Toward More Efficient Motion Planning with Differential Constraints

Maciej Kalisiak

Final Oral Exam
December 14^{th}, 2007

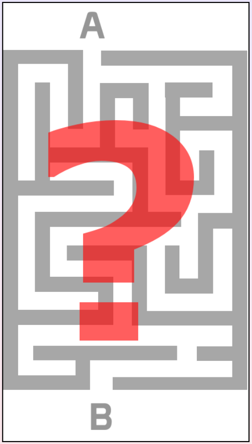
Outline

1. Motion Planning (MP)
   - What is MP?
   - Types of MP problems
   - MP is hard
2. Viability
3. Contributions
   - MP in highly constrained problems
   - MP w/viability filtering
   - Viability-based safety enforcement
4. Conclusion

Outline

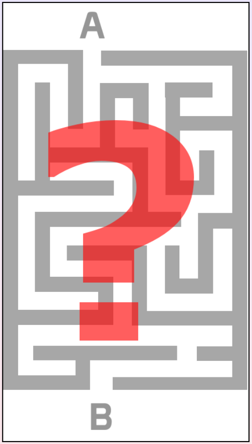
What is Motion Planning (MP)?

- in a nutshell:
  "how to get from A to B?"
- sometimes also:
  "... optimally?"
- example problems



What is Motion Planning (MP)?

- in a nutshell:
  "how to get from A to B?"
- sometimes also:
  "... optimally?"
- example problems

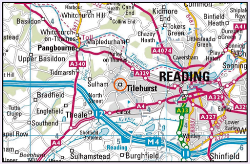


What is Motion Planning (MP)?

- in a nutshell:
  "how to get from A to B?"
- sometimes also:
  "... optimally?"
- example problems

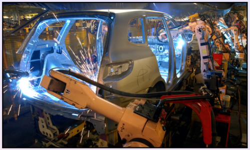
What is Motion Planning (MP)?

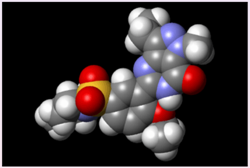
- in a nutshell:
  "how to get from A to B?"
- sometimes also:
  "... optimally?"
- example problems



What is Motion Planning (MP)?

- in a nutshell:
  "how to get from A to B?"
- sometimes also:
  "... optimally?"
- example problems

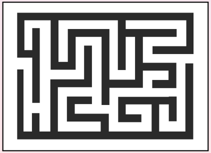


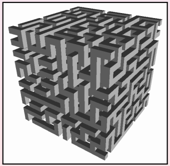


Approach

- solved by converting to dual problem (agent → point)
- complication: often cannot manipulate agent directly

$\mathbf{x} = (x, y, \theta)$





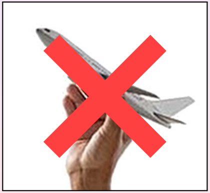
$$\mathcal{X} = x \times y \times \theta$$
$$\mathbf{x} \in \mathcal{X}$$

Approach

- solved by converting to dual problem (agent $\longrightarrow$ point)
- complication: often cannot manipulate agent directly

Approach

- solved by converting to dual problem (agent → point)
- complication: often cannot manipulate agent directly



Types of MP problems

common types:

- kinematic
- nonholonomic
- kinodynamic

e.g., “Piano Mover’s Problem”

Types of MP problems

common types:

- kinematic
- nonholonomic
- kinodynamic

e.g., agents w/rolling contacts

Types of MP problems

common types:

- kinematic
- nonholonomic
- kinodynamic

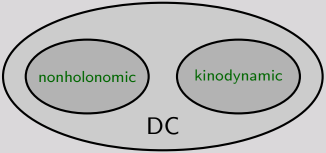
e.g., inertia & balance play big role

Types of MP problems

common types:

- kinematic
- nonholonomic
- kinodynamic

Differential Constraints (DC)

- DC: constraints on q' ($\frac{d}{dt}$ of agent configuration)
- 

- DCs very common, but make MP more difficult

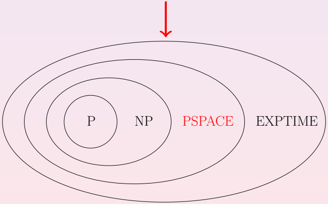
MP is hard

hardness

- Piano Mover's Problem:
  → PSPACE-complete
- MP problems w/DC: at least as hard

why?

- "curse of dimensionality"
- real world problems often high-D
- DCs complicate search space further



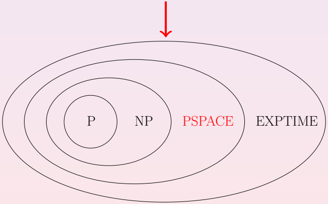
MP is hard

hardness

- Piano Mover's Problem:
  $\longrightarrow$ PSPACE-complete
- MP problems w/DC: at least as hard

why?

- "curse of dimensionality"
- real world problems often high-D
- DCs complicate search space further



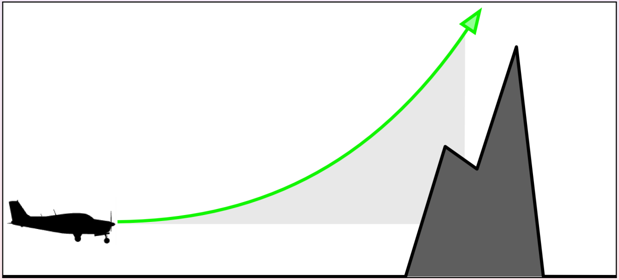
Outline

1. Motion Planning (MP)
   - What is MP?
   - Types of MP problems
   - MP is hard
2. Viability
3. Contributions
   - MP in highly constrained problems
   - MP w/viability filtering
   - Viability-based safety enforcement
4. Conclusion

What is Viability?

"definition"

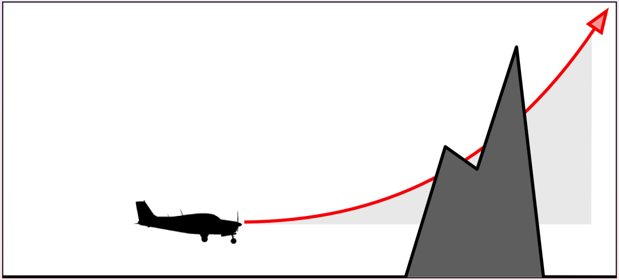
- viable state: ∃ an evasive action
- nonviable state: constraint violation unavoidable



What is Viability?

“definition”

- viable state: $\exists$ an evasive action
- nonviable state: constraint violation unavoidable



What is Viability?

“definition”

- viable state: $\exists$ an evasive action
- nonviable state: constraint violation unavoidable

why of interest?

- crops up in many contexts, useful
- exploited throughout thesis:
    - to expedite MP
    - to aid in user-control

Outline

Overall goal of thesis

- **aim:** explore some novel ideas in MP
- **focus:** improving MP speed
- **grand vision:** MP with motion "macro-primitives"

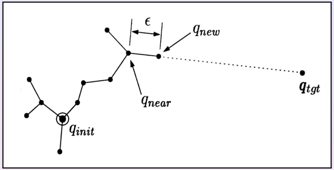
Outline

MP in highly constrained problems

key idea: *"any progress"* is better than *"no progress"*

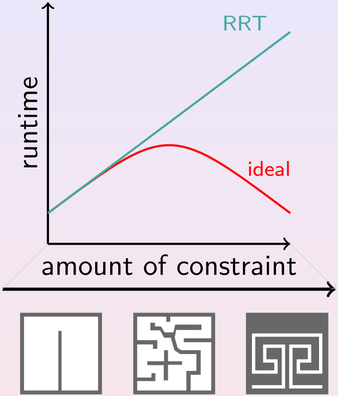
MP in highly constrained problems

- improvement to RRT algorithm
- highly-constrained problems: poor performance
- proposed: RRT-Blossom
- result: big speed ups (>10x)



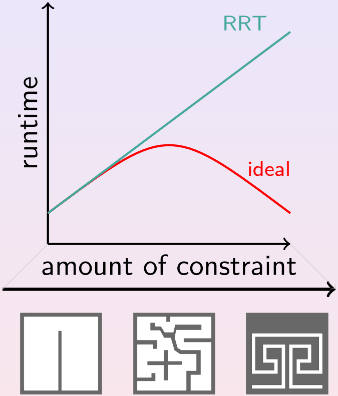
MP in highly constrained problems

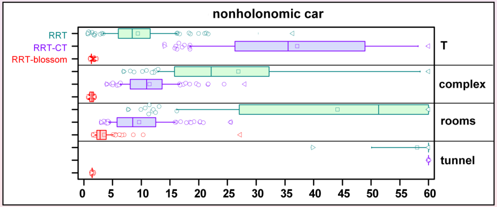
- improvement to RRT algorithm
- highly-constrained problems: poor performance
- proposed: RRT-Blossom
- result: big speed ups (>10x)



MP in highly constrained problems

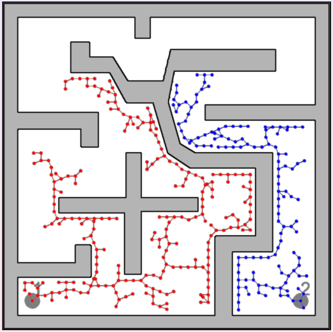
- improvement to RRT algorithm
- highly-constrained problems: poor performance
- proposed: RRT-Blossom
- result: big speed ups (>10x)





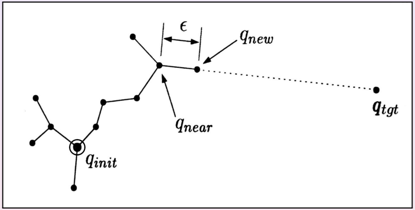
RRT operation review

- grows two trees
  (from q_{init} and q_{goal})
- each tree grows toward q_{tgt}



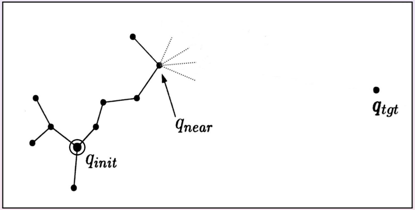
RRT operation review

- grows two trees
  (from q_{init} and q_{goal})
- each tree grows toward q_{tgt}



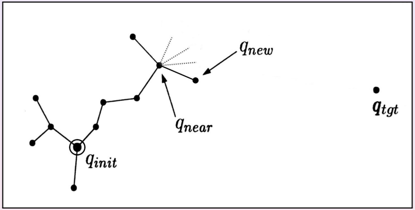
RRT operation review

- grows two trees
  (from q_{init} and q_{goal})
- each tree grows toward q_{tgt}



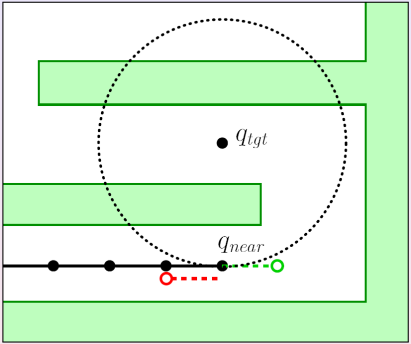
RRT operation review

- grows two trees
  (from q_{init} and q_{goal})
- each tree grows toward q_{tgt}



RRT-Blossom

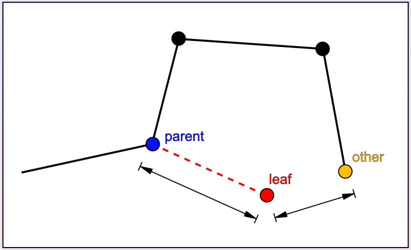
- allow receding edges...
- but not if regressing
- filter with regression test
- bottlenecks



key idea: *"any progress"* is better than *"no progress"*

RRT-Blossom

- allow receding edges...
- but not if regressing
- filter with regression test
- bottlenecks



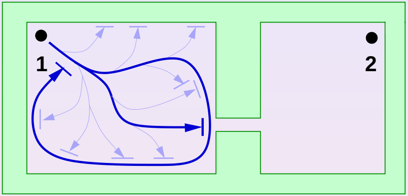
regression if:

$$\exists other \ | \ \rho(parent, leaf) > \rho(other, leaf)$$

key idea: *"any progress"* is better than *"no progress"*

RRT-Blossom

- allow receding edges...
- but not if regressing
- filter with regression test
- bottlenecks



key idea: *"any progress"* is better than *"no progress"*

Outline

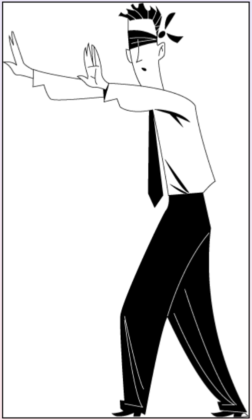
MP w/viability filtering

drawbacks of tree-based MP:

- tactile-only sensing
- search ignores prior attempts

general idea:

- *"work smarter, not harder"*
- add "sight" + "learning" $\longrightarrow$ faster MP



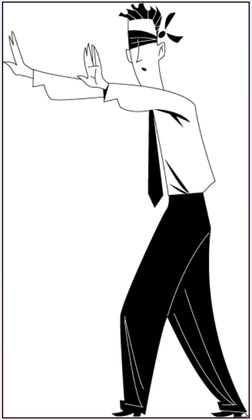
MP w/viability filtering

drawbacks of tree-based MP:

- tactile-only sensing
- search ignores prior attempts

general idea:

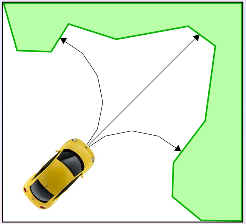
- *"work smarter, not harder"*
- add "sight" + "learning" $\longrightarrow$ faster MP



Key extensions

"sight"

- virtual sensors: distance along path
- yield "locally situated" state



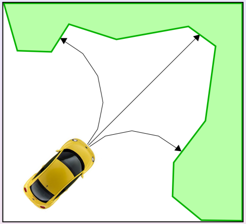
"learning"

- prior trajectories → viability models
- models parametrized using sensors
  - → local models
  - → transferrable
- ideally: bootstrapping

Key extensions

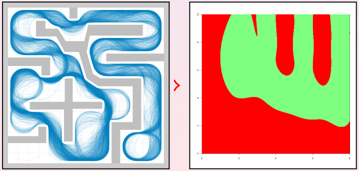
"sight"

- virtual sensors: distance along path
- yield "locally situated" state



"learning"

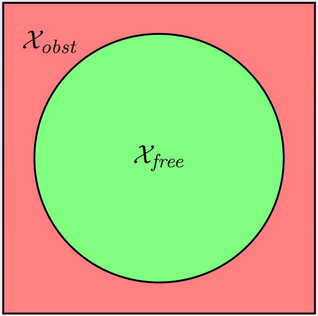
- prior trajectories → viability models
- models parametrized using sensors
  - → local models
  - → transferrable
- ideally: bootstrapping



Exploiting viability

observations

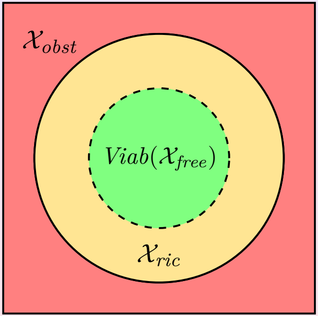
- currently: search in all of $\mathcal{X}_{free}$
- but $\mathcal{X}_{free}$ includes $\mathcal{X}_{ric}$
- x_{goal} usually unreachable from $x \in \mathcal{X}_{ric}$



Exploiting viability

observations

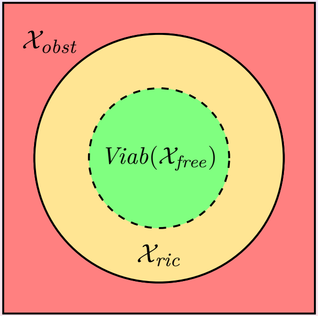
- currently: search in all of $\mathcal{X}_{free}$
- but $\mathcal{X}_{free}$ includes $\mathcal{X}_{ric}$
- x_{goal} usually unreachable from $x \in \mathcal{X}_{ric}$



Exploiting viability

observations

- currently: search in all of $\mathcal{X}_{free}$
- but $\mathcal{X}_{free}$ includes $\mathcal{X}_{ric}$
- x_{goal} usually unreachable from $x \in \mathcal{X}_{ric}$



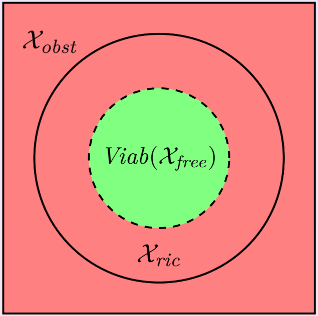
Exploiting viability

observations

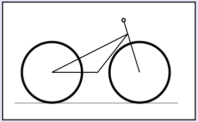
- currently: search in all of $\mathcal{X}_{free}$
- but $\mathcal{X}_{free}$ includes $\mathcal{X}_{ric}$
- x_{goal} usually unreachable from $x \in \mathcal{X}_{ric}$

$\Rightarrow$ **avoid futile searching!**

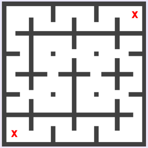
- model agent viability
- keep MP search within $Viab(\mathcal{X}_{free})$
- observed: speed-up of up to 10x



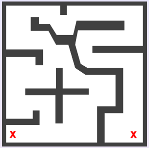
Results: model transfer



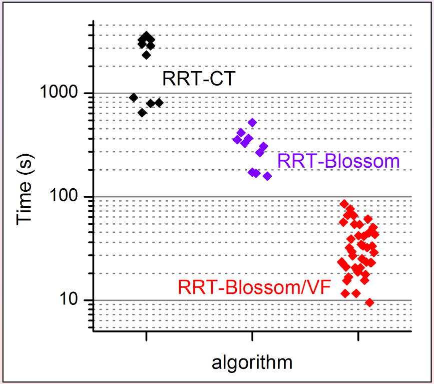
agent



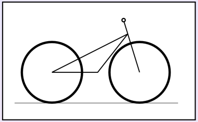
problem posed



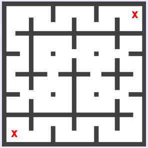
model trained on



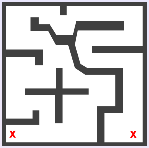
Results: model transfer



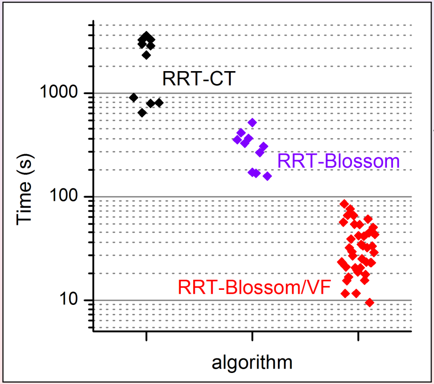
agent

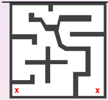
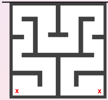
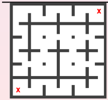


problem posed

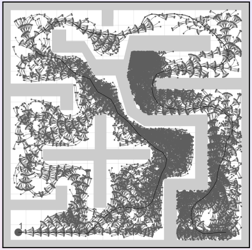
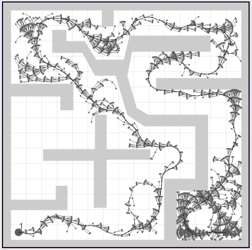
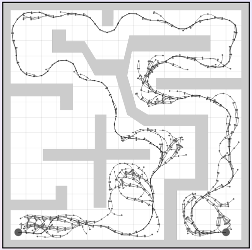


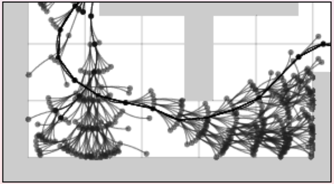
model trained on



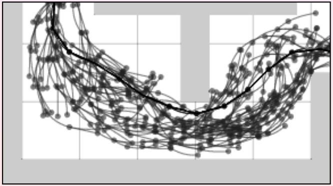
problem	algo.	runtimes
	RRT-CT	371.5s
	RRT-Blossom	21.0s
	RRT-Blossom-VF	5.6s
	RRT-CT	209.9s
	RRT-Blossom	13.5s
	RRT-Blossom-VF	3.6s
	RRT-CT	2148.6s
	RRT-Blossom	305.7s
	RRT-Blossom-VF	34.3s

Results: tree structure

RRT-CT	RRT-Blossom	RRT-Blossom w/VF
		



no filtering



viability filtering

Outline

Viability-based safety enforcement

- ⇒ assisted control:
  - inherently useful
  - facilitates obtaining user-demonstrated training data
  - helpful in user-assisted MP (future work)

- **key idea:** viability more reliable for detecting imminent danger

Viability-based safety enforcement

- ⇒ assisted control:
  - inherently useful
  - facilitates obtaining user-demonstrated training data
  - helpful in user-assisted MP (future work)

- **key idea:** viability more reliable for detecting imminent danger

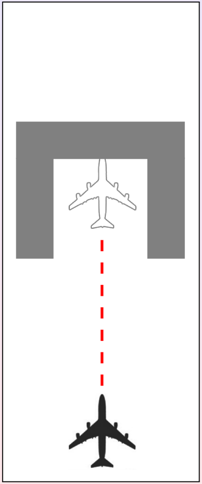
Collision avoidance

typical (collision-based)

- based on predictive lookahead (T_h seconds)
- weakness: T_h is finite
  - T_h may be too small
  - safety↑ as $T_h \to \infty$

better: viability-based safety enforcement

- only a minimal lookahead needed
- longer lookaheads: milder corrections



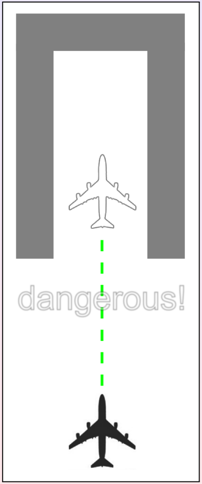
Collision avoidance

typical (collision-based)

- based on predictive lookahead (T_h seconds)
- weakness: T_h is finite
  - T_h may be too small
  - safety↑ as $T_h \to \infty$

better: viability-based safety enforcement

- only a minimal lookahead needed
- longer lookaheads: milder corrections



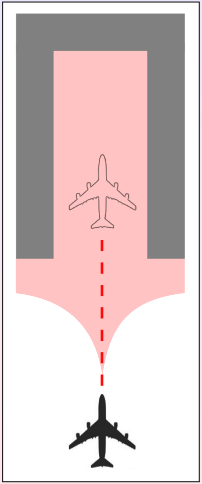
Collision avoidance

typical (collision-based)

- based on predictive lookahead (T_h seconds)
- weakness: T_h is finite
  - T_h may be too small
  - safety↑ as $T_h \to \infty$

better: viability-based safety enforcement

- only a minimal lookahead needed
- longer lookaheads: milder corrections



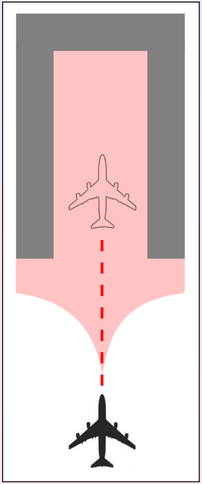
Collision avoidance

typical (collision-based)

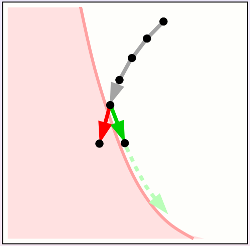
- based on predictive lookahead (T_h seconds)
- weakness: T_h is finite
  - T_h may be too small
  - safety↑ as $T_h \to \infty$

better: viability-based safety enforcement

- only a minimal lookahead needed
- longer lookaheads: milder corrections



Collision avoidance



typical (collision-based)

- based on predictive lookahead (T_h seconds)
- weakness: T_h is finite
  - T_h may be too small
  - safety↑ as $T_h \to \infty$

better: viability-based safety enforcement

- only a minimal lookahead needed
- longer lookaheads: milder corrections

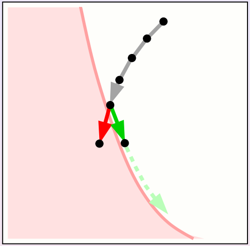
Collision avoidance

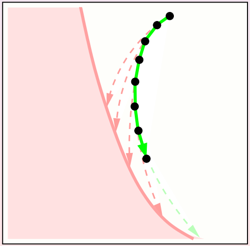
typical (collision-based)

- based on predictive lookahead (T_h seconds)
- weakness: T_h is finite
  - T_h may be too small
  - safety↑ as $T_h \to \infty$

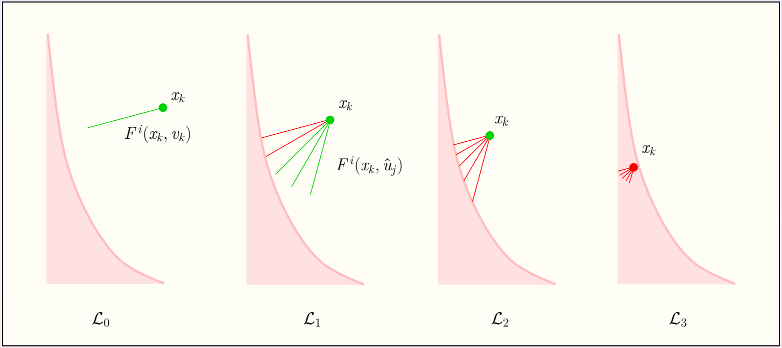
better: viability-based safety enforcement

- only a minimal lookahead needed
- longer lookaheads: milder corrections

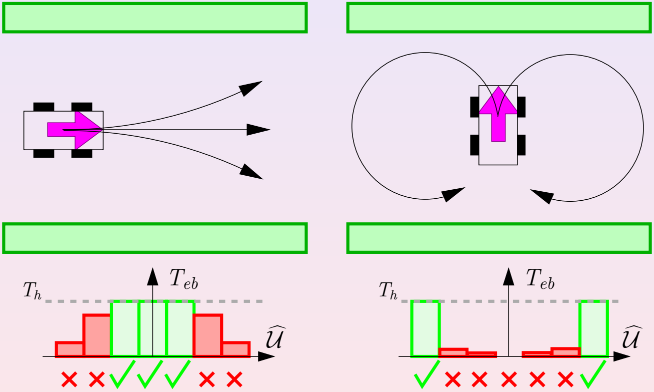




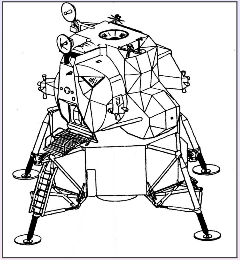
Operation

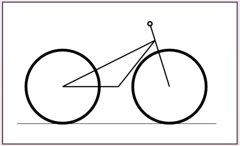

x_k
$F^i(x_k, v_k)$
$\mathcal{L}_0$
x_k
$F^i(x_k, \hat{u}_j)$
$\mathcal{L}_1$
x_k
$\mathcal{L}_2$
x_k
$\mathcal{L}_3$

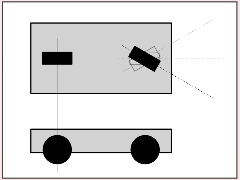
Viability of control actions



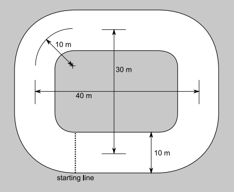
agents

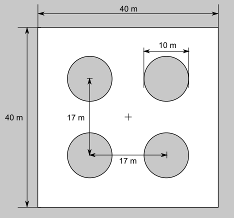


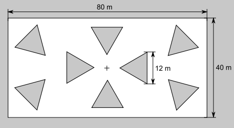




environments

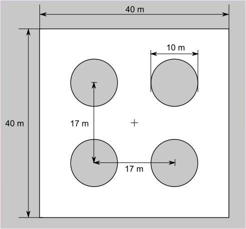






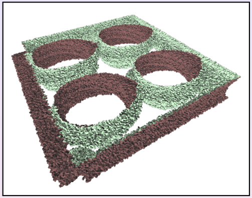
Results

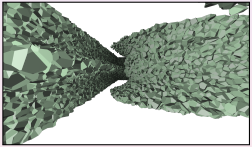
environment

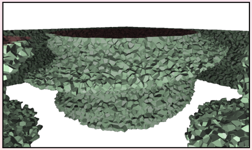


→

$Viab(\mathcal{X}_{free})$ model

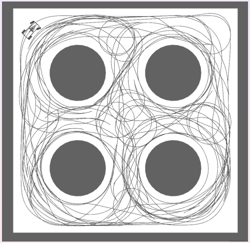






→

enforcement



Conclusion

- **contributions**
  - better handling of constrained environments in RRT
  - more efficient MP by narrowing search to $Viab(\mathcal{X}_{free})$
  - more robust threat avoidance in computer-assisted control

- **future work:**
  - learning appropriate *actions* from motion data
  - MP w/motion “macro primitives”
  - evaluate viability filtering with other MPs
  - *local* viability models for safety enforcement
  - (near-)optimal solutions for MP w/DC
  - human-derived motion data (e.g., style content)
  - human-guided MP: selection of style or topology

Conclusion

- **contributions**
  - better handling of constrained environments in RRT
  - more efficient MP by narrowing search to $Viab(\mathcal{X}_{free})$
  - more robust threat avoidance in computer-assisted control
- **future work:**
  - learning appropriate *actions* from motion data
  - MP w/motion "macro primitives"
  - evaluate viability filtering with other MPs
  - *local* viability models for safety enforcement
  - (near-)optimal solutions for MP w/DC
  - human-derived motion data (e.g., style content)
  - human-guided MP: selection of style or topology