SEVENTH FRAMEWORK PROGRAMME
THEME 3
Information and Communication Technologies

Grant agreement for:
Collaborative project, Small and medium-scale focused research project (STREP)

Deliverable D4.3:

First Report on the Unification of Results

Project acronym: FRONTS
Project full title: Foundations of Adaptive Networked Societies of Tiny Artefacts
Grant agreement no.: 215270

Responsible Partner: RACTI
Report Preparation Date: Friday, 27 February, 2009

Contents

1 Introduction

The aim of the FRONTS project is to establish the foundations of adaptive networked societies of small or tiny heterogeneous artefacts. In a nutshell, we are working towards a science of adaptive organization of large networks of small or tiny artefacts.

Our approach to this goal is foundational and is based on algorithmic expertise. We intend to apply our models, methods and results to the scrutiny of large-scale simulations and experiments, from which we expect to obtain valuable feedback.

Our methodology follows basic principles of System Science. The networks of our study are (rather complex) "systems" of small and restricted parts, that may be organized via local communication. Dynamicity (and change) is a main characteristic, both for the network and also of its environment. Any computation is locally restricted, and no central control exists. Thus, the nets (systems) considered are *distributed* and cooperation among the tiny parts is not automatically guaranteed. Rather, it *has to be established* (in ad-hoc situations) and *maintained*, while adapting to dynamically changing external situations.

Still, if such systems are to be designed, they should be *"programmable"* and *trusted*. FRONTS aims to deliver at its end a set of well defined design rules for such systems.

We have decomposed our effort into four technical workpackages. As the contract indicates, they do no step in isolation, but they interact in a designed way. Here, we rephrase their goals:

- **WP1:** Aims in stating Models, Laws and Complexity measures for our networks.
- **WP2:** Aims in answering how the "net" prepares internally in order to be "ready to adapt".
- **WP3:** Aims in answering how the "net" reacts successfully to a dynamic external environment (how it adapts).
- **WP4:** Aims in verifying our methods and models via experiments and simulations. To this end, we have built a simulation and experiments environment.

Such a foundational project is expected to, (and must!), initially investigate *a multitude of possible approaches* (per issue) that seem suitable for meeting the goals of the project. The task of "Integration of Results", considered here, is thus, important, crucial for the coordination of the effort and basic for letting the partners of the project to shape a unified way of working towards the project's goals. Our consortium discussed the unification of results issued thoroughly during the first year. For each workpackage, we came up with some *basic questions* to be answered. This allowed us to identify the basic *issues* (and *functionalities needed*) for the answers. Then, we identified *technical approaches* and we went in depth in some of them in our first results. Finally, we have identified *possible directions* that need to be explored in the future.

We also decided to characterize each model, method or algorithm proposed, by the degree of their *scientific soundness*, the promise of capturing many subproblems (*breadth*), and by their relative *importance* towards our major goal, namely a science of adaptive organization of networks of tiny, heterogeneous artefacts.

Important note: In the rest, we use very short descriptions (and typical names) for the models and methods, in order to avoid repetition. All such patterns are explained in the Deliverables D1.1, D2.1, D3.1, and the indexed literature here.

2 WP1: Principles of Adaptively Organized Societies of Artefacts

2.1 The system to be captured

Our “systems” under investigation are networks of tiny, possibly heterogeneous artefacts. Such artefacts can interact (communicate) only when they are close to each other (and/or share a common resource). The artefacts are seriously restricted with respect to local computational ability.

2.2 Main questions

Question 1 – How to “program” such a net? What is a suitable model that can explain global functionality? The model must capture scalability, self and emerging behavior, ad hoc situations, adaptiveness, simplicity, and local restrictions on devices.

Question 2 – How to capture “dynamicity” (e.g., movement, faults, etc.)? We understood that this leads to the question of “how to capture the evolution of local interactions” between the artefacts.

Question 3 – What are the efficiency measures that one would like to *optimize* in such networks?

2.3 The structure of the possible models

We understood that any model that explains global behavior must have a submodel for local interactions (in fact, the local interactions form the network). We also concluded that Optimization is an issue in both. We found out that there exists an issue of carefully building the behavior and the interaction models. This structure is represented in Figure 1.

2.4 Models for Programming behavior and global Functionality

Our investigation indicated that the suitable models are those that focus on *Populations* whose members interact. Two such models (in fact related to each other) came out from the literature:

1. The “Population Protocols” of Aspnes et al. (see D1.1).
2. The Evolutionary Games in populations (inspired by Biology and Economy).

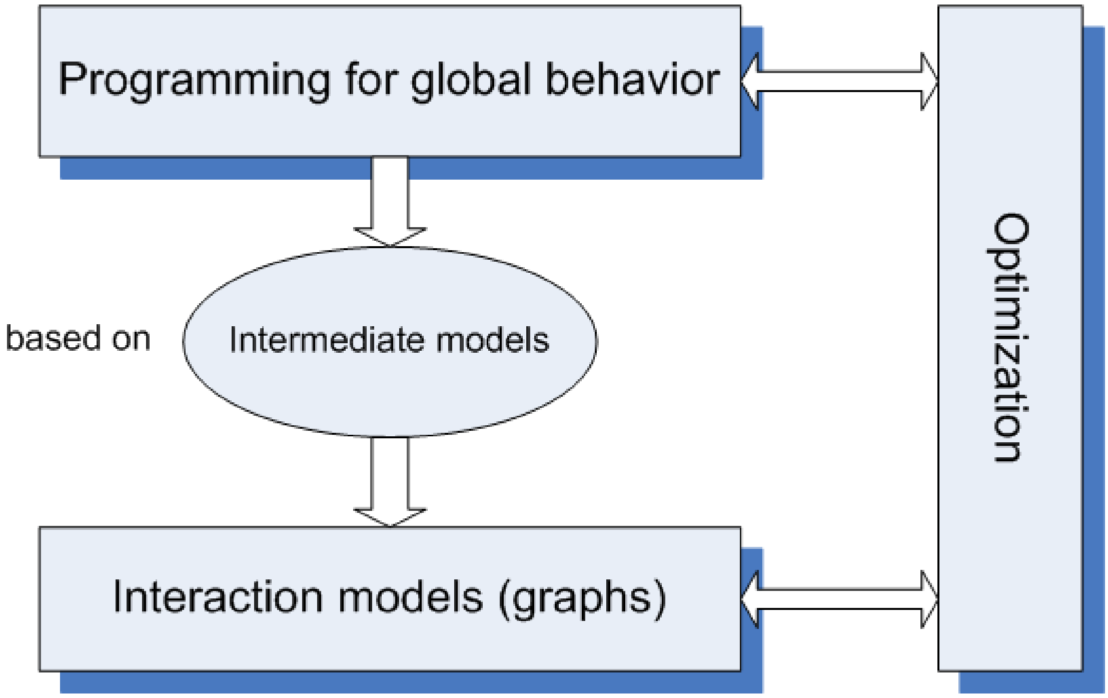

Figure 1: Structure of the possible models

Table 1: Models for programming (for global functionality) – all models allow for heterogeneity of artefacts.

Model	**Comment**
Population Protocols (and our mediated ones)	• New, more general. • Dynamic. • Adaptive. • Can explain cooperation
Evolutionary Games in Populations	• Old, (biology, competition/cooperation of species). • Adaptive. • Dynamic. • They are related.
Sensor Fields	• Sense & Act Communicating Automata. • New. • Adaptive and Dynamic. • Proposed by our Consortium.

During the first year we have worked on both, and we already have a concrete proposal for extending (1). In addition, our Consortium came up with a *new model* (based on the ideas of communicating automata). Table 1 summarizes our findings.

In the rest of the project we will compare (and possibly unify) the “Population Protocols” model with the “Sensor Fields” model.

2.5 Submodels for Local Interactions

Two such models (static) have been studied extensively by many researchers: the *Random Geometric Graphs* and the *Random Intersection Graphs*. In both, random-

Table 2: Local interaction models (various graphs)

STATIC	
Random Geometric Graphs	Proximity = Geometry
Random Intersection Graphs	Proximity = Common property (Combinatorial)
DYNAMIC	
Dynamic Geometric Graphs Evolving Graphs (new)	"Markov Chain" type evolutions of static models (motion)

ness is used to capture the ad hoc shape of such models (Networks). Geometric Graphs model the ability for local interactions between artefacts via "physical proximity". In contrast, Intersection Graphs explain the ability for interaction via shared properties (e.g., keys, channels, etc.) between artefacts.

During the first year we added dynamicity to the Geometric Graphs model. We also came up with a new model for dynamic interactions, called "Evolving Graphs". Both versions of dynamic interactions use Markov Chains to model possible evolutions of the network structure. Our findings are listed in Table 2.

2.6 An abstract intermediate model

A central problem in any foundational approach to nets of small artefacts is how to model the process of propagation of local changes until a global state is reached. This is exactly the focus of a model called "Markov Field", which is quite well-known to the Physics and Probability communities (see e.g., [Hek06] of WP1). We view this as an intermediate model and we intend to judge its value for FRONTS in the future.

The "Markov Field" model tries to capture convergent behavior. It comes with a general algorithmic idea from Physics, called *Survey Propagation*. The efficiency and applicability of this idea to dynamic environments is an open issue.

2.7 Optimization in the Net

While the models presented in section 2.4 can capture usual efficiency measures (e.g., local memory, local computation, time to achieve a global task, fairness) they fail to grasp the issue of *how to measure data flow* in the network. This is why we have also considered Data Flow / Data gathering (sub)models in WP1. Such models allow the designer to focus on *scheduling* and *synchronization* scenes, as well as on *Energy Consumption* issues. This submodel can be plugged in to any of our "abstract machines" of Section 2.4.

3 WP2: Designing the Adaptable Network Infrastructure

3.1 The issue to be captured

FRONTS has already postulated that any distributed adaptive system should devote part of its functionality to self-organize internally. This self-organization activity is viewed to continuous and it is a very necessary overhead that the system pays, in order to *be ready to adapt* (i.e., to react to changes in the external environment). Thus, WP2 focuses on the methods and network procedures that perform this internal preparation.

Such network procedures need to be *scalable*, *distributed*, *simple to implement* and should use only *local information*. A part of them should concern on how to keep the net being *secure*.

3.2 Main questions

Question 1 – What are the main parts of our "System" for which the internal organization is crucial?

Question 2 – Self-stabilization allows the system to automatically re-gain its consistency from any arbitrary state (that could the result of transient faults). What are the important primitives for self-stabilization?

Question 3 – Which, in more generality, are the *basic functionalities* needed to be always present so that the system can adapt its internal communication? What are the basic functionalities that allow the system to re-organize?

3.3 The structure of the internal organization "layer"

Already in the contract of FRONTS we had distinguished three parts of the internal organization activity:

Task 1: How to re-organize the *Communication Infrastructure*?

Task 2: What are the *roles of artefacts* that can be re-adjusted? What are the methods to do this?

Task 3: What is the approach to re-organize the *security* of the net?

For each part, FRONTS came out by proposing some basic *functionalities*. They are listed in Table 3.

Table 3: Continuous Re-organization

Part of the System	Functionalities needed
Internal Communication	▷ Reorganize *data gathering* ▷ Allow for *redundancy in connectivity* ▷ Maintain *hierarchy* in the communication graph that can keep a consistent status of the network
Redefine Roles of Artefacts for internal Reorganization	▷ Local Learning Methods ▷ Trust enhancement ▷ "Equilibria" and economic approaches for self-organization ▷ (Node "colors" are important to indicate and code local constraints)
Security	▷ Privacy Protection (keys and identities management) ▷ How to secure the routing?

3.4 Our initial approaches to the required functionalities

Internal communication

1. **Re-organization of data gathering**: we propose "multi-hop" adaptive schemes.
2. **Redundancy in Connectivity**: we have examined spanners (redundant path structures) of low stretch.
3. **Self-organized hierarchy**: we gave *self-stabilized* clustering methods that maintain *local leaders* in various areas of the net. We also provided a novel *on-demand* stabilization method that always produces a correct snapshot of the net.

Roles of artefacts

1. **"Economic" approaches:** The general idea is to allow artefacts to "play" strategic games and to go into sequences of local *best responses* to changing situations. This requires to define and maintain *utilities* and *payoffs* (also pure *strategies*) in each artefact. Such approaches allow the system to converge to equilibria, via local "selfish" moves. We took into account the fact that artefacts are very restricted devices; thus they cannot be very intelligent players (they are *automata*; in general their *rationality is bounded*).

   We can encode payoffs and strategies via automata *states* (and counters). We provided the following approaches:

   - *Enforcement*: Improper behavior of some artefacts results in punishments from the rest the system.
   - *Selfish local search*: We gave a distributed "game" via which the artefacts manage to achieve good proper colorings of the whole net. Each artefact is viewed as a very simple device, with a *color*. The continuous derivation

of global proper colorings allows for many important global operation (like scheduling without conflicts, etc.).

2. **Learning methods:** We use them to *purify* arriving data at an artefact. The required Roles are now "certainty levels".
3. **Roles for Trust enhancement:** We combined *self-stabilization* and *enforcement*. Roles are similar to (1).

We also examined self-adaptive *recommendation* methods. The artefacts now must maintain views of "local past history" of the system.

Security

1. **Privacy protection:** We propose *Combinatorial cryptography for low-end devices*.
2. **Secure Routing:** We propose routing based on probability. This overcomes the problem of "changing links" due to dynamic faults or mobility.

4 WP3: Adapting to the Dynamic Environment

4.1 The issue to be captured

Any adaptive network must react properly to external influences induced by changes of the environment. Our system (society of tiny artefacts) is assumed too large to be handled in a centralized manner. We have to design *local strategies* that are capable of handling the required global tasks. Such strategies should benefit from the internal re-organization methods of WP2.

4.2 Main Questions

Question 1 – What is the logical structure of the set of issues identified in the contract for WP3?

Question 2 – What are the crucial external "changes" to which the system should adapt?

Question 3 – Reaction to a sudden external change may result in chaotic internal competition for resources. How do we handle it?

4.3 The structure of the "react to the external" layer

We came up with the following conceptual structure (arrows mean "is based on") indicated in Figure 2.

4.4 Crucial external changes

We recognized the following issues:

1. The terrain of the deployment of artefacts is *unknown* and the *obstacles* (for the artefacts' movement or communication) may *move* in an uncontrolled way.
2. A "target" that moves through the net must be tracked (here the external request is to track the target).
3. The environment may "move" our artefacts in an uncontrolled way.
4. An external request comes for the net to offer some service. A subgroup of the artefacts has to be chosen. Which subgroup is the most economical (i.e., an efficient answer to an external request)?
5. How can an artefact trust other artefacts? Here the external change is an attack to a predefined trust structure.

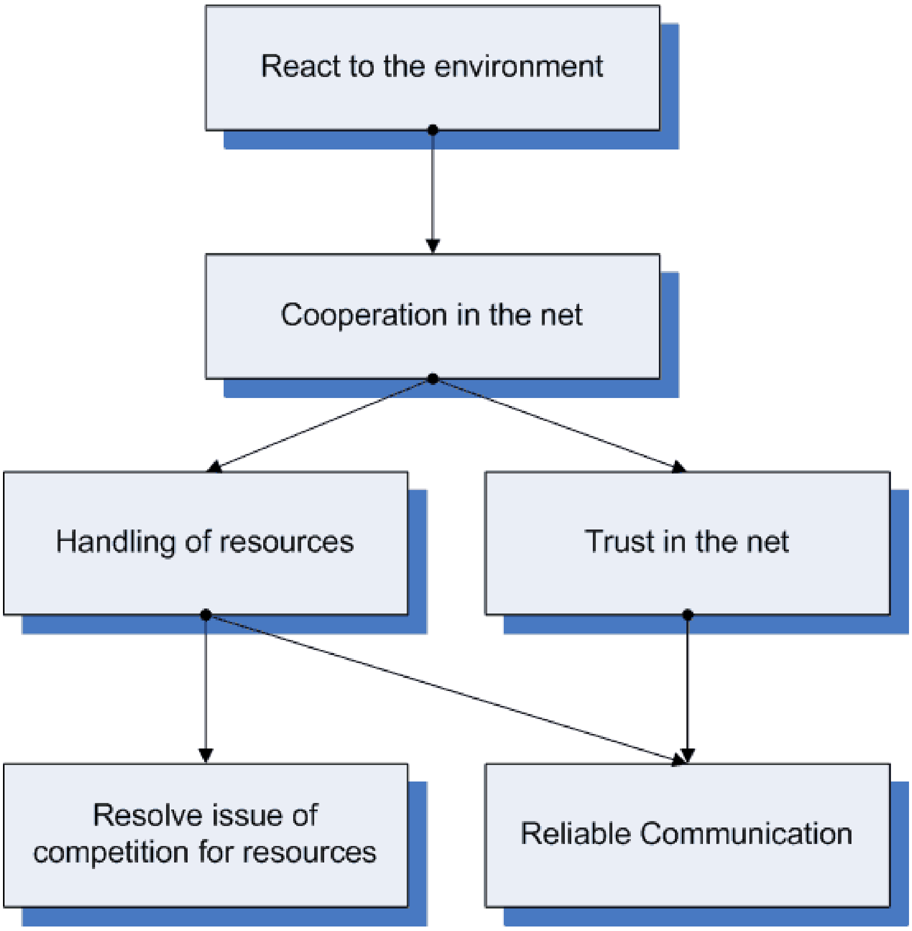

Figure 2: Conceptual structure

6. The external change is a failure in some communication links, either accidental or on purpose (Byzantine).

4.5 Our initial approaches

Our initial approaches are indicated in Tables 4-8.

Table 4: Approach to distributed cooperation

Functionality	Reacts to	Approach
Distributed Cooperation	Low energy levels	Hierarchies and clusters Colony algorithms
	Defectors (artefacts) exist, due to an external attack	Correlated punishment
	External terrain changes	Maintain geometric "formations" of artefacts (similar to robots' coordination

Table 5: Approach to tracking of resources

Functionality	Reacts to	Approach
Tracking of Resources	Unknown terrains	Terrain exploration (similar to Robotics)
	Target tracking	Use traces (our THTP protocol)
	Bypass obstacles	Probabilistic data propagation
	Obstacles appear (unknown)	Adaptive routing with back-tracking

Table 6: Approach to physical changes

Functionality	Reacts to	Approach
React to physical changes	Passive movement of artefacts	Maintain the net via moving (actively) some relay nodes
	A request that can be met by some subnetwork while the net changes	Dynamic Facility Location
	Dynamic appearance of obstacles	Adaptive routing via "sensing the obstacles" (similar to previous figure)

Table 7: Approach to trust

Functionality	Reacts to	Approach
Trust	An attack that may change id's of artefacts	Multi-level dynamic key distribution
	Behavior of some artefacts is altered	Emerging Trust via rational selfish choices (a behavior equilibria approach)
	System grows too large	Emerging Trust from statistical properties

Table 8: Approach to reliable communication

Functionality	Reacts to	Approach
Reliable Communication	Random faults - unknown distributions	Learning methods
	Adversary captures some nodes	Dynamic altering of communication paths
	Nodes sleep / messages not received	Redundancy, probabilistic restarts

5 Some conclusions and further unifying actions

1. A common "denominator" exists in the basic algorithmic approaches. Common techniques include:
   - Probabilistic methods (e.g., random walks in the net) to fight against dynamic changes.
   - Equilibria building (in approaches from games, economy, self-stabilization).
   - Redundancy structures.
   - Local Greedy methods with backtracking.
2. Many methods of WP3 are similar in spirit to those of WP2 but with different objectives. It seems that the "internal preparation" layer of WP2 will help a lot to reduce effort in the methods of WP3. We should mark the overlaps and point out the synergy of the approaches.
3. Till now, we have conducted some experiments mainly to verify individual methods. We expect that integration of methods will be helped a lot by the common Experimentation base and environment.
4. Almost all of our methods are local and simple (e.g., can be implemented by devices of restricted functionality. It seems that "global behavior" can indeed be base on simple local methods.

References

The union of references are included in the WP1, WP2 and WP3 reports (deliverables D1.1, D2.1, D3.1). We do not repeat them here.