Strategy RV: A Tool to Approximate ATL Model Checking under Imperfect Information and Perfect Recall

Angelo Ferrando[1] and Vadim Malvone[2]

1 University of Genova, Italy

2 Télécom Paris, France

UNIVERSITÀ DEGLI STUDI DI GENOVA

TELECOM Paris

IP PARIS

System Correctness

- A very important problem in critical systems:
  - Safety: errors cost lives (e.g. Therac-25).
  - Mission: errors cost in terms of objectives (e.g. Arianne 5).
  - Business: failure cost in loss of money (e.g. Denver airport).
- In such systems failure is not an option.

The Model Checking Approach to System Verification

- Model system S as a transition system M_S
- Specify property P as formula φ_P
- Check that $M_S \models \varphi_P$

Problem: for Multi-agent Systems the model checking problem is undecidable in many cases of interest

Multi-Agent Systems (MAS): Key aspects

- There are many agents (players) interacting among them.
- Each agent has a set of *strategies*.
- A *strategy* is a conditional plan that at each step of the game prescribes an action.
- The composition of strategies, one for each player, induces an unique computation.

The Role Played by Memory and Information

Depending on the memory, we distinguish between:

- *imperfect recall strategies (IR)* $\implies \sigma : St \to Act$;
- *perfect recall strategies (PR)* $\implies \sigma : St^+ \to Act$.

Depending on the players' information, we distinguish between:

- *perfect information games (PI)*;
- *imperfect information games (II)*.

Specification: Alternating-time temporal logic

State (φ) and path (ψ) formulas in ATL^* are:

$$\varphi ::= q \mid \neg\varphi \mid \varphi \wedge \varphi \mid < \Gamma > \psi$$
$$\psi ::= \varphi \mid \neg\psi \mid \psi \wedge \psi \mid X\psi \mid (\psi U \psi)$$

The strategy operator $< \Gamma >$ is read as "the agents in coalition Γ have a strategy to achieve ..."

Problem: Undecidability Imperfect information

	perfect information	imperfect information
imperfect recall	*PSPACE*-complete	*PSPACE*-complete
perfect recall	2*EXPTIME*-complete	undecidable

Our contribution

A tool to approximate the verification of Alternating-time Temporal Logic (ATL) under imperfect information and perfect recall, which is known to be undecidable, by using Runtime Verification.

High level idea

Given a model M and a formula φ in ATL^*, we need:

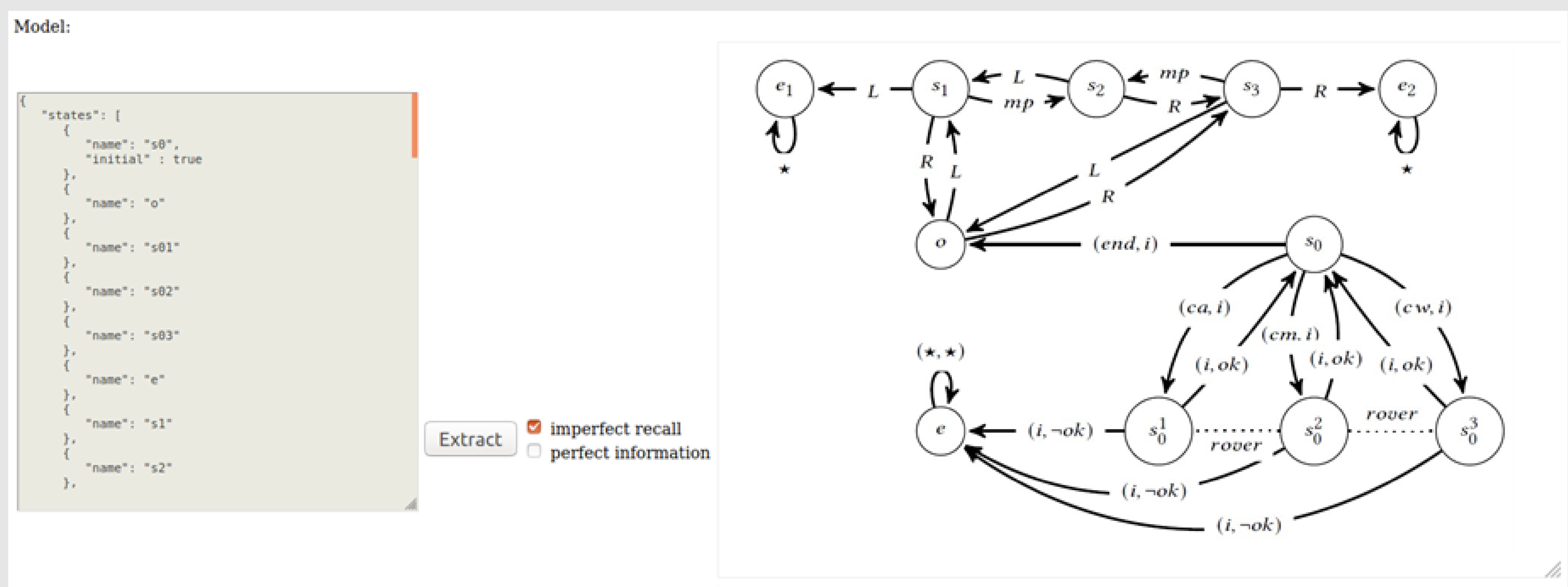
1. to find the sub-models of M in which there is perfect information (resp., imperfect recall strategies) and a sub-formula φ' of φ is satisfied;
2. to use monitors to check whether the temporal remaining part ψ of φ can be satisfied and a sub-model M' identified by (1) can be reached.

Our verification procedure

Data: a Model M, a property φ, and a variable *choice*
Result: the verification result

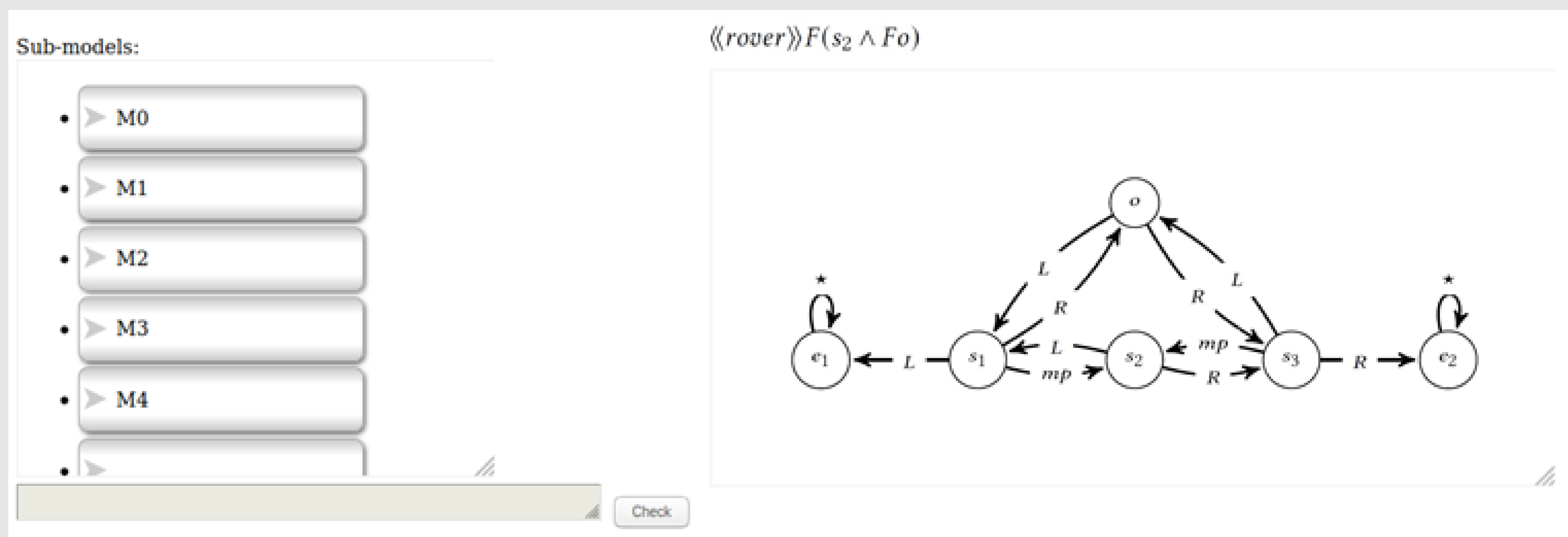
$c_{IR} = \{\}$;
$c_{ir} = \{\}$;
if $choice = 0$ **then**
  $c_{IR} =$ **FindSubModelsWithPerfectInfo**(M, φ) ;
end
else if $choice = 1$ **then**
  $c_{ir} =$ **FindSubModelsWithImperfectRecall**(M, φ) ;
end
else
  $c_{IR} =$ **FindSubModelsWithPerfectInfo**(M, φ) ;
  $c_{ir} =$ **FindSubModelsWithImperfectRecall**(M, φ) ;
end
return **GenerateAndRunMonitors**$(M, \varphi, c_{IR} \cup c_{ir})$;

Parsing of the input model



- The user inputs the Json model M and formula φ (left);
- The tool shows the graphical representation of M (right).

Extraction, visualisation, and RV of sub-models



- Each sub-model is translated into its equivalent ISPL (Interpreted Systems Programming Language) program, and verified by MCMAS;
- The list of sub-models (M_1, M_2, ...) satisfying a sub-formula φ' of φ is shown to the user (top left);
- By clicking a sub-model, its visualisation, along with the verified sub-formula φ', are displayed (right);
- Finally, an execution trace can be reported by the user, and checked by a monitor on M using the selected sub-model and the remaining part ψ of φ (bottom left).

Conclusions and future works

- We presented Strategy RV, a tool that, first extracts sub-models with perfect information and/or imperfect recall that satisfy a strategic objective; and then, it uses runtime verification to check the remaining temporal objectives and to reach one of the sub-models so generated.
- In future work we intend to improve the sub-models extraction and monitors generation.
- We plan to extend the approximation and monitoring techniques to more expressive languages for strategic reasoning.

contacts: angelo.ferrando@unige.it, vadim.malvone@telecom-paris.fr