Refining Process Descriptions from Execution Data in Hybrid Planning Domain Models

Alan Lindsay[3], Santiago Franco Aixela[2], Rubiya Reba[1], Thomas L. McCluskey[1]

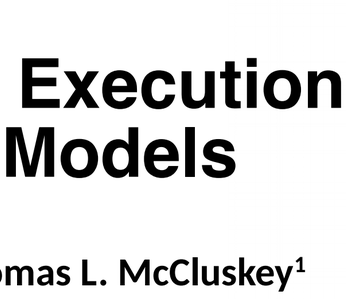
[1]University of Huddersfield, UK
[2] Royal Holloway, University of London, UK
[3]Heriot-Watt University, Edinburgh, UK
alan.lindsay@hw.ac.uk

Abstract

- Domain modelling is a recognised bottleneck in the use of automated planning.
- Engineering domain models using a hybrid representation is particularly challenging
- We consider the problem of the refinement of an engineered hybrid domain model
- We use the problem of modelling traffic flows in an Urban Traffic Management setting as a case study
- We demonstrate that the refined domain models provide more accurate simulation, which can lead to higher quality plans

UTM: Urban Traffic Management

- UTM problem modelled using PDDL+
- Process used to model flow of PCUs through green light:



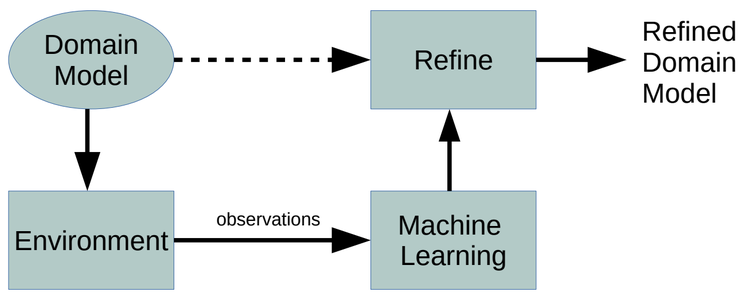
```
(:process flowrun_green
 :parameters (?p - stage ?r1 ?r2 - link)
 :precondition (and (> (occupancy ?r1) 0.0)
   (active ?p) …
 :effect (and
   (increase (occupancy ?r2) (* (#t (turnrate ?p ?r1 ?r2)))
   (decrease (occupancy ?r1) (* #t (turnrate ?p ?r1 ?r2)))))
```

Describing PCU flow through a green light

- Modelled turn rate of PCUs through junctions is constant

- Observed behaviour varies
- E.g., graph indicates "warm-up" period, as PCUs accelerate.

Reprocess: Approach



- Our aim is to refine a PDDL+ Domain Model.
- Using observations from executions in the environment.
- Learn a predictive model to explain the difference between the model's expectations and the observed reality.
- This model is used to refine the original planning model.

Target Representation: More Focused Processes

```
(:process flowrun_green-leaf-i
  :parameters (?p - stage ?r1 ?r2 - link)
 :precondition
 (and
    (> (occupancy ?r1) 0.0) (active ?p)
    (>= (turnrate ?p ?r1 ?r2) 0.0)
    (< (occupancy ?r2) (capacity ?r2))
    (refinement-condition-1 ?p ?r1 ?r2)
    …
    (refinement-condition-n ?p ?r1 ?r2))
 :effect
 (and
    (increase (occupancy ?r2)
        (* (#t (* refinement-factor-i-1 (turnrate ?p ?r1 ?r2))) )
    (decrease (occupancy ?r1)
        (* #t (* refinement-factor-i-2 (turnrate ?p ?r1 ?r2))))))
```

More processes

More specific

Refined behaviour for the specific situation

Using factor exploits information content of original model.

Separate factor learned for each process effect.

Reprocess Algorithm

Generate/select features
- Set of features extracted from domain – including numeric terms: e.g.,(/ (occupancy ?l) (capacity ?l))
- Hand/expert selected
- Filter based feature selection

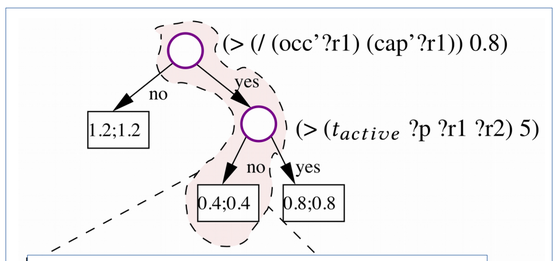
Set as Machine Learning problem

```
function Reprocess(obsTrn,obsVal,P,DM) :
  F = chooseFeatures(DM)
  trnData = makeProcessOrientated(trnObs, P)
  valData = makeProcessOrientated(valObs,P)
  t ← Root()
  GrowTree(trnData,t,F)
  PruneTree(valData,t)
  RevalueTree(trnData +valData,t)
  P+ = extractProcesses(t,P)
  extendModel(DM, P+)
end function
```

Multi-target (Tree Learning)
- Greedy selection of splitting conditions
- Fitness function sums error terms for each target

Compile back into PDDL+

Compiling the Predictive Model



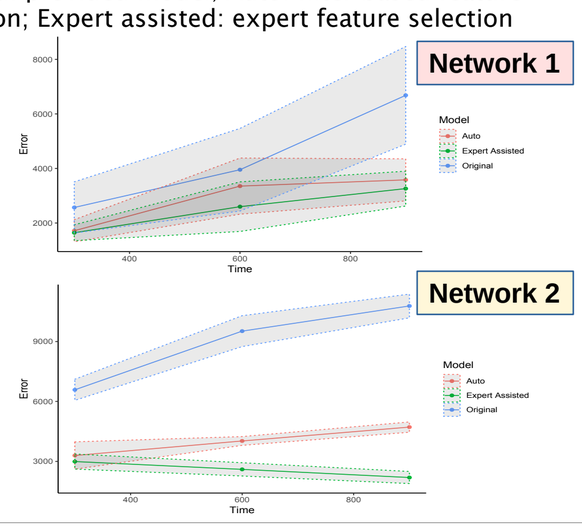

```
(:process flowrun_green-leaf-2
 :parameters (?p – stage ?r1 r2 – link)
 :precondition (and …
   (> (/ (occ' ?r1) (cap' ?r1)) 0.8)
   (<= (greentime ?p) 5))
 :effect (and
   (increase (occ' ?r2)
     (* #t (* 0.4 (turnrate ?p ?r1 ?r2)))
   (decrease (occ' ?r1)
     (* #t (* 0.4 (turnrate ?p ?r1 ?r2)))
```

- Each path from the root to a leaf identifies a specific situation
- Each is used to define a new process
- The values at the leaf are used as factors on the process's effects

Results

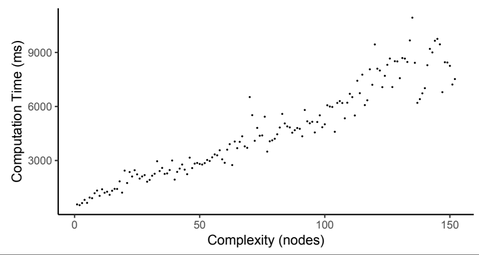
Plan Simulation Accuracy: Observation vs. Model

- Both learned models reduce error – the expert selected features perform best.
- Original: provided model; Auto: filter based feature selection; Expert assisted: expert feature selection



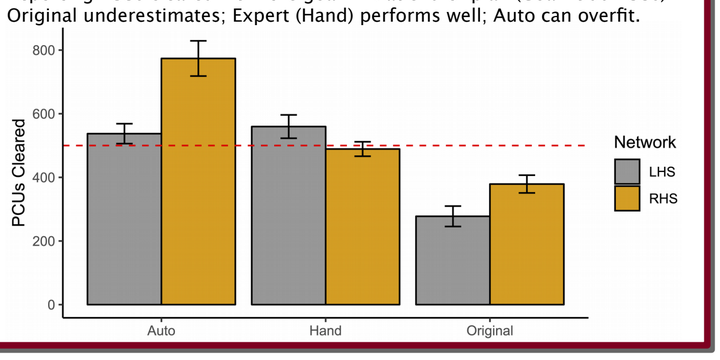
Efficiency of Refined Models: Simulation Time

- As the complexity of the refinement is increased, the computational effort of simulating the planning model increases.



Planning: Plan simulation in Traffic Simulator

- Reporting PCUs cleared from the goal link at end of plan (Goal=500 PCUs)
- Original underestimates; Expert (Hand) performs well; Auto can overfit.



Conclusion

- A general approach for refining hybrid planning models
- Exploiting observation data from executions
- Reduces knowledge engineering effort → Compiles predictive model into PDDL+
- Examined feature space –generated by combining domain functions– is limited
- Demonstrated on a real world application domain → UTM
- Improved accuracy of simulation and planning using the refined models - particularly expert selected features
- **FW**: Alternative feature languages
- **FW:** Examine the balance between model accuracy, planning time and feature computation cost