Universidad Carlos III de Madrid

IBaCoP and IBaCoP2 Planner

Isabel Cenamor, Tomás de la Rosa and Fernando Fernández

June 10, 2017

Who am I?

- I completed my studies as a master in Computer Science and Technology with specialization in Artificial Intelligence in the University Carlos III de Madrid.
- I am member of the Planning and Learning Group as Ph.D. student in Computer Science and Technology under the supervision of Dr. Fernando Fernández and Dr. Tomás de la Rosa.
- My current Ph.D. thesis focuses on the notion of dynamic sequential portfolios. I have applied this concept to classical automated planning using machine learning to create a good combination of planners.

Who am I?

This year, my planner won the Sequential Satisficing track in the 8th IPC and other obtained the second position at Sequential Multi Core.

⋆ More information about me: `www.plg.inf.uc3m.es/~icenamor/`

Outline

Introduction

IBaCoP

Instance Based Configured Portfolios follow two different strategies

- IBaCoP: static configuration. Same configuration for all planning task

Introduction

IBaCoP

Instance Based Configured Portfolios follow two different strategies

- IBaCoP: static configuration. Same configuration for all planning task
- IBaCoP2: dynamic configuration. Different configuration for each planning task

Motivation

- Since none of the existing planners dominates all others in every domains
- The portfolios achieved successful results in the previous International Planning Competitions (IPC)

Domain	Planner
barman	fd-autotune-1
elevators	forkuniform
floortile	fd-autotune-2
nomystery	arvand
openstacks	fd-autotune-2
parcprinter	arvand
parking	lama-2011

Domain	Planner
pegsol	lama-2011
scanalyzer	arvand
sokoban	fd-autotune-1
tidybot	lamar
transport	roamer
visitall	daeyahsp
woodworking	fdss1

The best planners for each domain in sequential satisfying track (IPC-2011)

Definition

Intuition: "Assign the available time to a sub-set of the available planners, and run this configuration"

The "Definition" of Planner Portfolio

Given a set of base planners, $\{pl_1, \ldots, pl_n\}$, and a maximum execution time, T, a planning portfolio can be considered as a sequence of m pairs $< pl_1, t_1 >, \ldots, < pl_m, t_m >$, where $pl_i \in \{pl_1, \ldots, pl_n\}$ and $\sum_{j=1}^{m} t_j = T$.

Portfolio Approaches

The portfolio configuration can be done:

- Over all seen benchmarks: unique configuration
- Per domain: same configuration per domains
- Per problem: different configuration per instance

Portfolio Approaches	Configuration
FD-Stone-Soup [Helmert, 2006]	Over all
PbP [Gerevini et al., 2014]	Domain
IBaCoP2	Instance

First Approach: IBaCoP (I)

IBaCoP

It is configured a priori following a Pareto efficiency approach to select a sub-set of planners (baseline strategy)

- The result of applying the Pareto efficiency to select a sub-set of planners from the planners in sequential satisficing track plus LPG-td (28 planners in total)
- We select the planners that dominates all others in at least one domain (from a set of training domains), taking into account quality and time

First Approach: IBaCoP (II)

Pareto selection

It computes for each planner and task a tuple (Q, T):

- where Q stands for the quality of the best solution found by the same planner
- T is the time (in seconds) it took for the planner to find it

$\star$ (Q, T) is said to pareto-dominate (Q', T') if and only if $Q \leq Q'$ and $T \leq T'$

- The same time per each planner

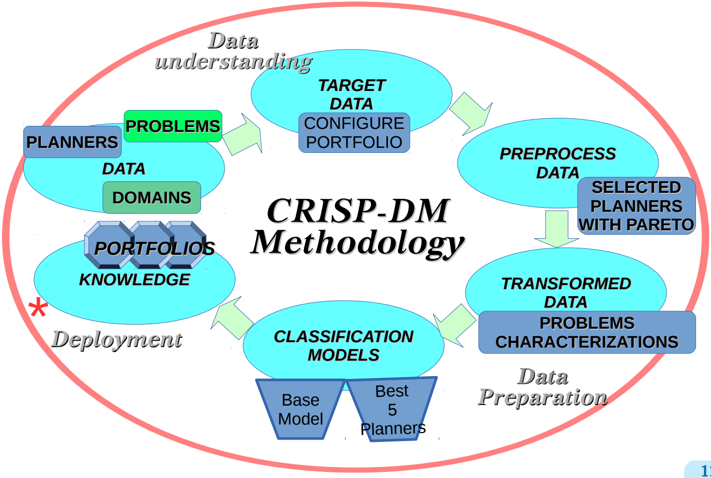
Second Approach: IBaCoP2

IBaCoP2

It decides for each problem the sub-set of planners to use. Such decisions are based on predictive models learnt also with training instances gathered from previous executions of the base planners

- It is a portfolio auto configurable with a classification model
- It performs a second planner selection using predictive models
- These models are the result of learning processes
- The planners with higher confidence are selected and ordered following such confidence
- Same time for 5 selected planners

General System Diagram



Selected Planners

- Arvand [Nakhost et al., 2011]
- Fd-autotune 1 & 2 [Fawcett et al., 2011]
- FD Stone Soup (FDSS) 1 & 2 [Helmert et al., 2011]
- Lama 2008 & 2011 [Richter et al., 2011]
- Probe [Lipovetzky and Geffner, 2011]
- Madagascar [Rintanen, 2011]
- Randward [Olsen and Bryce, 2011]
- Yahsp2-mt [Vidal, 2011]
- LPG-td [Gerevini et al., 2004]

Selected Domains

- IPC5: openstack, pathways, rover, storage, tpp and trucks.
- IPC6: cybersec, elevators, openstack, parcprinter, pegsol, pipesworld, scanalyzer, sokoban, transport and woodworking.
- IPC7: barman, elevators, floortile, nomystery, visitall, tidybot, openstacks, parcprinter, parking, pegsol, sokoban, scanalyzer, transport and woodworking
- Learning 2008: gold-miner, matching-bw, n-puzzle, parking and sokoban.
- Learning 2011: barman, blockworld, depots, gripper, parking, rover satellite, spanner and tpp.

How extracts Features?

- PDDL representation
- SAS+ representation
- Heuristics
- Other variables from Fast Downward Preprocess

Example

Example in Transport domain

Each vehicle can transport move packages depending on its capacity and moving has a cost depending on the length of the road. Picking up or dropping a package.

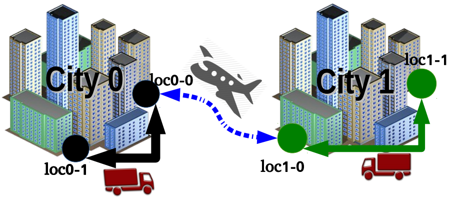
PDDL Domain

Initial State

(in-city loc0-0 city0)
(in-city loc0-1 city0)
(in-city loc1-0 city1)
(in-city loc1-1 city1)
(at truck0 loc0-0)
(at truck1 loc1-1)
(at p0 loc1-0)
(at airplane0 loc0-0)

Goal State

(at p0 loc0-0)

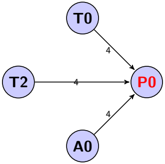


SAS+ Representation

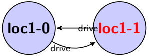
- Unlike STRIPS, it allows state variables to have non-binary (finite) domains
- There are two graph
  - Causal Graph (CG): which is a graph that captures the causal dependencies between the state variables of a given problem
  - Domain Transition Graph (DTG): which encodes the allowed transitions between different values of a variable

SAS+ Representation

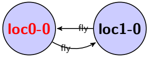
- CG



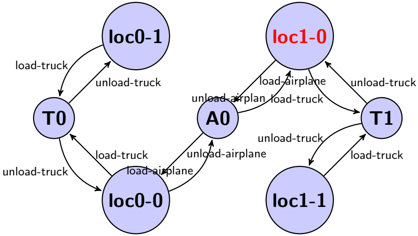
- DTG Trucks



- DTG Airplane



- DTG package



How extracts Features?

- PDDL representation
- SAS+ representation
- Heuristics
- Other variables from Fast Downward Preprocess

PDDL & Fast Downward Preprocess

- types of objects
- functions

- auxiliary atoms in the translation
- implied effects removed

SAS+ Features

- variables CG
- ratio HV $/\#$CG
- σ iEdgesCG
- $\bar{x}$ oEdgesCG
- $max, \bar{x}$ woEdgesCG
- σ oEdgesHV
- max wiEdgesHV
- variables DTG
- Edges DTG
- max wiEdfesDTG

Features (II)

- Heuristics initial state unit cost: h_{add}, h_{max}, h_{FF}, h_{cea}, goal count, h_{LM} and h_{LMCUT}
- the ratio h_{FF}/h_{max}
- *fact balance* of the relaxed plan (RP)

Model

We use some algorithms from Weka [Hall et al., 2009] machine learning toolkit to train models and make predictions.

- The class to predict if a planner solve a planning task (*true* or *false*)

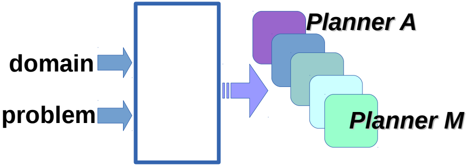
The final selected algorithm is:

Random Forest

We selected the model with best accuracy (99.83% in training phase) This model is a combination of tree predictors such that each tree depends on the values of a random vector and with the same distribution.

Deployment

IBaCoP



IBaCoP2

Sequential Satisficing Planner

- IBaCoP: 12 planners $\leq$ 150 seconds
- IBaCoP2: selects the 5 planners with the highest confidence of solving the problem. The execution order of the planners is based on their confidence. $t \leq 360$ seconds

Sequential Multi-core Planner

For Multi-core track, the planners are the same as for the sequential satisficing track, but taking into account that we have more time (1800 *seconds* $\times$ 4 *cores*). Memory is divided equally among all the planners running in the different cores.

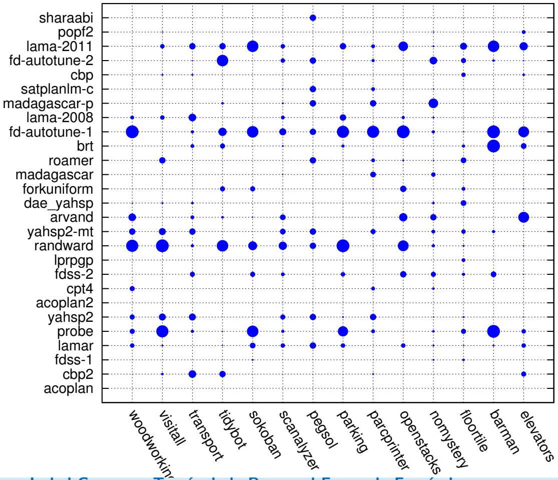
Sequential Agile Planner

- IBaCoP: 7 planners. We assigned as running time the average time to find the first solution in 300 seconds. The execution order for the planners is given by this average from less time to greater values.
- IBaCoP2: selects the 5 planners with the highest confidence of solving the problem. The execution order of the planners is based on their confidence. $t \leq 60$ seconds

Summary Planners

	Planner	seq-sat	seq-agl	seq-mco
1	yahsp2-mt	150	5	600
2	randward	150	50	600
3	arvand	150	55	600
4	fd-autotune-1	150	50	600
5	lama-2008	150	45	600
6	probe	150	–	600
7	madagascar	150	45	600
8	lpg-td	150	50	600
9	fdss-1	150	–	600
10	lama-2011	150	–	600
11	fd-autotune-2	150	–	600
12	fdss-2	150	–	600

Idea of Planner Selection



Learning Planners

- LIBaCoP: Same configuration that IBaCoP but different training set
- LIBaCoP2: Includes regression model.

Score Sat

	Tetris	Barman	Cave	Childsnack	Citycar	Floortile	Hiking
ibacop2	4,02	16,38	7,00	14,98	6,95	**18,23**	18,04
ibacop	6,28	16,27	7,00	15,29	7,32	15,18	18,65
mercury	14,38	13,94	3,00	0,00	3,99	2,00	16,46
miplan	7,46	16,54	7,00	18,22	4,69	4,10	18,14
jasper	9,52	**19,78**	**8,00**	0,00	8,89	2,00	17,19
uniform	11,52	17,83	7,00	1,23	12,74	1,53	17,01
cedalion	3,20	16,85	7,00	0,71	7,71	7,98	18,74
arvandherd	**14,63**	18,12	7,00	5,57	**19,39**	2,00	**19,00**
fdss-2014	9,87	11,64	7,00	1,36	5,00	2,00	17,44
dpmplan	1,80	16,57	7,00	**18,46**	5,82	1,97	16,28
use	3,76	15,12	0,00	0,00	1,65	9,17	8,79

Score Sat (II)

	Main.	Opens.	Parking	Thou	Trans.	Visitall	GED	Total
ibacop2	16,71	5,15	5,28	**15,75**	7,01	13,32	17,40	**166,21**
ibacop	**16,81**	3,58	1,74	13,76	9,94	14,18	16,73	162,73
mercury	5,06	**19,69**	**15,64**	0,00	**20,00**	**19,88**	**19,01**	153,04
miplan	16,62	9,07	11,08	11,18	0,00	8,18	17,72	150,00
jasper	9,28	17,28	12,91	0,00	7,59	15,18	17,27	144,89
uniform	9,36	10,83	9,74	0,00	9,25	19,56	15,65	143,25
cedalion	14,15	17,08	4,29	0,00	5,14	19,49	15,01	137,34
arvandherd	13,17	14,22	0,69	0,00	4,53	0,68	18,09	137,10
fdss-2014	16,63	17,63	10,86	0,00	4,06	8,78	15,61	127,89
dpmplan	15,07	10,15	0,00	13,18	0,00	3,41	15,79	125,50
use	15,03	15,76	3,65	0,00	6,27	14,80	13,15	107,14

Solved Problems

	ibacop2	ibacop	jasper	mercury	uniform	miplan	use
Tetris	5	9	11	**17**	12	8	10
Barman	20	20	20	20	20	20	20
Cave	7	7	**8**	3	7	7	0
Childsnack	**20**	**20**	0	0	2	19	0
Citycar	9	10	13	5	**14**	5	2
Floortile	**20**	16	2	2	2	5	17
Hiking	**20**	**20**	**20**	18	**20**	**20**	18
Maintenance	**17**	**17**	11	7	12	**17**	**17**
Openstacks	6	4	19	**20**	14	10	19
Parking	7	2	19	**20**	12	15	6
Thoughtful	**19**	15	0	0	0	12	0
Transport	13	**20**	10	**20**	17	0	14
Visitall	15	16	**20**	**20**	**20**	10	**20**
GED	20	20	20	20	20	20	20
Total	**198**	196	173	172	172	168	163

Fawcett, C., Helmert, M., Hoos, H., Karpas, E., Röger, G., and Seipp, J. (2011).
Fd-autotune: Automated configuration of fast downward.
The 2011 International Planning Competition, pages 31–37.

Gerevini, A., Saetti, A., Serina, I., and Toninelli, P. (2004).
Lpg-td: a fully automated planner for pddl2. 2 domains.
In *In Proc. of the 14th Int. Conference on Automated Planning and Scheduling (ICAPS-04) International Planning Competition abstracts*. Citeseer.

Gerevini, A. E., Saetti, A., and Vallati, M. (2014).
Planning through automatic portfolio configuration: The pbp approach.
Journal of Artificial Intelligence Research, 50:639–696.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009).
The weka data mining software: an update.
ACM SIGKDD Explorations Newsletter, 11(1):10–18.

Helmert, M. (2006).
The fast downward planning system.
Journal of Artificial Intelligence Research, 26:191–246.

Helmert, M., Röger, G., Seipp, J., Karpas, E., Hoffmann, J., Keyder, E., Nissim, R., Richter, S., and Westphal, M. (2011).
Fast downward stone soup.
The 2011 International Planning Competition, page 38.

Lipovetzky, N. and Geffner, H. (2011).
Searching with probes: The classical planner probe.
The 2011 International Planning Competition, 30(29):71.

Nakhost, H., Valenzano, R., and Xie, F. (2011).
Ary
Th

Olsen, A. and Bryce, D. (2011).