An Automaton-based Formalism for Cooperative Augmented Reality Systems

Felix G. Hamza-Lup\textsuperscript{1) }\quad Ferucio Laurenţi\textsuperscript{u}u Țiplea\textsuperscript{2) }

\textsuperscript{1) Department of Computer Science, Armstrong Atlantic State University, Savannah, Georgia 31419-1997, USA}
\textsuperscript{2) Department of Computer Science, “Al.I.Cuza” University, Iasi, Romania}

NCMA 2009
Outline

1. (Cooperative) Augmented Reality Systems
2. Modeling CARSs
3. Basic Properties of CARSs
4. Conclusions
Augmented Reality Systems

- The basic idea of augmented reality is to superimpose graphics, audio and other sense enhancements over a real-world environment in real-time, and to change them to accommodate a user’s head- and eye-movements, so that the graphics always fit the perspective;

- Augmented reality is still in an early stage of research and development at various universities and high-tech companies;

- Three basic components needed to make an augmented-reality system work:
  - head-mounted display;
  - tracking system;
  - mobile computing power.
There are hundreds of potential applications for augmented reality, such as:

- medicine;
- maintenance and construction;
- military;
- gaming;
- instant information

More details about AR systems and related projects:

http://cs.armstrong.edu/felix/
Example of Cooperative Augmented Reality System:

**Endo-tracheal Intubation (ETI)**

*Figure*: Instructors visualizing the 3D models relative position (left), while a remote student performs the ETI procedure (right)
Example of Cooperative Augmented Reality System: Remote Telerobotic Manipulation

*Figure*: Multi-Modal Interaction System
An Automaton-based Formalism for CARSs

F.L. Tiplea

Augmented Reality Systems
Modeling CARSs
Basic Properties of CARSs
Conclusions

Modeling CARSs

- **Actors** — entities that are able to perform complex operations on a given set of variables.\[ A = \{A_1, \ldots, A_k\} \] is a given set of actors;

- **Objectives** — sequence of actions that actors are to perform in order to drive the system from its initial state \( \gamma_0 \) to some final state \( \gamma_f \).
  
  An o-state (observation state) is a valuation \( \gamma \) of a given set \( V = \{x_1, \ldots, x_m\} \) of (typed) variables.
  
  \( \Gamma \) is the set of all o-states.

- **Environments and actions** —
  
  \( read_A : Q \rightarrow T \) is the read-time function of \( A \);

  \( write_A : Q \rightarrow T \) is the write-time function of \( A \);
Modeling CARSs

- **Modeling actors** — An actor is a 4-tuple \( A = (Q, \Sigma, \delta, q_0) \), where
  \[ \delta : Q \times \Sigma \rightarrow \mathcal{P}(Q \times \Sigma) \]
  (\( \delta \) may be a partial function).
  Infinite input sets are allowed;

- **Time constraints** — A time-constraint is any function
  \[ C : \Gamma \rightarrow T \cup \{\infty\} \]
  which gives the maximum delay permitted to the actors to trigger their actions in a state \( \gamma \).
  \( C(\gamma) = \infty \) means that no time-constraint is imposed.
Cooperative system (CS) — is a 5-tuple

\[ S = (\mathcal{V}, \mathcal{A}, \text{read}_\mathcal{A}, \text{write}_\mathcal{A}, C) \]

Computation — transition relation between configurations

\[(t, q^1_1, \ldots, q^k_1, \gamma) \vdash (t', q^1_2, \ldots, q^k_2, \gamma')\]

iff there exists an \( i \) such that:

- \( \text{read}_{\mathcal{A}_i}(q^i_1) \leq C(\gamma) \) (i.e., \( A_i \) satisfies the time-constraint \( C(\gamma) \));

- \( A_i \) performs an action, i.e.
  - \( \delta_i(q^i_1, \gamma) = (q^i_2, \gamma') \);
  - \( t' = t + \text{read}_{\mathcal{A}_i}(q^i_1) + \text{write}_{\mathcal{A}_i}(q^i_1) \);

- \( q^j_2 = q^j_1 \), for all \( j \neq i \) (i.e., the other actors do not perform any action).
Objectives again — variations of the reachability problem:

**Reachability Problem**
- Instance: cooperative system $S$, initial o-state $\gamma_0$, and final o-state $\gamma_f$;
- Question: is $\gamma_f$ reachable from $\gamma_0$?

**P-Reachability Problem**
- Instance: cooperative system $S$, initial o-state $\gamma_0$, final o-state $\gamma_f$, and predicate $P$ over $\Gamma$;
- Question: is $\gamma_f$ $P$-reachable from $\gamma_0$?

**Time-reachability Problem**
- Instance: cooperative system $S$, initial o-state $\gamma_0$, final o-state $\gamma_f$, predicate $P$ over $\Gamma$, and time value $t$;
- Question: is $\gamma_f$ $P$-reachable from $\gamma_0$ in time $t' \leq t$?
An Automaton-based Formalism for CARSs

F.L. Tiplea

Augmented Reality Systems

Modeling CARSs

Basic Properties of CARSs

Conclusions

Modeling CARSs

Example of an actor in ETI:

![Diagram of an automaton for modeling CARSs](image)

\[ read_{A_2}(q) = write_{A_2}(q) = \tau, \forall q \]

- \( q_0 \): br and hr normal
- \( q_1 \): br at least once modified
- \( q_2 \): br and hr at least once modified
- \( q_3 \): hr at least once modified

Figure: Actor \( A_2 \)
From PN to CS:

\[ M[t] M' \iff (0, q_0, \ldots, q_0, \gamma_M) A_t \vdash (0, q_0, \ldots, q_0, \gamma_{M'}) \]

*Figure:* a) A transition \( t \); b) The actor \( A_t \)
From CS to PN:

- A cooperative system $S$ is called **monotonic** if:
  - for any variable $x$, its domain is $\mathbb{N}$;
  - for any actor $A$ and any transition $(q', \gamma') \in \delta(q, \gamma)$ of $A$, the following property holds true
    $$(q', \bar{\gamma} + (\gamma' - \gamma)) \in \delta(q, \bar{\gamma}),$$
    for any $\bar{\gamma} \geq \gamma$ (the inequality between functions is component-wise defined).

- A monotonic cooperative system $S$ is called **locally finite** if for any actor $A$ and any states $q$ and $q'$ of $A$, there exists a finite set of vectors with integer components, $\{V_1, \ldots, V_p\}$, such that for any transition $(q', \gamma') \in \delta(q, \gamma)$ of $A$ there exists $i$ with $\gamma' - \gamma = V_i$. 

Theorem 1

For any monotonic and locally finite cooperative system $S$ without time-constraints, there exists a Petri net $\Sigma$ such that for any configurations $c$ and $c'$ of $S$ there are two markings $M_c$ and $M_{c'}$ and a transition $t_{c,c'}$ satisfying

$$c \vdash c' \iff M_c[t_{c,c'}]M_{c'}.$$
From CS to PN (example):

**Figure:** a) A CS with only one actor A; b) A Petri net associated to the CS in a)
Theorem 2
The reachability problem for cooperative systems is undecidable.

Proof.
The halting problem for counter machines can be reduced to the reachability problem for CS.
Reachability Problem for Cooperative Systems

**Theorem 3**

The polynomial time-reachability problem for finite-domain cooperative systems is NP-complete.

**Proof.**

Membership to NP \((S, \gamma_0, \gamma_f, \text{predicate } P \text{ verifiable in polynomial time, and time value } t \text{ of polynomial size (w.r.t. } |S|)|):

- guess a sequence of transitions of length at most \(t\) such that the first one rewrites \(\gamma_0\) and the last one ends up with \(\gamma_f\);
- if the sequence induces a computation
  - then if each configuration in computation verifies \(P\)
    - then “yes” else “no”;

NP-hardness: reduction from the Hamiltonian circuit problem.
This work proposes an automaton-based formalism for CARSs;

Future work:
- in-depth study of the basic properties of the model;
- verification techniques (based on automata theory (reachability-based techniques, model checking etc.));
- accommodate delays in the formalism.