Reconciling Rationality and Stochasticity: Rich Behavioral Models in Two-Player Games

Mickael Randour

Computer Science Department, ULB - Université libre de Bruxelles, Belgium

July 24, 2016

GAMES 2016 - 5th World Congress of the Game Theory Society

The talk in one slide

Two traditional paradigms for agents
in complex systems

Fully rational ⇓ System = (multi-player) game

Fully stochastic ⇓ System = large stochastic process

↓

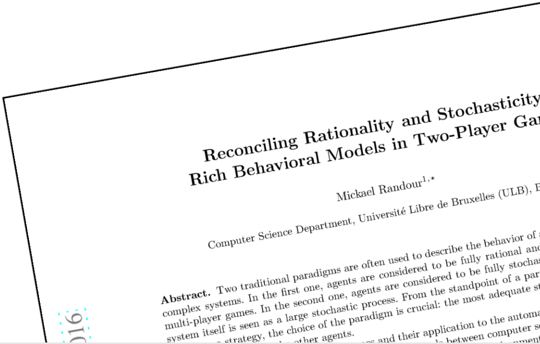
In some fields (e.g., computer science), need
to go beyond: **rich behavioral models**

⇓

Illustration: planning a journey
in an uncertain environment

Advertisement

Full paper available on arXiv [Ran16a]: abs/1603.05072



Rationality hypothesis

Rational agents [OR94]:

- clear personal objectives,
- aware of their alternatives,
- form sound expectations about any unknowns,
- choose their actions coherently (i.e., regarding some notion of optimality).

$\Longrightarrow$ In the particular setting of *zero-sum* games: antagonistic interactions between the players.

$\hookrightarrow$ **Well-founded abstraction in computer science.**
E.g., processes competing for access to a shared resource.

Stochasticity

Stochastic agents:

- often a *sufficient abstraction* to reason about macroscopic properties of a complex system,
- agents follow stochastic models that can be based on experimental data (e.g., traffic in a town).

Several models of interest:

- fully stochastic agents $\implies$ Markov chain [Put94],
- rational agent against stochastic agent $\implies$ **Markov decision process** [Put94],
- two rational agents + one stochastic agent $\implies$ stochastic game or competitive MDP [FV97].

Choosing the appropriate paradigm matters!

As an agent having to choose a strategy, **the assumptions made on the other agents are crucial**.

$\Longrightarrow$ **They define our objective hence the adequate strategy.**

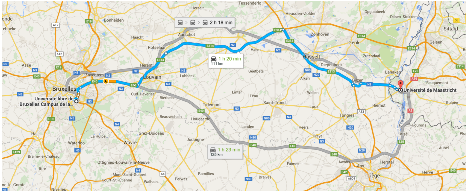
$\Longrightarrow$ **Illustration: planning a journey.**

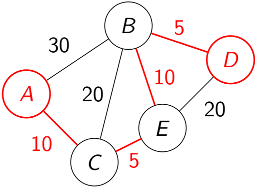
Aim of this illustration

Flavor of ≠ types of **useful strategies** in stochastic environments.

- ▷ Based on a series of papers, most in a computer science setting (more on that later) [Ran13, BFRR14b, BFRR14a, RRS15a, RRS15b, BCH+16].

Applications to the **shortest path problem**.





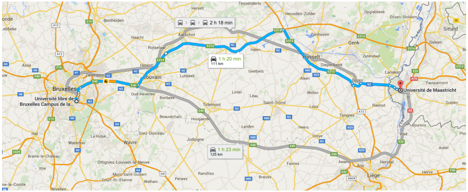
↪ Find a **path of minimal length** in a weighted graph (Dijkstra, Bellman-Ford, etc) [CGR96].

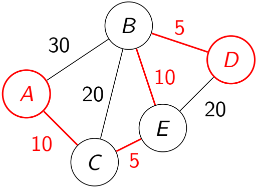
Aim of this illustration

Flavor of ≠ types of **useful strategies** in stochastic environments.

▷ Based on a series of papers, most in a computer science setting (more on that later) [Ran13, BFRR14b, BFRR14a, RRS15a, RRS15b, BCH+16].

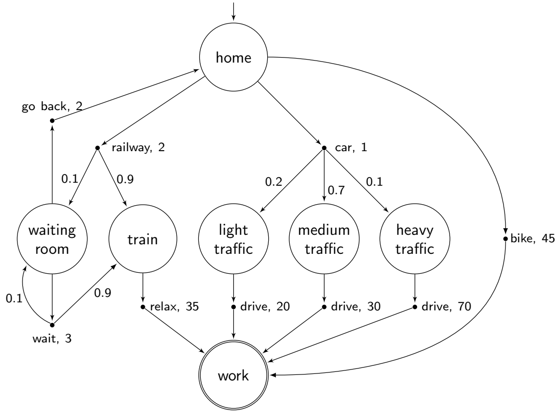
Applications to the **shortest path problem**.





What if the environment is **uncertain**? E.g., in case of heavy traffic, some roads may be crowded.

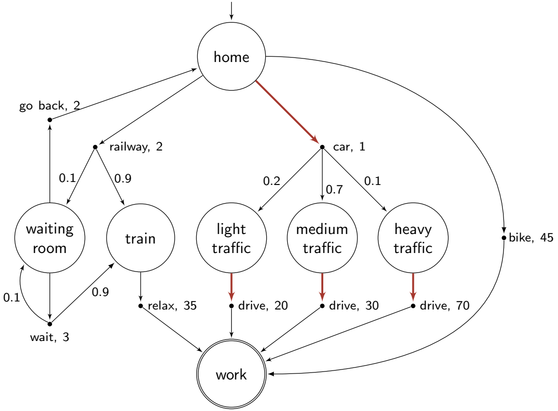
Planning a journey in an uncertain environment



Each action takes time, target = work.

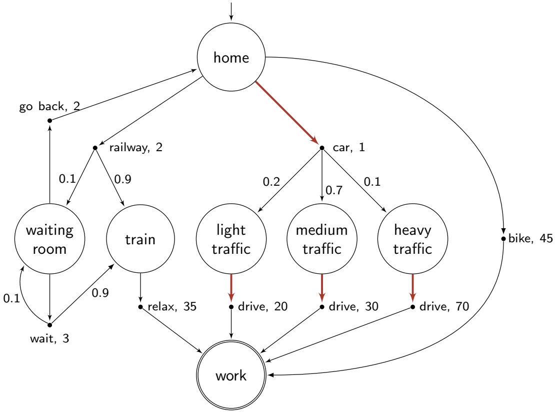
▷ What kind of strategies are we looking for when the environment is **stochastic** (MDP)?

Solution 1: minimize the *expected* time to work



- ▷ "Average" performance: meaningful when you journey often.
- ▷ **Simple strategies** suffice: no memory, no randomness.
- ▷ Taking the **car** is optimal: $\mathbb{E}_D^{\sigma}(\mathsf{TS}^{\mathsf{work}}) = 33$.

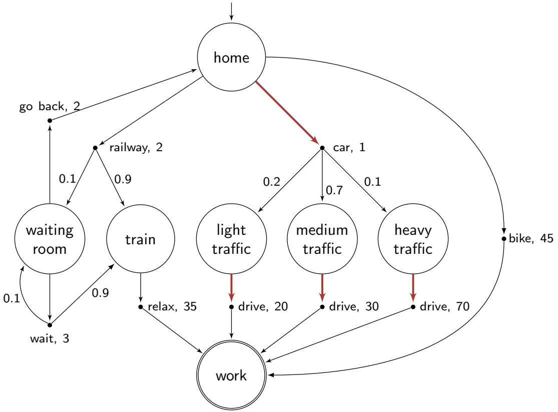
Solution 2: traveling without taking too many risks



Minimizing the *expected time* to destination makes sense **if** we travel often and **it is not a problem to be late**.

With car, in 10% of the cases, the journey takes 71 minutes.

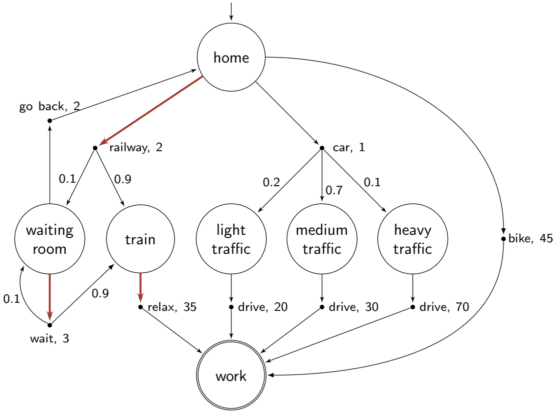
Solution 2: traveling without taking too many risks



Most bosses will not be happy if we are late too often. . .

⇝ what if we are risk-averse and want to avoid that?

Solution 2: maximize the *probability* to be on time

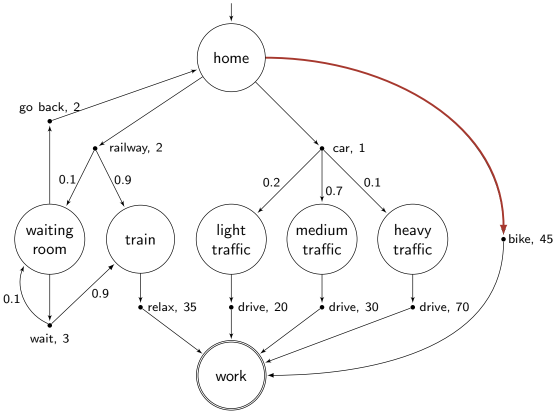


Specification: reach work within 40 minutes with 0.95 probability

Sample strategy: take the **train** $\rightsquigarrow \mathbb{P}^{\sigma}_{D}\left[\mathsf{TS}^{\mathsf{work}} \leq 40\right] = 0.99$

Bad choices: car (0.9) and bike (0.0)

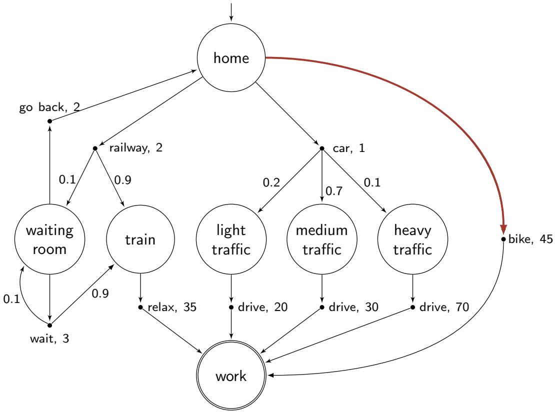
Solution 3: strict worst-case guarantees



Specification: *guarantee* that work is reached within 60 minutes (to avoid missing an important meeting)

Sample strategy: **bike** $\rightsquigarrow$ worst-case reaching time $= 45$ minutes.
Bad choices: train ($wc = \infty$) and car ($wc = 71$)

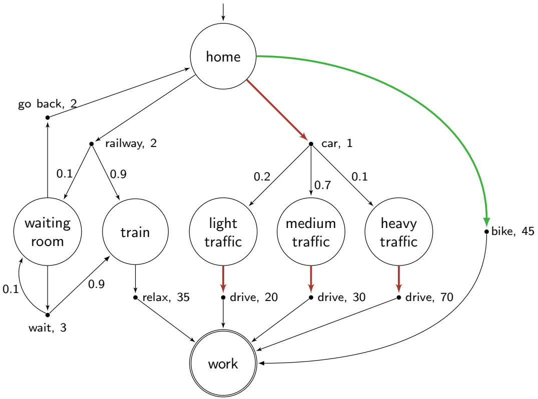
Solution 3: strict worst-case guarantees



Worst-case analysis ⇝ **two-player zero-sum game** against a rational antagonistic adversary (*bad guy*)

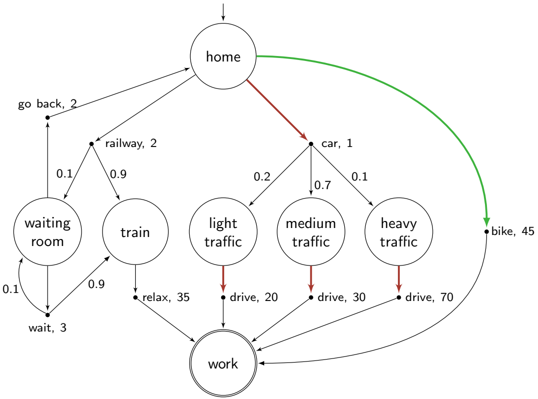
▷ forget about probabilities and give the choice of transitions to the adversary

Solution 4: minimize the *expected* time under strict worst-case guarantees



- Expected time: car $\rightsquigarrow \mathbb{E} = 33$ but $wc = 71 > 60$
- Worst-case: bike $\rightsquigarrow wc = 45 < 60$ but $\mathbb{E} = 45 >>> 33$

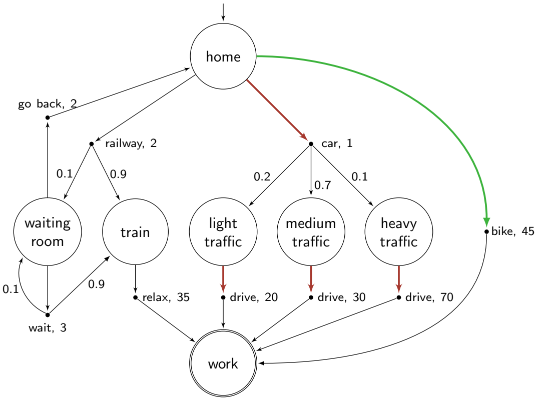
Solution 4: minimize the *expected* time under strict worst-case guarantees



In practice, we want both! Can we do better?

▷ **Beyond worst-case synthesis** [BFRR14b, BFRR14a]: minimize the expected time under the worst-case constraint.

Solution 4: minimize the *expected* time under strict worst-case guarantees

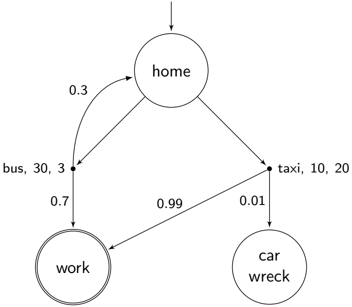


Sample strategy: try train up to 3 delays then switch to bike.

$\rightsquigarrow$ $wc = 58 < 60$ and $\mathbb{E} \approx 37.34 << 45$

$\rightsquigarrow$ Strategies need **memory** $\rightsquigarrow$ more complex!

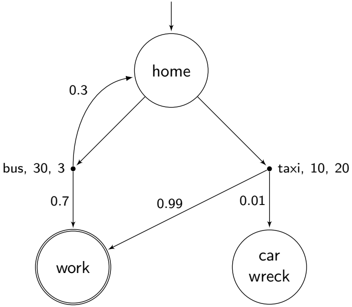
Solution 5: multiple objectives ⇒ trade-offs



Two-dimensional weights on actions: *time* and *cost*.

Often necessary to consider trade-offs: e.g., between the probability to reach work in due time and the risks of an expensive journey.

Solution 5: multiple objectives ⇒ trade-offs

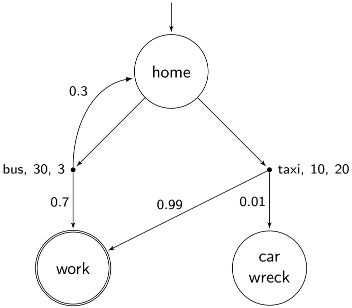


Solution 2 (probability) can only ensure a **single constraint**.

- **C1**: 80% of runs reach work in at most 40 minutes.
  - ▷ Taxi $\rightsquigarrow \leq 10$ minutes with probability $0.99 > 0.8$.
- **C2**: 50% of them cost at most 10$ to reach work.
  - ▷ Bus $\rightsquigarrow \geq 70\%$ of the runs reach work for 3$.

Taxi $\not\models$ C2, bus $\not\models$ C1. What if we want C1 $\wedge$ C2?

Solution 5: multiple objectives ⇒ trade-offs

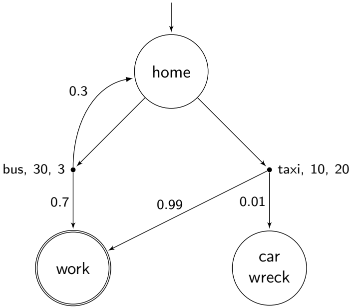


- **C1**: 80% of runs reach work in at most 40 minutes.
- **C2**: 50% of them cost at most 10$ to reach work.

Study of **multi-constraint percentile queries** [RRS15a].

- ▷ Sample strategy: bus once, then taxi. Requires *memory*.
- ▷ Another strategy: bus with probability 3/5, taxi with probability 2/5. Requires *randomness*.

Solution 5: multiple objectives ⇒ trade-offs



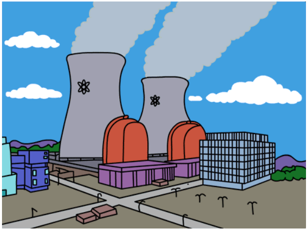
- **C1**: 80% of runs reach work in at most 40 minutes.
- **C2**: 50% of them cost at most 10$ to reach work.

Study of **multi-constraint percentile queries** [RRS15a].

In general, *both* memory *and* randomness are required.

$\neq$ previous problems $\rightsquigarrow$ more complex!

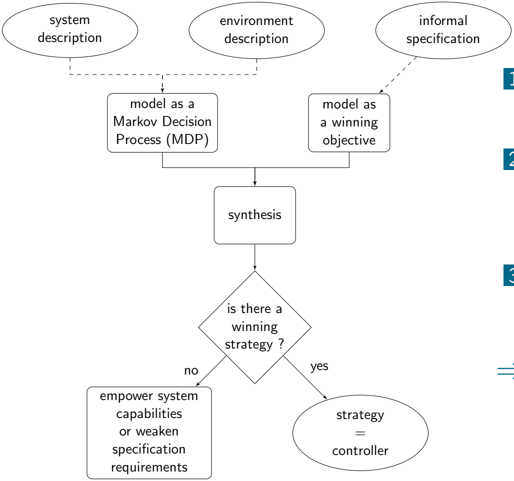
Controller synthesis



- Setting:
  - a reactive **system** to *control*,
  - an *interacting* **environment**,
  - a **specification** to *enforce*.
- For **critical** systems (e.g., airplane controller, power plants, ABS), testing is not enough!
  - ⇒ Need **formal methods**.
- **Automated synthesis** of provably-correct and efficient controllers:
  - mathematical frameworks,
    - ↪ e.g., games on graphs [GTW02, Ran13, Ran14]
  - software tools.

Strategy synthesis in stochastic environments

Strategy = formal model of how to control the system



1. How complex is it to decide if a winning strategy exists?
2. How complex such a strategy needs to be? **Simpler is better**.
3. Can we synthesize one efficiently?

⇒ Depends on the winning objective, the exact type of interaction, etc.

Some other objectives

The example was about **shortest path objectives**, but there are many more! Some examples based on energy applications.

- **Energy**: operate with a (bounded) fuel tank and never run out of fuel [BFL+08].
- **Mean-payoff**: average cost/reward (or energy consumption) per action in the long run [EM79].
- **Average-energy**: energy objective + optimize the long-run average amount of fuel in the tank [BMR+15].

Also inspired by economics:

- **Discounted sum**: simulates interest or inflation [BCF+13].

Conclusion

Our research aims at:

- defining meaningful *strategy concepts*,
- providing *algorithms* and *tools* to compute those strategies,
- classifying the *complexity* of the different problems from a theoretical standpoint.
  - ↪ Is it mathematically possible to obtain efficient algorithms?

Take-home message

Rich behavioral models are natural and important in computer science (e.g., synthesis).

Maybe they can be useful in other areas too. E.g., in economics: combining sufficient risk-avoidance and profitable expected return, value-at-risk models.

Thank you! Any question?

References I

T. Brázdil, T. Chen, V. Forejt, P. Novotný, and A. Simaitis.
Solvency Markov decision processes with interest.
In Proc. of FSTTCS, volume 24 of LIPIcs, pages 487–499. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013.

R. Brenguier, L. Clemente, P. Hunter, G.A. Pérez, M. Randour, J.-F. Raskin, O. Sankur, and M. Sassolas.
Non-zero sum games for reactive synthesis.
In Proc. of LATA, LNCS 9618, pages 3–23. Springer, 2016.

P. Bouyer, U. Fahrenberg, K.G. Larsen, N. Markey, and J. Srba.
Infinite runs in weighted timed automata with energy constraints.
In Proc. of FORMATS, LNCS 5215, pages 33–47. Springer, 2008.

V. Bruyère, E. Filiot, M. Randour, and J.-F. Raskin.
Expectations or guarantees? I want it all! A crossroad between games and MDPs.
In Proc. of SR, EPTCS 146, pages 1–8, 2014.

V. Bruyère, E. Filiot, M. Randour, and J.-F. Raskin.
Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games.
In Proc. of STACS, LIPIcs 25, pages 199–213. Schloss Dagstuhl - LZI, 2014.

P. Bouyer, N. Markey, M. Randour, K.G. Larsen, and S. Laursen.
Average-energy games.
In Proc. of GandALF, EPTCS 193, pages 1–15, 2015.

References II

B.V. Cherkassky, A.V. Goldberg, and T. Radzik.
Shortest paths algorithms: Theory and experimental evaluation.
Math. programming, 73(2):129–174, 1996.

A. Ehrenfeucht and J. Mycielski.
Positional strategies for mean payoff games.
International Journal of Game Theory, 8:109–113, 1979.

J. Filar and K. Vrieze.
Competitive Markov decision processes.
Springer, 1997.

E. Grädel, W. Thomas, and T. Wilke, editors.
Automata, Logics, and Infinite Games: A Guide to Current Research, LNCS 2500. Springer, 2002.

M.J. Osborne and A. Rubinstein.
A Course in Game Theory.
MIT Press, 1994.

M.L. Puterman.
Markov Decision Processes: Discrete Stochastic Dynamic Programming.
John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994.

M. Randour.
Automated synthesis of reliable and efficient systems through game theory: A case study.
In Proceedings of the European Conference on Complex Systems 2012, Springer Proceedings in Complexity XVII, pages 731–738. Springer, 2013.

References III

M. Randour.
Synthesis in Multi-Criteria Quantitative Games.
PhD thesis, Université de Mons, Belgium, 2014.

M. Randour.
Reconciling rationality and stochasticity: Rich behavioral models in two-player games.
CoRR, abs/1603.05072, 2016.

M. Randour.
Reconciling rationality and stochasticity: Rich behavioral models in two-player games.
Talk at GAMES 2016 - 5th World Congress of the Game Theory Society, 2016.

M. Randour, J.-F. Raskin, and O. Sankur.
Percentile queries in multi-dimensional Markov decision processes.
In Proc. of CAV, LNCS 9206, pages 123–139. Springer, 2015.

M. Randour, J.-F. Raskin, and O. Sankur.
Variations on the stochastic shortest path problem.
In Proc. of VMCAI, LNCS 8931, pages 1–18. Springer, 2015.

Algorithmic complexity: hierarchy of problems

For shortest path