Algorithmic Generation of Path Fragment Covers for Mobile Robot Path Planning

3rd IEEE Conference On Intelligent Systems
4–6 September 2006

Jan Willemson, Maarja Kruusmaa

Tartu University, Estonia

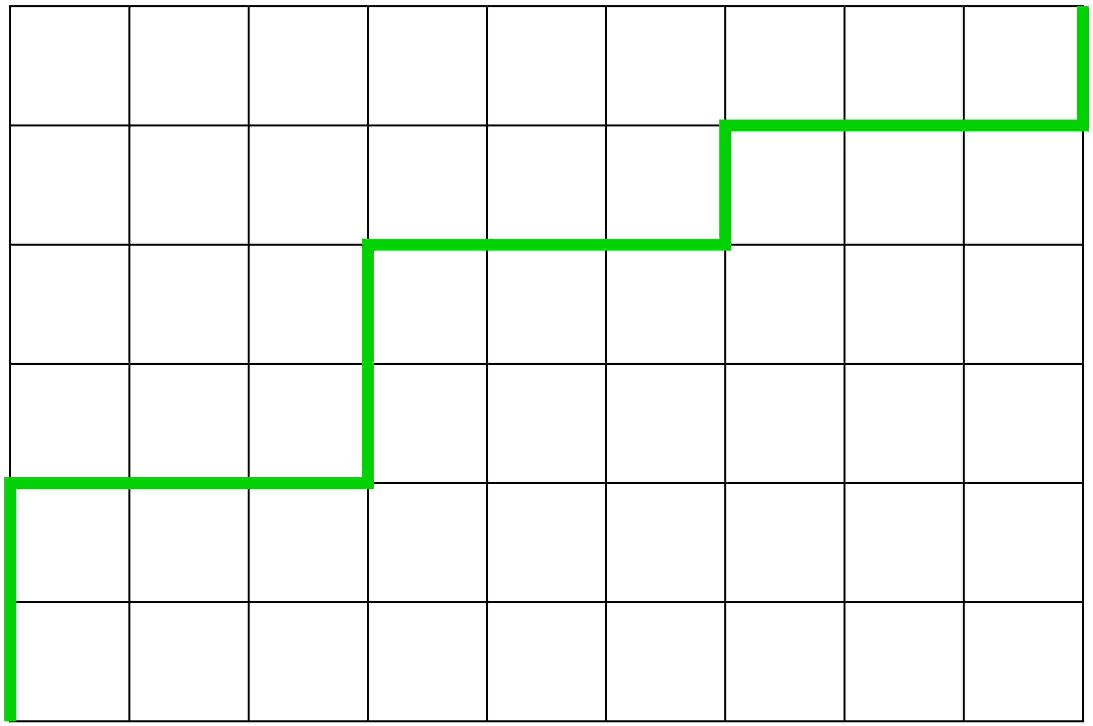
Institute of Computer Science and Institute of Technology

Motivation

- We consider mobile robot path planning in (semi-)dynamic environments
- The task of the robot is to traverse repetedly between predefined points, covering some given area, e.g. for
  - surveillance
  - cleaning tasks
  - demining applications
- At the same time, the robot must complete its task safely, minimizing the number of replannings and collisions with obstacles

The Model

- We model the robot's environment as an $m \times n$ grid map, where the robot moves from lower left to upper right corner moving only right and up along the grid lines



The Problem

- How to select a subset of all grid paths so that they would cover the grid as well as possible?

The Problem

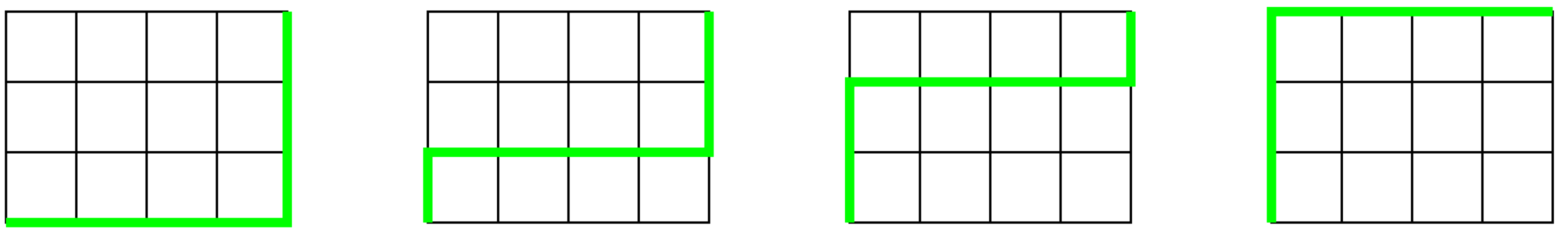
- How to select a subset of all grid paths so that they would cover the grid as well as possible?
- What does it mean to cover a grid?

The Problem

- How to select a subset of all grid paths so that they would cover the grid as well as possible?
- What does it mean to cover a grid?
- All grid points? No, we would get an unsatisfactory cover

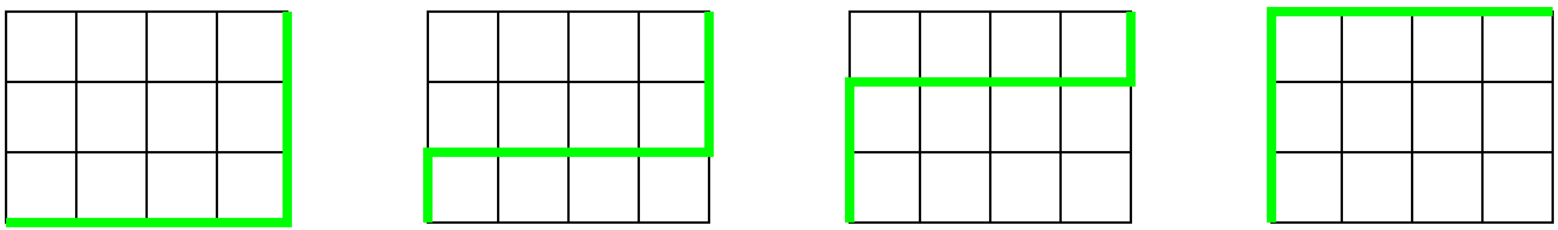
The Problem

- How to select a subset of all grid paths so that they would cover the grid as well as possible?
- What does it mean to cover a grid?
- All grid points? No, we would get an unsatisfactory cover



The Problem

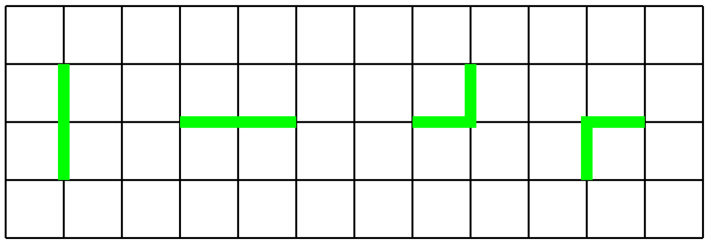
- How to select a subset of all grid paths so that they would cover the grid as well as possible?
- What does it mean to cover a grid?
- All grid points? No, we would get an unsatisfactory cover



- It is possible to prove that the approach of defining distance between grid paths and trying to cover balls of paths does not scale well

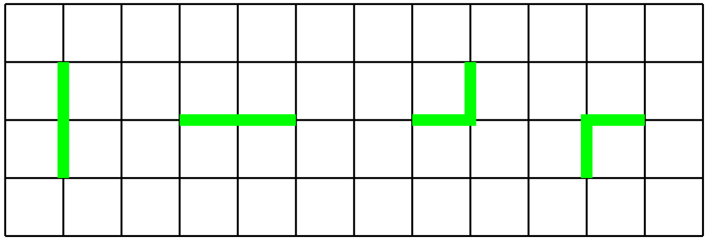
Covering Path Fragments

- We define the path fragments (of length 2) to be the following unions of grid fragments:



Covering Path Fragments

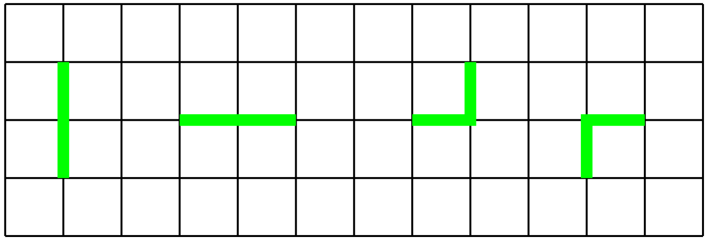
- We define the path fragments (of length 2) to be the following unions of grid fragments:



- Let $\mathcal{P}_{m,n}$ denote the set of all legal grid paths
- A subset $X \subseteq \mathcal{P}_{m,n}$ is called a *path fragment cover* if for every possible path fragment there is a path in X containing it

Covering Path Fragments

- We define the path fragments (of length 2) to be the following unions of grid fragments:



- Let $\mathcal{P}_{m,n}$ denote the set of all legal grid paths
- A subset $X \subseteq \mathcal{P}_{m,n}$ is called a *path fragment cover* if for every possible path fragment there is a path in X containing it
- **The main objective of the paper:** *Find the minimal size of a path fragment cover for the* $m \times n$ *grid. Describe all the path fragment covers of minimal size and design the means for generating them*

Minimal Path Fragment Cover

- **Theorem 1.** Cardinality $\tau(m,n)$ of the minimal path fragment cover of the the $m \times n$ grid is

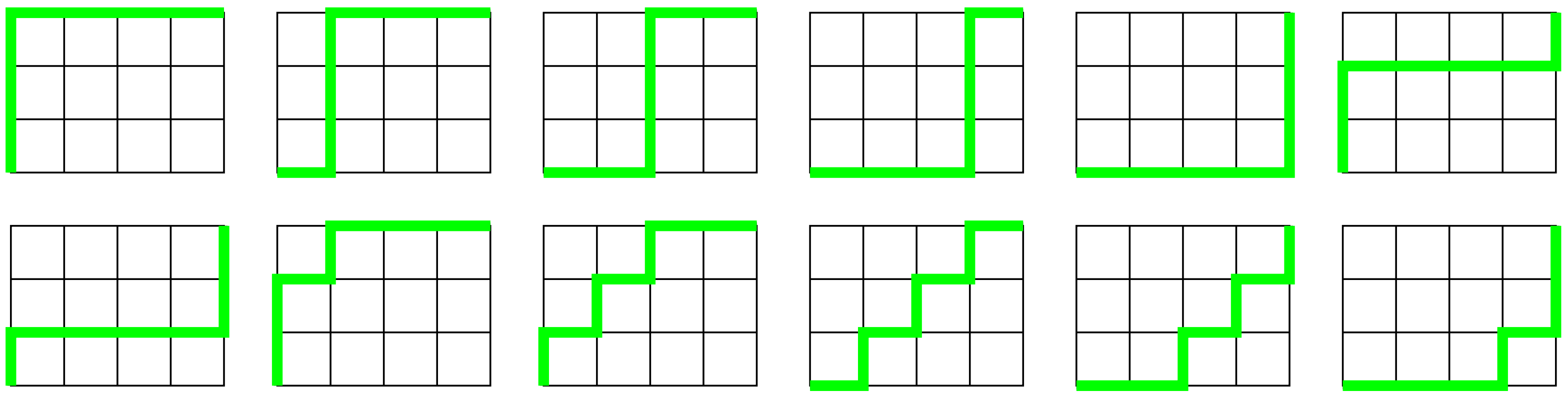
$$\tau(m,n) = \begin{cases} 1, & \text{if } m = 0 \vee n = 0; \\ m+n, & \text{if } (m = 1 \,\&\, n \geq 1) \vee \\ & (n = 1 \,\&\, m \geq 1); \\ 2m+2n-2, & \text{if } m,n \geq 2 \end{cases}$$

Minimal Path Fragment Cover

- **Theorem 1.** Cardinality $\tau(m,n)$ of the minimal path fragment cover of the the $m \times n$ grid is

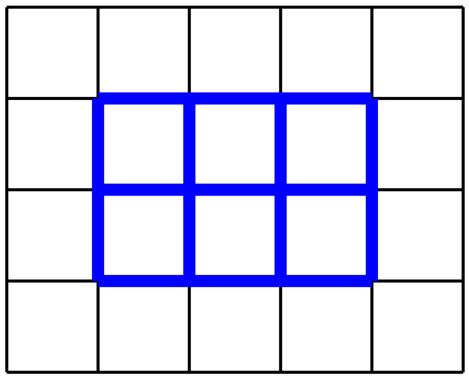
$$\tau(m,n) = \begin{cases} 1, & \text{if } m = 0 \vee n = 0; \\ m+n, & \text{if } (m = 1 \,\&\, n \geq 1) \vee \\ & (n = 1 \,\&\, m \geq 1); \\ 2m+2n-2, & \text{if } m,n \geq 2 \end{cases}$$

- An example of the *trivial* cover:



Describing the Minimal Covers

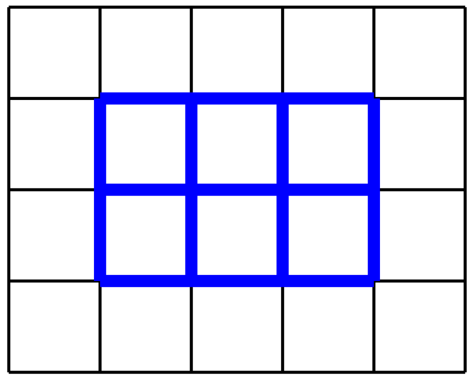
- It turns out that every minimal cover can be described by its behaviour on the set $\mathcal{S}_{m,n}$ of grid segments that have no endpoint on the grid's boundary:



- Clearly, $|\mathcal{S}_{m,n}| = (m-1)(n-2) + (m-2)(n-1)$

Describing the Minimal Covers

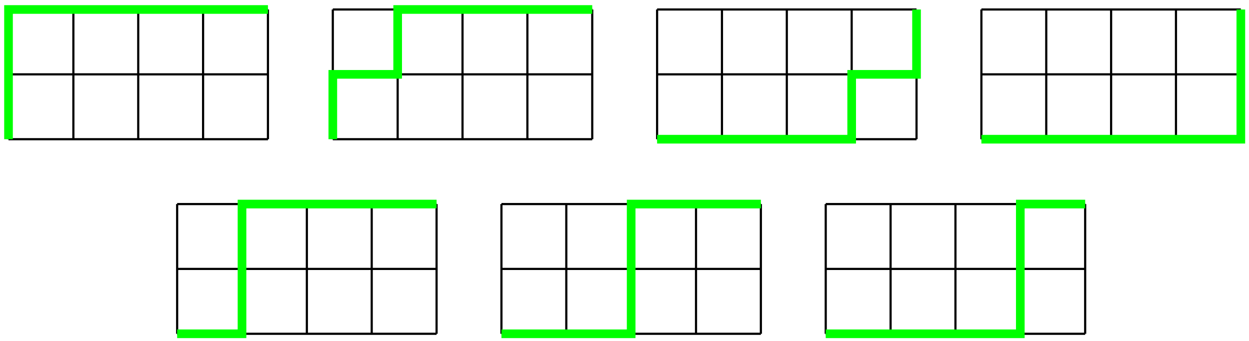
- It turns out that every minimal cover can be described by its behaviour on the set $\mathcal{S}_{m,n}$ of grid segments that have no endpoint on the grid's boundary:



- Clearly, $|\mathcal{S}_{m,n}| = (m-1)(n-2) + (m-2)(n-1)$
- We will establish a one-to-one correspondence between minimal path fragment covers and functions from the set $F_{m,n} = \{f : \mathcal{S}_{m,n} \to \{0,1\}\}$

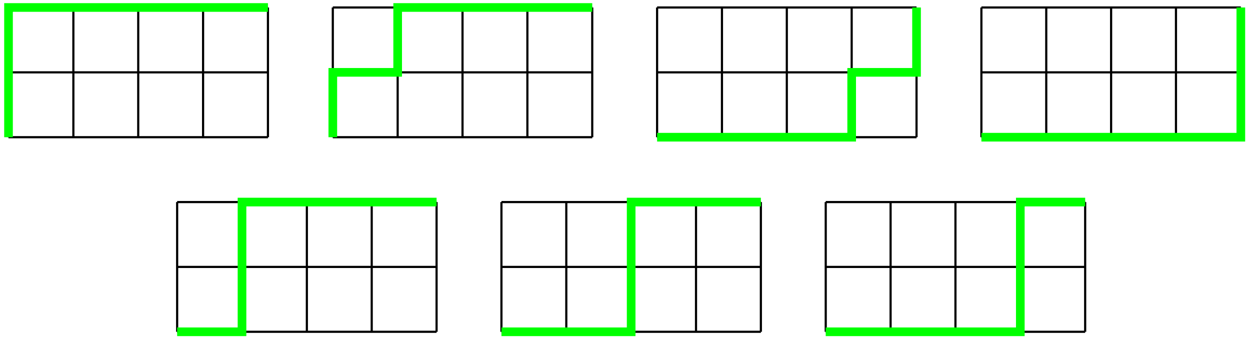
The Algorithm: Initialization

- First we find the paths common to all minimal covers – these are the ones that do not have an edge in $\mathcal{S}_{m,n}$
- For $m = 4$, $n = 2$ we get the following 7 paths:

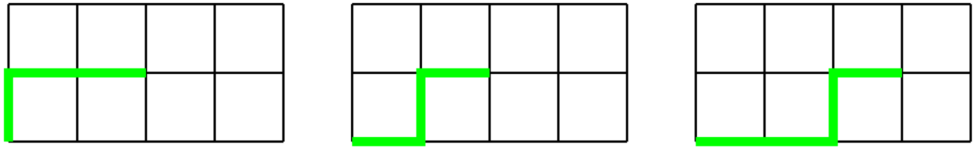


The Algorithm: Initialization

- First we find the paths common to all minimal covers – these are the ones that do not have an edge in $\mathcal{S}_{m,n}$
- For $m = 4$, $n = 2$ we get the following 7 paths:



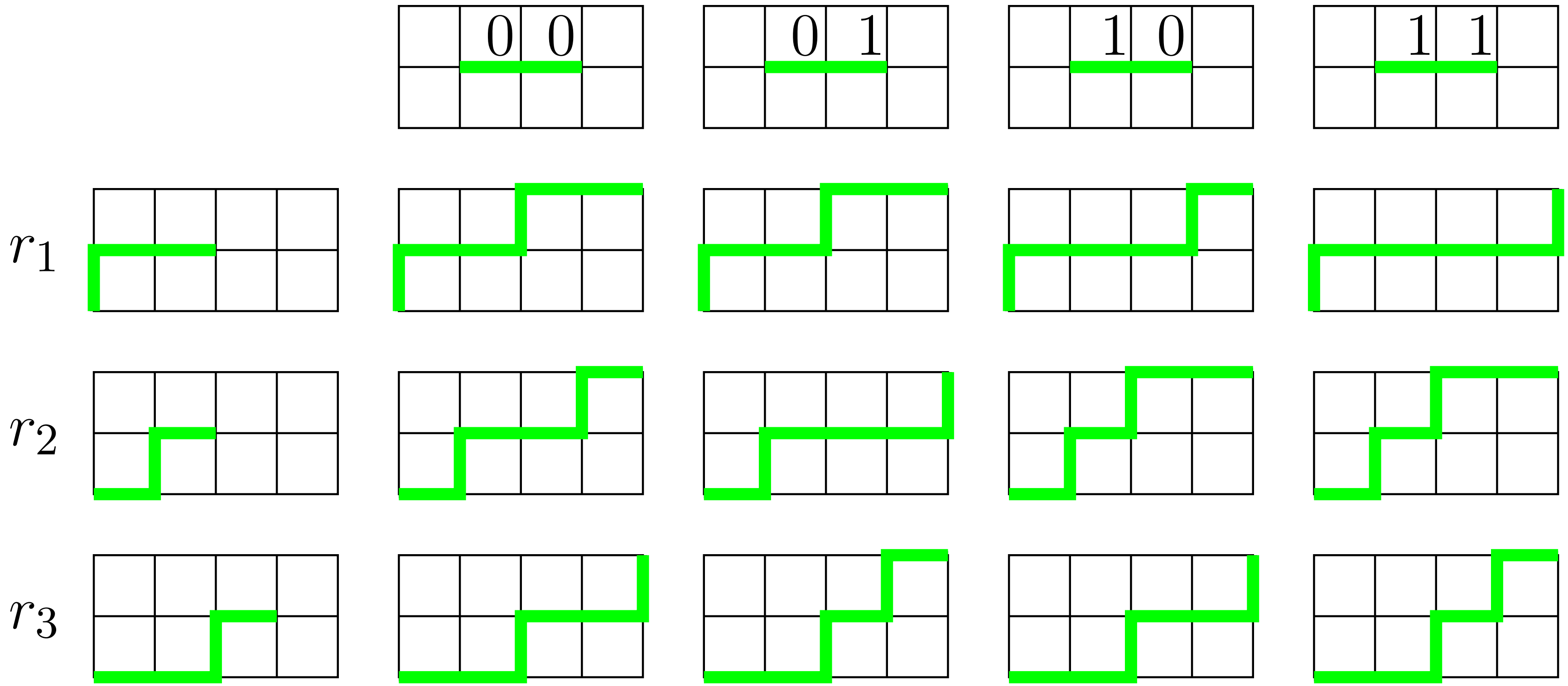
- Next we choose initial parts $r_1, r_2, \ldots, r_k$ of the other paths up to the first segment in $\mathcal{S}_{m,n}$
- For $m = 4$, $n = 2$ we get the following 3 parts:



The Algorithm: Main Cycle

- INPUT: Function $f \in F_{m,n}$ and path parts $r_1, r_2, \ldots, r_k$
- FOR each path part r_i ($i = 1, 2, \ldots, k$) DO
  - WHILE the last segment e of the current path part belongs to $\mathcal{S}_{m,n}$ DO
    - IF $f(e) = 0$ and e has not been traversed before OR $f(e) = 1$ and e has been traversed before
      - THEN the next segment is upwards
      - ELSE the next segment is rightwards
  - Completing the WHILE cycle we have reached the boundary of the grid; complete the path to the upper right corner and OUTPUT it

The Algorithm: Illustration



The Algorithm: Propositions

- The Algorithm produces a minimal path fragment cover and conversely, all the minimal path fragment covers are produced this way
- There are

$$|F_{m,n}| = 2^{|\mathcal{S}_{m,n}|} = 2^{(m-1)(n-2)+(m-2)(n-1)}.$$

minimal path fragment covers of an $m \times n$ grid.

- Time complexity of the Algorithm is $O((m+n)^2)$ and its space complexity is $O(mn)$
    - Note that the same time complexity is needed for the result printout alone, thus the Algorithm is very efficient

Conclusions and Further Work

- We defined a notion for grid covering that provides linear cardinality covers and is thus very scalable
- We presented an efficient algorithm for minimal path fragment generation
- As a next step, measures for minimal cover evaluation must be defined, the best theoretical covers must be found and experimentally tested

Thank You!

Questions?