

COMPETITIVE ONLINE COVERAGE OF GRID ENVIRONMENTS

2 ONLINE ALGORITHMS COMPETITIVELY ANALYSED

KARIM DJEMAI

UNIVERSITÄT HAMBURG

30.07.2023

TEASER

Figure: Not very efficient, but cute.

OUTLINE

1 Introduction

2 Spiral STC

3 Scan STC

4 Analysis of the Algorithms

5 Universal Lower Bound

6 Conclusion

INTRODUCTION

DEFINITIONS

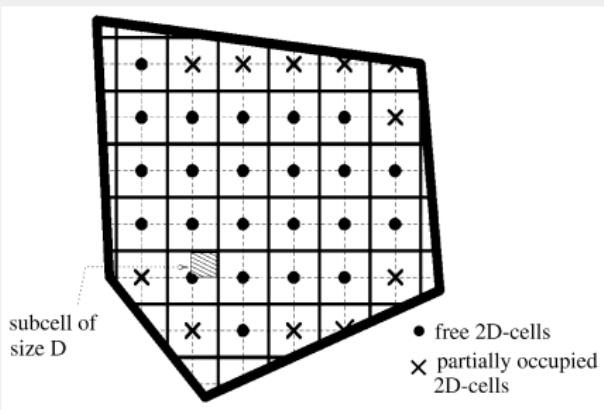


Figure: The work-area grid

- When online: robot has no knowledge about environment
- Let D be the tool size
- Subdivide the work-area into a grid of cells of size D
- Let n be the number of free cells
- Obstacles are allowed

DEFINITIONS

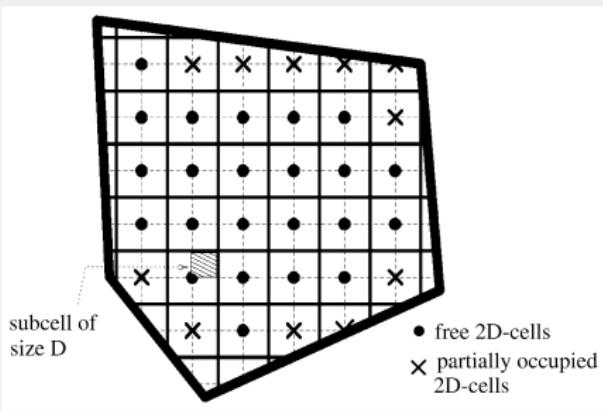
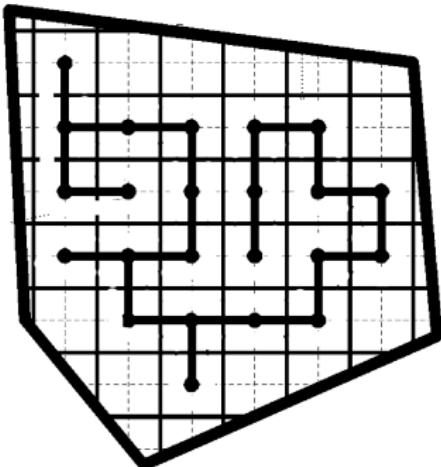


Figure: The work-area grid

- Boundary cells share at least one point with grid boundary
- Let m be number of boundary cells
- Let $l_{\mathcal{A}}$ be length of path, generated by Algorithm \mathcal{A}
- $l_{\mathcal{A}}$ defines cost for \mathcal{A}

TRIVIAL SOLUTION: DFS



- l_{DFS} is always $2nD$
- STC Algorithms are upper bounded by $(n + m)D$
- Both have comp. ratio of $2 - \epsilon$
- In practice $m \ll n$

Figure: A trivial solution: DFS

RELATED WORK

- 1994 Kalyanasundaram and Pruhs: Constructing competitive tours from local information
 - ▶ 16-competitive algorithm: ShortCut
- 2002 Icking et al.: On the Competitive Complexity of Navigation Tasks
 - ▶ 2-competitive algorithm: URC

SPIRAL STC

IDEA

- Constructing spanning trees online
- Grid of coarser 2D cells
- Traversal of spanning tree edges
- Internal covering of 2D cells
- Spiral-like patterns

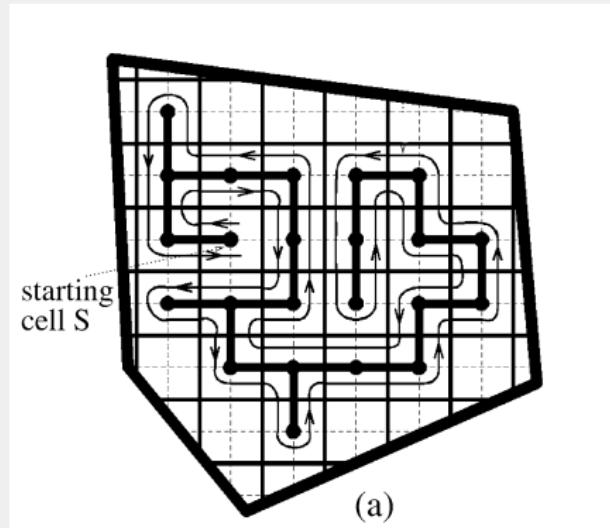


Figure: 2D Spiral STC

2D SPIRAL STC

Recursive Function $STC1(w, x)$:

1. Mark the current cell x as an old cell.
2. While x has a new obstacle-free neighboring cell:
 - 2.1 Scan for first **new neighbor of x in ccw order**, start with parent cell w . Call this y .
 - 2.2 Construct spanning-tree edge from x to y .
 - 2.3 Move to subcell of y **following right-side of spanning tree edges**.
 - 2.4 Execute $STC1(x, y)$.
3. If $x \neq S$, move back from x to a subcell of w **along right-side of spanning tree edges**.

VISUALIZATION

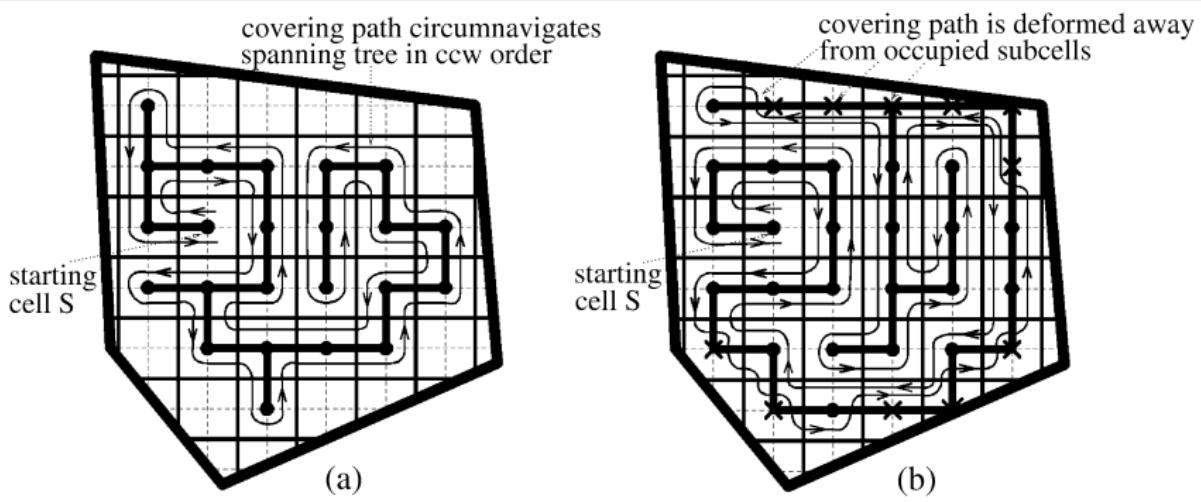


Figure: Visualization of the Spiral STC Algorithms

FULL SPIRAL STC

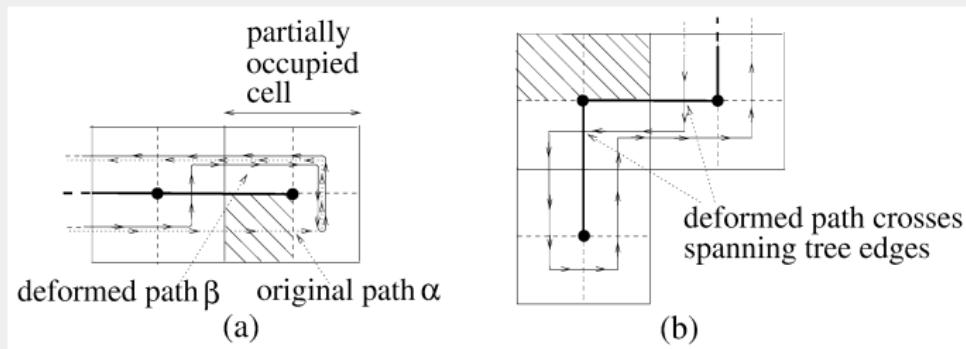


Figure: Deforming the path to circumvent boundaries

- Keep distance of $\frac{D}{2}$ to boundary

VISUALIZATION

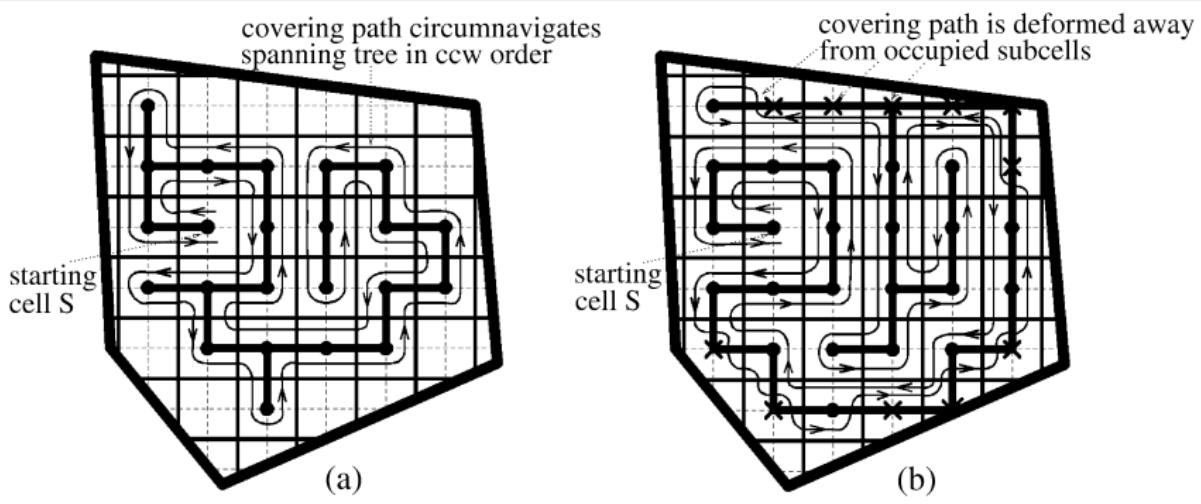


Figure: Visualization of the Spiral STC Algorithms

SCAN STC

VISUALIZATION

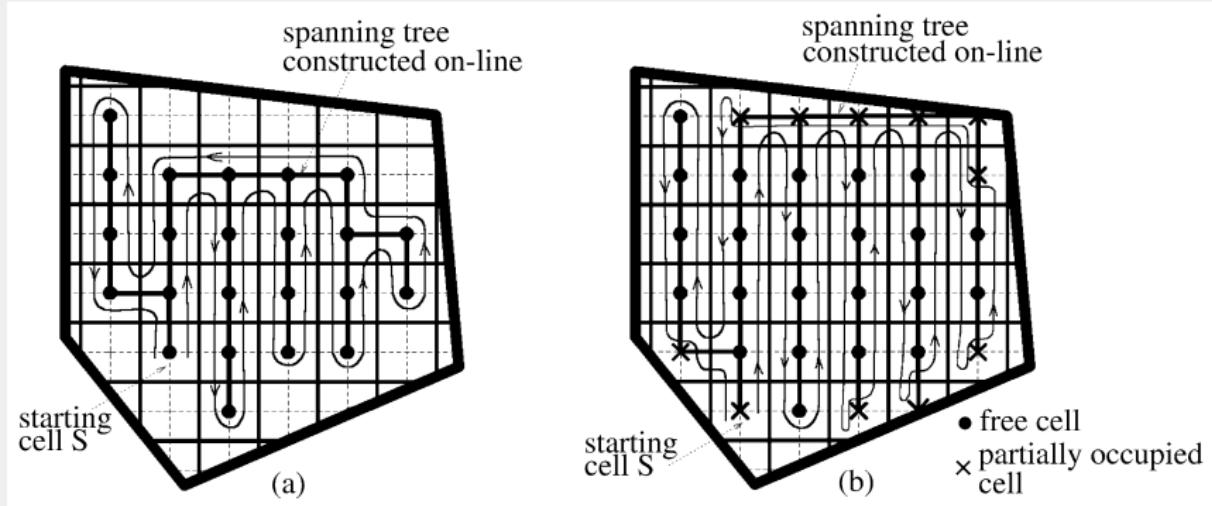
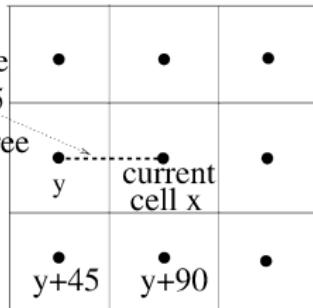


Figure: Visualization of the Scan STC Algorithms

DIFFERENCES TO SPIRAL STC

this edge can be
skipped if $y+45$
and $y+90$ are free



this edge can be
skipped if $y+45$
and $y+90$ are free

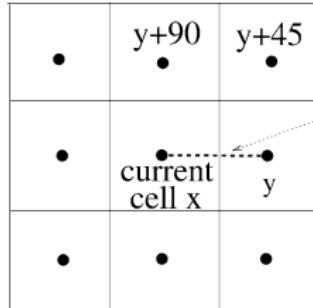


Figure: How Scan STC achieves scan like patterns

- Define $y + 45$ and $y + 90$
- Cells that form 45 or 90 degree angles
- ST edge from x to y in CCW direction
- **skip ST edge construction** if both are free

VISUALIZATION

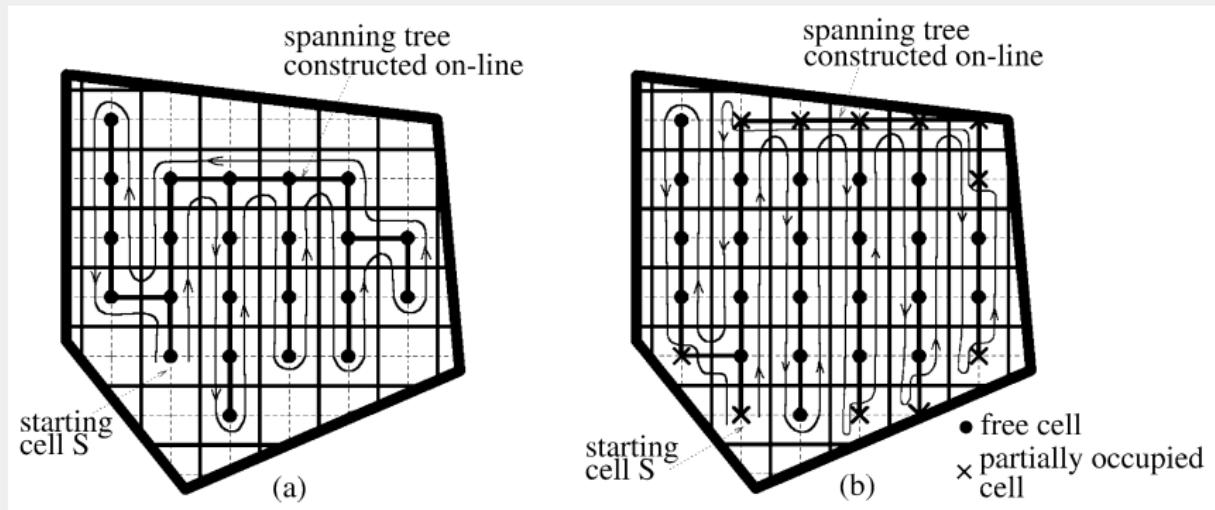


Figure: Visualization of the Scan STC Algorithms

ANALYSIS OF THE ALGORITHMS

BOUNDS

- $l_{2D-STC} \leq nD$ but ignores partially occupied 2D cells
- $l_{Full-STC} \leq (n + m)D$
- STC Algorithms run in $O(n)$ time and have $O(n)$ space complexity

THEOREM 1

Theorem 1

Spiral-STC and Scan-STC cover the work-area grid using a path of total length $l \leq (n + m)D$

THEOREM 1 - PRELIMINARIES

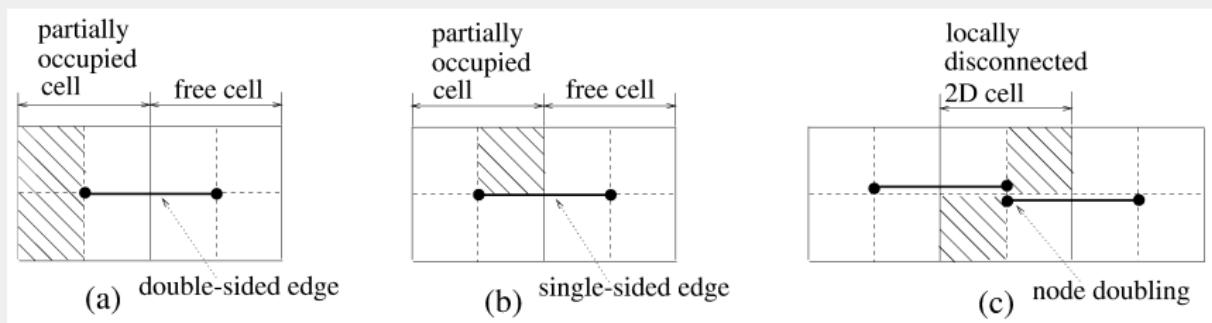


Figure: Single sided edges versus double sided edges.

- Entry Edges, Exit Edges
- Single Sided Edge, Double Sided Edge

THEOREM 1 - REPETITIVE COVERAGES

- When a subcell is covered, then uncovered and later covered again
- Inter-cell repetitive coverage: Between two 2D cells
- Intra-cell repetitive coverage: Within a 2D cell

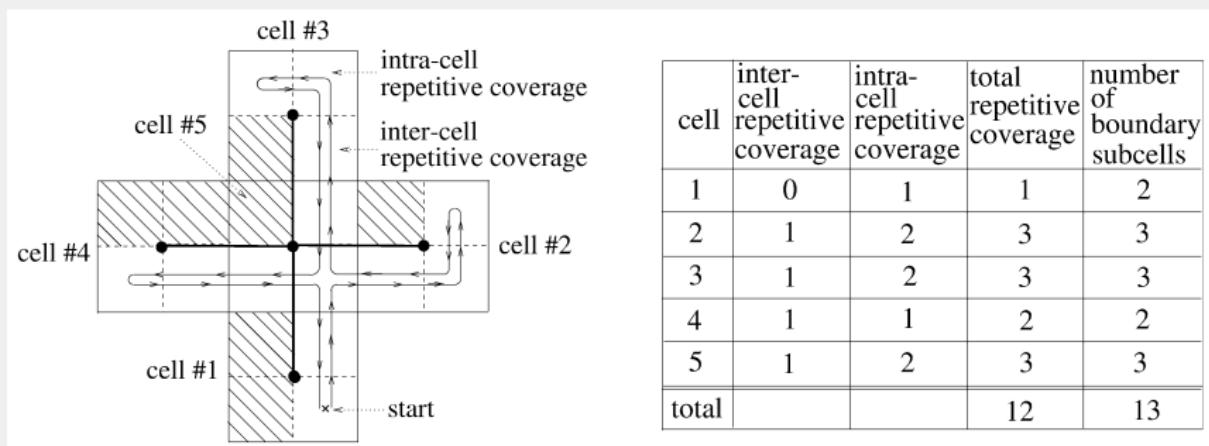
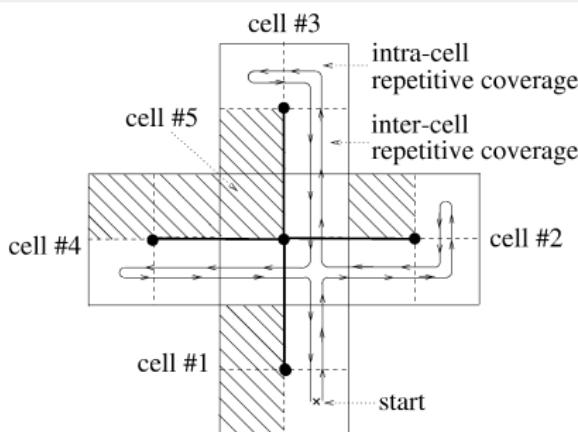


Figure: Repetitive coverage counting of a 2D cell.

THEOREM 1 - PRELIMINARIES

- Single sided edge \implies inter-cell repetitive coverage in cell **from which it exits**
- Counting convention: Such rep. coverage is counted for cell **into which it enters**
- No change in total amount of repetitive coverages



cell	inter-cell repetitive coverage	intra-cell repetitive coverage	total repetitive coverage	number of boundary subcells
1	0	1	1	2
2	1	2	3	3
3	1	2	3	3
4	1	1	2	2
5	1	2	3	3
total			12	13

Figure: Repetitive coverage counting of a 2D cell.

THEOREM 1

Claim 1: Total repetitive coverages bounded by m

- Suffices to show for each 2D cell:
Total amount repetitive coverages \leq Total amount of boundary cells
- Consider 4 cases
 - ▶ 1 free subcell
 - (Double sided entry edge)
 - Single sided entry edge
 - ▶ 2 free subcells
 - Double sided entry edge
 - Single sided entry edge

...

CLAIM 1 - CASE 1



CLAIM 1 - CASE 1



- 1 free subcell
- Only single sided entry edge
- 1 inter-cell repetitive coverage
- 1 boundary cell

CLAIM 1 - CASE 2

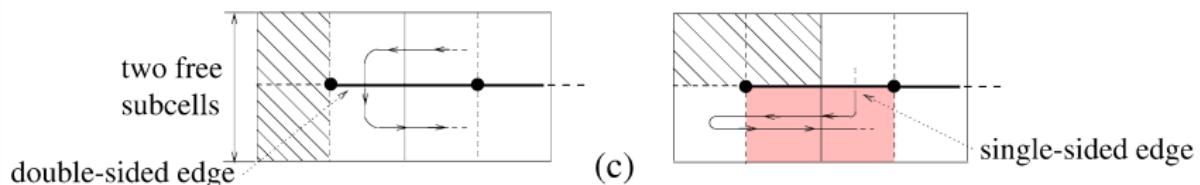


Figure: 2 free subcells, double sided entry edge (left) and single sided entry edge (right).

- 0 repetitive coverages
- 2 boundary cells
- 2 repetitive coverages
- 2 boundary cells

CLAIM 1 - CASE 3

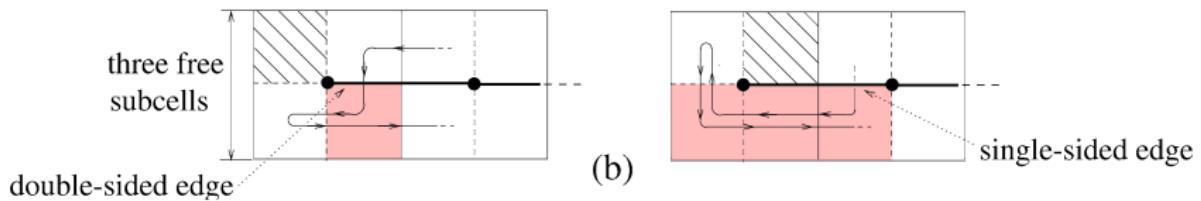


Figure: 3 free subcells, double sided entry edge (left) and single sided entry edge (right).

- 1 repetitive coverages
- 3 boundary cells
- 3 repetitive coverages
- 3 boundary cells

CLAIM 1 - CASE 4

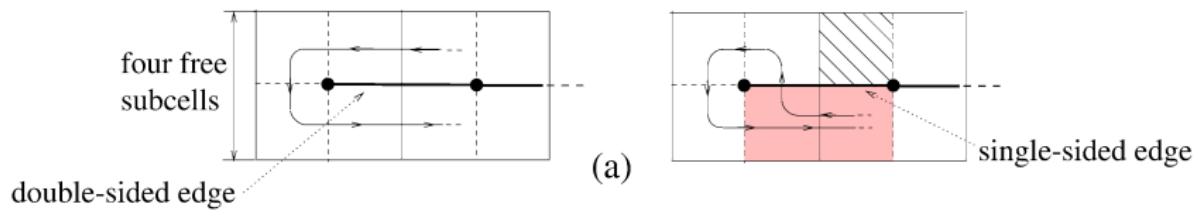


Figure: 4 free subcells, double sided entry edge (left) and single sided entry edge (right).

- No repetitive coverages
- No boundary cells
- 2 repetitive coverages
- 2 boundary cells

CLAIM 1

Proof.

Claim 1: Total repetitive coverages bounded by m

- Suffices to show for each 2D cell:
Total amount repetitive coverages \leq Total amount of boundary cells
- Consider 4 cases
 - ▶ ...
 - ▶ 3 free subcells
 - Double sided entry edge
 - Single sided entry edge
 - ▶ 4 free subcells
 - Double sided entry edge
 - Single sided entry edge



THEOREM 1

Proof.

- Every cell covered at least once: n
- (Claim 1) Number of repetitive coverages is at most m
- $\implies l_{STC} \leq (n + m)D$



OPTIMALITY AND COMPETITIVE RATIO

- Area of grid: $A := n \cdot D^2$
- Area of boundary cells: $\delta A := m \cdot D^2$
- $l_{opt} \geq \frac{A}{D} = \frac{nD^2}{D} = n \cdot D$
- $l_{stc} \leq (n + m) \cdot D$

$$\frac{l_{stc}}{l_{opt}} \leq \frac{(n + m) \cdot D}{n \cdot D} = \dots = 1 + \frac{mD^2}{nD^2} = 1 + \frac{\delta A}{A} = 2 - \epsilon$$

with $\epsilon = (1 - \frac{\delta A}{A})$ and $0 \leq \epsilon \leq 1$ as $A \geq \delta A$ and $A, \delta A \geq 0$

UNIVERSAL LOWER BOUND

UNIVERSAL LOWER BOUND

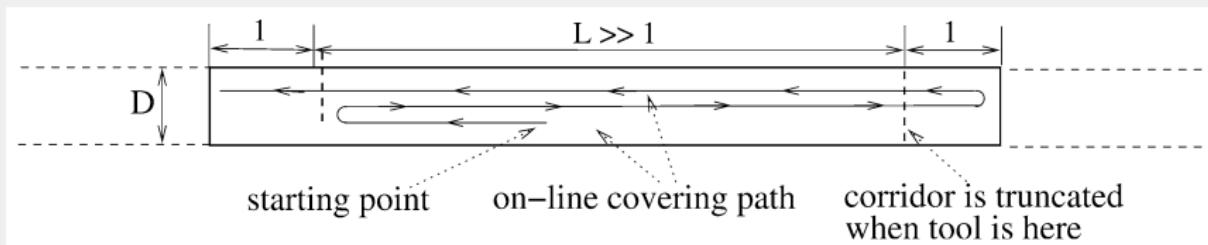
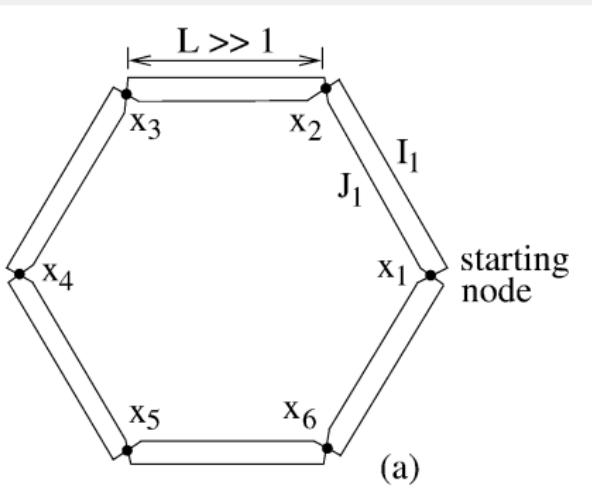


Figure: Trivial corridor example.

- Simple Corridor with width D
- Corridor truncated when L portion is covered
- $l \geq 2L + 3$
- $l_{opt} = L + 2$
- Does not show bounds for tour req.
- Different starting positions

THEOREM 2



- Lower bound for any online algorithm
- W.l.o.g point robot on graph
- Detection range $1 - \delta, \delta > 0$
- Let x_1, \dots, x_k nodes of ring (ccw. order)
- I_i, J_i parallel edges connecting x_i and x_{i+1}
- L length of I_i and J_i

Figure: Double ring environment.

THEOREM 2

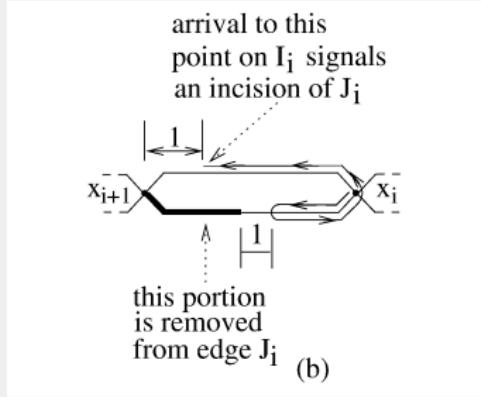


Figure: Incision procedure

- At instant when robot is 1 distance away from x_{i+1} on I_i
- Truncate J_i at distance 1 of already visited part
- $H_i :=$ length of J_i that was covered before incision
- $H := \sum_{i=1}^k H_i$
- Incision special case at starting position

THEOREM 2

- Consider best case scenario (say ccw. order)
- First Loop: $I_1, \dots, I_k + 2 \cdot H_1, \dots, 2 \cdot H_k$
- Remaining part of J_1, \dots, J_k covered in second loop
- Cost on I_1, \dots, I_k is: $2kL$
- Cost on J_1, \dots, J_k is: $2H$ in first round
- And $2(H + k)$ in second round

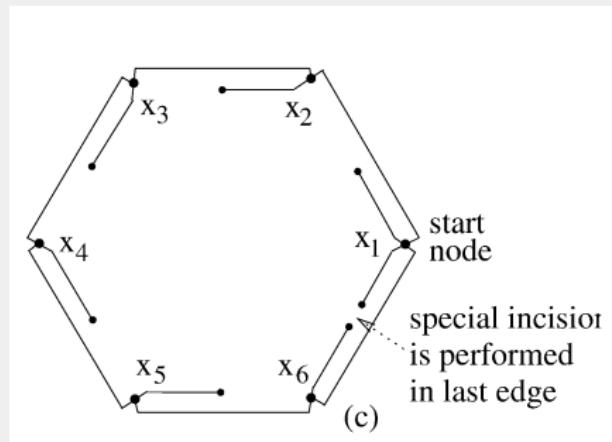


Figure: Incisions in double ring environment.

THEOREM 2

Proof.

- Online Lower Bound: $l_{\mathcal{A}} \geq 2kL + 4H + 2k$
- Offline Upper Bound: $l_{opt} \leq kL + 2(H + k)$
- $\implies l_{\mathcal{A}} \geq (2 - \epsilon)l_{opt}$ where $\epsilon = \frac{2k}{kL+2(H+k)} \ll 1$ as $L \gg 1$

□

CONCLUSION

CONCLUSION

- Spiral STC and Scan STC are $2 - \epsilon$ competitive online algorithms for the coverage problem
- In practice, where $m \ll n$ the algorithms are close to optimal (comp. ratio $2 - \epsilon$ and ϵ approaches 1)
- Online Algorithm can not get better than $2 - \epsilon$ competitive
- No online algorithm is better than Spiral STC and Scan STC in worst case analysis
- But the same holds for DFS

THANKS!

REFERENCES

-  YOAV GABRIELY AND ELON RIMON.
COMPETITIVE ON-LINE COVERAGE OF GRID ENVIRONMENTS BY A MOBILE ROBOT.
Computational Geometry, 24(3):197–224, 2003.
-  CHRISTIAN ICKING, THOMAS KAMPHANS, ROLF KLEIN, AND ELMAR LANGETEPE.
ON THE COMPETITIVE COMPLEXITY OF NAVIGATION TASKS.
In Gregory D. Hager, Henrik Iskov Christensen, Horst Bunke, and Rolf Klein, editors, *Sensor Based Intelligent Robots*, pages 245–258, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
-  BALA KALYANASUNDARAM AND KIRK R. PRUHS.
CONSTRUCTING COMPETITIVE TOURS FROM LOCAL INFORMATION.
Theoretical Computer Science, 130(1):125–138, 1994.

OPTIMALITY AND COMPETITIVE RATIO - DETAILS

- Area of grid: $A := n \cdot D^2$
- Area of boundary cells: $\delta A := m \cdot D^2$
- $l_{opt} \geq \frac{A}{D} = \frac{nD^2}{D} = n \cdot D$
- $l_{stc} \leq (n + m) \cdot D$

$$\begin{aligned}\frac{l_{stc}}{l_{opt}} &\leq \frac{(n + m) \cdot D}{n \cdot D} = \frac{(n + m)D^2}{nD^2} = \frac{nD^2 + mD^2}{nD^2} = \frac{nD^2 \cdot (1 + \frac{mD^2}{nD^2})}{nD^2} \\ &= 1 + \frac{mD^2}{nD^2} = 1 + \frac{\delta A}{A} = 2 - \epsilon\end{aligned}$$

with $\epsilon = (1 - \frac{\delta A}{A})$ and $0 \leq \epsilon \leq 1$ as $A \geq \delta A$ and $A, \delta A \geq 0$