APPLICATION OF PROPOSITIONAL SATISFIABILITY TO SPECIAL CASES OF COOPERATIVE PATH-PLANNING

Pavel Surynek

Charles University in Prague, Czech Republic
and **Kobe University, Japan**

Problem of Cooperative Path-planning (CPP)

- **Abstraction** for tasks of motion of multiple (autonomous or passive) entities in a certain environment (real or virtual).
  - Entities are given an **initial** and a **goal** arrangement in the environment.
  - We need to **plan movements of entities in time**, so that entities reach the goal arrangement while **physical limitations are observed**.

- **Physical limitations** are:
  - Entities must **not collide with each other**.
  - Entities must **not collide with obstacles** in the environment.

CPP – Formal definition (1)

Wilson, 1974; Kornhauser et al., 1984; Ryan, 2008

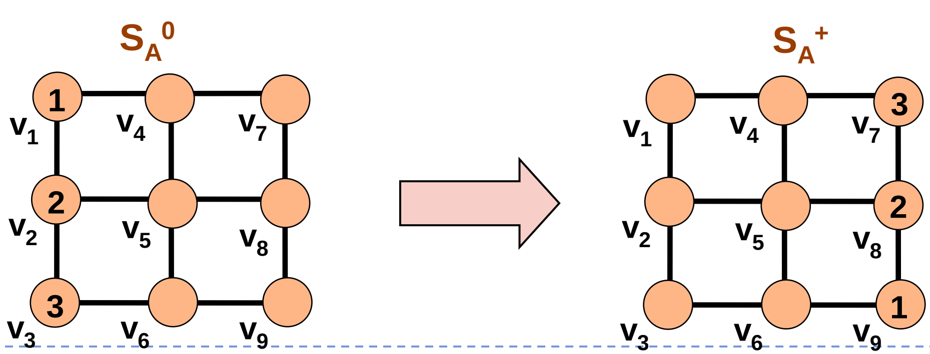
- The environment is modeled as an **undirected graph** where **vertices represent locations** in the environment occupied by agents and **edges** enable agents to go to the **neighboring location**.

- **Formal definition** of the task of CPP
  - It is a quadruple $\Pi = (G, A, S_A^0, S_A^+)$, where:
    - $G=(V,E)$ is an **undirected graph**,
    - $A = \{a_1,a_2,\ldots,a_\mu\}$, where $\mu<|V|$ is a **set of agents**,
    - $S_A^0: A \rightarrow V$ is a uniquely invertible function determining the **initial arrangement of agents** in vertices of G, and
    - $S_A^+: A \rightarrow V$ is a uniquely invertible function determining the **goal arrangement of agents** in vertices of G.

CPP – Formal Definition (2)

Wilson, 1974; Kornhauser et al., 1984; Ryan, 2008

- The **dynamicity** of the task is as follows:
  - An agent occupying a vertex at time step *i* can move into a neighboring vertex (the move is finished at time step *i+1*) if the target vertex is **unoccupied** at time step *i* and **no other agent** is moving simultaneously into the same target vertex
- For the given $\Pi = (G, A, S_A^0, S_A^+)$, we need to find:
  - A sequence of moves for every agent such that dynamicity constraint is satisfied and every agent reaches its goal vertex.



Solution of an instance of cooperative path-planning on a graph with A={1,2,3}

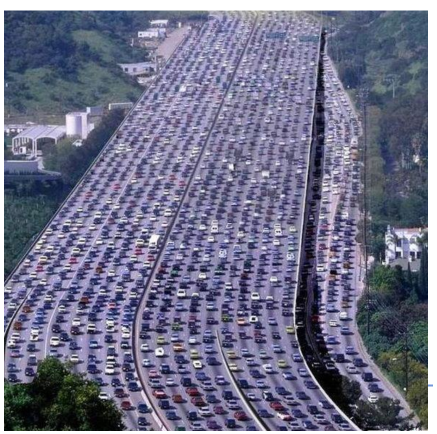
makespan=7

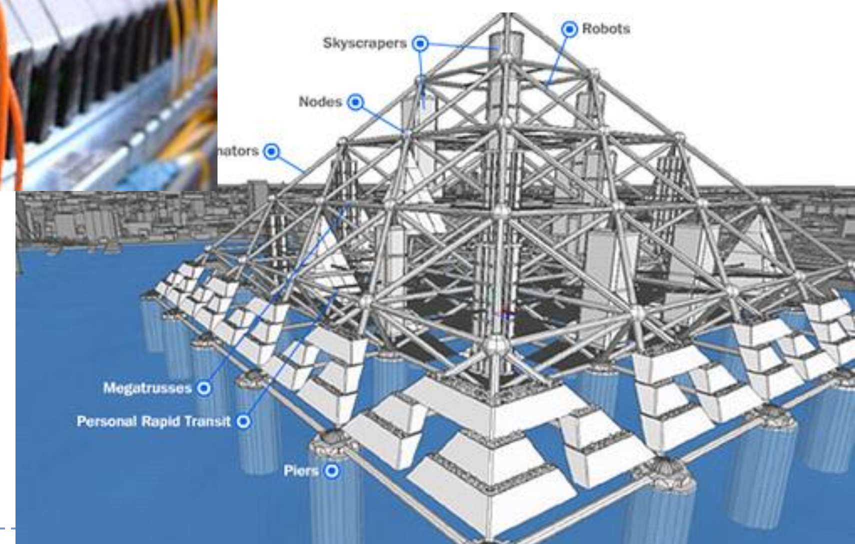
$\mathbf{M_1}$=[$\mathbf{v_1}$, v_4, v_7, v_8, v_9, v_9, $\mathbf{v_9}$]
$\mathbf{M_2}$=[$\mathbf{v_2}$, v_2, v_1, v_4, v_7, v_8, $\mathbf{v_8}$]
$\mathbf{M_3}$=[$\mathbf{v_3}$, v_3, v_3, v_2, v_1, v_4, $\mathbf{v_7}$]

Time step: 1 2 3 4 5 6 7

Motivation

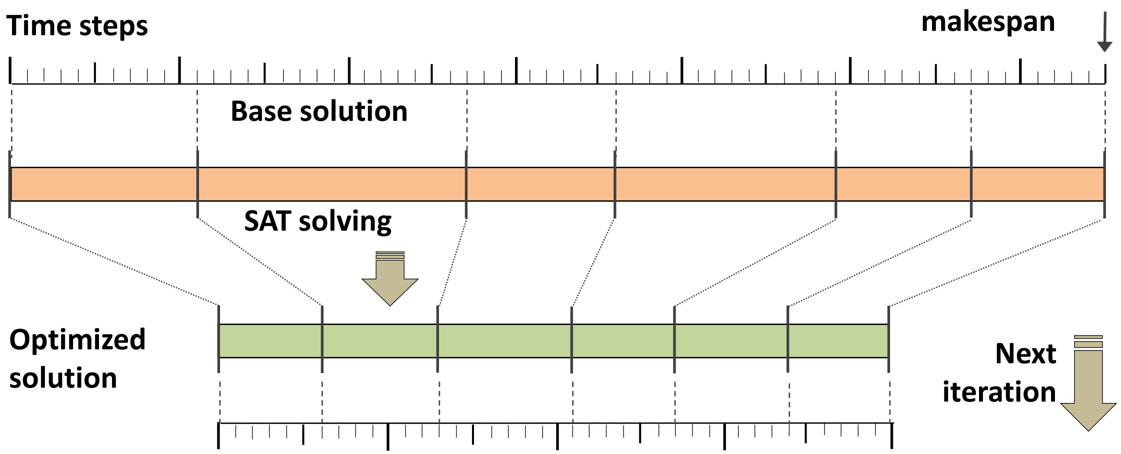
- Container rearrangement
  (entity = **container**)
- Heavy traffic
  (entity = **automobile** (in jam))
- Data transfer
  (entity = **data packet**)
- Generalized lifts
  (entity = **lift**)





COBOPT – CPP as Propositional Satisfiability

- Suppose that we are able to construct a **propositional formula** such that
  - **satisfiable** iff there exists a solution to CPP of a given **makespan**
- Next suppose that we are provided with **makespan suboptimal solution** (base solution – can be generated in polynomial time)
  - we can find **makespan optimal** replacement of the given sub-sequence of the base solution



All-Different Encoding of CPP

- Encodes "**where is the given agent**"
- The state at the given time step *i* is described by the following **integer variables** for each $a \in A$:
  - $L_a^i \in \{1,2,\ldots, |V|\}$ with the interpretation that $L_a^i = j$ iff the agent a is located in the j-th vertex of the graph G
- The requirement that there is at most one agent per vertex is modeled as All-Different(L_{a1}^i, L_{a2}^i, …, $L_{a\mu}^i$)
- Other constraints are more complicated
  - it is necessary to express that agents can move along edges only
  - and that target vertex of the movement must be empty
- Augmenting by heuristics
  - some vertices are unreachable by the agent in the given time step

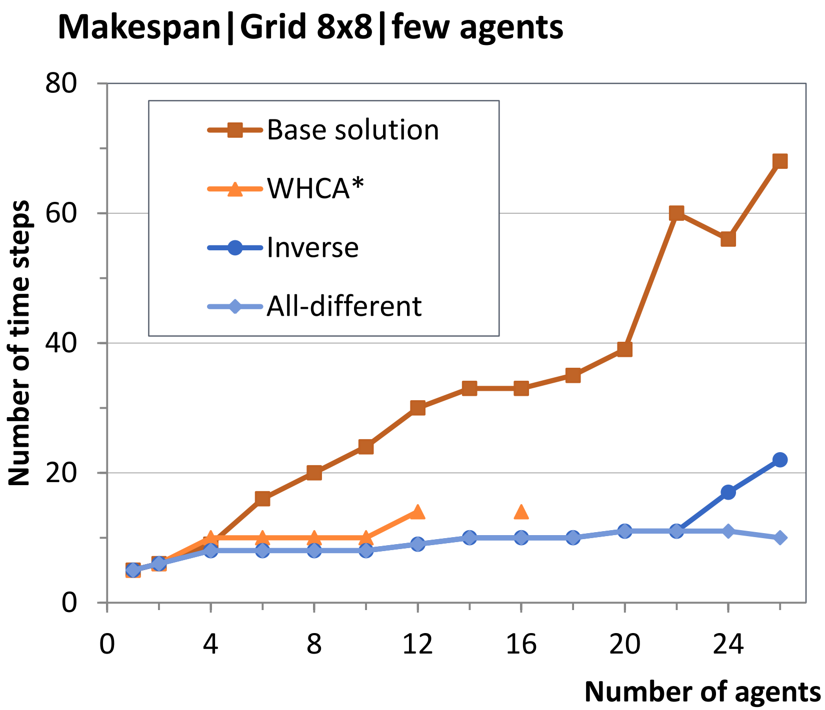
Encoding Size Comparison

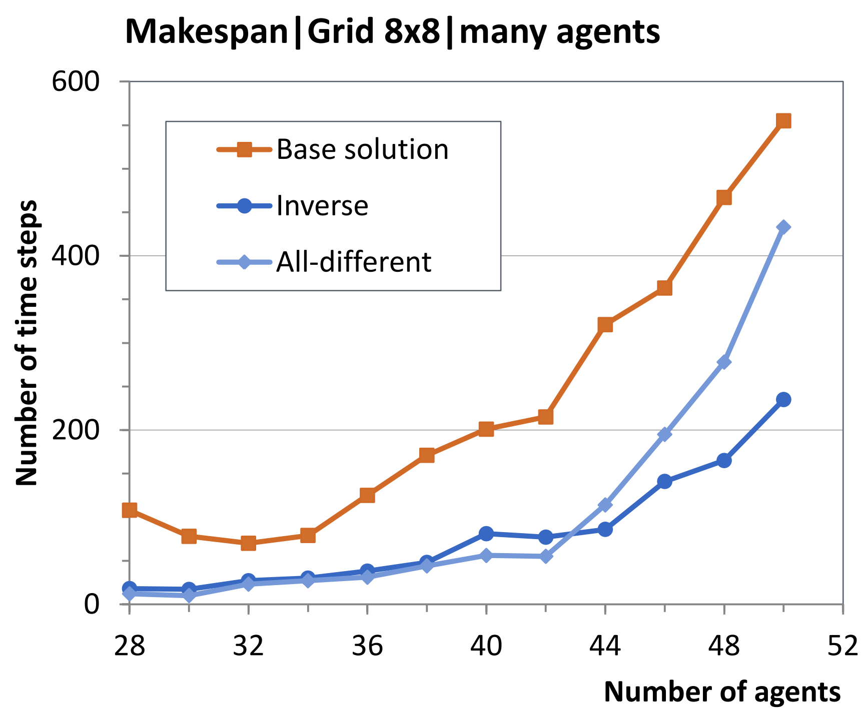
- Experimental setup:
  - 4-connected **grid** of size 8x8
  - **random** initial and goal arrangement of agents

|A| in the 4-connected grid 8×8	Number of layers	SATPLAN encoding		SASE encoding		INVERSE encoding		ALL-DIFFERENT encoding	
		|Variables|	|Clauses|	|Variables|	|Clauses|	|Variables|	|Clauses|	|Variables|	|Clauses|
4	8	5864	55330	11386	53143	5400	38800	11128	54356
8	8	10022	165660	19097	105724	5920	48224	25136	114952
12	8	14471	356410	26857	168875	5920	46176	42024	181788
16	10	30157	1169198	51662	372140	8122	76192	79008	326736
24	10	43451	2473813	73101	588886	8122	71072	140400	537528
32	14	99398	8530312	157083	1385010	12396	137120	309824	1120672

Makespan Comparison – grid 8x8

- Compared against WHCA*
  - WHCA* is decoupled
    - often produces near makespan optimal solution





Concluding Remarks

- Improving sub-optimal solutions of cooperative path-planning by modeling the problem as propositional satisfiability.

- COBOPT: short subsequences of a sub-optimal solution are replaced by the makespan optimal ones.

- SAT encoding (and its variants)
  - All-Different encoding

- **COBOPT** solution optimization represents state-of-the-art in generating short solutions to CPP