Reasoning about Knowledge and Strategies

Bastien Maubert and Aniello Murano

Università degli Studi di Napoli
Federico II

Program synthesis

Basic idea:

"Program synthesis is the task to automatically construct a program that satisfies a given high-level specification."

We are interested in programs that:

- Interact with an environment
- May run forever

Example: operating systems, controllers in power plants. . .

Specification language: LTL

Propositional logic +

- $\mathbf{X}\varphi$: "φ holds at next step"
- $\varphi\mathbf{U}\psi$: "φ will hold until ψ holds"
- $\mathbf{G}\varphi$: "φ always holds"
- $\mathbf{F}\varphi$: "φ eventually holds"

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

i_0

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

i_0

o_0

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{ll} i_0 & i_1 \\ o_0 & \end{array}$$

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{cc} i_0 & i_1 \\ o_0 & o_1 \end{array}$$

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{ccc} i_0 & i_1 & i_2 \\ o_0 & o_1 & \end{array}$$

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{ccc} i_0 & i_1 & i_2 \\ o_0 & o_1 & o_2 \end{array}$$

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{cccc} i_0 & i_1 & i_2 & \dots \\ o_0 & o_1 & o_2 & \dots \end{array}$$

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{cccc} i_0 & i_1 & i_2 & \dots \\ o_0 & o_1 & o_2 & \dots \end{array}$$

LTL synthesis problem

Given a specification $\varphi \in$ LTL over $I \cup O$, synthesize a strategy $\sigma : (2^I)^* \to 2^O$ such that all resulting behaviours satisfy φ.

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{cccc} i_0 & i_1 & i_2 & \dots \\ o_0 & o_1 & o_2 & \dots \end{array}$$

LTL synthesis problem

Given a specification $\varphi \in$ LTL over $I \cup O$, synthesize a strategy $\sigma : (2^I)^* \to 2^O$ such that all resulting behaviours satisfy φ.

Synthesize a finite representation of this infinite object.

LTL synthesis (Pnueli and Rosner, 1989)

- I: input variables,
- O: output variables

Game between system and environment

- Environment chooses valuations for I
- System chooses valuations for O

$$\begin{array}{cccc} i_0 & i_1 & i_2 & \dots \\ o_0 & o_1 & o_2 & \dots \end{array}$$

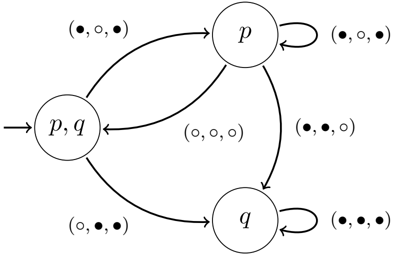
LTL synthesis problem

Given a specification $\varphi \in \mathrm{LTL}$ over $I \cup O$, synthesize a strategy $\sigma : (2^I)^* \to 2^O$ such that all resulting behaviours satisfy φ.

Synthesize a finite representation of this infinite object.

What about synthesis of distributed systems?

Distributed synthesis



p, q are *atomic propositions*

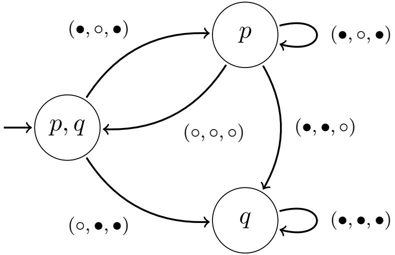
$\circ, \bullet$ are *actions*

strategies σ : Histories $\to$ Actions

Input: A concurrent game structure and a formula $\varphi \in$ LTL

Output: A distributed strategy to enforce φ

Distributed synthesis



p, q are *atomic propositions*

$\circ, \bullet$ are *actions*

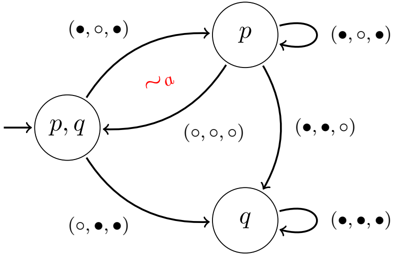
strategies σ : Histories $\to$ Actions

indistinguishability relations $\sim_a$

Input: A concurrent game structure and a formula $\varphi \in$ LTL

Output: A distributed strategy to enforce φ

Distributed synthesis



p, q are *atomic propositions*

$\circ, \bullet$ are *actions*

strategies σ : Histories $\to$ Actions

indistinguishability relations $\sim_a$

Input: A concurrent game structure and a formula $\varphi \in$ LTL

Output: A distributed strategy to enforce φ

Imperfect information

1. Strategies must be consistent with players' information

Constraint on strategies:

If $h \sim_a h'$, then $\sigma_a(h) = \sigma_a(h')$.

2. Makes epistemic reasoning meaningful and useful

Example: opacity

A system is *opaque* for property P if a spy never knows whether the current execution is in P.

Classic definition:

$$\forall h, \exists h' \text{ s.t. } h \sim_{\mathsf{spy}} h' \text{ and } h' \notin P$$

With epistemic temporal logic:

$$\mathbf{G}\neg K_{\mathsf{spy}} P$$

Imperfect information

1. Strategies must be consistent with players' information

Constraint on strategies:

If $h \sim_a h'$, then $\sigma_a(h) = \sigma_a(h')$.

2. Makes epistemic reasoning meaningful and useful

Example: opacity

A system is *opaque* for property P if a spy never knows whether the current execution is in P.

Classic definition:

$$\forall h, \exists h' \text{ s.t. } h \sim_{\mathsf{spy}} h' \text{ and } h' \notin P$$

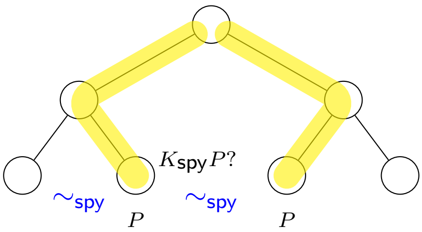
With epistemic temporal logic:

$$\exists \sigma (c, \sigma) \mathbf{G} \neg K_{\mathsf{spy}} P$$

Semantics of knowledge when reasoning about strategies

Yellow subtree: controller's strategy

$\sim_{\text{spy}}$: spy's indistinguishability relation



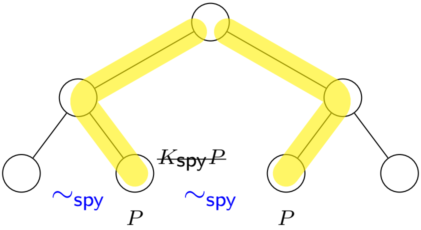
Two possible semantics:

- spy ignores controller's strategy
  $\rightarrow K_{\text{spy}} P$ does not hold
- spy knows controller's strategy
  $\rightarrow K_{\text{spy}} P$ holds

Semantics of knowledge when reasoning about strategies

Yellow subtree: controller's strategy

$\sim_{\mathsf{spy}}$: spy's indistinguishability relation



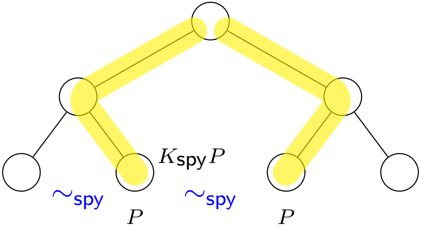
Two possible semantics:

- spy ignores controller's strategy
  $\rightarrow K_{\mathsf{spy}} P$ does not hold
- spy knows controller's strategy
  $\rightarrow K_{\mathsf{spy}} P$ holds

Semantics of knowledge when reasoning about strategies

Yellow subtree: controller's strategy

$\sim_{\text{spy}}$: spy's indistinguishability relation



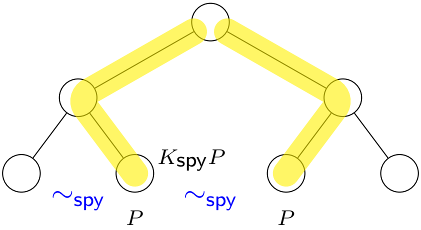
Two possible semantics:

- spy ignores controller's strategy
  $\rightarrow K_{\text{spy}}P$ does not hold
- spy knows controller's strategy
  $\rightarrow K_{\text{spy}}P$ holds

Semantics of knowledge when reasoning about strategies

Yellow subtree: controller's strategy

$\sim_{\textsf{spy}}$: spy's indistinguishability relation



Two possible semantics:

- spy ignores controller's strategy — Uninformed semantics
  $\rightarrow K_{\textsf{spy}}P$ does not hold
- spy knows controller's strategy — Informed semantics
  $\rightarrow K_{\textsf{spy}}P$ holds

In the litterature

Both semantics have been used, but implicitely.

Informed semantics:

Distributed synthesis from epistemic temporal specifications

- van der Meyden and Vardi, 1998
- van der Meyden and Wilke, 2005

Uninformed semantics:

All epistemic extensions of ATL and SL (that we know of)

One paper talks about this issue: Puchala, 2010

In the litterature

Both semantics have been used, but implicitely.

Informed semantics:

Distributed synthesis from epistemic temporal specifications

- van der Meyden and Vardi, 1998
- van der Meyden and Wilke, 2005

Uninformed semantics:

All epistemic extensions of ATL and SL (that we know of)

One paper talks about this issue: Puchala, 2010

What is known about distributed synthesis?

Peterson and Reif (1979), Pnueli and Rosner (1990)

Distributed synthesis for reachability objective is undecidable.

Two known ways of retrieving decidability for temporal objectives:

1. Public actions
2. Hierarchical information

For epistemic temporal objectives and

- informed semantics:
  - decidable for public actions
  - undecidable for hierarchical information

  [van der Meyden and Wilke, 2005]
- uninformed semantics:
  - decidable for public actions [Belardinelli et al., 2017]
  - decidable for hierarchical information [Puchala, 2010]

Overview

Peterson and Reif (1979), Pnueli and Rosner (1990)

Distributed synthesis for reachability objective is undecidable.

Two known ways of retrieving decidability for temporal objectives:

1. Public actions
2. Hierarchical information

For epistemic temporal objectives and

- informed semantics:
  - decidable for public actions
  - undecidable for hierarchical information

  [van der Meyden and Wilke, 2005]
- uninformed semantics:
  - decidable for public actions [Belardinelli et al., 2017]
  - decidable for hierarchical information [Puchala, 2010]

SL with imperfect information and knowledge

SL (Chatterjee et al. 2010, Mogavero et al. 2014)

LTL +

- $\exists \sigma\, \varphi$

  "there exists a strategy σ s.t. φ"

- $(a, \sigma)\varphi$

  "when player a plays strategy σ, φ"

SL with imperfect information and knowledge

SL$_{\mathrm{ii}}$ (Berthon et al. 2017)

LTL $+$

- $\exists^{o}\sigma\,\varphi$
  "there exists a strategy σ with observational power o s.t. φ"
- $(a,\sigma)\varphi$
  "when player a plays strategy σ, φ"

SL with imperfect information and knowledge

ESL (M. and Murano, 2018)

LTL +

- $\exists^{o}\sigma\,\varphi$
  "there exists a strategy σ with observational power o s.t. φ"
- $(a, \sigma)\varphi$
  "when player a plays strategy σ, φ"
- $K_a\varphi$
  "player a knows that φ"
- $\mathbf{A}\varphi$
  "in all outcomes, φ"

What can ESL express?

- Distributed synthesis:

$$\exists^{o_1} x_1 \, \dots \exists^{o_n} x_n \, (a_1, x_1) \dots (a_n, x_n) \forall^{o_e} y \, (e, y) \, \psi$$

- Existence of Nash equilibria:

$$\exists^{o_a} x \, \exists^{o_b} y \, \exists^{o_c} z \, (a, x)(b, y)(c, z) \bigwedge_{d \in \{a,b,c\}} \exists^{o_d} x' \, (d, x') \, \mathsf{Win}_d \to \mathsf{Win}_d$$

- Players changing observation:

$$\exists^{o_1} x_1 \, (a, x_1) \mathbf{AF} \exists^{o_2} x_2 \, (a, x_2) \mathbf{AF} \, \mathsf{Win}_a$$

"First I find my glasses, then I play for real."

Main result

Hierarchical instances

An ESL formula Φ is *hierarchical* if:

- innermost strategies observe better than outermost ones
- epistemic subformulas do not talk about current strategies

Considering the uninformed semantics of knowledge:

Theorem

Model-checking hierarchical instances of ESL is decidable.

Corollaries:

On systems with hierarchical information,
for epistemic temporal specifications with uninformed semantics,

We can solve

- distributed synthesis,
- module checking,
- synthesis of Nash equilibria,
- rational synthesis,
- ...

Interested? Come to Napoli!

Rational distributed synthesis

Rational synthesis

Fisman et al. (2010), Condurache et al. (2016), Kupferman et al. (2016)

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i
- System made of one component a
- LTL specification ψ_g

$$\Phi_{\text{c-RS}} := \exists x\, \exists y_1 \ldots \exists y_m\, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y})\, \varphi_{\mathsf{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\text{nc-RS}} := \exists x\, \forall y_1 \ldots \forall y_m\, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y})\, \varphi_{\mathsf{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\mathsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i'\, (e_i, y_i')\, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i
- System made of one component a
- LTL specification ψ_g

$$\Phi_{\textsf{c-RS}} := \exists x \, \exists y_1 \, \ldots \exists y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\textsf{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\textsf{nc-RS}} := \exists x \, \forall y_1 \, \ldots \forall y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\textsf{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\textsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of one component a
- LTL specification ψ_g

$$\Phi_{\textsf{c-RS}} := \exists x \, \exists y_1 \, \ldots \exists y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\textsf{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\textsf{nc-RS}} := \exists x \, \forall y_1 \, \ldots \forall y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\textsf{NE}} \rightarrow \mathbf{A}\psi_g$$

where

$$\varphi_{\textsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \rightarrow \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g

$$\Phi_{\textsf{c-RS}} := \exists x\, \exists y_1\, \ldots \exists y_m\, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y})\, \varphi_{\textsf{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\textsf{nc-RS}} := \exists x\, \forall y_1\, \ldots \forall y_m\, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y})\, \varphi_{\textsf{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\textsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i'\, (e_i, y_i')\, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists x \, \exists y_1 \, \ldots \exists y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\text{nc-RS}} := \exists x \, \forall y_1 \, \ldots \forall y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\text{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists x \, \exists^{o_1^e} y_1 \; \ldots \exists^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\text{nc-RS}} := \exists x \, \forall y_1 \; \ldots \forall y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\text{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists^{o_1} x_1 \ldots \exists^{o_n} x_n \, \exists^{o_1^e} y_1 \ldots \exists^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\mathsf{NE}} \wedge \mathbf{A}\psi_g$$

$$\Phi_{\text{nc-RS}} := \exists x \, \forall y_1 \ldots \forall y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\mathsf{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\mathsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists^{o_1} x_1 \ldots \exists^{o_n} x_n \, \exists^{o_1^e} y_1 \ldots \exists^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \wedge \mathbf{A}\psi_g$$

$$\Phi_{\text{nc-RS}} := \exists x \, \forall^{o_1^e} y_1 \ldots \forall^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \to \mathbf{A}\psi_g$$

where

$$\varphi_{\text{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \to \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists^{o_1} x_1 \ \ldots \exists^{o_n} x_n \, \exists^{o_1^e} y_1 \ \ldots \exists^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\text{nc-RS}} := \exists^{o_1} x_1 \ \ldots \exists^{o_n} x_n \, \forall^{o_1^e} y_1 \ \ldots \forall^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\text{NE}} \rightarrow \mathbf{A}\psi_g$$

where

$$\varphi_{\text{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \rightarrow \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists^{o_1} x_1 \ \ldots \exists^{o_n} x_n \, \exists^{o_1^e} y_1 \ \ldots \exists^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\mathsf{NE}} \wedge \mathbf{A}\psi_g$$

$$\Phi_{\text{nc-RS}} := \exists^{o_1} x_1 \ \ldots \exists^{o_n} x_n \, \forall^{o_1^e} y_1 \ \ldots \forall^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\mathsf{NE}} \rightarrow \mathbf{A}\psi_g$$

where

$$\varphi_{\mathsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists^{o_i^e} y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \rightarrow \mathbf{A}\psi_i \right]$$

Rational distributed synthesis

Rational distributed synthesis

- Environment made of several components $\{e_1, \ldots, e_m\}$
- individual LTL goals ψ_i and observations o_i^e
- System made of several components $\{a_1, \ldots, a_n\}$
- LTL specification ψ_g and observations o_i

$$\Phi_{\text{c-RS}} := \exists^{o_1} x_1 \ \ldots \exists^{o_n} x_n \, \exists^{o_1^e} y_1 \ \ldots \exists^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\mathsf{NE}} \wedge \mathbf{A}\psi_g$$
$$\Phi_{\text{nc-RS}} := \exists^{o_1} x_1 \ \ldots \exists^{o_n} x_n \, \forall^{o_1^e} y_1 \ \ldots \forall^{o_1^e} y_m \, (\boldsymbol{a}, \boldsymbol{x})(\boldsymbol{e}, \boldsymbol{y}) \, \varphi_{\mathsf{NE}} \rightarrow \mathbf{A}\psi_g$$

where

$$\varphi_{\mathsf{NE}} := \bigwedge_{i \in [m]} \left[\left(\exists^{o_p} y_i' \, (e_i, y_i') \, \mathbf{A}\psi_i \right) \rightarrow \mathbf{A}\psi_i \right]$$