

# RECUA: WSN におけるエージェントを用いた仮想マシン拡張手法 RECUA:Extensible Virtual Machine using Agent in WSN

浦島賢治<sup>†</sup>, 福田浩章<sup>‡</sup>

Kenji Urashima<sup>†</sup>, and Hiroaki Hukuda<sup>‡</sup>

<sup>†</sup> 芝浦工業大学 工学部情報工学科

<sup>‡</sup> 芝浦工業大学 分散ソフトウェアシステム研究室

<sup>†</sup>Information Science and Engineering, Shibaura Institute of Technology.

<sup>‡</sup>Distributed Software System Lab, Shibaura Institute of technology.

## 要旨

WSN では、アプリケーションの更新や新規アプリケーションの追加などが必要になる場合には既存の設置してあるノードを使い回すことが望ましい。そのためには設置してあるノードを回収し、インストールし直す必要がある。

しかし、この作業にはコストが掛かるため頻繁に行うべきではない。従って本研究では、設置してあるノードに実装されているプログラムでは本来処理することが不可能なアプリケーションの追加であっても、実装済みの処理を組み合わせることで、無線経路で動的に処理可能にする手法である RECUA を提案する。

RECUA の実現には、Mate のようなバイナリコードを処理するための仮想マシンを利用したりプログラミングと、モバイルエージェントの技術を利用する。

## 1. 研究背景

WSN(Wireless Sensor Network) は山間部などの有線ネットワーク環境が整っていないような場所であっても、ノードと呼ばれる小型機器を設置することでセンサデータを活用したネットワーク構築を可能にする技術である。

表 1 のように、ノードは資源が限られているため、それらの使用を抑制することが重要視されている。従来はノード毎に 1 つのアプリケーション (以降 APP) を実装していたが、これでは必要な APP の数だけ、ネットワーク構築するためのノードが必要になり、コストが増加してしまう。従って、現在は 1 つのノードに対して複数の APP を実装することが主流となっている [1]。

しかし、この場合設置してあるノードのプログラムを更新する必要があり、広範囲に散布している全てのノードを回収することは現実的ではない。

そこで Mate[?] では、設置したノードに対してバイナリコードを送信し、オペレーティングシステム (以降 OS) が理解できるように翻訳する仮想マシンを予め実装しておくことで、プログラムを無線経路で更新する (リプログラミング [?])。このような仮想マシンの実装により、今まで実装されていなかった APP を実行可能にする。

しかし、一般に WSN の APP は多岐にわたるため、必要な命令セットを予めすべて決めることは難しい [3]。例えば、ノードを設置する前に、商品の状態管理 APP に必要な命令セットしか実装していなかった場合、新たに在庫管理 APP を追加する必要が発生しても、個数を数えるための算術演算命令が実装されていないため、ノードを回収して新たに必要な命令 (以降新規命令) を仮想マシンに組み込む必要がある。

DVM[?] では、実行時にメモリにアクセスするという OS の機能を利用することで、再インストールせずにプログラム更新を可能にするが、特定の OS に依存した機構のため、それ以外の OS を搭載したノードには対応できない。ネットワーク中に複数存在するノードは、それぞれが異なる OS を搭載している場合が殆どのため、OS に依存せずにプログラム更新可能にする方が望ましい。

そこで本研究では、新規命令を既存命令の組み合わせで表現するモジュール機構を導入し、モジュー

表 1: micaZ ノードにおける資源

プログラムメモリ	データメモリ	CPU	電源
128kB	512kB	7.4MHz	単三電池 2 本

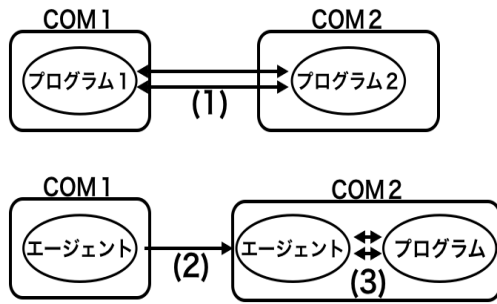


図 1: モバイルエージェント技術

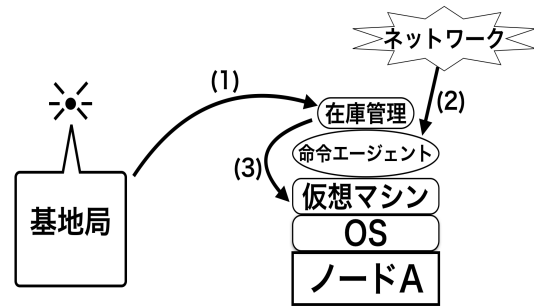


図 2: RECUA を使用した在庫管理 APP 実行方法

ルをモバイルエージェント [?] として実現する RECUA(Reusable Existing Codes Using Agent) を提案する。RECUA では、新規 APP で必要である命令が仮想マシンに実装されていない場合、WSN で動作するエージェントを即座に探索し、必要なノードに移動させることで新規命令の実行を可能にする。

## 2. モバイルエージェント

モバイルエージェント (以降エージェント) とはコンピュータ間をプログラムコードやスタック、プログラムカウンタなどの実行状態を保持し、移動しながら処理を行う事のできるソフトウェアのことである。図 1 に示すように、自コンピュータが他コンピュータと通信によるデータのやり取りをする必要がある場合、一般には自コンピュータ内で他コンピュータのデータが必要になる度に通信する必要がある (図 1.(2))。

一方、エージェントを送信することで (図 1.(2))、それ以降はエージェントとのやり取り (図 1.(3)) をすることで他コンピュータとの通信する必要がなくなる。これにより、通信時間の遅延や通信コストの低減を図ることを可能とする。

## 3. RECUA アーキテクチャ

RECUA では、ノードを設置する際に、必要最低限の命令セット (算術演算命令、スタック命令など) を処理可能な仮想マシンを用いる。そして命令エージェントを導入することで新規 APP の処理を実行可能にする。例えば、基地局がノード A に新規 APP である在庫管理 APP を送信する (図 2.(1))。しかし、ノード A は新規 APP に必要な命令を処理するための命令セットを保持していないため、命令エージェントをネットワーク上から探索する。そして、発見した命令エージェントを必要としているノードに移動させ (図 2.(2))、その命令エージェントを介することで (図 2.(3))、OS に依存せずに新たな命令語が含まれている処理を実行可能にする。

RECUA では実際には数バイトのバイナリコードになる処理をモジュール化し、関数のように扱うことを可能にする (図 3)。具体的な例としては、「温度がある一定以上になった場合に LED を緑色から赤色に点灯させて通知する」という処理を 1 バイトのバイナリコードで表現することが可能となる。それにより、温度の異常検知処理を使用するサーバ管理 APP や工場内の製品管理 APP に共通して利用可能にする (図 4)。

WSN では 1 ビットパケットの送信に掛かる消費電力と、1000 命令を実行する際の消費電力が等しいとされているため [?], 送信するデータ量は小さいほうが好ましい。RECUA ではエージェントの性質として、一度受け取ったエージェントを保持しているため、命令エージェントが必要な処理が発生した場合でも二度目からは通信する必要がなく、消費電力や待ち時間を削減できる。

また、1 回の送信で送ることができるバイト数は限られている為、それ以上のバイト数が必要になる APP では複数回に分割して送信しなければならない。そのため、分割して送信すると、その度に付与させ

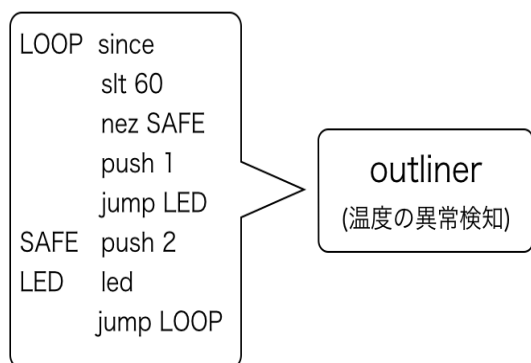


図 3: モジュール化の例

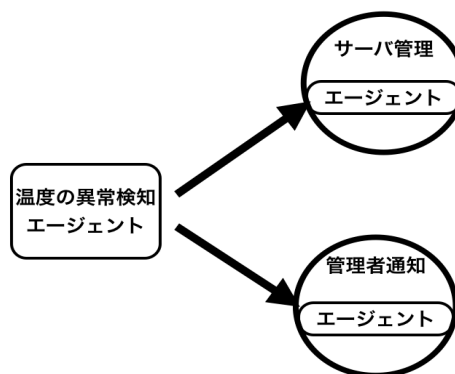


図 4: APP でのエージェント利用性

れヘッダーなどの余分な送信コストが発生してしまうという問題も存在する。この問題に対し、RECUA では上記で説明した、モジュールとして複数の命令をまとめる (図 3) ことで、分割して送信する必要がなくなり、送信する度に付与されるヘッダーなどに掛かる余分な送信コスト消費電力を削減できる。

#### 4. まとめと今後の展望

新規に APP を追加する場合、従来ではノードを回収して再インストールし直す必要があった。これを RECUA を用いることで OS などの環境に依存せずに APP の処理が可能となる。

現在の研究段階としては、RECUA を用いた機構での簡単な APP の実装をすることで、RECUA の実現性を確認した。今後は RECUA を用いた APP での消費電力や処理速度の測定をすることで既存手法との比較を行い、改善点の検討を行う。

#### 参考文献

- [1] P. Levis and D. Culler, “Mate: A tiny virtual machine for sensor networks”, ACM SIGPLAN, Volume 37, 2002.
- [2] Qiang Wang et al. “Reprogramming Wireless Sensor Networks: Challenges and Approaches”, IEEE Network, 2006.
- [3] Balani R. et al, “Multi-Level Software Reconfiguration for Sensor Networks”, Volume 22-25, ACM & IEEE, 2006, pp.112-121.
- [4] 佐藤一郎, “モバイルエージェント技術と研究動向,” NII Journal No. 3, 2001.