Operational space control in the context of model learning

First workshop for young researchers on Human-friendly Robotics
Napoli, October 24th, 2008

Vincent Padois
vincent.padois@upmc.fr

Université Pierre et Marie Curie (UPMC)
Institut des Systèmes Intelligents et de Robotique (ISIR)
75252 Paris, FRANCE

A word about me

A word about me

Background

- Studied Engineering and Automatic Control at ENI in Tarbes.
- Exchange student at NTU (Singapore).
- M. Sc. and PhD in Automatic Control from INP in Toulouse.
- Postdoc in Prof. O. Khatib research Group at Stanford University (CA, USA).

A word about me

Background

- Studied Engineering and Automatic Control at ENI in Tarbes.
- Exchange student at NTU (Singapore).
- M. Sc. and PhD in Automatic Control from INP in Toulouse.
- Postdoc in Prof. O. Khatib research Group at Stanford University (CA, USA).

Current status

- Assistant Professor at Université Pierre et Marie Curie since Sept. 2007 : Teaching Computer Science and Control for Robotics
- Research at l'Institut des Systèmes Intelligents et de Robotique :
  - Reactive control and motion generation for redundant systems
  - Simulation and evaluation software | Collision detection
  - iCub...

A word about UPMC and ISIR

A word about UPMC and ISIR

Université Pierre et Marie Curie

- A "Science" University :

  From Astrophysics to Zoology...

- Around 38000 students and 120 research laboratories.

A word about UPMC and ISIR

Université Pierre et Marie Curie

- A "Science" University :

  From Astrophysics to Zoology...

- Around 38000 students and 120 research laboratories.

Institut des Systèmes Intelligents et de Robotique

- A "young" lab. born from the fusion of three different labs in jan. 2007.
- 100 people among which 30 faculties.
- Lab. director : Prof. Philippe Bidaud.
- 3 research groups :
  - Interactive Systems : micro/nano-manipulation, surgical robotics
  - Mobile and autonomous Systems : bio-inspired robotics, mobile robotics, autonomous systems
  - Perception and Motion : Human and artificial perception, motion analysis, rehabilitation, humanoid robots control

Outline

How are robots controlled (in 1 slide) ? ... And what's the impact on safety ?

How are robots controlled (in 1 slide) ? ... And what's the impact on safety ?

Stiff PID control

- Inherited from industrial Robotics tasks : repeatability
- Not much model needed
- Usually : stiff actuators, high embedded inertias, no natural back-drivability
  $\Longrightarrow$ This is obviously dangerous if Humans are in the robots' environment.

How are robots controlled (in 1 slide) ? ... And what's the impact on safety ?

Stiff PID control

- Inherited from industrial Robotics tasks : repeatability
- Not much model needed
- Usually : stiff actuators, high embedded inertias, no natural back-drivability
  $\Longrightarrow$ This is obviously dangerous if Humans are in the robots' environment.

Model-based control

- Provides a sound framework to the control problem
- Allows to generate controlled compliance
- Requires a good knowledge of the systems dynamics
  $\Longrightarrow$ Ultimately, if the control fails, the actuation type and inertias localization will condition the robot's dangerousness

How are robots controlled (in 1 slide) ? ... And what's the impact on safety ?

Stiff PID control

- Inherited from industrial Robotics tasks : repeatability
- Not much model needed
- Usually : stiff actuators, high embedded inertias, no natural back-drivability
  $\Longrightarrow$ This is obviously dangerous if Humans are in the robots' environment.

Model-based control

- Provides a sound framework to the control problem
- Allows to generate controlled compliance
- Requires a good knowledge of the systems dynamics
  $\Longrightarrow$ Ultimately, if the control fails, the actuation type and inertias localization will condition the robot's dangerousness

Model free, learning-based (NN, RL) control

- No a priori knowledge required, on-line adaptation
- Stability of the resulting controllers is hard to prove
- Deriving a generic control method is not easy
  $\Longrightarrow$ Same conclusion !

How should we control robots (in 1 slide) ?

How should we control robots (in 1 slide) ?

It depends...

How should we control robots (in 1 slide) ?

It depends...

Within the framework of personal Robotics applications :

How should we control robots (in 1 slide) ?

It depends...

Within the framework of personal Robotics applications :

- The environment is not structured and dynamically evolving
- Humans are moving around
- Tasks are highly complex and of very different natures

How should we control robots (in 1 slide) ?

It depends...

Within the framework of personal Robotics applications :

- The environment is not structured and dynamically evolving
- Humans are moving around
- Tasks are highly complex and of very different natures

So we need : aware (perceptive), adaptive, safe, and versatile robots.

How should we control robots (in 1 slide) ?

It depends...

Within the framework of personal Robotics applications :

- The environment is not structured and dynamically evolving
- Humans are moving around
- Tasks are highly complex and of very different natures

So we need : aware (perceptive), adaptive, safe, and versatile robots.

First steps...

- Equip our robots with a redundant number of sensors and actuators (humanoids ?)
- Stop actuating robots with DC motors and high gear ratios.
- Use the nice framework offered by model-based control.
- Don't trust models and put some learning in the loop.

What is nice about model-based control ?

What is nice about model-based control ?

Well, let's focus : What is nice about operational space control ?

What is nice about model-based control ?

Well, let's focus : What is nice about operational space control ?

FIG.: A CAD view of the iCub robot - credit : the RobotCub Consortium

What is nice about model-based control ?

Well, let's focus : What is nice about operational space control ?

What is nice about model-based control ?

Well, let's focus : What is nice about operational space control ?

It relates the task space (what I want to control) to the joints space (what I can actually control)

- At the geometric level : $\mathbf{x} = h(\mathbf{q}) \rightarrow$ non-linear : too complex
- At the kinematic level : $\dot{\mathbf{x}} = J(\mathbf{q})\dot{\mathbf{q}}$
- At the dynamic level : $\mathbf{\Gamma} = M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{\Gamma}_{ext}$
- In a combined fashion : $\mathbf{f} = \Lambda(\mathbf{q})\ddot{\mathbf{x}} + \mu(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{p}(\mathbf{q}) + \mathbf{f}_{ext}$ [Khatib 87]

What is nice about model-based control ?

What is nice about model-based control ?

- **The focus is put on the performance of the control in the task space**

What is nice about model-based control ?

- **The focus is put on the performance of the control in the task space**
- Linear Algebra and Matrix computation tools : Image, Kernel, Null space, Projectors, Generalized inverses, SVD...

What is nice about model-based control ?

- **The focus is put on the performance of the control in the task space**
- Linear Algebra and Matrix computation tools : Image, Kernel, Null space, Projectors, Generalized inverses, SVD...
- In the case of redundant systems, ∞ of solutions to the inversion problem : $\dot{\mathbf{q}} = J(\mathbf{q})^{\sharp}\dot{\mathbf{x}} + (I - J(\mathbf{q})^{\sharp}J(\mathbf{q}))\dot{\mathbf{q}}_0$ [Liégeois 77], [Nakamura 91], [Doty 93] ...

What is nice about model-based control ?

- **The focus is put on the performance of the control in the task space**
- Linear Algebra and Matrix computation tools : Image, Kernel, Null space, Projectors, Generalized inverses, SVD...
- In the case of redundant systems, ∞ of solutions to the inversion problem : $\dot{\mathbf{q}} = J(\mathbf{q})^{\sharp}\dot{\mathbf{x}} + (I - J(\mathbf{q})^{\sharp}J(\mathbf{q}))\dot{\mathbf{q}}_0$ [Liégeois 77], [Nakamura 91], [Doty 93] ...
- $\dot{\mathbf{x}}$ can be anything :
  - Hands, feet, head, CoG velocities...
  - Whatever velocity of the system that can be related to the joints velocities by a linear relation.

What is nice about model-based control ?

- **The focus is put on the performance of the control in the task space**
- Linear Algebra and Matrix computation tools : Image, Kernel, Null space, Projectors, Generalized inverses, SVD...
- In the case of redundant systems, ∞ of solutions to the inversion problem : $\dot{\mathbf{q}} = J(\mathbf{q})^{\sharp}\dot{\mathbf{x}} + (I - J(\mathbf{q})^{\sharp}J(\mathbf{q}))\dot{\mathbf{q}}_0$ [Liégeois 77], [Nakamura 91], [Doty 93] ...
- $\dot{\mathbf{x}}$ can be anything :
  - Hands, feet, head, CoG velocities...
  - Whatever velocity of the system that can be related to the joints velocities by a linear relation.
- Very general framework, powerfull mathematical tools

What is nice about model-based control ?

- **The focus is put on the performance of the control in the task space**
- Linear Algebra and Matrix computation tools : Image, Kernel, Null space, Projectors, Generalized inverses, SVD...
- In the case of redundant systems, ∞ of solutions to the inversion problem : $\dot{\mathbf{q}} = J(\mathbf{q})^\sharp \dot{\mathbf{x}} + (I - J(\mathbf{q})^\sharp J(\mathbf{q}))\dot{\mathbf{q}}_0$ [Liégeois 77], [Nakamura 91], [Doty 93] ...
- $\dot{\mathbf{x}}$ can be anything :
  - Hands, feet, head, CoG velocities...
  - Whatever velocity of the system that can be related to the joints velocities by a linear relation.
- Very general framework, powerfull mathematical tools

Some state of the art literature results for humanoids :
<u>**Reactive approach**</u> : J. Park, L. Sentis, O. Khatib at Stanford University (2004-2008).

<u>**Motion planning based approach**</u> : E. Yoshida, M. Poirier, J.P.Laumond, O. Kanoun, F. Lamiraux, R. Alami and K. Yokoi at the French-Japanese Joint Robotics Lab. at LAAS CNRS (Toulouse, France) (2008).

What is nice about model-based control ? An Example...

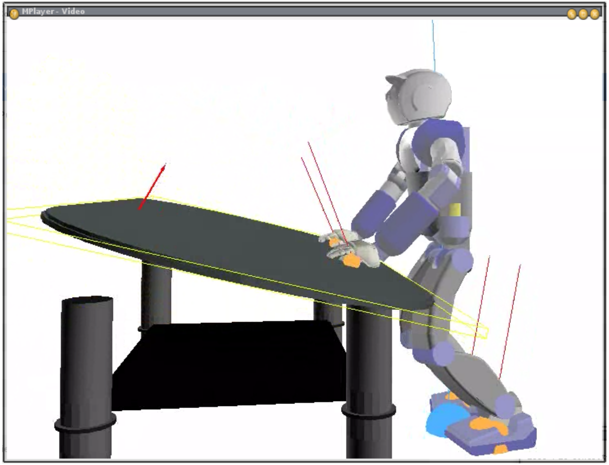


FIG.: A virtual humanoid in the SAI environment - credit : Luis Sentis

Why does learning has to be included ?

Why does learning has to be included ?

1. Having a good model of a complex robot is costly

Why does learning has to be included ?

1. Having a good model of a complex robot is costly
2. Some effects are hard to model (friction)

Why does learning has to be included ?

1. Having a good model of a complex robot is costly
2. Some effects are hard to model (friction)
3. Inverting models with high dimensions is computationally expensive whereas using a learnt inverse model can be very efficient

Why does learning has to be included ?

1. Having a good model of a complex robot is costly
2. Some effects are hard to model (friction)
3. Inverting models with high dimensions is computationally expensive whereas using a learnt inverse model can be very efficient

How should we learn ?

- On-line (incrementally)
- Using Simple data representation
- Efficiently (O(n) complexity in high number of dimensions)

Why does learning has to be included ?

1. Having a good model of a complex robot is costly
2. Some effects are hard to model (friction)
3. Inverting models with high dimensions is computationally expensive whereas using a learnt inverse model can be very efficient

How should we learn ?

- On-line (incrementally)
- Using Simple data representation
- Efficiently (O(n) complexity in high number of dimensions)

Does it exist already ?

- Yes : Locally Weighted Projection Regression [Vijayakumar et al. 00]

Locally Weighted Projection Regression

Locally Weighted Projection Regression

LWPR is a function approximator which provides accurate approximation in very large spaces in a $O(n)$ complexity, where n is the number of sample data.

Locally Weighted Projection Regression

LWPR is a function approximator which provides accurate approximation in very large spaces in a $O(n)$ complexity, where n is the number of sample data.

- Uses a combination of linear models (called Receptive Fields (RFs)) (RLS)
- Each model is valid on an elliptic zone of the inputs space
- The prediction made by LWPR for a given input vector is a weighted sum of the results of all the active surroundings RFs.
- RFs are created and destroyed incrementally
- Only relevant combination of the input vector are kept in the learnt model (PLS)

Locally Weighted Projection Regression

LWPR is a function approximator which provides accurate approximation in very large spaces in a $O(n)$ complexity, where n is the number of sample data.

- Uses a combination of linear models (called Receptive Fields (RFs)) (RLS)
- Each model is valid on an elliptic zone of the inputs space
- The prediction made by LWPR for a given input vector is a weighted sum of the results of all the active surroundings RFs.
- RFs are created and destroyed incrementally
- Only relevant combination of the input vector are kept in the learnt model (PLS)

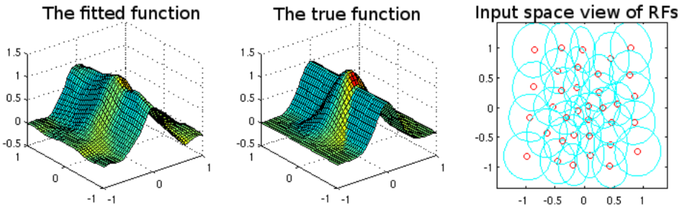


FIG.: Example of LWPR learning on a sample function - credit : Sethu Vijayakumar

Our approach to combine OSC and learning

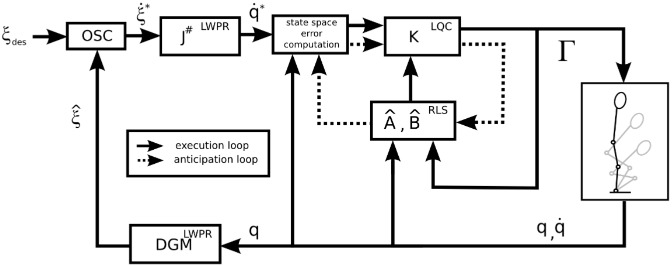


FIG.: Some example of OSC and learning combination - credit : Camille Salaün (Abials, 2008)

Some results in simulation and using a toy-like robot

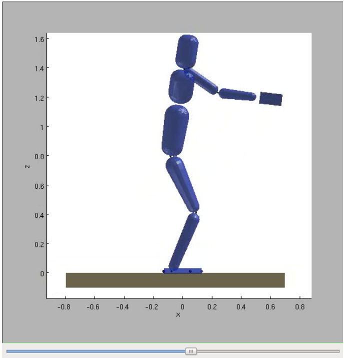


Fig.: A standing and reaching task using learnt models - credit : Camille Salaün (Abials, 2008)

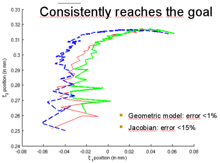


Fig.: A standing task using learnt models on a Bioloid robot - credit : Paul Tonelli

Perspectives

Perspectives

From a theoretical point of view

- Combine multiple tasks and constraints using learnt models
- Increase the number of DoFs
- Combine coarse hand-crafted models with on-line learnt models

Perspectives

From a theoretical point of view

- Combine multiple tasks and constraints using learnt models
- Increase the number of DoFs
- Combine coarse hand-crafted models with on-line learnt models

From a practical point of view

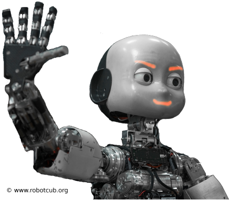
Perspectives

From a theoretical point of view

- Combine multiple tasks and constraints using learnt models
- Increase the number of DoFs
- Combine coarse hand-crafted models with on-line learnt models

From a practical point of view

- Use this robot as a real demonstrator !



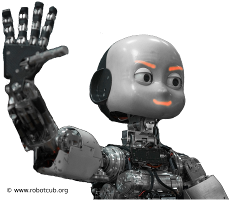
Perspectives

From a theoretical point of view

- Combine multiple tasks and constraints using learnt models
- Increase the number of DoFs
- Combine coarse hand-crafted models with on-line learnt models

From a practical point of view

- Use this robot as a real demonstrator !
- Guide learning using Human/Robot interaction (teaching, imitation, curiosity)



Perspectives

Perspectives

Some possible extensions

Perspectives

Some possible extensions

Bernstein in 1967 raised the following paradox : thanks to the redundancy of their motor system, human beings can realize a task from an infinity of ways. However, for a given task, they always reproduce the same kind of stereotypic motion. The problem consists in explaining where these regularities come from, given the diversity of possible solutions.

Perspectives

Some possible extensions

Bernstein in 1967 raised the following paradox : thanks to the redundancy of their motor system, human beings can realize a task from an infinity of ways. However, for a given task, they always reproduce the same kind of stereotypic motion. The problem consists in explaining where these regularities come from, given the diversity of possible solutions.

Can we get an idea of what is the optimal motion criteria using motion capture data and learnt direct and inverse models ? This vision of human motion is a bit simplistic. But we may try !

Acknowledgements

- Prof. Philippe Bidaud (always thank the boss !)
- Camille Salaün (PhD student)
- Prof. Olivier Sigaud (Senior colleague)
- Paul Tonelli (Master Student)
- The RobotCub consortium and Giorgio Metta in particular (always thank the robot provider !)

Questions ?