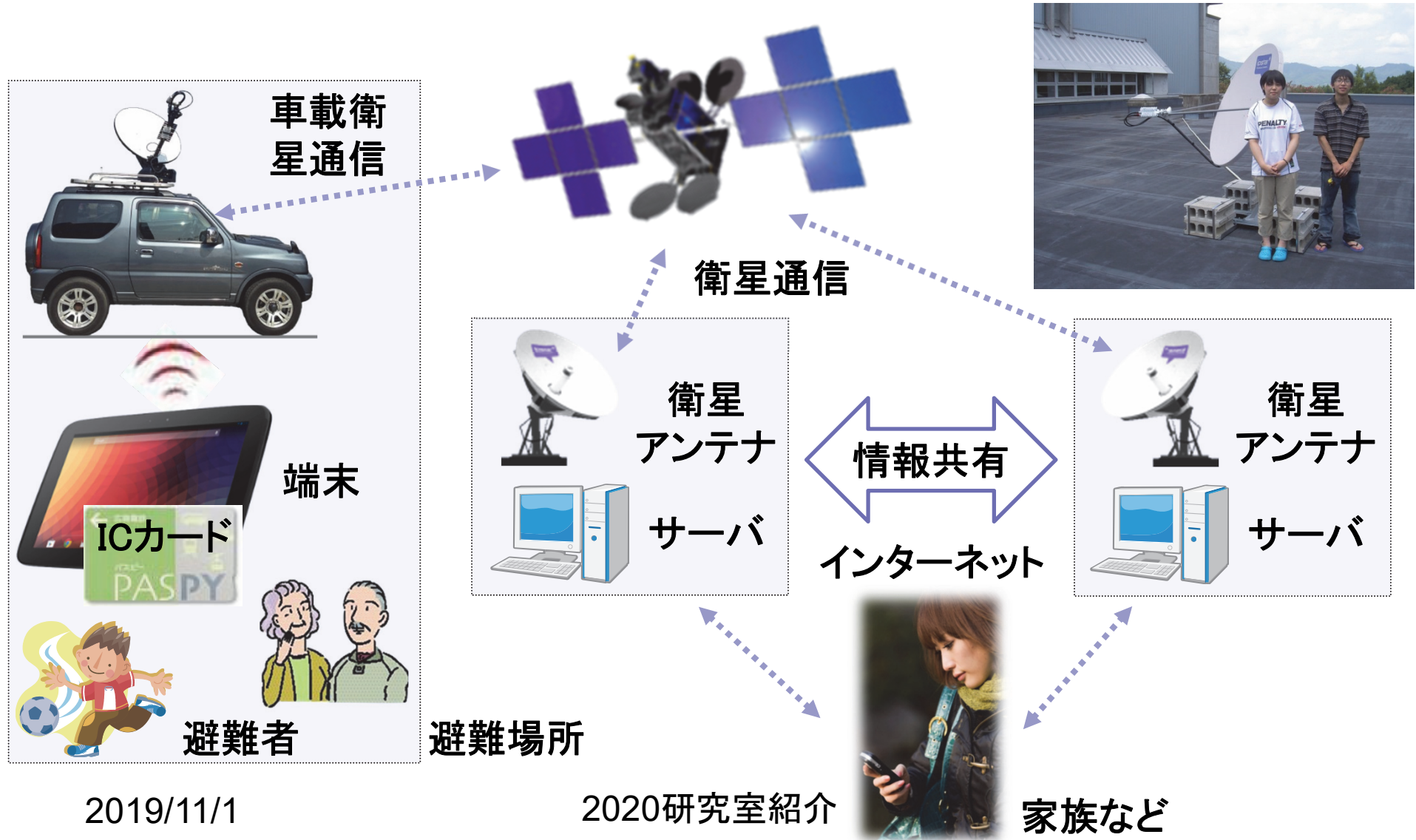


# 土砂災害の前兆現象検知





# 衛星通信を用いた安否確認



准教授 双紙正和(そうし まさかず)

■ 専門: 情報セキュリティ

研究分野	研究内容
IoT (Internet of Things) セキュリティ	鍵共有プロトコル, ホワイトボックス暗号
軽量認証法	リソースが限られたデバイス向けの, 効率のよい認証法
車々間通信セキュリティ	車同士が通信する際のセキュリティ確保

准教授 村田佳洋(むらた よしひろ)

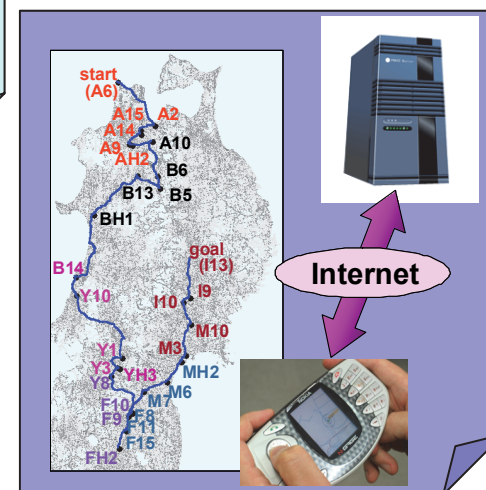
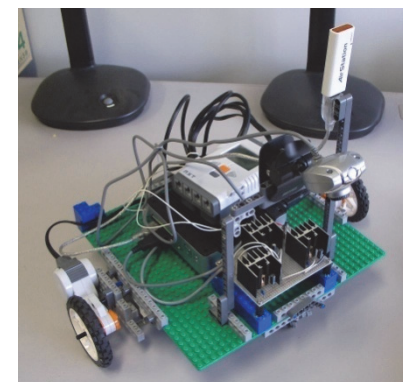
専門: 進化アルゴリズム・組み合わせ最適化

観光用パーソナルナビゲーションシステム

- ・遺伝アルゴリズムを用いた観光経路の組み合わせ最適化
  - ・グループ観光補助システム
  - ・**ポケモンGoナビ・経路探索システム**

可動ロボットを用いたワイヤレスセンサネットワークシステム

- ・省電力のための移動先最適化



## ポケモンGO用のナビ・経路探索アルゴリズム(過去事例)

### ■ 問題の定義, 定式化, 近似アルゴリズムの提案

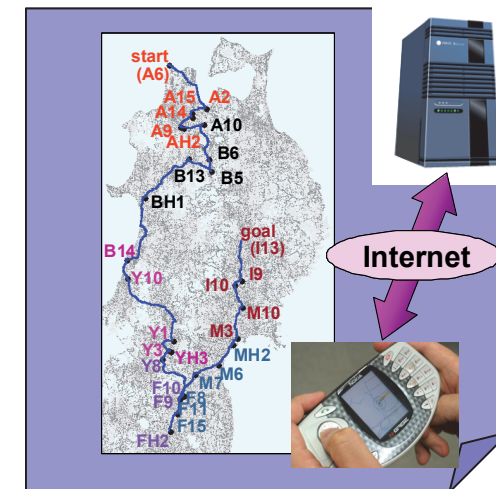
- 「スケジュールの採点方法」の机上での決めつけ
- 今回は「出現周期ごとの捕獲数最大化」
- 新規性・創造性・有用性が必要

### ■ 文献調査, 既存技法の習得

- スケジューリング手法研究の調査
- ポケモン出現周期の实地確認

### ■ プログラム実装とシミュレーション実験

- 進化アルゴリズムの応用が基本
- 問題に特化しての高性能化を目指せ



助教 佐藤康臣 (さとう やすおみ)

■ 専門: インターネットコンピューティング

- インターネット上の評判情報を利用した商品推薦
  - レビュー文の極性に基づくアイテム評価値予測
  - 印象を考慮した映画推薦