Context-Driven Self-Configuration of Mobile Ad Hoc Networks

A. Malatras, G. Pavlou

Networks Research Group
Centre for Communication Systems Research
Dept. of Electronic Engineering
University of Surrey, UK

http://ee.surrey.ac.uk/CCSR/Networks/

Presentation Outline

- Introduction
- Research Objective
- Related Work
- System Architecture and Design
  - MANET Management
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

Presentation Outline

- **Introduction**
- Research Objective
- Related Work
- System Design and Architecture
  - MANET Management
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

UniS

Introduction

- **Mobile Ad Hoc Networks (MANETs)**
  - Any device with wireless interface can join a MANET
  - Self-creating, self-organising, self-administrating
  - Dynamic nature and lack of centralisation brings benefits
  - Emerging paradigm for future wireless communications
- **Open challenges**
  - Mobile Nodes (MNs) move randomly, leading to continuously changing and unpredictable topologies
  - Unpredictable behaviour of radio channels
  - Time-varying bandwidth availability
  - Energy considerations
  - Scalability issues (MN number and density)
  - Diverse capabilities of MNs
  - Management issues

UniS

Presentation Outline

- Introduction
- **Research Objective**
- Related Work
- System Design and Architecture
  - MANET Management
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

Research Objective

- **Self-creating, infrastructure-less, spontaneous network**
  - Some form of management necessary
  - Need to rapidly create/deploy/manage services/protocols in response to context
  - Network-wide MN functionality alignment in a distributed fashion
  - Programmability is necessary for self-alignment purposes - download and activate required protocols/services on-the-fly
- **Self-configuration for MANETs is a necessity**
  - Context awareness a cornerstone to enable autonomy
  - Context-based adaptive self-configuration & optimisation
  - Leads to Autonomic Communication solutions in MANETs

Presentation Outline

- Introduction
- Research Objective
- **Related Work**
- System Design and Architecture
  – MANET Management
  – Context Management
  – MANET Self-Configuration
  – Middleware Architecture
- Evaluation
  – Testbed Experimentation
- Closing Remarks & Future Work

Related Work

- **Autonomic Communications**
  – Self-managed operation of communication networks
  – High-level objectives dictate system operation, trying to achieve self-configuration & optimisation
  – Most solutions/research approaches target system with sufficient resources and stability
  – Limited AC solutions for MANETs
  – Initial work presented as poster in IEEE ICAC 2005
- **Programmability**
  – Many different viewpoints
  – Means for (re-)configuration in our proposed approach
  – Initial work presented in IFIP/IEEE IM 2005

Presentation Outline

- Introduction
- Research Objective
- Related Work
- **System Design and Architecture**
  - MANET Management
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

System Architecture and Design

- **Basic elements of proposed system**
  - **Context-awareness to monitor surroundings (physical/computational)**
  - **Lightweight middleware platform present in every MN**
    - **Monitors context**
    - **Acts upon pre-conditions for (re-)configuration changes**
  - **Distributed management approach**
    - **Management Body (MB): Dynamic set of MNs**
    - **High-level predefined rules guide the decision-making process**
  - **Functionality of configuration changes (new services/protocols or alterations to existing ones) is carried out through software plugins**

Presentation Outline

- Introduction
- Research Objective
- Related Work
- **System Design and Architecture**
  - **MANET Management**
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

MANET Management (1/3)

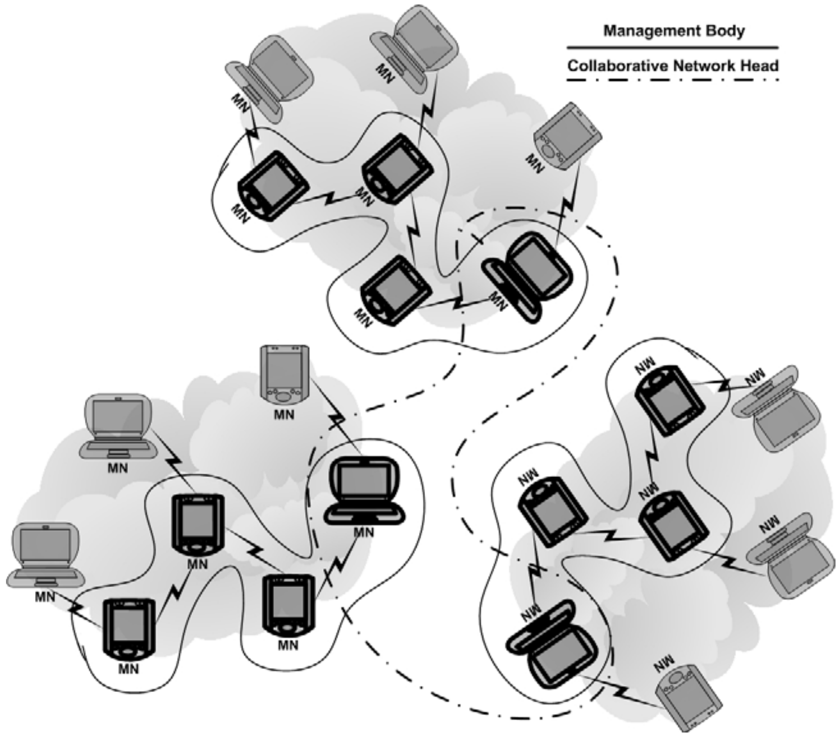
- **Hierarchical Management**
  - **MANET grouped into clusters each with an elected cluster head (CH)**
  - **CHs cooperate and elect a network head (NH)**
  - **Centralised approach not well suited for highly dynamic / volatile MANETs**
  - **Applies well to longer-term, relatively stable MANETs**
- **Distributed Management**
  - **All MNs have equal rights**
  - **Collaborative management decision making**
  - **Deployment procedure may be time-consuming**
  - **Prone to node misbehaviour**
  - **Very demanding as far as control traffic is concerned**

MANET Management (2/3)

- **Hybrid approach is adopted**
  - **MANET divided into clusters**
  - **Collaborative Management Body (MB) replaces CH and NH entities**
  - **MB collectively more stable than single CH (node movement changes mitigated inside the MB)**
  - **Motivated by virtual backbones for MANET routing and suits well the dynamic nature of MANETs achieving better robustness**
  - **MB is constructed in a distributed fashion as a Connected Dominating Set (CDS) of the underlying MANET graph**
    - **Capability Function (CF) calculated for every MN to indicate suitability to join the MB**
    - **Depends on computational resources (Memory, Processing Power, Battery, Current Load) and the location stability (Mobility Ratio) of the MN**

$$CF(x) = \frac{(w_1 \times MEM(x)) \times (w_2 \times PP(x)) \times (w_3 \times BP(x))}{(w_4 \times MR(x)) \times (w_5 \times CL(x))}, \quad \sum_{i=1}^{5} w_i = 1$$

MANET Management (3/3)

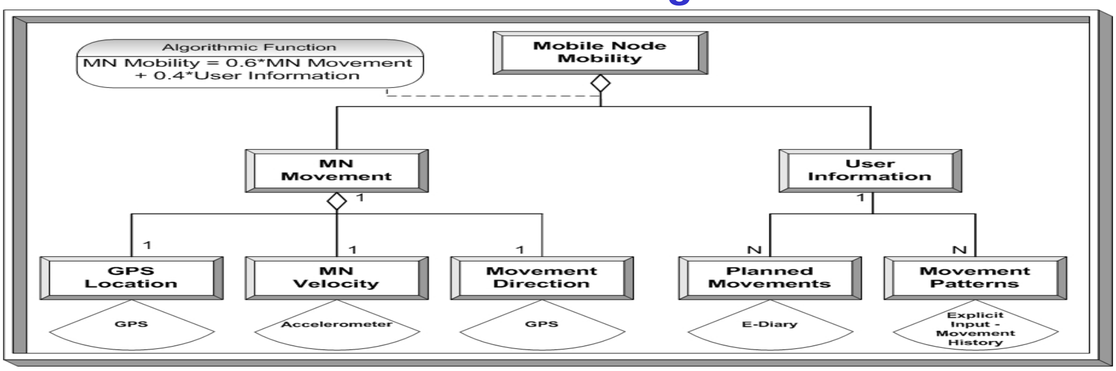


Presentation Outline

- Introduction
- Research Objective
- Related Work
- **System Design and Architecture**
  - MANET Management
  - **Context Management**
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

Context Management

- **Cross-layer context gathering**
  - **Context collected from physical & computational sensors on MNs i.e location, device characteristics, user profiles, network conditions, application requirements**
- **Higher-level contexts aggregated from "simpler" contexts are used for self-configuration**



Presentation Outline

- Introduction
- Research Objective
- Related Work
- **System Design and Architecture**
  - MANET Management
  - Context Management
  - **MANET Self-Configuration**
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- Closing Remarks & Future Work

MANET Self-Configuration

- Middleware on every MN processes local context to aggregate/higher-level contexts and advertises it to MB
- Only aggregate context is transmitted across the MANET
- MB nodes calculate MANET-wide values for aggregate context information
  - Every MB node calculates a value for the MNs it dominates in the CDS and then informs accordingly the rest of the MB
- Predefined rules located on the MB are matched against the predefined aggregated context to establish the need for configuration changes
- Uniform adaptation of MANET is guaranteed since all MB nodes possess the same values for context and the same predefined rules

Presentation Outline

- Introduction
- Research Objective
- Related Work
- **System Design and Architecture**
  – MANET Management
  – Context Management
  – MANET Self-Configuration
  – **Middleware Architecture**
- Evaluation
  – Testbed Experimentation
- Closing Remarks & Future Work

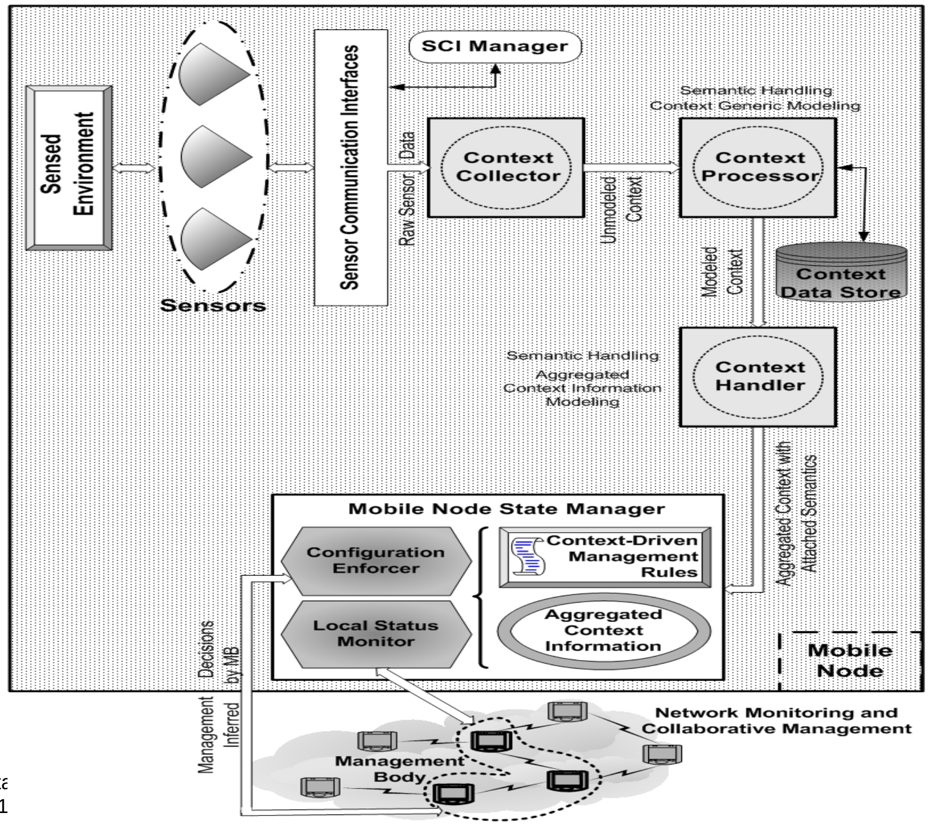
UniS

Middleware Architecture (1/2)

- **Lightweight and modular design/implementation with the following fundamental entities**
  – **Context Monitoring**
    - **Sensor Communication Interfaces (SCIs) to interact with sensors**
    - **Pruning of abundant context data using filters**
    - **Sensor semantics tagged to the data**
  – **Context Handling**
    - **Context Processor (CP) transforms the sensor data to our generic context model**
    - **Context Handler (CH) infers the aggregate context required for management reasons (hard-coded models)**
  – **Self-Configuration**
    - **MN State Manager module collects aggregated context information and advertises it to the MB**
    - **Configuration Enforcer (CE) listens for "commands" from the MB to execute locally by deploying the appropriate plugin**

UniS

Middleware Architecture (2/2)



UniS

Presentation Outline

- Introduction
- Research Objective
- Related Work
- System Design and Architecture
  - MANET Management
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- **Evaluation**
  - **Testbed Experimentation**
- Closing Remarks & Future Work

UniS

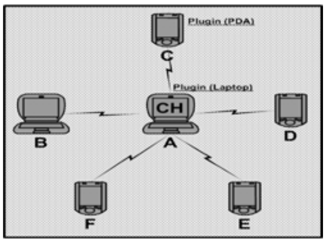
Evaluation
Application Scenario

- Selection of routing protocols available for MANETs
- Selected routing protocol is dependent on MANET stability
  - Proactive protocols (i.e. OLSR) suitable for relatively stationary MANETs
  - Reactive protocols (i.e. AODV) suitable for dynamic MANETs
- The routing protocol deployed on a MANET is typically fixed
- Context information regarding the mobility of MNs forming a MANET can yield an indication as to whether it is better to switch dynamically between two different routing protocols/strategies

Evaluation
Testbed Experimentation (1/2)

- XML-RPC protocol used for communication
  - Lightweight subset of SOAP
  - Platform/hardware independence possible with its use of remote procedure calls (RPCs)
  - XML-RPC uses the HTTP for transport and XML encodings for the RPC
- Java 2 Micro Edition (J2ME) is used with the Connected Device Configuration (CDC) framework
- Ad hoc testbed comprised of 2 laptops and 4 PDAs, running Linux Debian OS and Familiar Linux respectively
- Various scenarios have been tested to assess the platform efficiency and scalability
- 802.11b / various time-varying topologies assessed
- No mobility was assumed but similar effects were achieved through emulation

Evaluation
Testbed Experimentation (2/2)



Star Topology

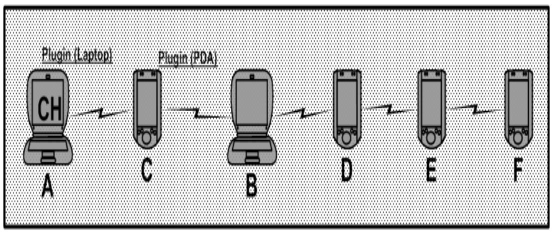
Time required for convergence: 41.96 sec

Routing related traffic: 7736 bytes

Inter-MN traffic: 41742 bytes

TFTP traffic: 1064880 bytes

The MB is formed of 1 node, solely A and the plugin to be distributed is owned by nodes A and C.



Line Topology

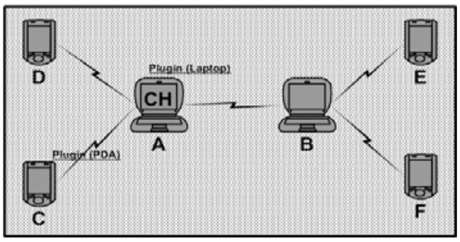
Time required for convergence: 47.94 sec

Routing related traffic: 14332 bytes

Inter-MN traffic: 83145 bytes

TFTP traffic: 1530924 bytes

The MB is formed of 4 nodes, C, B, D and E and the plugin to be distributed is owned by nodes A and C.



Random Topology

Time required for convergence: 44.43 sec

Routing related traffic: 12068 bytes

Inter-MN traffic: 51491 bytes

TFTP traffic: 1366896 bytes

The MB is formed of 2 nodes, A and B and the plugin to be distributed is owned by nodes A and C.

Presentation Outline

- Introduction
- Research Objective
- Related Work
- System Design and Architecture
  - MANET Management
  - Context Management
  - MANET Self-Configuration
  - Middleware Architecture
- Evaluation
  - Testbed Experimentation
- **Closing Remarks & Future Work**

Closing Remarks

- We presented the foundations and design principles of a context-aware programmable middleware platform that enables self-configuration in MANETs through a hybrid management strategy
- Our architecture follows a lightweight approach with XML-RPC as the basis of communication among middleware components
- We evaluated the performance of our approach through extensive testbed experiments
  - Our initial performance results are encouraging
  - The platform supports the expected functionality
  - Custom emulator built for scenarios incorporating MN mobility

Future Work

- To test the performance, scalability and quantify the benefits of our proposed approach on MANETs using simulation tools
- The capture and use of accurate context information is paramount to the correct operation of our approach
  - We plan to use context history and sensor accuracy metrics to overcome ambiguities in context
- Study security and self-protection in addition to autonomic self-configuration / self-optimisation