Stefan Edelkamp, University of Dortmund

(joint work with

Jörg Hoffmann University of Freiburg)
Overview

- IPC History
- PDDL
- Evaluation
- Benchmarks
- Organisation
- Summary
Overview

- **IPC History**
- PDDL
- Evaluation
- Benchmarks
- Organisation
- Summary, and CFP
IPC-1, IPC-2, and IPC-3

- **AIPS-98** McDermott  PDDL (STRIPS & ADL);
  5 competitors: IPP, HSP, STAN, Blackbox, SGP; no clear-cut winner

- **AIPS-00** Bacchus; automated vs. hand-coded,
  16 competitors, awards: FF/TALplanner, STAN, System R, **MIPS**, HSP-2 / Shop

- **AIPS-02** Fox & Long; PDDL2.1 (numbers, duration and optimization metrics),
  15 competitors, awards: LPG / TLplan, **MIPS** / Shop2, VHPOP
IPC-4

- **Co-Chairs**
  *Edelkamp & Hoffmann*: Classical Planning
  *Littman & Younes*: Probabilistic Planning

- **Committee**
  *Maria Fox, Derek Long, Fahiem Bacchus, Drew McDermott, David Smith, Daniel Weld, Alessandro Cimatti, Jussi Rintanen, Enrico Giunchiglia, Blai Bonet*
Overview

- IPC History
- PDDL
- Evaluation
- Benchmarks
- Organisation
- Summary, and CFP
Example

Startzustand

Zielzustand
Domain Description

Predicates & Functions

(define (domain zeno-travel)
  (:predicates
    (at ?x - aircraft ?c - city)
    (in ?p - person ?a - aircraft))
  (:functions (fuel ?a - aircraft)
    (distance ?c1 - city ?c2 - city)
    (slow-speed ?a - aircraft)
    (fast-speed ?a - aircraft)
    (slow-burn ?a - aircraft)
    (fast-burn ?a - aircraft)
    (capacity ?a - aircraft)
    (refuel-rate ?a - aircraft)
    (total-fuel-used)
    (boarding-time)
    (debarking-time))

Actions

(:durative-action zoom
  :parameters (?a - aircraft
    ?c1 ?c2 - city)
  :duration
    (= ?duration (/ (distance ?c1 ?c2)
      (fast-speed ?a)))
  :condition
    (and (at start (at ?a ?c1))
      (at start (>= (fuel ?a)
        (* (distance ?c1 ?c2) (fast-burn ?a)))))
  :effect
    (and (at start (not (at ?a ?c1)))
      (at end (at ?a ?c2))
      (at end (increase total-fuel-used
        (* (distance ?c1 ?c2) (fast-burn ?a))))
      (at end (decrease (fuel ?a)
        (* (distance ?c1 ?c2) (fast-burn ?a))))))
Problem Description

define (problem zeno-travel-1)
 (:domain zeno-travel)
 (:objects plane - aircraft ernie scott dan – person
   city-a city-b city-c city-d city)
 (:init
  (= total-fuel-used 0) (= debarking-time 20) (= boarding-time 30)
  (= (distance city-a city-b) 600) (= (distance city-b city-a) 600)
  (= (distance city-b city-c) 800) (= (distance city-c city-b) 800)
  (= (distance city-a city-c) 1000) (= (distance city-c city-a) 1000)
  (= (distance city-c city-d) 1000) (= (distance city-d city-c) 1000)
  (= (fast-speed plane) (/ 600 60)) (= (slow-speed plane) (/ 400 60))
  (= (fuel plane) 750) (= (capacity plane) 750)
  (= (fast-burn plane) (/ 1 2)) (= (slow-burn plane) (/ 1 3))
  (= (refuel-rate plane) (/ 750 60))
  (at plane city-a) (at scott city-a) (at dan city-c) (at ernie city-c))
 (:goal (and (at dan city-a) (at ernie city-d) (at scott city-d)))
 (:metric minimize total-time))
PDDL 2.2: Being Conservative

- **Consolidate existing techniques** for PDDL2.1, so that planners can catch up
- **Understand results** in IPC-3 by making sense of data
- **Handle existing features**, in particular complex optimization criteria
- **Widen applicability focus**
PDDL Extensions

- **Axioms**: derived predicates

\[ (:\text{derived} \ (\text{above} \ ?x \ ?y) \]
\[ (\text{or} \ (\text{on} \ ?x \ ?y) \]
\[ \quad (\text{exists} \ (?z) \ (\text{and} \ (\text{above} \ ?x \ ?z) \]
\[ \quad \quad (\text{above} \ ?z \ ?y)))\\]
PDDL Extensions

- **Timed initial facts**: propositions that occur independently of actions taken by planner

  (at 9 (open shop))
  (at 17 (close shop))
Overview

- IPC History
- PDDL
- Evaluation
- Benchmarks
- Organisation
- Summary, and CFP
(Sub)optimal Planning

- **Hardness** [Helmert] Optimisation in many domains much harder than satisfiability
- **Special prices** for optimal planners or ones that give performance guarantees
- Comparison with existing *special-purpose* optimal *solvers*
- ...
Effort of Hand-Coding

- Measuring *coding time* invested for adding control knowledge
- Measure (compressed) *ASCII text length*
- Let *users other than the programmers* generate the knowledge
- …
Overview

- IPC History
- PDDL
- Evaluation

**Benchmarks**

- Organisation
- Summary, and CFP
Benchmarks...

...have an important impact on the field.

collections

...are more and more taken from the IPC.

should:

be realistic

cover diverse problem structures
Problem Structure – wrpto $h^+$

„Heuristic planners are fast.“ --- „Across certain classes of domains!“
Problem Structure, ctd.

- The benchmarks ought to cover the „range of interesting problem structures“
  - ... in particular, they shouldn’t all have the same structure
- How to measure „structure“?
  - Intuitively: Logistics ≠ Blocksworld
  - Wrpto $h^+$
  - Wrpto computational complexity
Realistic...

- ...is a word with two meanings:
  1. “not restricted to a too simplified model“
  2. “close to applications“
- 1. is a prerequisite of 2.
- IPC-3 made great progress on 1., as well as some steps towards 2.
- There is more to be done on 2.
Close to Applications

- Airport ground traffic control [Hoffmann et al.]
- Biological Pathway Discovery? [Khan et al.]
- Miconic (fully fledged) [Koehler]
- Pipesworld [Milidiu et al.]
- Promela Verification [Edelkamp]
- PSR [Thiebaux]
- Space applications [NASA]
- Pressure networks [Kurien]
IPC-4 Benchmarks

How to combine “close to applications“ and “cover broad range of structures“?

□ With application-oriented domains that cover a broad range of structures...

□ Maybe: one group for very application-oriented domains (eg. Miconic, Pipes, PSR,...), another group for structurally very characteristic domains (eg. Blocks, Log, Mystery, $n^2-1$ puzzle, Sokoban, Settlers,...)
Overview

- IPC History
- PDDL
- Evaluation
- Benchmarks
- Organisation
- Summary, and CFP
Infrastructure

- Test Domains: Blocks + Derived, ZenoTravel + Timed-Initial Facts (compiled and explicit)
- Validator VAL: PDDL 2.1 -> PDDL 2.2 [Durham]
- Converter: ADL -> STRIPS [Hoffmann]
- Grounder: PDDL2.2 -> Grounded PDDL 2.2 [Edelkamp]
Time Table

- Now: Language extensions fixed (PDDL 2.2), first testing examples out
- 01/04: registration deadline for planners
- 02/04 – 03/04: data collection
- 05/04: data evaluation
- 06/04: detailed results presentation at ICAPS’04
- Summer 05: competitors publication
IPC-4 at ICAPS‘04

- Detailed results presentation:
  - extra room throughout conference, maybe after award ceremony
  - in form of posters, assisted with (visualisation, eg.) tools

- Competitors handout:
  - extended abstracts describing all the competitors, distributed to ICAPS participants
Overview

- IPC History
- PDDL
- Evaluation
- Benchmarks
- Organisation

**Summary, and CFP**
Summary

There will be classical and probabilistic (and non-deterministic?) parts of IPC-4.

The ingredients for the classical part are:

- Moderate PDDL extensions
- Careful choice of benchmark domains
- Separate prize(s) for best optimal planner(s)
- Detailed information available online
CFP

- A competition without competitors is boring
- There will be terrific prizes
- We explicitly invite the optimal approaches to come back to us
- ...everybody else can come, too
Mips
domain.pddl  problem.pddl

static analyzer

intermediate representation

symbolic pattern databases  explicit pattern databases

symbolic heuristic search  explicit heuristic search
  BDDA*Weak Strong  A* IDA* EHC

numerical relaxed planning heuristic

relaxed plan

relaxed planning heuristic

scheduling relaxed planning heuristic

sequential plan

PERT scheduler

temporal plan

International Planning Competition