Reduced Time-Expansion Graphs and Goal Decomposition for Solving Cooperative Path Finding Sub-optimally

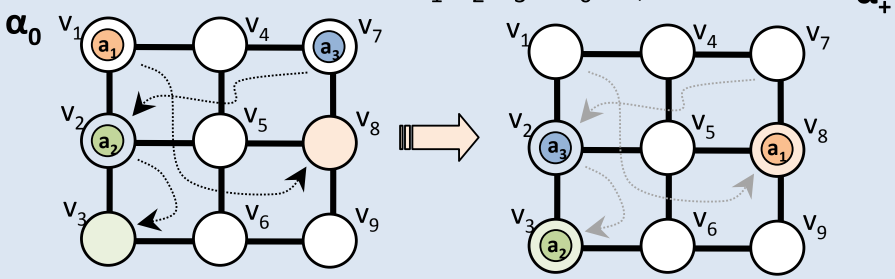
Pavel Surynek
Charles University in Prague
Czech Republic

24th International Joint Conference on Artificial Intelligence (**IJCAI 2015**), Buenos Aires, Argentina

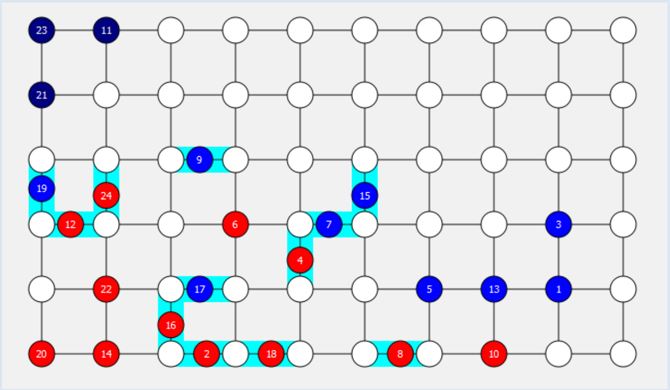
Cooperative Path Finding

- **CPF**
  - a group of agents (robots, cars, units in RTS, ...)
  - each agent has unique **start** and **goal** location
  - **collisions** must be avoided
  - environment - undirected graph

CPF $\boldsymbol{\Sigma}$=(G, {a_1,a_2,a_3}, α_0, α_+)

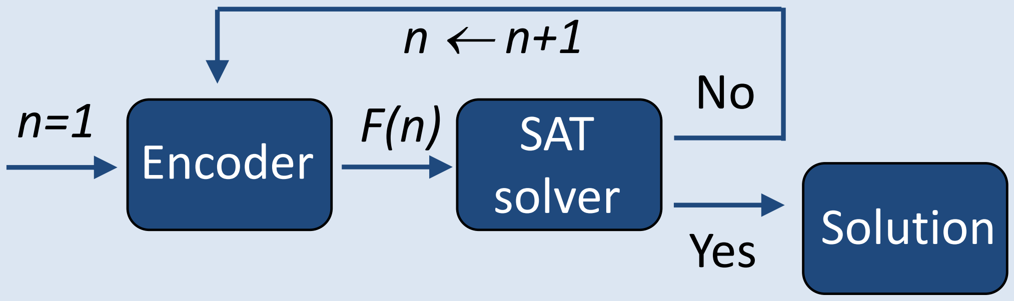


	α_0	α_1	α_2	α_3	$\alpha_4 = \alpha_+$
a_1	v_1	v_1	v_2	v_5	v_8
a_2	v_2	v_3	v_3	v_3	v_3
a_3	v_7	v_4	v_4	v_1	v_2



Solving CPF by Reducing it to SAT

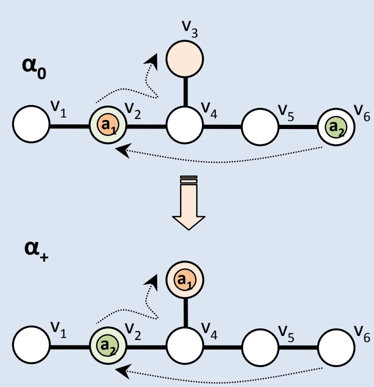
- **expand** (copy) graph *G* over time
  - the number of expansions *n* is specified
  - **represent arrangements** of agents in time
  - **encode** relocation of agents through expanded graph as a propositional formula *F(n)*
    - constraints to check validity of transitions between arrangements at time-steps
  - **ask SAT solver** whether *F(n)* **is solvable**



Standard Time Expansion

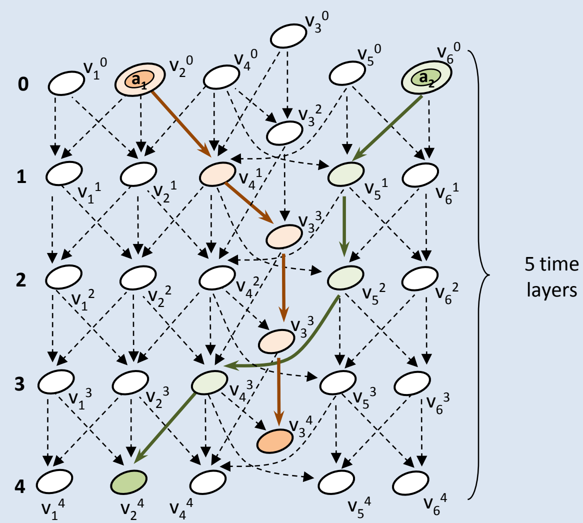
- each **expansion corresponds to a time step**
  - **placement** of each agent at each time step is explicitly represented
    - too **many expansions** in case of long makespan
  - can be used for **makespan optimal** solving of CPF

CPF $\mathbf{\Sigma}$=(G=(V,E), {a_1,a_2}, α_0, α_+)



	α_0	α_1	α_2	α_3	$\alpha_4 = \alpha_+$
a_1	v_2	v_4	v_3	v_3	v_3
a_2	v_6	v_5	v_5	v_4	v_2

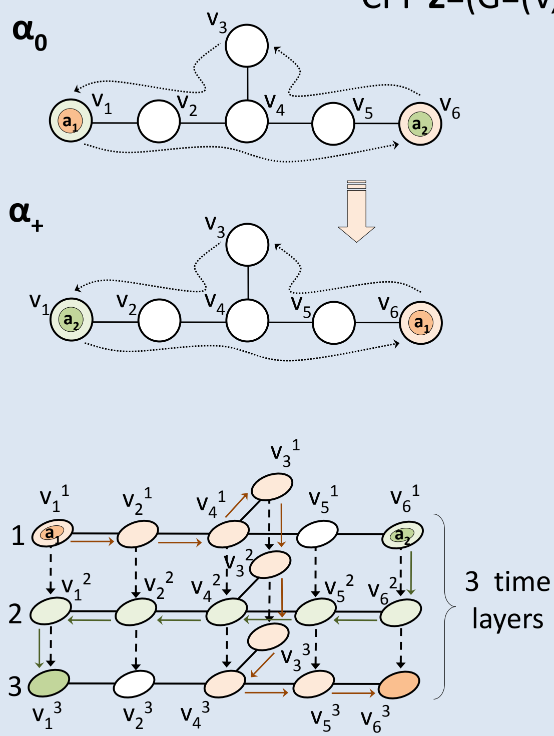
Exp_T(G, 4)



Reduced Time Expansion

- expansions correspond **avoidance** among agents
  - movements of agents are represented as **vertex disjoint** paths
  - **few expansions** for small interaction among agents
    - even if makespan is large
  - can used for makespan **suboptimal** CPF solving

CPF $\boldsymbol{\Sigma}=(G=(V,E), \{a_1,a_2\}, \alpha_0, \alpha_+)$

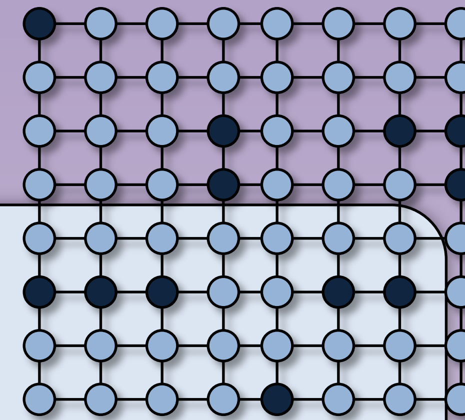


	a_1	a_2
α_0	v_1	v_6
α_1	v_2	v_6
α_2	v_4	v_6
α_3	v_3	v_6
α_4	v_3	v_5
α_5	v_3	v_4
α_6	v_3	v_2
α_7	v_3	v_1
α_8	v_4	v_1
α_9	v_5	v_1
$\alpha_{10} = \alpha_+$	v_6	v_1

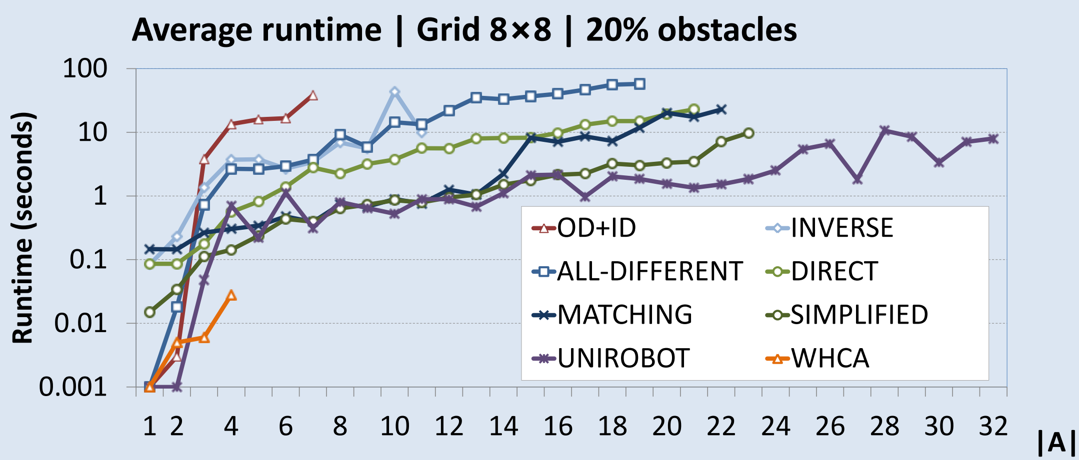
Goal Decomposition

- **observation**
  - few expansions are needed if there is **little difference** between the initial and goal arrangement
- place agents **one by one** (UniROBOT)
  - solve a separate CPF for single agent placement
    - few expansions ⇒ small propositional formula ⇒ **easy SAT**
    - merge solutions into an overall solution of the original CPF

Experimental Evaluation



- **setup**
  - **4-connected** grid, with obstacles
  - **SAT-based solving** with various propositional encodings is compared with **A*-based algorithms**



Makespan

|A|	1	4	8	12	16	20	24	28	32
optimal	5.3	8.4	11.0	11.7	12.4	12.3	-	-	-
WHCA*	5.6	9.3	-	-	-	-	-	-	-
UniROBOT	9.3	15.8	33.0	49.3	83.4	96.1	131.4	154.1	201.7