LTL and Past LTL on Finite Traces for Planning and Declarative Process Mining

SAPIENZA
UNIVERSITÀ DI ROMA

Francesco Fuggitti

Master of Science in
Engineering in Computer Science
Sapienza, University of Rome

Advisor: Prof. Giuseppe De Giacomo

A.Y. 2017/2018

Outline

1. Introduction
2. Objectives
3. LTL_f2DFA

Introduction

- Classic Reinforcement Learning:
  - An *agent* interacts with an *environment* by taking *actions* so to maximize *rewards*;
  - No knowledge about the transition model, but assume Markov property (history does not matter): Markov Decision Process (MDP)
  - Solution: (Markovian) policy $\rho : S \to A$

- RL for Non-Markovian Decision Process (NMRDP):
  - Rewards depend from history, not just the last transition;
  - Specify proper behaviours by using temporal logic formulas;
  - Solution: (Non-Markovian) policy $\rho : S^* \to A$
  - Reduce the problem to MDP (with extended state space)

- In (Brafman et al. 2018) specify reward using:
  - Linear-time Temporal Logic on Finite Traces LTL_f
  - Linear-time Dynamic Logic on Finite Traces LDL_f

Objectives

- Provide an efficient technique to transform LTL$_f$/PLTL formulas into DFAs
- Provide an approach to FOND Planning for LTL$_f$/PLTL goals:
  - reduce the problem standard FOND planning
  - working with FOND domains instead of automata
- Provide a generalization of the Janus approach to declarative process mining:
  - generalization of the constraint formula

PLTL and LTL_f (De Giacomo and Vardi, 2013)

- Linear Temporal Logic on finite traces: LTL_f
  - exactly the same syntax of LTL
  - interpreted over *finite* traces
    - next: $\bigcirc$*happy*
    - eventually: $\Diamond$*rich*
    - until: *reply* $\mathcal{U}$ *acknowledge*
    - always: $\Box$*safe*
- Past Linear Temporal Logic: PLTL
  - same syntax of LTL_f, but looks into the past
    - yesterday: $\ominus$*happy*
    - once: $\Diamond\!\!\!\!-$*rich*
    - since: *reply* $\mathcal{S}$ *acknowledge*
    - hystorically: $\boxminus$*safe*
- Reasoning in LTL_f/PLTL:
  - transform formulas φ into DFAs $\mathcal{A}_\varphi$
  - for every trace π an LTL_f/PLTL formula φ:

$$\pi \models \varphi \iff \pi \in \mathcal{L}(\mathcal{A}_\varphi)$$