Self-organizing mobile surveillance security

Duco Ferro & Alfons Salden, Almende B.V. Rotterdam, The Netherlands

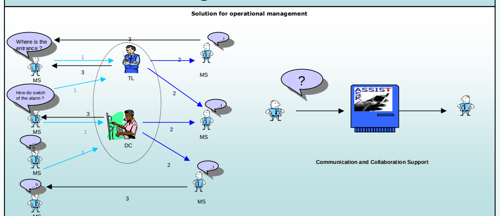
INTRODUCTION

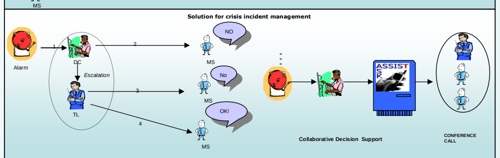
Mobile Surveillance Security

- Guards perform patrols

Problem

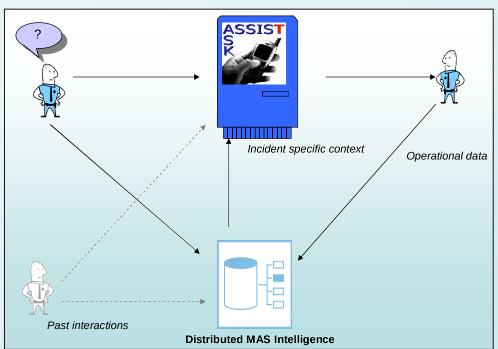
- Incidents require patrols to be coordinated:
  - Communication and collaboration
  - Decision making





Towards a solution

Automated reconfiguration by **ASK-ASSIST** following a dynamic scale-space paradigm



Reconfiguration Framework

DYNAMIC COALITION FORMATION ENVIRONMENT

Coalition Environment Elements

A coalition formation environment consists of :

- $\mathcal{A}$: agents
- A = A(t): active elements (roles mapping to agents)
- B = B(t): passive elements (contextual elements)
- T = T(t): tasks
- t : time

Assignment of roles to task

An assignment of roles to tasks in a situational context at given time t:

- $\gamma: 2^{A(t)} \times 2^{B(t)} \rightarrow 2^{T(t)}$

Coalition

A coalition is a time-ordered composite task assignment to agent roles in contexts, labeled by a route number r in R:

- $\gamma^r = \gamma^{A(t_n),B(t_n)} \circ \ldots \circ \gamma^{A(t_0),B(t_0)}|_r$

Configuration

Having a coalition formation environment, the goal is not only to generate a configuration of elements, so-called coalitions, that can handle common mobile surveillance circumstances, e.g. a daily patrol, but also that can handle unexpected security incidents by providing improvisation support to humans by newly added elements, e.g. ASK-ASSIST.

- A ***configuration*** $\Gamma_0 \in \Gamma$ for the routes $r \in R$ consists of $\gamma^{r_0} \cup \ldots \cup \gamma^{r_n}$

RECONFIGURATION

Reconfiguration function

When security incidents, like alarms, occur at a certain point in time, assistance is required. This means that the configuration needs to be altered. The reconfiguration of the affected coalitions is described by the reconfiguration function:

- $f : \Gamma \times t \rightarrow \Gamma$

$$f(\Gamma,t) = \begin{cases} \Gamma(t) = \Gamma_s(t), & t_0 \le t < t^i \\ \Gamma(t) = \Gamma_s^i(t), & t^i \le t < t_n \end{cases}$$

$$\Gamma_s = \cup_r \gamma^r$$
$$\Gamma_s^i = \cup_r \bar{\gamma}_{t_n}^{A_n,B_n} \circ \ldots \circ \bar{\gamma}_{t_{i+1}}^{A_{i+1},B_{i+1}} \circ \bar{\gamma}_s^i \circ \gamma_{t_i}^{A_i,B_i} \circ \ldots \circ \gamma_{t_0}^{A_0,B_0}|_r$$

A reconfiguration represents the required reassignment of agents, e.g. security guards, to agent roles and those in turn to tasks given new situational contexts, such as an unexpected alarm or the introduction of ASK-ASSIST.

Required reconfiguration features

Reconfigurations should limit e.g. (cognitive) workload in an organization to acceptable levels in case of incidents or introduction of ASK-ASSIST, and should at least sustain communication, collaboration and decision processes amongst humans in resolving / benefiting from such cases. Therefore reconfigurations should be assessed by means of consistent value functions M for:

- Robustness
- Performance

Dynamic SCALE SPACE PARADIGM

Context dependent reconfiguration hierarchies induced by value functions

The robustness and improvement of performance of reconfigurations – the acquisition of value functions - have to be accounted for by empirical models. Such accounts may come about after statistically analyzing the logged agent network history of reconfiguration patterns and the associated values induced by humans explicitly as feedback or implicitly as usage. This allows MAS to predict, to automate and to recommend ranked lists of reconfigurations, so-called context dependent hierarchies CMH on the reconfiguration space parameterized by contexts and ordered by the value functions M.

$$CMH(f) = \{(\Gamma_q^{opt}(t_0^r), \ldots, \Gamma_q^{opt}(t_n^r)) \,|q, n, r \in \mathbb{N}|\}$$

MAS as self-organized criticalities

Empirical models for reconfiguration can be distilled upon dynamic learning performance metrics of multi-agent systems (MAS) over situational context-specific reconfiguration spaces. These models manifest themselves as self-organized criticalities of MAS interacting and co-evolving with distributed ICT systems and organizations of humans, following a so-called dynamic scale-space paradigm (Salden, Ter Haar Romeny and Viergever, 2001). A dynamic scale space paradigm enables rapidly acquiring robust and sustainable context-dependent reconfiguration schemes as self-organized criticalities of MAS at a certain limited level of operational or evolutionary scale. This self-organization of MAS interacting and co-evolving with those systems and organizations is governed by:

$$j^F = -\frac{\nabla_{v_s}^{\Pi} F}{\kappa^2(\sqrt{g(\nabla_{v_s}^{\Pi} F, \nabla_{v_s}^{\Pi} F)}}$$
$$Z = \exp\left[-F\left[V_i(x)\right]\right]$$
$$F\left[V_i(x)\right] = \sum_{i,k,p} dv^p \left(\tilde{V}_{i;\pi_k(g_1 \ldots g_k)}(x, \tau_{i;\pi_k(g_1 \ldots g_k)})\right)$$

Agent Development Environment

Common Hybrid Agent Platform (CHAP) taking human in the loop
http://chap.sourceforge.net

Design Approaches	***Features***
Customer	Ease of distribution, maintenance, self-learning
Developer	Reusability, agent libraries, coordination libraries, protocol libraries
	Distributed contributions, community standards, interoperability, user tool support, public relations
	Domain libraries, domain tool support, rewriting logic, process algebra, concurrent models
Aspect	Modularity, separation of concerns, aspects
Complexes	Emergence, aggregation, escalation, hybridization, coevolution

Distributed Self-Learning -- Self-Organization – Self-Management to address Heterogeneity, Concurrency, Robustness, Scalability and Sustainability

Communication, interaction, collaboration, monitoring, contextualization, decision making, reconfiguration - coalition formation, planning, scheduling

MSS Application