Multi-Objective AI Planning: Evaluating DaE on a tunable benchmark

Mostepha Khouadjia, Marc Schoenauer, Vincent Vidal, Johann Dréo, Pierre Savéant

INRIA Saclay
ONERA, Toulouse
Thalès R&D, Palaiseau

EMO 2013
An AI Planning problem

• Domain:
  – State space $S$ (set of predicates)
  – Set of actions $A$
    with preconditions and effects

• Instance:
  – List of objects (instanciate predicates)
  – Initial state $I$
  – Goal state $G$

Find an optimal sequence of actions
mapping $I$ to $G$
Teaser: MiniZeno (best makespan 8)

- **Domain**: unique predicate at

```prolog
(:action fly :duration (= ?duration (time ?c1 ?c2))

(:action flyVide :duration (= ?duration (time ?c1 ?c2))
```

- **Instance**: 3 cities, 2 planes, 3 passengers

```prolog
(:objects plane1 plane2, person1 person2 person3 city0 city1 city2)
(= (time city0 city1) 2) (= (time city1 city2) 2)
(= (time city1 city0) 2) (= (time city2 city1) 2)
(:init (at plane1 city0) (at plane2 city0) (at person1 city0)
   (at person2 city0) (at person3 city0))
(:goal (at person1 city2) (at person2 city2) (at person3 city2))
```

![Diagram of cities connected by edges](image)
Agenda

• AI Planning
• Multi-objective AI Planning and benchmarks
• Divide-and-Evolve (DaE)
• Multi-objective DaE
• Experiments
• Conclusions
Agenda

• AI Planning
• Multi-objective AI Planning and benchmarks
• Divide-and-Evolve (DaE)
• Multi-objective DaE
• Experiments
• Conclusions
AI Planning

• Yearly ICAPS conference since 1990
• Biennial IPC (International Planning Competition)
  – Since 1998 (7th in 20...11)
  – Drive for PDDL design/improvements

• Lots of exact or satisficing single-objective planners
• Either cost-based (purely sequential) or temporal (actions can be run in parallel)
Agenda

- AI Planning
- Multi-objective AI Planning and benchmarks
- Divide-and-Evolve (DaE)
- Multi-objective DaE
- Experiments
- Conclusions
Multi-objective AI Planning

- PDDL 3.0 (2006) allows for several objectives
- But existing strategies/heuristics not applicable
- → aggregation of objectives

- A multi-objective track in IPC 5 and IPC 6
  ... not in IPC 7
- + recent approach [Sroka & Long, STAIRS 2012]
  using LPG [Geverini et al., AI 08]
Multi-objective Zeno Benchmark

- 3n passengers
- 2 planes
- from city0
- to city 4

Pareto fronts for 6 passengers and varying cost/durations values
Agenda

• AI Planning
• Multi-objective AI Planning and benchmarks
• Divide-and-Evolve (DaE)
• Multi-objective DaE
• Experiments
• Conclusions
DaE: the Paradigm

- Slicing the original problem into a series of (hopefully simpler) sub-problems
- Using a 'dumb' solver on each sub-problem

(variable length) Genotype = (c_1, c_2, c_3)
DaE-YAHSP

Problem

\[ <S, A, I, G> = P_{D(I,G)} \]

Representation

Ordered list of partial states \( S_0 = I, S_1, ..., S_n, S_{n+1} = G \)

Evaluation

Solve consecutive sub-problems \( P_D(S_k, S_{k+1})/ k \in [0,n] \)

with embedded **single-objective planner** YAHSP [Vidal, ICAPS 04]

Fitness

All problems solved: concatenate partial plans

Fails solving \( P_D(S_i, S_{i+1}) \): Penalization

Crossover: One-point crossover

Mutations: AddGoal, delGoal, addAtom, delAtom
Single-objective DAE-YAHSP

• An original (intricate) memetization strategy
• A very noisy fitness
but
• YAHSP is both cost- and temporal planner
• DAE-YASHP: state-of-the-art performance in both domains [Bibai et al., ICAPS 2010]
• Silver medal, Humies Awards 2010
• Ranked 1st, temporal satisficing, IPC 2011
Agenda

• AI Planning
• Multi-objective AI Planning and benchmarks
• Divide-and-Evolve (DaE)
• Multi-objective DaE
• Experiments
• Conclusions
Multi-objective DAE-YAHSP

• YAHSP is both cost- and temporal planner... can compute one while optimizing the other (since 2010)

• « Only » need to change the EC engine! [Schoenauer, Saveant, Vidal, EvoCOP'06]

• Two problems
  – **Cost**: additive (tax at every landing)
  – **Risk**: max (only highest value matters)
YAHSP strategy

- Optimize cost or makespan?

Noisy fitness: objectives of a single individual computed by YAHSP with both pure strategies

→ randomize, and use weights (individual level)
Parameter Tuning

Parameters ranges/discretization

Tuner

Calls with different parameter settings

Target Algorithm

Solves

Problem instances

Returns solution quality (here hypervolume)

UBC ParamILS [Hutter et al., JAIR 2009]
### Parameter Tuning (2)

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Range</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W$-makespan</td>
<td>$0, 1, 2, 3, 4, 5$</td>
<td>Weighting for optimizing makespan during the search</td>
</tr>
<tr>
<td>$W$-cost</td>
<td>$0, 1, 2, 3, 4, 5$</td>
<td>Weighting for optimizing cost during the search</td>
</tr>
<tr>
<td>Pop-size</td>
<td>$30, 50, 100, 200, 300$</td>
<td>Population Size</td>
</tr>
<tr>
<td>Proba-cross</td>
<td>$0.0, 0.1, 0.2, 0.5, 0.8, 1.0$</td>
<td>Probability (at population level) to apply crossover</td>
</tr>
<tr>
<td>Proba-mut</td>
<td>$0.0, 0.1, 0.2, 0.5, 0.8, 1.0$</td>
<td>Probability (at population level) to apply one mutation</td>
</tr>
<tr>
<td>$w$-addAtom</td>
<td>$0, 1, 3, 5, 7, 10$</td>
<td>Relative weight of the addAtom mutation</td>
</tr>
<tr>
<td>$w$-addGoal</td>
<td>$0, 1, 3, 5, 7, 10$</td>
<td>Relative weight of the addGoal mutation</td>
</tr>
<tr>
<td>$w$-delAtom</td>
<td>$0, 1, 3, 5, 7, 10$</td>
<td>Relative weight of the delAtom mutation</td>
</tr>
<tr>
<td>$w$-delGoal</td>
<td>$0, 1, 3, 5, 7, 10$</td>
<td>Relative weight of the delGoal mutation</td>
</tr>
<tr>
<td>Proba-change</td>
<td>$0.0, 0.1, 0.2, 0.5, 0.8, 1.0$</td>
<td>Probability to change an atom in addAtom mutation</td>
</tr>
<tr>
<td>Proba-delatom</td>
<td>$0.0, 0.1, 0.2, 0.5, 0.8, 1.0$</td>
<td>Average probability to delete an atom in delAtom mutation</td>
</tr>
<tr>
<td>Radius</td>
<td>$1, 3, 5, 7, 10$</td>
<td>Number of neighbour goals to consider in addGoal mutation</td>
</tr>
</tbody>
</table>

→ $1.5 \times 10^9$ Possible configurations
Agenda

• AI Planning
• Multi-objective AI Planning and benchmarks
• Divide-and-Evolve (DaE)
• Multi-objective DaE
• Experiments
• Conclusions
Experimental Conditions

- **EC engines**: NSGA-II, SPEA2, IBEA-ε, IBEA-Hv
- **Implementation**: ParadisEO
- **Instances**: Zeno3, Zeno6, Zeno9
- **11 independent runs (also within ParamILS)**
- **Stopping criterion**
  - ParamILS: 48h (Zeno3 and 6), 72h (Zeno9)
  - Optimization: 300, 600 and 900s
- **Statistical tests**: Wilcoxon signed rank test with 95% confidence
Comparative Results

Evolution of hypervolume / reference set for all 4 MOEAs
Ibea-Hv performs significantly better
Pareto Front Attainability

(a) $\text{MultiZenO6}_{\text{cost}}$

(b) $\text{MultiZenO6}_{\text{risk}}$

Hitting plots for Ibea-Hv on Zeno6 (Cost and Risk)
Influence of YAHSP strategy

(a) YAHSP optimizes makespan

(b) YAHSP optimizes cost

Hitting plots for Ibea-Hv on Zeno6 for the 2 'pure' strategies
Comparison with aggregation

Pareto Fronts (from 11 runs) for Zeno9 (scales are different)
See [EvoCOP'13]
Summary

• MO-DAE-YAHSP : a multi-objective evolutionary planner based on a single-objective classical planner
• A simple MO benchmark for AI Planning
• Randomized YAHSP strategy (confirmed:-)
• IBEA-Hv best choice (on Zeno benchmarks)
• Outperforms aggregation
  – Both DAE [EvoCop'13] and LPG [submitted]
Perspectives

- Comparison with LPG-based approach
  - [submitted]
- Extended benchmarks from IPC domains
  - [submitted] in parts
- **Adaptive** choice of YAHSP strategy
  - Individual or sub-goal level?
- On-line parameter setting
  - **Adaptive** operator selection/tuning
- Better handling of risk