My speech at

VINO ‘11

Holger Schlingloff
Topic of this talk

Should I speak about

- Liebherr and maritime cranes?
  - nice mountain story, but too little content
- Copreci and the modelling of gas burners?
  - nice engineering flow, but not yet complete
- Berlin Heart and ventricular assistant devices?
  - nice application, but not yet enough formal methods
- Thales Rail Automation and software product lines?
  - very relevant topic, but preliminary results only
- Opel GM and specification of fuel cells?
  - interesting questions, but Hartmut already did
SOS: Werewolves!

- Self-Organizing Systems
- Winning strategies in games
- Temporal logics, model checking, ...
- Program synthesis

- Work in collaboration with Jan Calta
  - Finding uniform strategies for multi-agent systems
  - Synthesizing strategies for homogenous multi-agent systems with incomplete information
Problem Description

- Consider a network of computing nodes (agents)
  - e.g. smart sensors in earthquake detection
- They have to synchronize (consent)
  - e.g. decide on whether an earthquake occurred
- They are distributed (incomplete information)
  - no agents has complete knowledge of the situation
  - e.g. can only ask his (few) neighbours about their state
- They are all alike (homogeneous)
  - e.g. all pre-programmed in the same way
- The overall behaviour must be “correct”
  - e.g. earthquake warning iff there is an immanent earthquake
• **Strategy:**
  - repeat:
    - ask your right and left neighbour about their colour
    - if they both are different from your own, swap colour

• **Termination? Correctness?**

• **How to find such strategies?**
Definition 1. A modular model $\mathcal{M}$ is a tuple

$$\mathcal{M} = \langle \text{Agt}, \text{Act}, \text{St}, \Pi, \pi, \text{neig}, k, \Sigma, \text{man}, \text{tran} \rangle,$$

where

- $\text{Agt} = \{1, \ldots, n\}$ is a set of agents;
- $\text{Act}$ is a finite nonempty set of actions; arbitrary actions will be denoted by lower-case Greek letters from the beginning of the alphabet, such as $\alpha, \beta, \ldots$;
- $\text{St}$ is a finite nonempty set of agent's states (the set of system states is then $\text{St}^n$); we denote the members of $\text{St}$ using lower-case Latin letters such as $q$ and members of $\text{St}^n$ using upper-case Latin letters such as $Q$;
- $\Pi = \Pi_1 \cup \cdots \cup \Pi_n$ is the union of sets of atomic propositions, one for each agent,
- $\pi : \Pi \rightarrow \mathcal{P}(\text{St})$ is a valuation function,
- $k$ is the maximal number of neighbors an agent can have,
- $\text{neig} : \text{Agt} \times \{1, \ldots, k\} \rightarrow \text{Agt} \cup \{\#\}$ is a neighborhood function (thus, if $a \in \text{Agt}$ and $1 \leq i \leq k$, then $\text{neig}(a, i)$ is the $i^{th}$ neighbor of $a$; $\text{neig}(a, i) = \#$ means that $a$ does not have the $i^{th}$ neighbor),
- $\Sigma$ is a finite nonempty set of manifestation symbols,
- $\text{man} : \text{St} \rightarrow \Sigma$ is a manifestation function; for technical reasons, $\text{man}(Q[\#]) = \lambda.$

where $\lambda \in \Sigma$ is a manifestation of an "absent" neighbor;
- $\text{tran} : \text{St} \times \Sigma^k \times \text{Act} \rightarrow \text{St}$ is a transition function.
Strategies and Outcomes

Definition 2. A homogenous strategy is a function $S : St \times \Sigma^k \rightarrow Act$ assigning actions to perceptions.

Definition 5. A path is an infinite sequence of system states $\Lambda = Q_1, Q_2, Q_3 \ldots$ that can be effected by subsequent action vectors; that is, for every $j \geq 1$, there exists an action vector $A \in Act^n$ leading from $Q_j$ to $Q_{j+1}$. The $j^{th}$ component of $\Lambda$ is denoted by $\Lambda[j]$.

Definition 6. The outcome of a homogenous strategy $S$ at system state $Q$, denoted by $\text{out}(Q, S)$, is the path $\Lambda$ such that $\Lambda[1] = Q$ and

$$
\Lambda_{j+1} = \prod_{a \in \text{Agt}} \text{tran}(\text{per}(\Lambda_j, a), S(\text{per}(\Lambda_j, a))),
$$
ATL and LTL

\( \varphi ::= p | \neg p | \varphi \land \varphi | \varphi \lor \varphi | \langle A \rangle X \varphi | \langle A \rangle G \varphi | \langle A \rangle \varphi U \varphi \)

\( \gamma ::= \top | p_a | \neg p_a, \text{ where } a \in Agt \text{ and } p_a \in \Pi_a \)

\( \varphi ::= \gamma | \varphi \land \varphi | \varphi \lor \varphi | X \varphi | G \varphi | \varphi U \varphi \)

• Questions
  
  ▪ for a given strategy, formula, and state, does the outcome satisfy the formula? (i.e. does the strategy enforce the formula?)
  
  ▪ for a given formula and state, does there exist a strategy enforcing the formula?
  
  ▪ for a given formula and strategy, for which states does the strategy enforce the formula?
  
  ▪ for a given formula, does there exist a strategy enforcing the formula for all (or „many“) states?
Synthesis of Maximal Strategies

• Naive solution:
  ▪ generate all possible strategies for the model
  ▪ for each strategy check at which system states its outcome satisfies the formula
  ▪ check the resulting strategies for maximality

• Complexity: $O(|Act|^2 \cdot |St|^{k+1})$

• Incremental solution:
  ▪ generate only those strategies which enforce the formula
  ▪ partial strategies (perception – action)
Results?

- Unfortunately, the incremental solution has an even worse worst-time complexity $O(2^{\mid St \mid^n})$
- Hopefully, in „non-degenerate“ cases this will usually not be the case
- However, as we don‘t have an implementation, we can‘t know...