Application of Interval Analysis : safe path planning

Sébastien Lengagne, Nacim Ramdani, Philippe Fraisse

DEMAR / LIRMM / INRIA

Small Workshop on Interval Methods 2008 19-20 juin 2008

Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier

LIRMM

INRIA

Plan

1. Context

Plan

Plan

Plan

Plan

Context

DEMAR : Artificial movement and gait restoration (project-team)

Context

DEMAR : Artificial movement and gait restoration (project-team)

- Safety

Context

DEMAR : Artificial movement and gait restoration (project-team)

- Safety
- few sensors

Context

DEMAR : Artificial movement and gait restoration (project-team)

- Safety
- few sensors $\Rightarrow$ No feedback , No control loop.

Context

DEMAR : Artificial movement and gait restoration (project-team)

- Safety
- few sensors $\Rightarrow$ No feedback , No control loop.
- Validation on humanoid robots.

Modeling

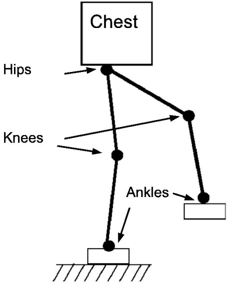


Fig.: 2D model of paraplegian patient under FES

We modelize the patient as a serial chain with 6 degrees of freedom in the saittal plane.

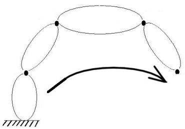
modeling

Forward recurrency (i =1 to N) Given $q, \dot{q}, \ddot{q}, X_0, \dot{X}_0, \ddot{X}_0$

$$\begin{aligned} X_i &= f_1(X_{i-1}, q_i) \\ \dot{X}_i &= f_2(\dot{X}_{i-1}, q_i, \dot{q}_i) \\ \ddot{X}_i &= f_3(\ddot{X}_{i-1}, q_i, \dot{q}_i, \ddot{q}_i) \end{aligned} \tag{1}$$

With

$$X_i = [x_i, y_i, \theta_i]^T \tag{2}$$



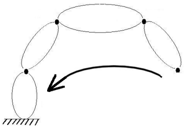
modeling

Backward recuurency : (i = N to 0)

$$F_i = g(F_{i+1}, q_i, \dot{q}_i, \ddot{q}_i, X_i, \dot{X}_i, \ddot{X}_i) \tag{1}$$

With

$$F_i = [Fx_i, Fy_i, \Gamma_i]^T \tag{2}$$



Path planning problem

A motion is defined through the vector $\mathbf{P}$

Path planning problem

A motion is defined through the vector **P**
The path planning problem is to find the best **P** that :

Path planning problem

A motion is defined through the vector $\mathbf{P}$
The path planning problem is to find the best $\mathbf{P}$ that :

$$\min \int_0^T F(\mathbf{P}, t)dt$$

Path planning problem

A motion is defined through the vector $\mathbf{P}$
The path planning problem is to find the best $\mathbf{P}$ that :

$$\min \int_0^T F(\mathbf{P}, t)dt$$

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0$$

Path planning problem

A motion is defined through the vector $\mathbf{P}$
The path planning problem is to find the best $\mathbf{P}$ that :

$$\begin{aligned} &\min \int_0^T F(\mathbf{P},t)dt \\ &\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P},t) \leq 0 \\ &\forall j \quad h_j(\mathbf{P}) = 0 \end{aligned} \tag{3}$$

Path planning problem

A motion is defined through the vector $\mathbf{P}$
The path planning problem is to find the best $\mathbf{P}$ that :

$$\begin{aligned} &\min \int_0^T F(\mathbf{P}, t)dt \\ &\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \\ &\forall j \quad h_j(\mathbf{P}) = 0 \end{aligned} \tag{3}$$

Semi-Infinite Programming [Hettich and Kortanek(1993)]

Path planning problem

Set of equality constraints :

$$\forall j \quad h_j(\mathbf{P}) = 0 \tag{4}$$

Path planning problem

Set of equality constraints :

$$\forall j \quad h_j(\mathbf{P}) = 0 \tag{4}$$

Used to define the motion.

Path planning problem

Set of equality constraints :

$$\forall j \quad h_j(\mathbf{P}) = 0 \tag{4}$$

Used to define the motion.
Usually this constraints must be satisfied for discrete instants

Path planning problem

Set of inequality constraints :

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{5}$$

Path planning problem

Set of inequality constraints :

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{5}$$

- joint position, velocity and torque

Path planning problem

Set of inequality constraints :

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{5}$$

- joint position, velocity and torque
- balance (ZMP [Vukobratović and Borovac(2004)])

Path planning problem

Set of inequality constraints :

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{5}$$

- joint position, velocity and torque
- balance (ZMP [Vukobratović and Borovac(2004)])

This constraints must be satisfied over whole motion duration

Path planning problem

Set of inequality constraints :

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{5}$$

- joint position, velocity and torque
- balance (ZMP [Vukobratović and Borovac(2004)])

This constraints must be satisfied over whole motion duration

We present a new method to deal with the inequality constraints.

Usual Discretization $\Rightarrow$ Time-point Discretization

The algorithms cannot deal with continuous functions.

Usual Discretization ⇒ Time-point Discretization

The algorithms cannot deal with continuous functions.
The constraint functions must be discretized.

Usual Discretization $\Rightarrow$ Time-point Discretization

The algorithms cannot deal with continuous functions.
The constraint functions must be discretized.
We change

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{6}$$

Usual Discretization $\Rightarrow$ Time-point Discretization

The algorithms cannot deal with continuous functions.
The constraint functions must be discretized.
We change

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{6}$$

Into

$$\forall i, \forall t_k \in \{t_0, t_1, ..., t_{N-1}, t_N\} \quad g_i(\mathbf{P}, t_k) \leq 0 \tag{7}$$

Usual Discretization ⇒ Time-point Discretization

The algorithms cannot deal with continuous functions.
The constraint functions must be discretized.
We change

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{6}$$

Into

$$\forall i, \forall t_k \in \{t_0, t_1, ..., t_{N-1}, t_N\} \quad g_i(\mathbf{P}, t_k) \leq 0 \tag{7}$$

Continuous time interval ⇒ Set of discrete time-point (grid)

Usual Discretization $\Rightarrow$ Time-point Discretization

The algorithms cannot deal with continuous functions.
The constraint functions must be discretized.
We change

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{6}$$

Into

$$\forall i, \forall t_k \in \{t_0, t_1, ..., t_{N-1}, t_N\} \quad g_i(\mathbf{P}, t_k) \leq 0 \tag{7}$$

Continuous time interval $\Rightarrow$ Set of discrete time-point (grid)

[Reemtsen(1998)] presents how to compute and adapt $\{t_0, t_1, ..., t_{N-1}, t_N\}$

Usual Discretization : Illustration of constraint violation

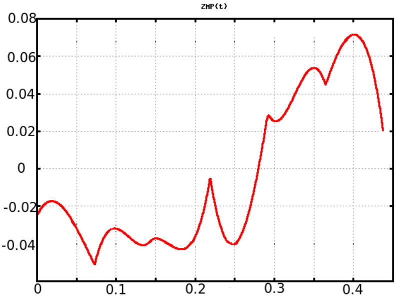


Fig.: Representation of ZMP(t)

Usual Discretization : Illustration of constraint violation

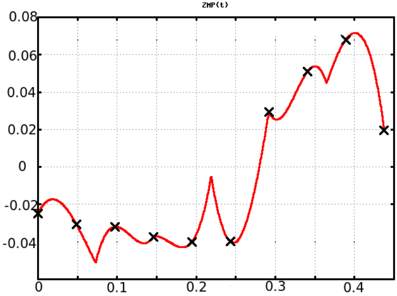


Fig.: Representation of ZMP(t)

We take 10 points to evaluate the constraint function

Usual Discretization : Illustration of constraint violation

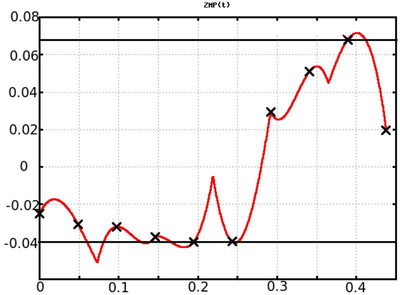
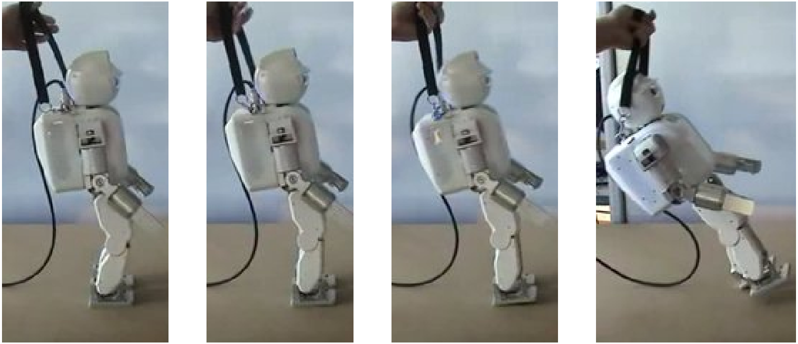


Fig.: Representation of ZMP(t)

The points satisfy the constraint whereas the continuous function violate it

Usual Discretization : Illustration of constraint violation



(a) $t = 0s$ (b) $t = 0.2s$ (c) $t = 0.4s$ (d) $t = 0.6s$

Fig.: Motion optimized with a time-point discretization

Safe path planning : Definition

Safe path planning is a path planning algorithm that uses

Safe path planning : Definition

Safe path planning is a path planning algorithm that uses

- the same algorithms that usual path planning,

Safe path planning : Definition

Safe path planning is a path planning algorithm that uses

- the same algorithms that usual path planning,
- safe discretization.

Safe discretization $\Rightarrow$ Time-interval Discretization

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{8}$$

Safe discretization ⇒ Time-interval Discretization

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{8}$$

$$[t_0, t_N] = [t_0, t_1] \cup [t_1, t_2] \cup \ldots \cup [t_{N-1}, t_N] \tag{9}$$

Safe discretization ⇒ Time-interval Discretization

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{8}$$

$$[t_0, t_N] = [t_0, t_1] \cup [t_1, t_2] \cup \ldots \cup [t_{N-1}, t_N] \tag{9}$$

$$\forall i, \forall j \in \{1, 2, ..N\} \quad \max_{\forall \tau \in [t_{j-1}, t_j]} g_i(\mathbf{P}, \tau) \leq 0 \tag{10}$$

Safe discretization ⇒ Time-interval Discretization

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{8}$$

$$[t_0, t_N] = [t_0, t_1] \cup [t_1, t_2] \cup \ldots \cup [t_{N-1}, t_N] \tag{9}$$

$$\forall i, \forall j \in \{1, 2, ..N\} \quad \max_{\forall \tau \in [t_{j-1}, t_j]} g_i(\mathbf{P}, \tau) \leq 0 \tag{10}$$

Continuous time interval ⇒ Set of time-interval

Safe discretization ⇒ Time-interval Discretization

$$\forall i, \forall t \in [t_0, t_N] \quad g_i(\mathbf{P}, t) \leq 0 \tag{8}$$

$$[t_0, t_N] = [t_0, t_1] \cup [t_1, t_2] \cup \ldots \cup [t_{N-1}, t_N] \tag{9}$$

$$\forall i, \forall j \in \{1, 2, ..N\} \quad \max_{\forall \tau \in [t_{j-1}, t_j]} g_i(\mathbf{P}, \tau) \leq 0 \tag{10}$$

Continuous time interval ⇒ Set of time-interval

The safe discretization is done through Interval Analysis.

Safe discretization : illustration

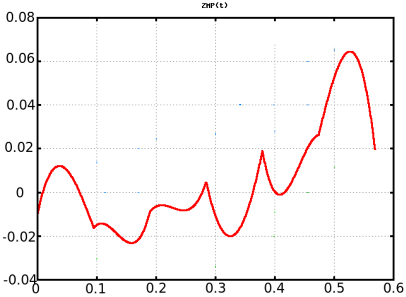


Fig.: Representation of ZMP(t)

Safe discretization : illustration

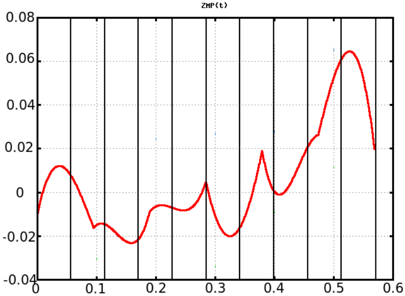


Fig.: Representation of ZMP(t)

Split into 10 time-intervals

Safe discretization : illustration

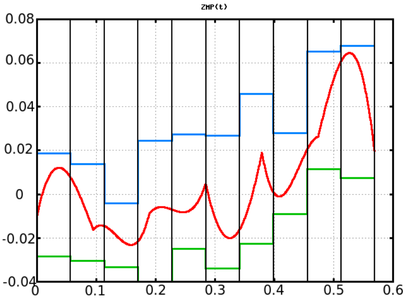


Fig.: Representation of ZMP(t)

Compute the extrema through interval analysis

Safe discretization : illustration

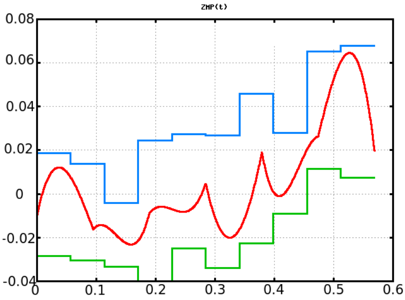


Fig.: Representation of ZMP(t)

Return the values to the algorithm

Safe discretization : illustration

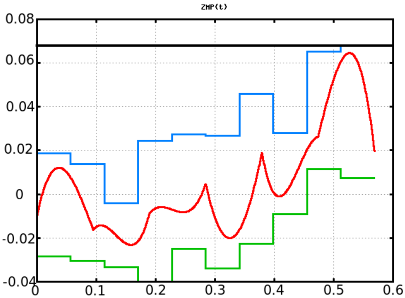


Fig.: Representation of ZMP(t)

The ZMP constraint is never violates ⇒ the robot keeps its balance

Safe discretization : illustration

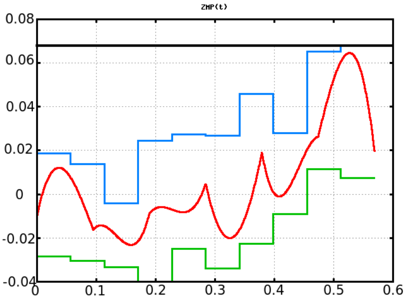


Fig.: Representation of ZMP(t)

Some Methods from Interval Analysis allow to get better computation [Hansen and Walster(2004)]

Safe discretization : Illustration

(a) $t = 0s$ (b) $t = 0.2s$ (c) $t = 0.4s$ (d) $t = 0.6s$

Fig.: Motion optimized with a time-interval discretization

Results : model for path planning

We did path planning :

Results : model for path planning

We did path planning :

- with a 2D model in the sagittal plane of the lower limbs of HOAP-3

Results : model for path planning

We did path planning :

- with a 2D model in the sagittal plane of the lower limbs of HOAP-3
- considering 6 dof

Results : model for path planning

We did path planning :

- with a 2D model in the sagittal plane of the lower limbs of HOAP-3
- considering 6 dof
- to achieve a step of 7 cm

Results : model for path planning

We did path planning :

- with a 2D model in the sagittal plane of the lower limbs of HOAP-3
- considering 6 dof
- to achieve a step of 7 cm
- minimizing the motion duration.

Results : model for path planning

We did path planning :

- with a 2D model in the sagittal plane of the lower limbs of HOAP-3
- considering 6 dof
- to achieve a step of 7 cm
- minimizing the motion duration.
- using C-FSQP

Results

	usual path planning	safe path planning
number of time-point/interval discretization	20	18

Results

	usual path planning	safe path planning
number of time-point/interval discretization	20	18
computation time	18 min 45s	55 min

Results

	usual path planning	safe path planning
number of time-point/interval discretization	20	18
computation time	18 min 45s	55 min
Motion Duration	0.35 s	0.44s

Results

	usual path planning	safe path planning
number of time-point/interval discretization	20	18
computation time	18 min 45s	55 min
Motion Duration	0.35 s	0.44s
Number of iteration of the algorithm	1510	232

Results

	usual path planning	safe path planning
number of time-point/interval discretization	20	18
computation time	18 min 45s	55 min
Motion Duration	0.35 s	0.44s
Number of iteration of the algorithm	1510	232

Tab.: Comparison of usual and safe path planning.

Ongoing works & Conclusion

Ongoing works :

Ongoing works & Conclusion

Ongoing works :

- improving interval computation

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to $\mathbf{P}$

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to $\mathbf{P}$
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to **P**
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to **P**
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Safe path planning is a new method wich :

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to **P**
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Safe path planning is a new method wich :

- uses the same algorithm than usual path planning,

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to **P**
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Safe path planning is a new method wich :

- uses the same algorithm than usual path planning,
- ensures the validity of the constraints for whole the motion duration,

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to $\mathbf{P}$
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Safe path planning is a new method wich :

- uses the same algorithm than usual path planning,
- ensures the validity of the constraints for whole the motion duration,
- can be generalized

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to $\mathbf{P}$
- creation of the Guaranteed Discretization Library `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Safe path planning is a new method wich :

- uses the same algorithm than usual path planning,
- ensures the validity of the constraints for whole the motion duration,
- can be generalized
  - from one dimension (time) to N dimensions

Ongoing works & Conclusion

Ongoing works :

- improving interval computation
- computation of the gradient of the constraint with respect to **P**
- creation of the Guaranteed Discretization Library
  `http://www.lirmm.fr/~lengagne/GDL/`

Conclusion

Safe path planning is a new method wich :

- uses the same algorithm than usual path planning,
- ensures the validity of the constraints for whole the motion duration,
- can be generalized
  - from one dimension (time) to N dimensions
  - to more complex systems : 3D.

Thank you for your attention !

E. Hansen and G.W. Walster.
Global optimization using interval analysis.
Marcel Dekker, 2nd edition, 2004.

R. Hettich and K. O. Kortanek.
Semi-infinite programming : theory, methods, and applications.
SIAM Rev., 35(3) :380–429, 1993.
ISSN 0036-1445.
doi : http://dx.doi.org/10.1137/1035089.
URL http://www.jstor.org/pss/2132425.

R. Reemtsen.
Semi-infinite programming : discretization methods, 1998.
URL citeseer.ist.psu.edu/282299.html.

Miomir Vukobratović and Branislav Borovac.
Zero-moment point : Thirty five years of its life.
International Journal of Humanoid Robotics, 1(1) :157–173, 2004.