Model-based Multi-agent Tracking

Abstract

Introduction

Multi-agent tracking is the process of locating multiple moving objects (agents) over time. In such scenario, the assumption of having fully observable trajectories of the moving objects means that the sensors are able to capture every state change at every instant, which typically is unrealistic. Normally, obtaining sensor feedback (or the processing of the sensor readings) is associated with a given sampling frequency that misses intermediate data between two subsequent sensor readings.

In this work we propose to exploit (1), a movement model of the agents to track and (2), the power of current classical planner, to produce reliable multi-agent tracking (even when there is a significant amount of missing sensor data). The paper formalizes the *Model-based Multi-Agent Tracking* task and shows that this task is addressable with off-the-shelf classical planners. Last but not least the paper shows the appliability of this approach for tracking ants.

Background

The Classical Planning Model

We define a *classical planning* problem as a tuple $P = \langle X, I, G, A \rangle$:

- X, is the set of *state variables* such that each variable $x \in X$ has an associated finite domain $D(x)$ defining the set of possible values that the variable can take. The set of possible states S in a classical planning problem P is then given by the *cartession product* of the size of the domain variables.
- I is the *initial state*, a full assignment to the variables in X that is compatible with their respective domains.
- G, is a conjunction of *goal conditions* where each $g \in G$ expresses as different Boolean condition over the state variables in X, that is $g : S \rightarrow \{0, 1\}$.
- A is the set of actions whose semantics is expressed with two functions:
  - The *applicability function* $\alpha : A \times S \rightarrow \{0, 1\}$ which determines if a given action $a \in A$ is applicable in a given state $s \in S$.
  - The *transition function* $\theta : A \times S \rightarrow S$ which determines the successor of a state $s \in S$ after an applicable action $a \in A$ is applied.

A *plan* for a classical planning problem P is an action sequence $\pi = \langle a_1, \ldots, a_n \rangle$ that induces the *state trajectory* $s = \langle s_0, s_1, \ldots, s_n \rangle$ such that $s_0 = I$ and, for each $1 \leq i \leq n$, a_i is applicable in s_{i-1} and generates the successor state $s_i = \theta(s_{i-1}, a_i)$. The *plan length* is denoted with $|\pi| = n$. A plan π *solves* P iff $G \subseteq s_n$, i.e., if all the goal conditions are satisfied at the last state reached after following the application of the plan π in the initial state I. A solution plan for P is *optimal* if it has minimum length.

The Observation Model

Given classical planning problem $P = \langle X, I, G, A \rangle$, we define the observation ω_s of a state $s \in S$ as a partial assigment of the state variables in X. In this work we assume that state observations are noiseless, this means that the partial assigment given in ω_s is correct while the value of some variables in X is unknown.

Now, we define the observation of the execution of a plan π that solves P as a sequence $\omega(\pi) = \langle \omega_0, \ldots, \omega_m \rangle$ of *partially-observed* states. Because of the partial observability $0 \leq |\omega_\pi| \leq |\pi| + 1$ and hence, the transitions between two consecutive observed states may involve the execution of more than a single action. Formally $\theta(s_i, \langle a_1, \ldots, a_k \rangle) = s_{i+1}$, where $k \geq 1$ is unknown and unbound. This means that having ω_π does not implies knowing the actual length of π.

We say that a given observation τ is *compliant* with a given plan π iff the sequence of states in ω_π is the same sequence of states traversed by π but:

1. with certain states omitted and/or,
2. the value of certain fluents omitted in that states, i.e. $|\omega_{s_i}| \leq |X|$ for every $0 \leq i \leq m$.

Model-based Multi-Agent Tracking

Model-based Multi-Agent Tracking is the task of identifying k moving objects not by its shape (the k objects are considered identical), but by the observation of their movement in

```
(at-x ant1 l1) (at-y ant1 l12)
(at-x ant2 l5) (at-y ant2 l4)
(at-x ant3 l15) (at-y ant3 l14)
(at-x ant4 l1) (at-y ant4 l1)
(at-x ant5 l2) (at-y ant5 l3)
(at-x ant6 l4) (at-y ant6 l5)
(at-x ant7 l8) (at-y ant7 l9)
(at-x ant8 l10) (at-y ant8 l2)
(at-x ant9 l11) (at-y ant9 l3)
(at-x ant10 l4) (at-y ant10 l9)
```

Figure 1: Initial value of the 2D coordinates for ten agents.

```
(:action inc-x
    :parameters  (?id - agent ?x1 ?x2 - loc)
    :precondition (and (at-x ?id ?x1) (next ?x1 ?x2))
    :effect (and (not (at-x ?id ?x1)) (at-x ?id ?x2)))

(:action inc-y
    :parameters  (?id - agent ?y1 ?y2 - loc)
    :precondition (and (at-y ?id ?y1) (next ?y1 ?y2))
    :effect (and (not (at-y ?id ?y1)) (at-y ?id ?y2)))
```

Figure 2: Operators specified in PDDL for increasing the X and Y coordinates of an agent.

time by building the most possible explanation that is compliant with a model of the objects movement.

The inputs to the *Model-based k-Agent Tracking* task are:

- A set of labels $\{\lambda_1, \ldots, \lambda_k\}$ to uniquely identify each of the k-agents.
- A sequence of state observations $\langle \omega_0, \ldots, \omega_m \rangle$ identifying the configuration of the different agents to track. In this observations agents are unlabeled. To illustrate this, Figure 1 shows an example of a state observation where the configuration of ten ants is given, in terms of their X and Y 2D-coordinates, for the initial time instant.
- A movement model. This model expresses the set of possible configuration changes of an agent between to subsequent time-stamps t and $t+1$. To illustrate this, Figure 2 shows an example of a possible movement for each of the ants to change its configuration between to subsequent time-stamps.

The output to the *Model-based Multi-Agent Tracking* task is, for each agent, a probability distribution of the possible labels that the agent can have given the observations and the agent models.

Tracking as Classical Planning

Here we show that the probability distribution that solves a *Model-based Multi-Agent Tracking* can be estimated using an off-the-shelf classical planning problem.

Evaluation: Ants tracking

Conclusions

References