Challenges of Humanoid Motion Planning for Navigation

Jasper Güldenstein

University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences
Department of Informatics

Technical Aspects of Multimodal Systems

05. November 2018

Outline

Introduction and Motivation

- humanoid robots are mobile robots
- approaches for traditional mobile robots (with wheels) only work for flat terrain
- humanoid robots can step on or over obstacles
- navigation space is limited by balancing criteria
- environment is dynamic $\rightarrow$ dynamic replanning is required

Dynamic Window Approach [FBT]

- global path has been computed by a standard pathfinding algorithm (A* etc.)
- motion capabilities of the robot are known
  - velocity limits (v, ω)
  - acceleration limits $(\dot{v}, \dot{\omega})$
- current position $x(t), y(t), \theta(t)$ and velocity $v(t), \omega(t)$ is known

future position can be calculated

$$x(t_n) = x(t_0) + \int_{t_0}^{t_n} v(t) \cdot cos\left(\theta(t)\right) dt \tag{1}$$

$$y(t_n) = y(t_0) + \int_{t_0}^{t_n} v(t) \cdot sin\left(\theta(t)\right) dt \tag{2}$$

Dynamic Window Approach [FBT]

velocities depend on current velocities $v(t), \omega(t)$ and accelerations $\dot{v}(\hat{t}), \dot{\omega}(\hat{t})$

$$x(t_n) = y(t_0) + \int_{t_0}^{t_n} \left(v(t_0) + \int_{t_0}^{t} \dot{v}(\hat{t}) d\hat{t} \right) \cdot cos \left(\theta(t) + \int_{t_0}^{t} \left(\omega(t_0) + \int_{t_0}^{\hat{t}} \dot{\omega}(\tilde{t}) d\tilde{t} \right) d\hat{t} \right) dt \tag{3}$$

velocities depend on current velocities $v(t), \omega(t)$ and accelerations $\dot{v}(\hat{t}), \dot{\omega}(\hat{t})$

$$x(t_n) = y(t_0) + \int_{t_0}^{t_n} \left(v(t_0) + \int_{t_0}^{t} \dot{v}(\hat{t}) d\hat{t} \right) \cdot cos \left(\theta(t) + \int_{t_0}^{t} \left(\omega(t_0) + \int_{t_0}^{\hat{t}} \dot{\omega}(\tilde{t}) d\tilde{t} \right) d\hat{t} \right) dt \tag{3}$$

$$y(t_n) = y(t_0) + \int_{t_0}^{t_n} \left(v(t_0) + \int_{t_0}^{t} \dot{v}(\hat{t}) d\hat{t} \right) \cdot sin \left(\theta(t) + \int_{t_0}^{t} \left(\omega(t_0) + \int_{t_0}^{\hat{t}} \dot{\omega}(\tilde{t}) d\tilde{t} \right) d\hat{t} \right) dt \tag{4}$$

Dynamic Window Approach [FBT]

- discrete simulation of possible trajectories
- evaluation of trajectories based on
  - target heading
  - clearance
  - velocity
  - distance to path
- finer granularity leads to
  - closer to optimal solution
  - computationally more expensive
- longer simulation time leads to
  - minimum should be the (maximum) deceleration time
  - computationally more expensive
  - longer reaction time to changes in the environment

Dynamic Window Approach [FBT]

Video Break [Tar]

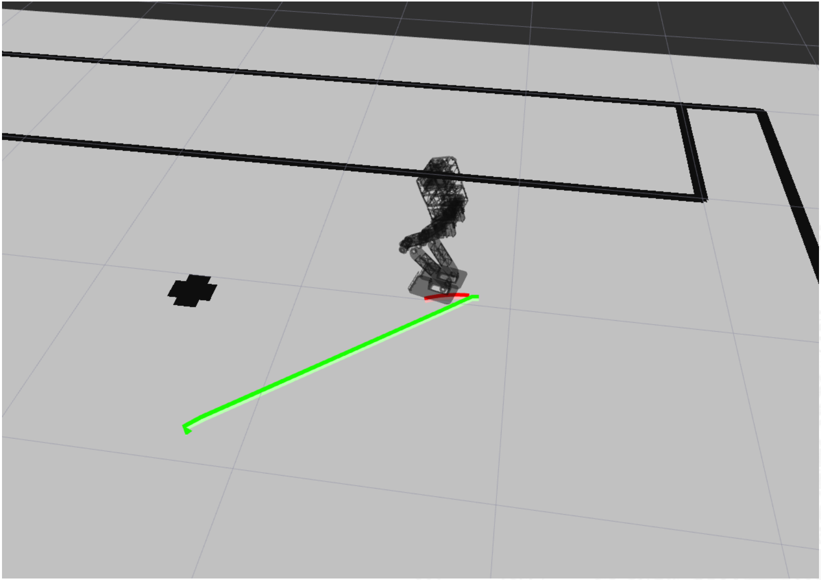


Figure 1: Visualization of local and global plan for a humanoid robot

Dynamic Window Approach [FBT]

Video Break 2

Dynamic Window Approach [FBT]

Conclusion

- quick reactions to changes in the environment (faster than global replanning)
- computationally inexpensive
- collision free trajectory
- used in many real world robots
- planning restricted to x, y, θ

Dynamic Footstep Planner [GHB]

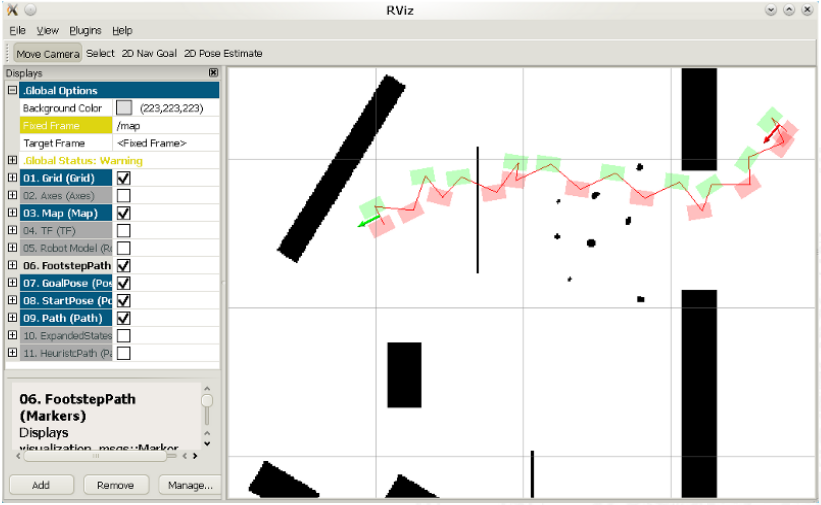
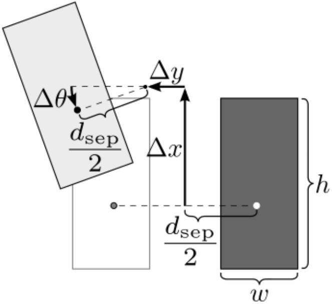


Figure 2: Visualization of planned footsteps between and above obstacles [GH]

Dynamic Footstep Planner [GHB]

- control feet position instead of velocities and accelerations
- walking engine needs to support this

Dynamic Footstep Planner [GHB]



Parameter	Value / Range
d_{sep}	9.5 cm
w	9.0 cm
h	16.0 cm
Δ x	[−4.0, 4.0] cm
Δ y	[−1.0, 4.0] cm
Δ θ	[−2.9, 20.0]°

Figure 3: possible parameters of the foot placement vector [GHB]

Dynamic Footstep Planner [GHB]

State is modeled as position of supporting foot:

$$s = (x, y, \theta) \tag{5}$$

State transition is modeled as taking one step:

$$a = (\Delta x, \Delta y, \Delta\theta) \tag{6}$$

Cost of state transition is modeled as:

$$c(s, s') = \|(x, y), (x', y')\| + k + d(s') \tag{7}$$

Where $\|(x, y), (x', y')\|$ is the distance travelled, k is a constant cost to minimize steps taken and $d(s')$ is distance to closest obstacle

Dynamic Footstep Planner [GHB]

Efficient collision checking between foot and environment is necessary

Dynamic Footstep Planner [GHB]

Heuristic for path evaluation was chosen statistically

$$h_1 = \omega_1 \|(x, y), (x_{start}, y_{start})\| + kS_1(s, s_{start}) \tag{8}$$

$$h_2 = \omega_1 \|(x, y), (x_{start}, y_{start})\| + kS_1(s, s_{start}) + \omega_2 |\theta - \theta_{start}| \tag{9}$$

$$h_3 = \omega_1 \mathcal{D}(s, s_{start}) + kS_2(s, s_{start}) \tag{10}$$

With ω_1, ω_2 as scaling factors, S_1 as expected number of footsteps based on distance, k as constant cost per step, $\mathcal{D}$ as length of 2D Path and S_2 as expected number of footsteps along $\mathcal{D}$

Dynamic Footstep Planner [GHB]

Heuristic for path evaluation was chosen statistically

$$h_1 = \omega_1 \|(x, y), (x_{start}, y_{start})\| + kS_1(s, s_{start}) \quad (8)$$

$$h_2 = \omega_1 \|(x, y), (x_{start}, y_{start})\| + kS_1(s, s_{start}) + \omega_2 |\theta - \theta_{start}| \quad (9)$$

$$h_3 = \omega_1 \mathcal{D}(s, s_{start}) + kS_2(s, s_{start}) \quad (10)$$

With ω_1, ω_2 as scaling factors, S_1 as expected number of footsteps based on distance, k as constant cost per step, $\mathcal{D}$ as length of 2D Path and S_2 as expected number of footsteps along $\mathcal{D}$

h_3 was chosen through statistical evaluation

Dynamic Footstep Planner [GHB]

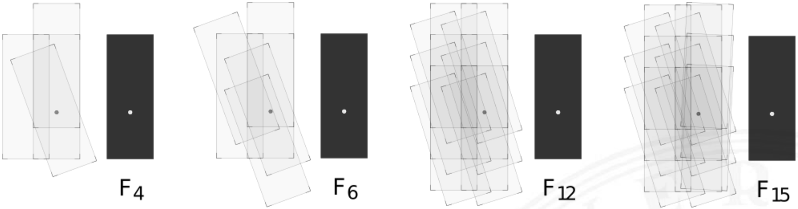


Figure 4: sets of possible foot placements [GHB]

Dynamic Footstep Planner [GHB]

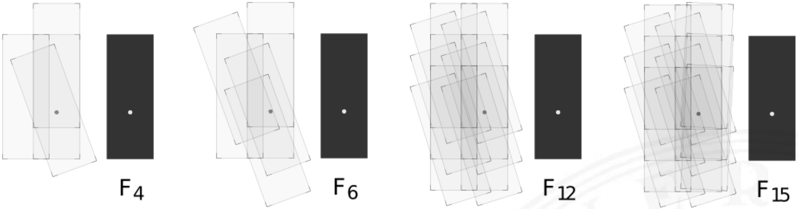


Figure 4: sets of possible foot placements [GHB]

F_{12} was chosen through statistical analysis

Dynamic Footstep Planner [GHB]

- foot slippage and inaccurate joints cause open loop execution to be infeasible
- adaptation of next step to incorporate information about current position
- efficient replanning using D* when derivation from plan is too great

Dynamic Footstep Planner [GHB]

- stepping over planar or near planar objects is achieved
- stepping onto objects is not achieved
- cluttered environments create local minima [HDLB]

Footstep Planning for 3D Environments [KDF+]

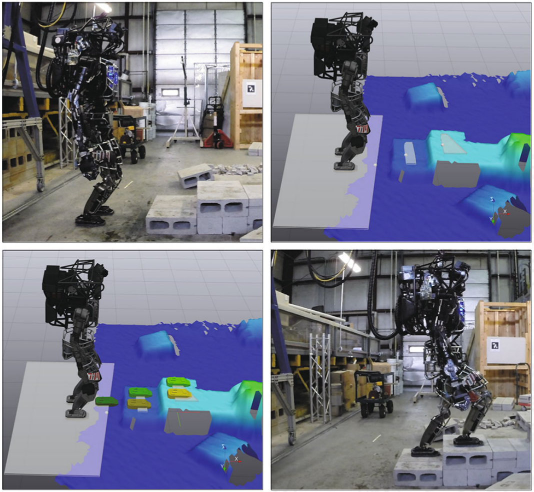


Figure 5: Planned Footsteps for climbing Stairs [KDF+]

Footstep Planning for 3D Environments [KDF+]

- sets of footsteps limit available position
- continuous searching for complete area is
- subdivide into convex regions with convex cost function
- easily solvable by optimization

Footstep Planning for 3D Environments [KDF+]

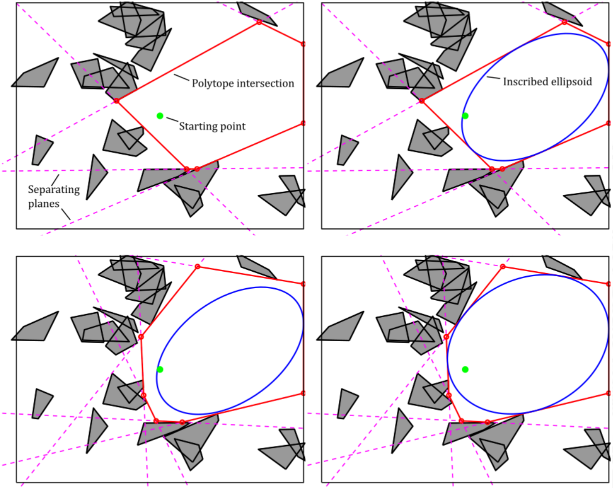


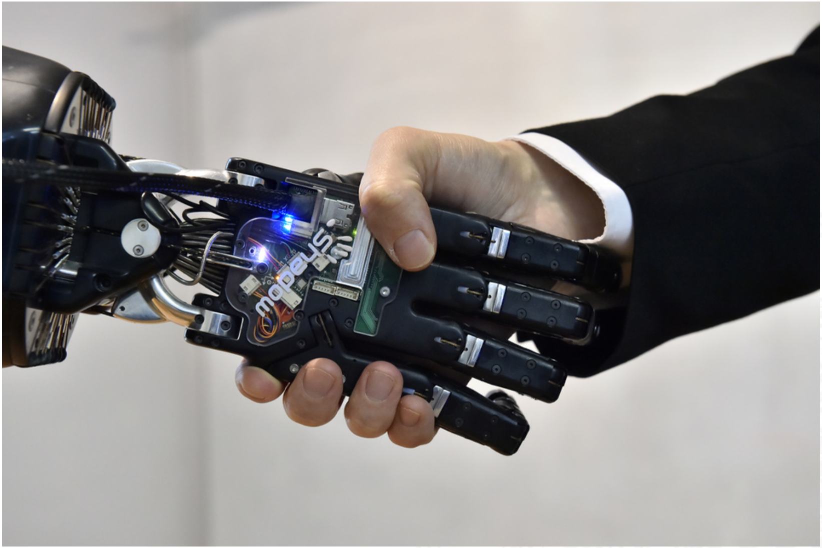
Figure 6: Convex region calculation [KDF+]

Footstep Planning for 3D Environments [KDF+]

- trajectory of center of mass and center of pressure needs to be planned
- reaching a stable pose after the last step
- placing feet on sloped terrain is not modeled

Human comparison

Video [Cel]



[noa]

Comparison

	DWA	Footstep Planning
dimensionality of navigation	3	3-6
computational difficulty	low	high
nav. on flat ground	yes	yes
nav. over obstacles	no	yes
stepping on obstacles	no	yes
sloped ground	(yes)	(yes)

Thank you for your attention.
Time for your questions.

References

[Cel] CELL PRESS:
Gaze and Gait When Walking in Natural Terrain/ Curr. Biol., Apr. 12, 2018 (Vol. 28, Issue 8).
https://www.youtube.com/watch?v=L90OH61-33c

[FBT] FOX, D. ; BURGARD, W. ; THRUN, S.:
The dynamic window approach to collision avoidance.
4, Nr. 1, S. 23–33.
http://dx.doi.org/10.1109/100.580977. –
DOI 10.1109/100.580977. –
ISSN 1070–9932

[GH] GARIMORT, Johannes ; HORNUNG, Armin:
footstep_planner - ROS Wiki.
http://wiki.ros.org/footstep_planner

References (cont.)

[GHB] Garimort, Johannes ; Hornung, Armin ; Bennewitz, Maren:
Humanoid navigation with dynamic footstep plans.
In: *2011 IEEE International Conference on Robotics and Automation*, IEEE. –
ISBN 978–1–61284–386–5, 3982–3987

[HDLB] Hornung, Armin ; Dornbush, Andrew ; Likhachev, Maxim ; Bennewitz, Maren:
Anytime search-based footstep planning with suboptimality bounds.
In: *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, IEEE. –
ISBN 978–1–4673–1369–8, 674–679

References (cont.)

[KDF+] Kuindersma, Scott ; Deits, Robin ; Fallon, Maurice ; Valenzuela, Andrés ; Dai, Hongkai ; Permenter, Frank ; Koolen, Twan ; Marion, Pat ; Tedrake, Russ:
Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot.
40, Nr. 3, 429–455.
http://dx.doi.org/10.1007/s10514-015-9479-3. –
DOI 10.1007/s10514–015–9479–3. –
ISSN 0929–5593, 1573–7527

[noa] *Get involved with the best in UK innovation.*
https://www.gov.uk/government/news/get-involved-with-the-best-in-uk-innovation

References (cont.)

[Tar] Tarik Kelestemur:
TurtleBot 2 Autonomous Navigation and Obstacle-avoidance,
https://www.youtube.com/watch?v=0eDFSXPnh2I