



# Society5.0 / Smart Community MECシステムソフトウェアグループ

IPSJ全国大会@Zoom 2020.3.6

## MEC (Multi-access Edge Computing)への挑戦

菅谷みどり<sup>1</sup> 大川猛<sup>2</sup>  
広渕崇宏<sup>3</sup> 高野了成<sup>3</sup> 石綿陽一<sup>4</sup>

1 芝浦工業大学 工学部情報工学科

2 東海大学 情報通信学部 組込みソフトウェア工学科

3 国立研究開発法人 産業技術総合研究所

4 VA Linux Japan

# Society 5.0 にて目指す方向

- 新たな価値で経済発展と社会的課題の解決を両立<sup>[1]</sup>

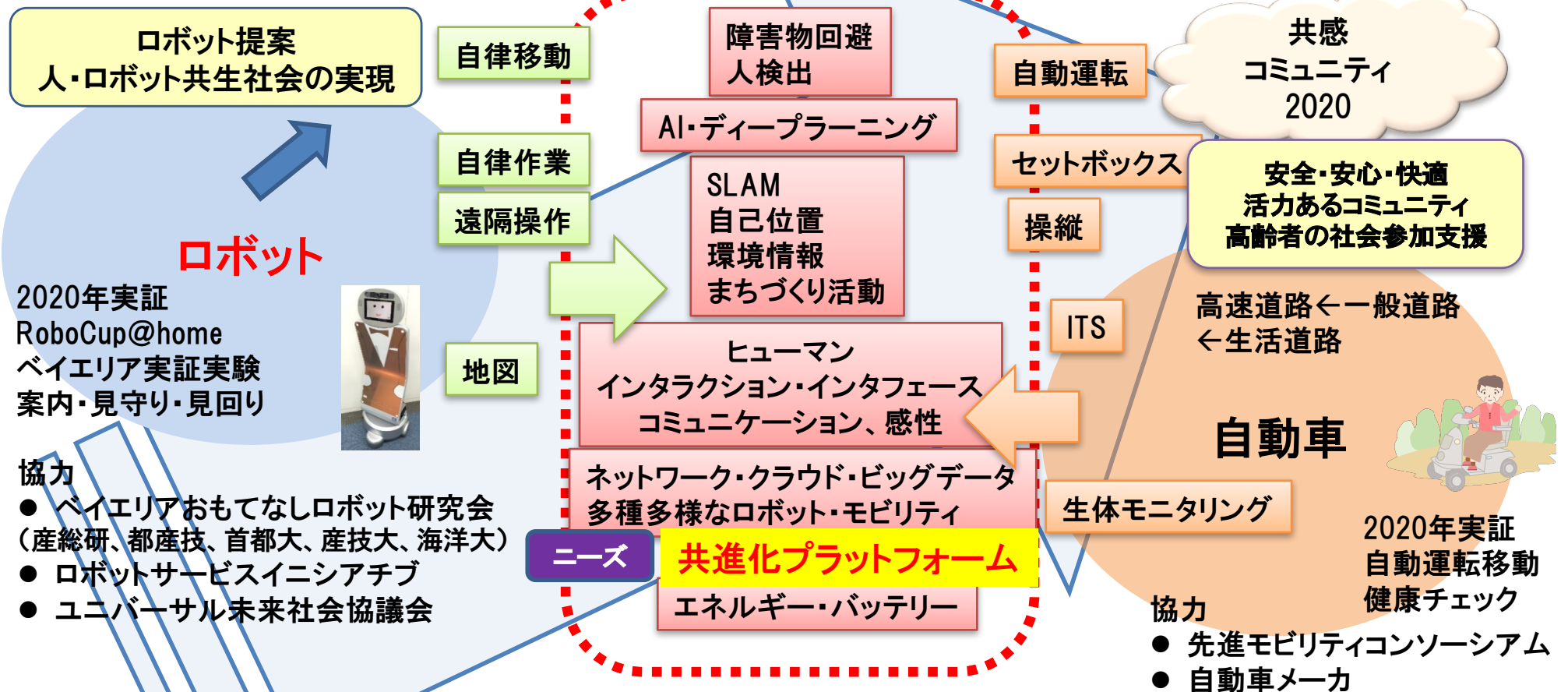


[内閣府作成]

[1] 内閣府 科学技術政策 Society 5.0 [https://www8.cao.go.jp/cstp/society5\\_0/index.html](https://www8.cao.go.jp/cstp/society5_0/index.html) より

# ロボットを日常的に利用する社会に向けて

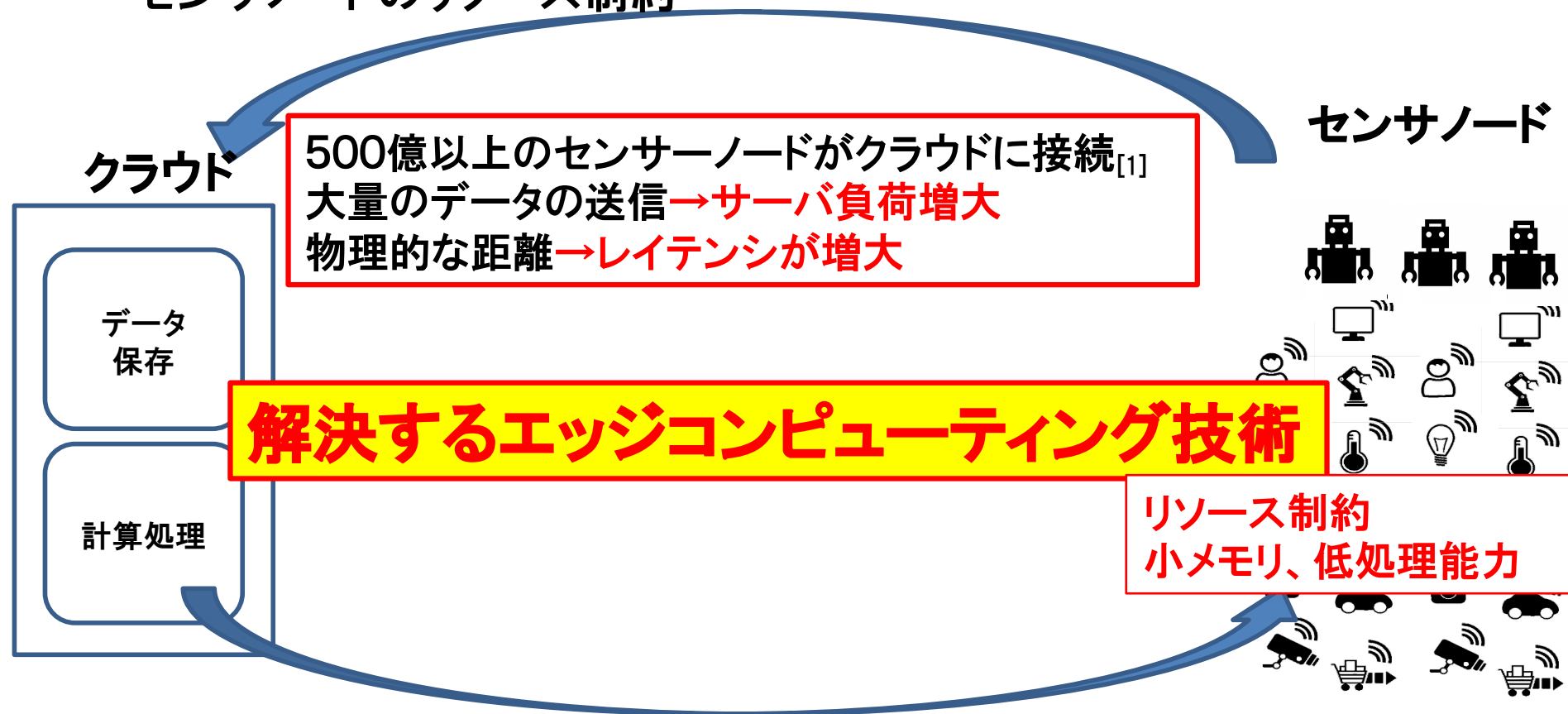
- ロボット・自動車共進化に向けて(芝浦工業大学ロボット・コンソーシアム)[2]



- 人・物・移動に関わる高度な物理シミュレーションが必要だが**計算機パワー不足**
  - 多様なハード・ソフト・ミドルウェアの**不統一**
- ロボット・自動車は**無線**が必要だが**干渉が多く、接続が不安定なことも多い**

# IoTの発展と課題

- IoT は大量のリアルタイム要求が異なるセンサノードとクラウドで構成される
  - クラウドサーバの過負荷
  - センサノードのリソース制約

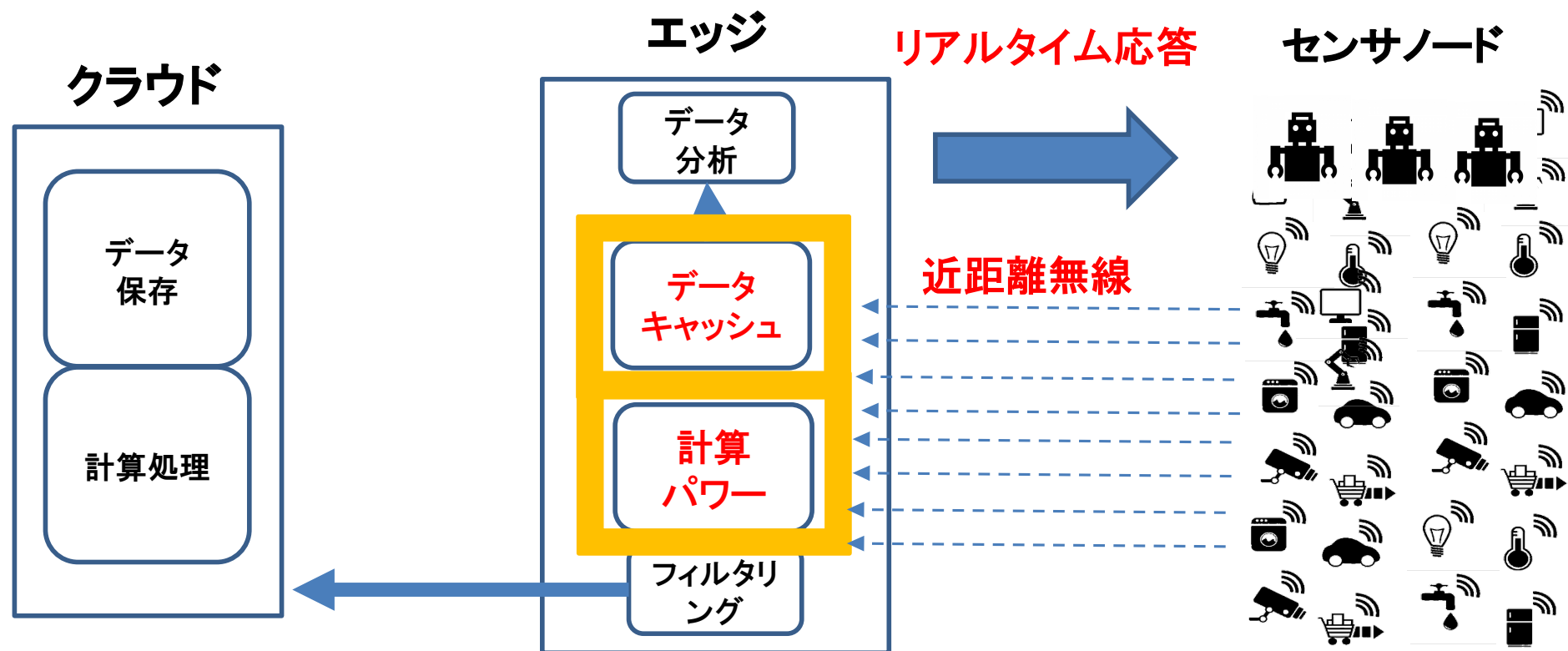


[3] Kashif Bilal<sup>a</sup>, Osman Khalid<sup>b</sup>, Aiman Erbada, Samee U. Khan .Potentials, trends, and prospects in edge technologies: Fog, cloudlet, mobile edge, and micro data centers . Computer NetworksVolume 130, 15 January 2018, pp 94-120

# エッジコンピューティング (MEC) によるIoT課題の解決

- センサノードとクラウドの間に位置し，課題に対応

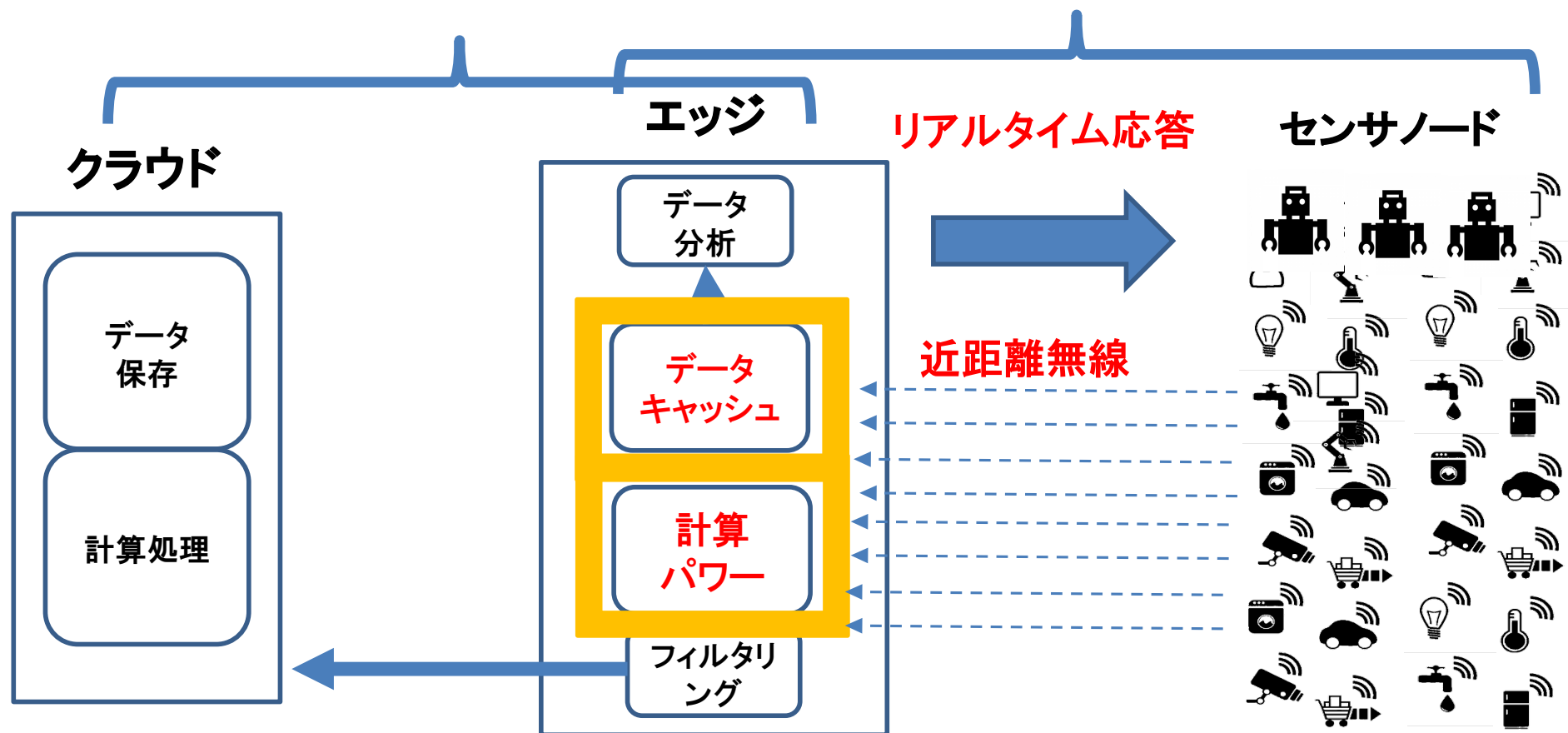
高速処理かつ，データストレージ(データキャッシュ)の役割を担うことで，センサノードとクラウドの課題を解決可能



# エッジコンピューティング (MEC) によるIoT課題の解決

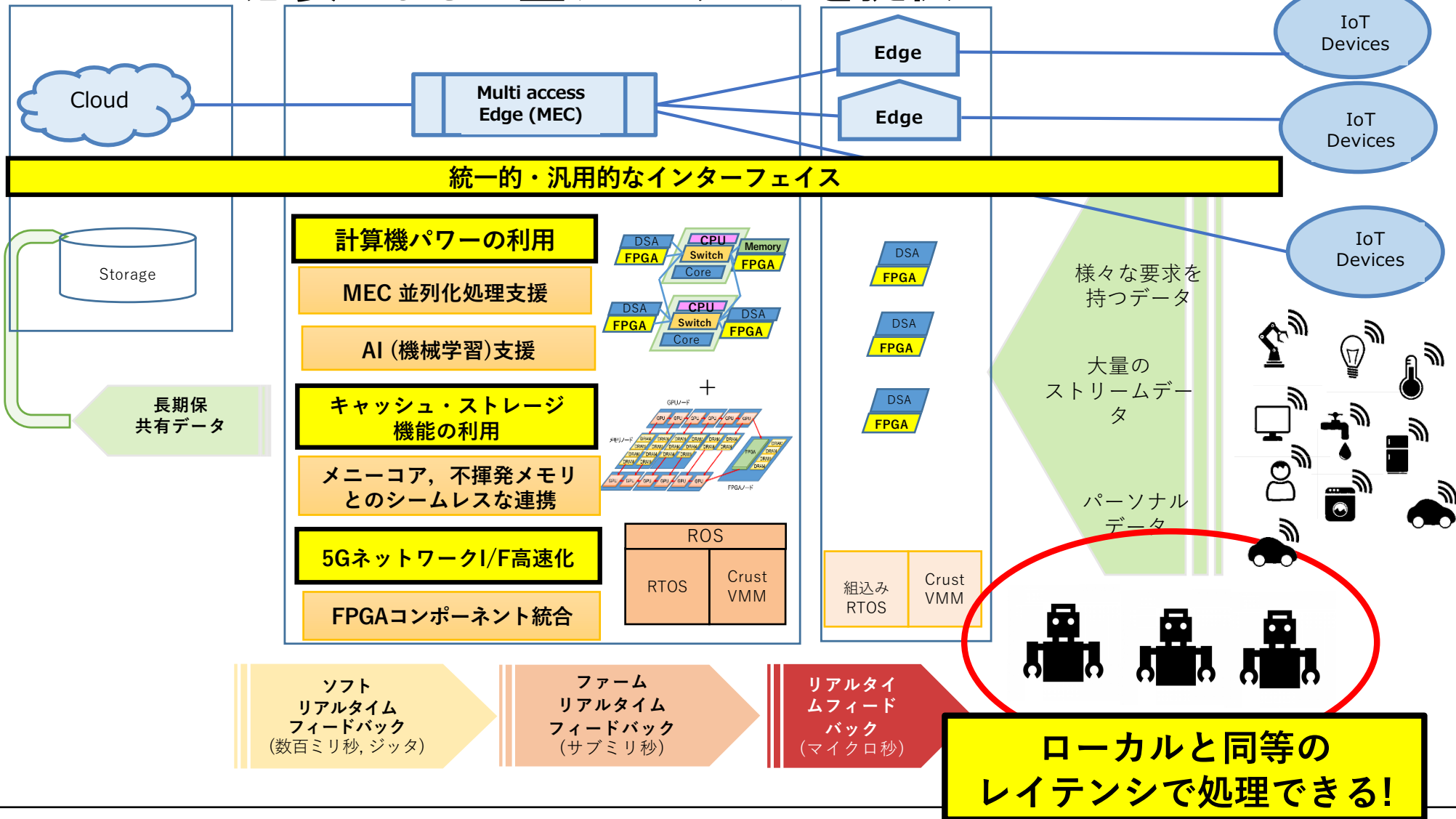
- センサノードとクラウドの間に位置し, 課題に対応

ハードウェア・ソフトウェア分割 → 汎用I/F, 統一的な記述I/F



# システムソフトウェアG

- MEC で必要となる基盤ソフトウェアを提供



# MEC による計算パワーの利用

## 複数台ロボットの消費電力の削減手法の研究 [4,5]

個々のロボの予測式 x ロボット数 → **最適化計算** → 動作再割り当て [4,5]

$$MC_x = CPD_x^V \times D_y + CPS_x^V \times S_y$$

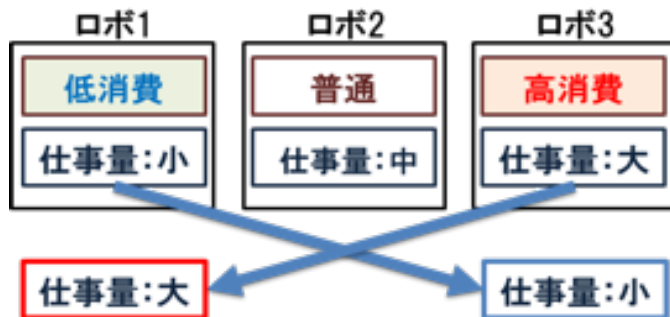
移動距離(D) と速度Vの時の消費電力

停止回数(S) と速度Vの時の消費電力

累積モデル

$$MC_{all} > MC_{min\_all} = \sum_i^x MC_i$$

最小化される組み合わせ(最適化)計算



近距離無線



燃費の良いロボットにより多くの仕事  
燃費の悪いロボットにより少ない仕事を割り当てる

[4] Toshiyuki Kantake and Midori Sugaya, "Power consumption reduction method by classifying individual differences using machine learning", Asia Pacific Conference on Robot IoT System Development and Platform (APRIS 2018), Oct30-Nov.2, 2018.

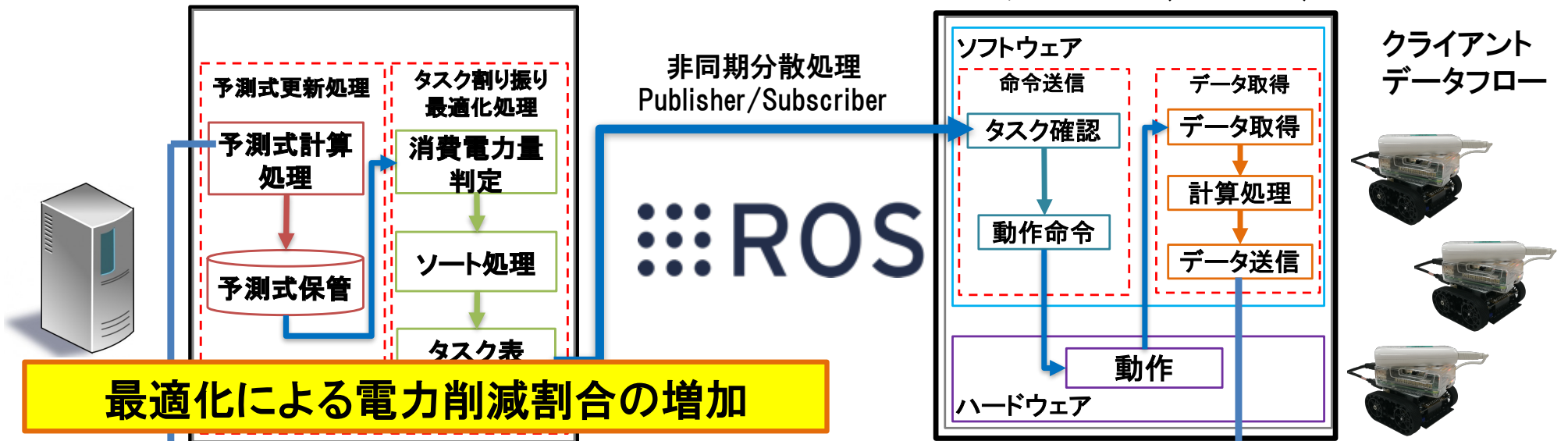
[5] 寒竹 俊之, 中野 美由紀, 菅谷 みどり, 複数台の移動ロボットのバッテリー残量を考慮した総電力削減手法, 組込みシステムシンポジウム 2017, 8月25日, 下呂, 2017年. 情報処理学会, 組込みシステムシンポジウム2017, 優秀論文賞



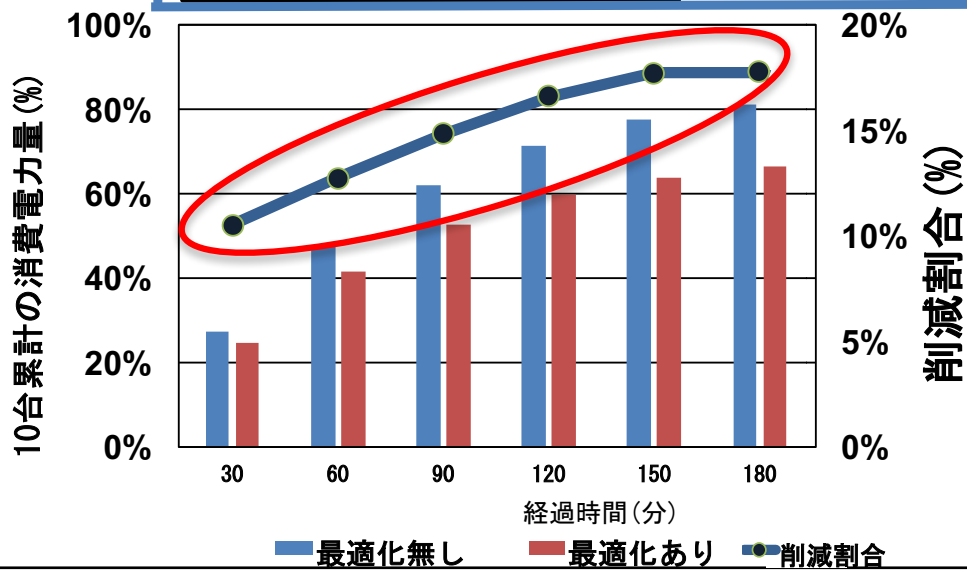
# ROS によるシステム実装と評価 [6]

エッジサーバー

クライアント(ロボット)



**最適化による電力削減割合の増加**



MEC による計算パワーの利用

複数台ロボットの消費電力の削減手法の研究 [4,5]

個々のロボの予測式 x ロボット数 → **最適化計算** → 動作再割り当て [4,5]

ロボット単体の消費電力モデル:  $MC_x = CPD_x^V \times D_y + CPS_x^V \times S_y$

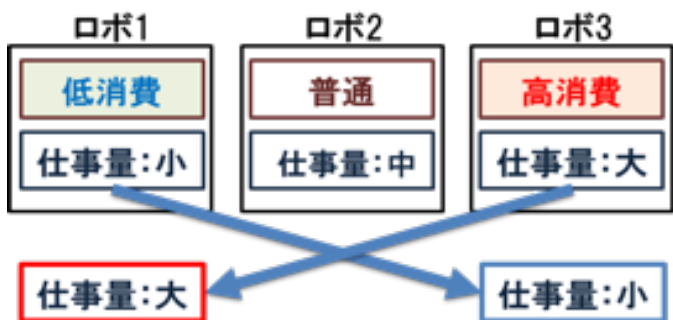
移動距離(D) と速度Vの時の消費電力

停止回数(S) と速度Vの時の消費電力

**累積モデル**

$$MC_{all} > MC_{min\_all} = \sum_{i=1}^x$$

最小化される組み合わせ(最適化)計算



近距離無線

計算機パワー不足による  
処理応答の遅延

無線通信干渉 2.4GHz 接続不能



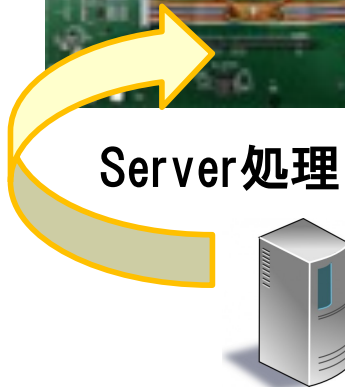
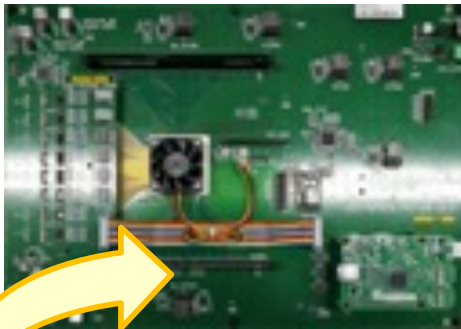
燃費の良いロボットにより多くの仕事  
燃費の悪いロボットにより少ない仕事を割り当てる

[4] Toshiyuki Kantake and Midori Sugaya, "Power consumption reduction method by classifying individual differences using machine learning", Asia Pacific Conference on Robot IoT System Development and Platform (APRIS 2018). Oct30-Nov.2, 2018.

[5] 寒竹 俊之, 中野 美由紀, 菅谷 みどり, 複数台の移動ロボットのバッテリー残量を考慮した総電力削減手法, 組込みシステムシンポジウム 2017, 8月25日, 下呂, 2017年. 情報処理学会, 組込みシステムシンポジウム2017, 優秀論文賞

# サーバでの最適化処理をMEC(FPGA)にて実現 [6]

FiC / MEC (FPGA)

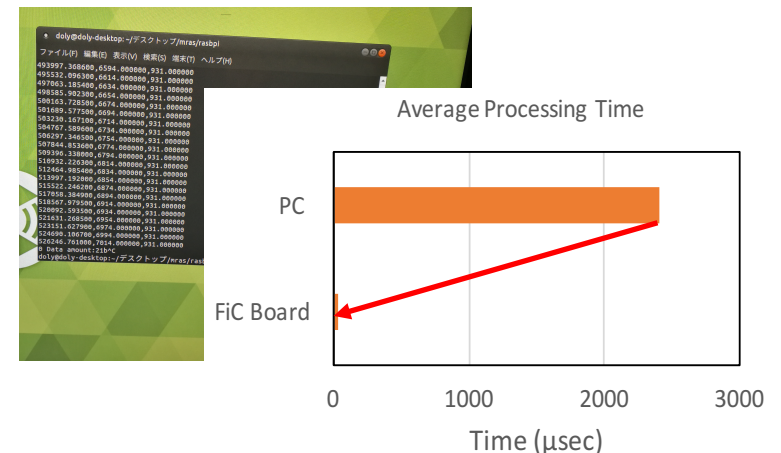
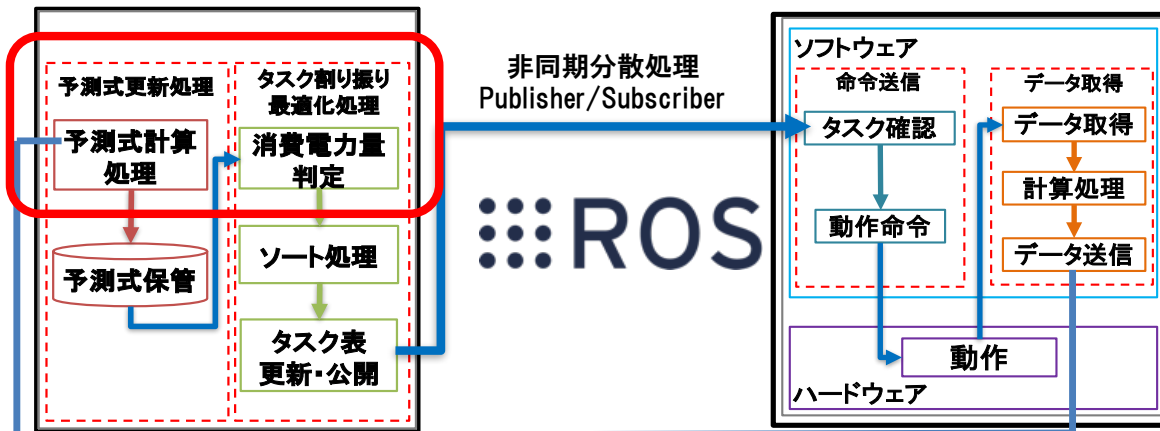


FiC ボード(MEC) を用いた FPGA による  
高速計算システムへ **サーバでの最適化**  
**計算処理を一部オフロード実装**

- データ数539個による重回帰分析プログラムの実行結果

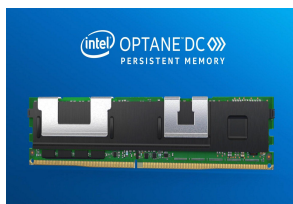
	FiCボード	PC
平均時間	32.28(μ秒)	2410.3(μ秒)

一般的なPCサーバと比較し、約**70倍**の性能達成



# エッジでのキャッシュ・ストレージ機能の実現

- 大容量高速**不揮発性メモリ(NVM)**を用いた Hybrid-KVS サーバの提案[5]



**NVM の特徴**  
 大容量化が容易  
 低レイテンシ、不揮発性メモリモジュール

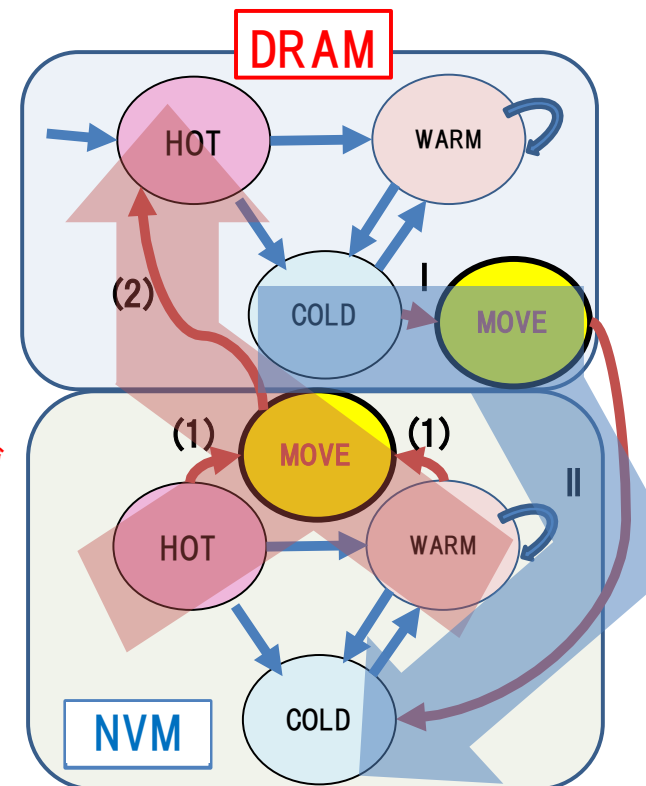
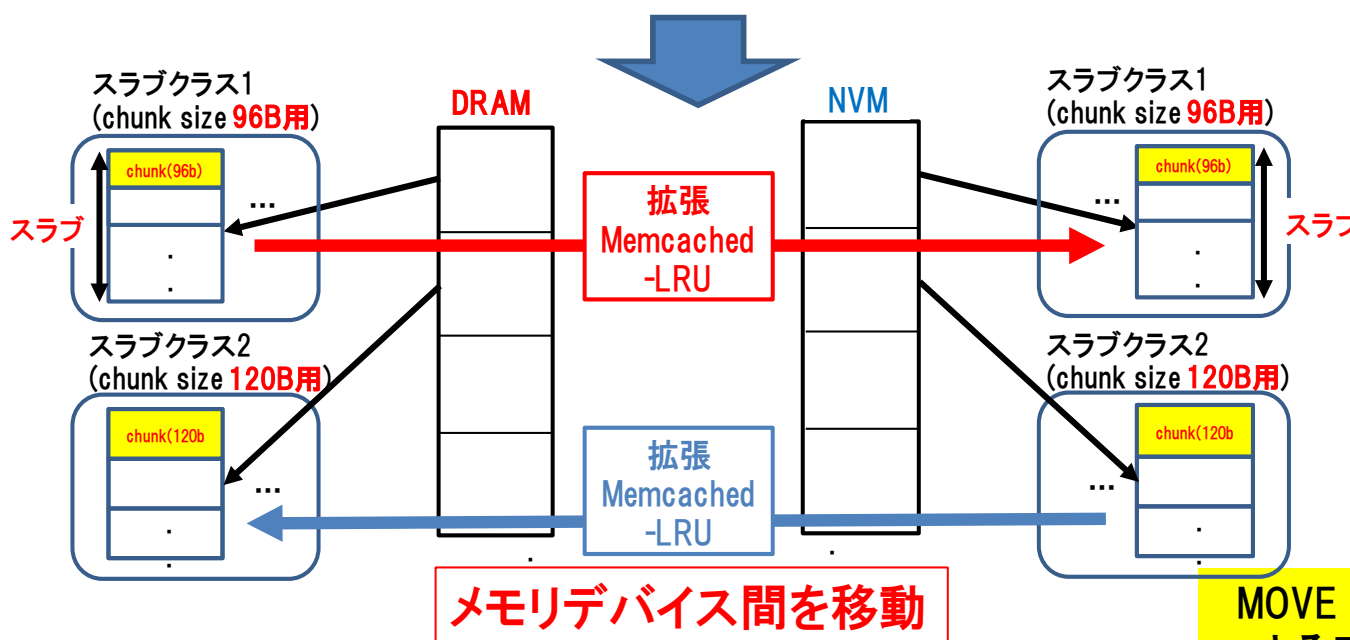
	読み込みレイテンシ
DRAM	約 80ns
DCPM	約 300ns

3~4倍

データ構造,  
アルゴリズムの拡張

KVS 既存手法:NVM の使用を想定しておらず, 性能が低い

KVS のDRAM+NVM への拡張 Dual-LRU の提案[5]



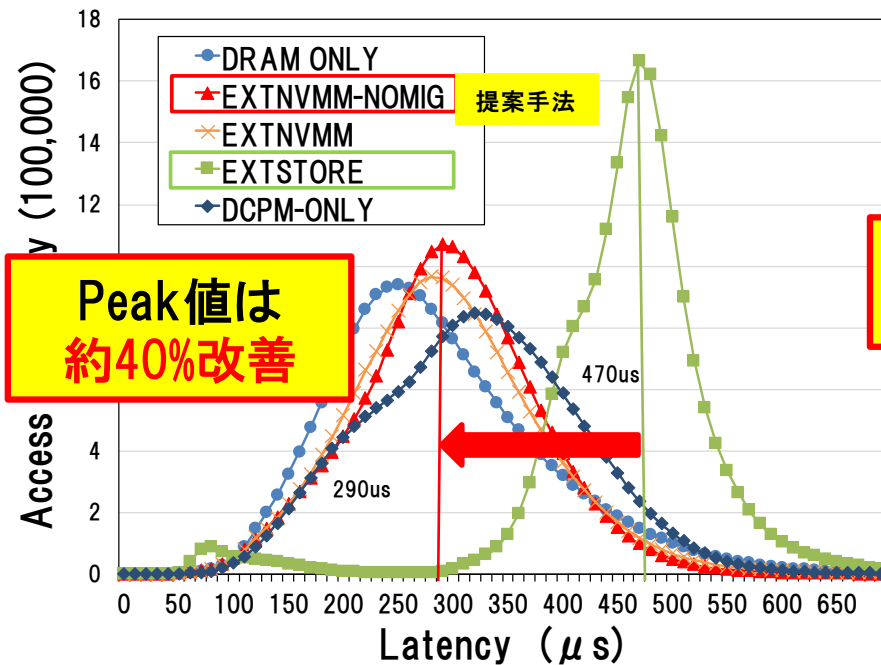
MOVE queue 拡張によるデータの移動

参照頻度に応じた DCPMの利用

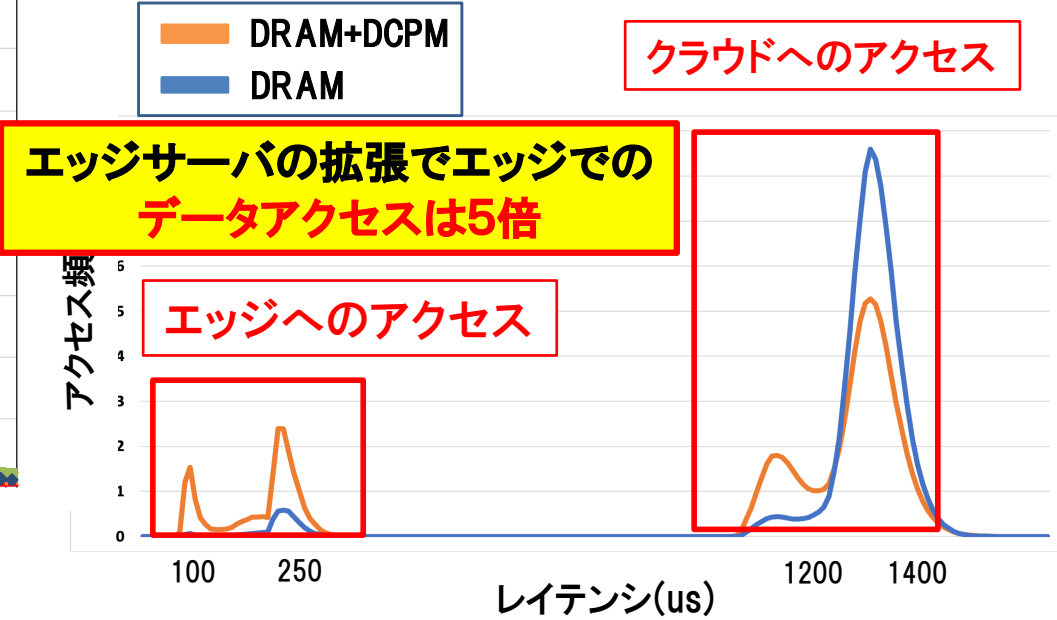
# 不揮発性メモリによるデータ保持, 応答性評価

- 評価: **大容量高速不揮発性メモリ(DCPM)**を用いた hybrid-KVS サーバの提案 [5]

提案(DRAM+DCPM)移動ありと, DRAM+DCPM(既存研究)の比較結果



Peak値は約40%改善

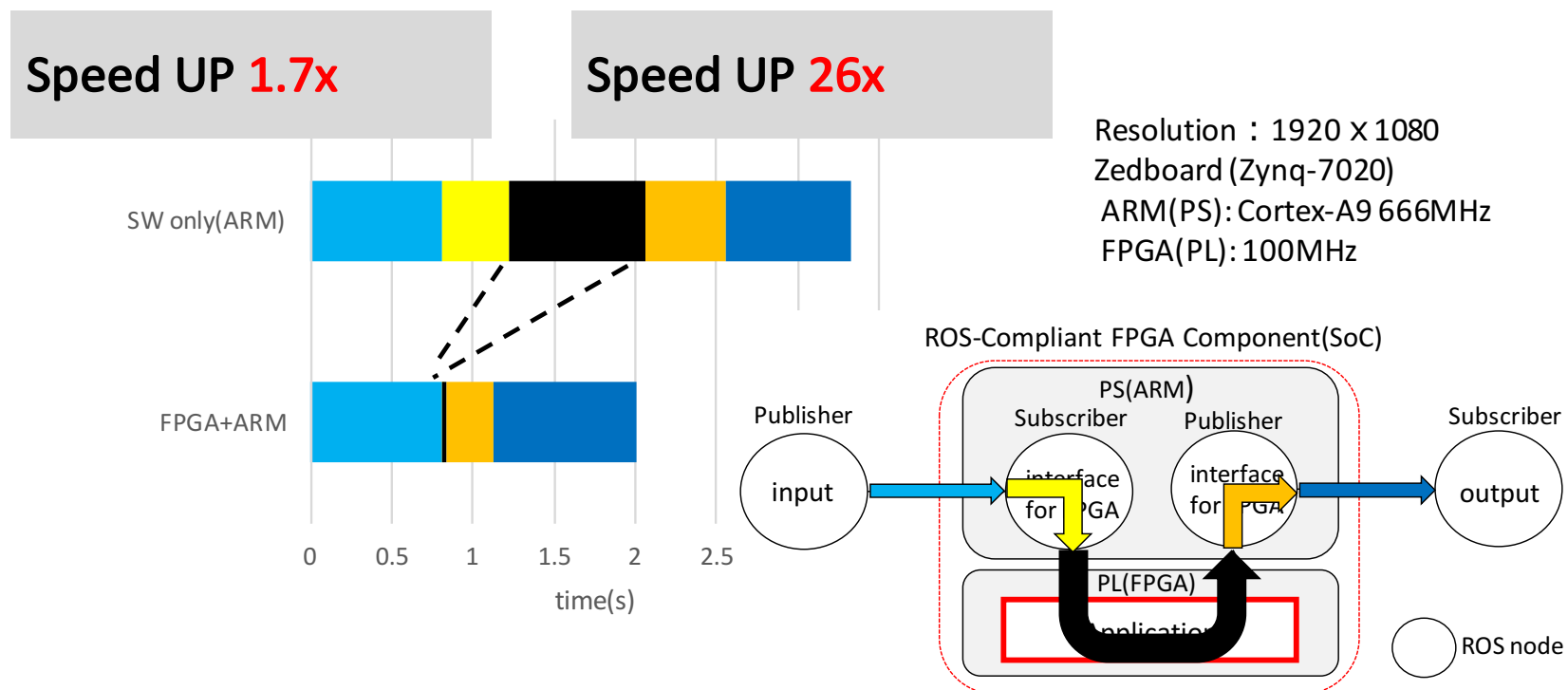


**KVS エッジサーバの拡張によりシステム全体のレイテンシ改善、スループット向上**

[5] 小沢 公基, 広瀬 崇宏, 高野 了成, 菅谷 みどり, ハイブリッド型メインメモリを対象とするKVSサーバの試作, コンピュータシステムシンポジウム2019 ComSys 2019

# ROS のTCP/IPコンポーネント (大川)

- ROSのハードウェア実装と発展



**Problem: TCP/IPの通信オーバーヘッドが大きい**

**TCP/IP のFPGA化により約26倍 の性能向上**

# MEC (FiC/FPGA) によるROSシステム試作 (大川)



USB (UVC) カメラ 入力\*

RGB画像\*\* (640x480)

画像縮小 および 二値化

2値化画像\*\* (28x28)

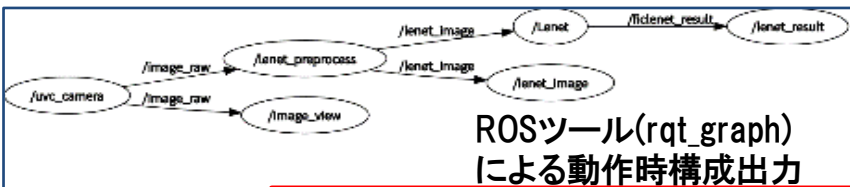
Lenet (認識)

結果 (float [10])

認識結果 表示

## ROS(Robot Operating System)準拠の利点

- ・IPネットワーク上どこでも配置可能
- ・\*オープンソースパッケージ(libuvc)および\*\*ROS標準形式データ(sensor\_msgs/Image)を用いることで**短期間での開発が可能**



ROSノード化



アプリ層:  
ソフトウェアによる  
ROS通信・データ変換



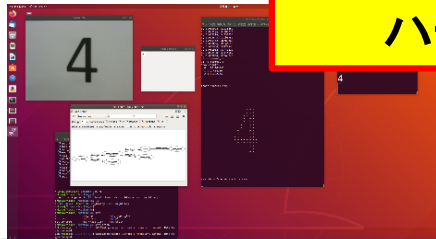
PC (Ubuntu 18.04) + FiCSES (KCU1500)

2値化画像\* (28x28)

Lenet (認識)

結果 (float [10])

## ハードウェア・ソフトウェア分割 → 統一的な記述インターフェイス



試作ROSシステムの動作確認画面写真

- Linux デバイスドライバ
- PCI express
- Aurora (高速シリアル通信)
- Crust-Core

**現状:** 試作ROSシステムの性能 = 約1.5Hz  
(通信を含む動作周期:0.69秒)

**今後:**

- ・性能ボトルネック分析・改善
- ・システム層・アプリ層の最適切り分けの検討
- ・アプリごとの通信・データ変換の自動生成

アプリ層:  
Lenetによる数字画像認識 (HLSモジュール)



FiCボード (FPGA)

# 今後の予定

## 統合ミドルウェアの研究開発

### ROS (Robot Operating System), KVSの統合ミドルウェアの開発

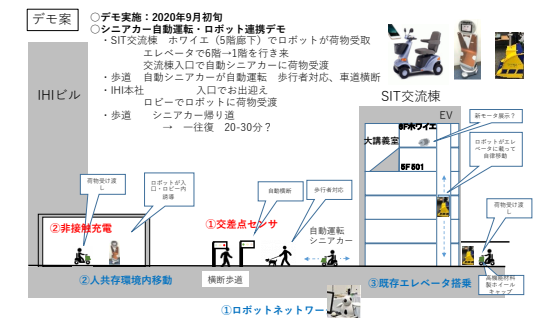
- クラウドとの連携も含めた仮想的なミドルウェア設計
- 並列化, AI(機械学習)支援を連携的に行う

### 5G技術との連携

- 5G Slice に応じたリアルタイム処理の設計
- RT要求の異なるタスクの動的最適化スケジューラ

### 統一的なインターフェイスの設計手法とツール開発

- サーバ分野, ロボット分野それぞれ目的の異なるミドルウェアが存在



## 実証実験

- 大量データ, 高速処理が必要となる実験, 5G, SLAM を利用する
- 複数台ロボットでの実証実験(芝浦工業大学, ロボットコンソーシアム)

## 規格化、標準化

### ソフトウェア設計の共通化(ROSベース)

- 人(センサ), 物(ロボット)全てをつなぐインターフェイス
- 複数台ロボットで実証実験しながら発展的な改善

