Relocation Tasks and A Hierarchical Subclass

Charles University in Prague
Czech Republic

Pavel Surynek

Kobe University
Japan

Relocation Problem

- Abstraction for **relocation of objects**
- **Components** of a relocation task:
  - G=(V,E) – an undirected graph modeling the **environment**
  - $Y=\{y_1, y_2, \ldots, y_m\}$ with $m \in \mathbb{N}$ a set of **entities**
  - $T=\{t_1, t_2, \ldots, t_n\}$ with $n \in \mathbb{N}$ a set of **entity types**
    - each entity has assigned a type by $\tau: Y \rightarrow T$
  - **Allowance** constraint $\alpha \subseteq \{1,2,\ldots,|Y|\}^n$
    - compatible numbers of entities of various types allowed to **reside together** in a vertex
    - **example**: 1 entity of type t_1 or two 1 entity of type t_2 with 1 entity of type t_2 can stay together in a vertex
  - **Mobility** constraint $\beta \subseteq \{1,2,\ldots,|Y|\}^n$
    - compatible numbers of entities **movable together**

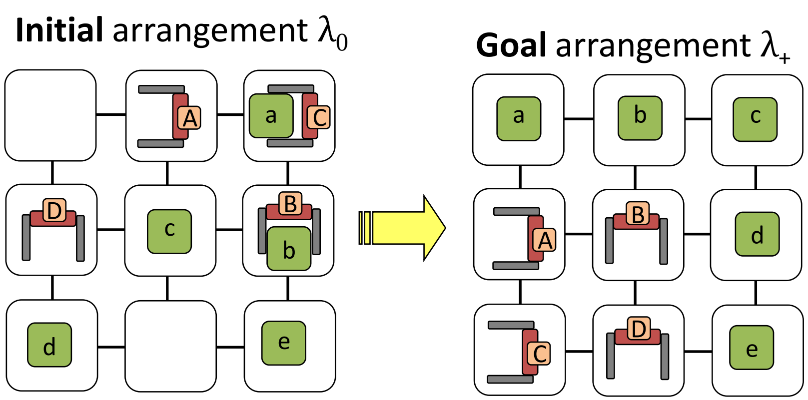
Example of Hierarchical Relocation Task

- At each discrete time-step entities are arranged in vertices of G
  - λ_0:Y→V **initial** arrangement
  - λ_+:Y→V **goal** arrangement
- If β={[1,0, ...,0]; [1,1,0,...,0]; ... [1,1, ..., 1]} (leading ones) ⇒ **hierarchical subclass**
- **Example:**

Y={A,B,C,D,a,b,c,d,e}
T ={robot, box}

τ(A)= τ(B)= τ(C)= τ(D)=robot

τ(a)= τ(b)= τ(c)= τ(d)= τ(d)=box



α={[0,0]; [1,0]; [0,1]; [1,1]}

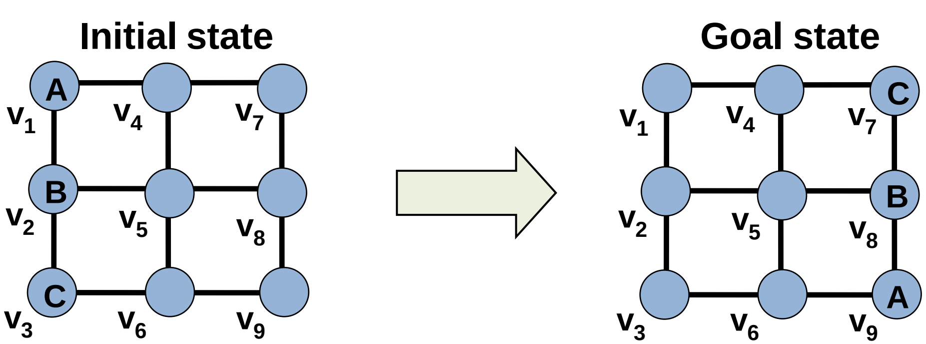
Empty vertex; one robot in a vertex; one box in a vertex; and one robot and one box in a vertex are allowed.

β={[1,0]; [1,1]}

Robots can move freely while boxes can be moved by robots only.

Relation to Cooperative Path-Finding

- Employ polynomial time/space algorithms for **cooperative path-finding (CPF)**
  - can be used to solve hierarchical subclass
- CPF as a special case of relocation task
  - only one type of entities (called agents)
  - constraints are trivial $\alpha=\{[1]\}$ and $\beta=\{[1]\}$
- **Example:**



Set of **agents** = {1,2,3}

plan for **agent A** = [$\mathbf{v_1}$, v_4, v_7, v_8, v_9, v_9, $\mathbf{v_9}$]
plan for **agent B** = [$\mathbf{v_2}$, v_2, v_1, v_4, v_7, v_8, $\mathbf{v_8}$]
plan for **agent C** = [$\mathbf{v_3}$, v_3, v_3, v_2, v_1, v_4, $\mathbf{v_7}$]
Time step: **1 2 3 4 5 6 7**

makespan = 7

Tractability of Hierarchical Sub-class

- Solve hierarchy by levels
  - place entities of the **top type** first
  - continue with entities of **lower types**
  - finally place entities of the **bottom type**
    - equivalent to cooperative path finding (CPF)
- Moving entity along edge on level *h*
  - requires at most |V| moves of entities on level *h-1*
    - vertex in front of the entity must be **freed**
    - supporting entities on level h-1 must be **prepared**
  - extra time of $|V|^2$ is needed to find paths in G
  - $T(h) \leq |V|T(h-1)+|V|^2$ and $T(1)=1$
  - We obtain that: $T(h)=O(|V|^h)$

Conclusions and Related Works

- Many real-life problems can be modeled as relocation task
- Hierarchical structure of the presented class is very typical in relocation problem
- Tractability of hierarchical class has been shown