Learning local trajectories for high precision robotics tasks

Joris Guérin, Olivier Gibaru, Eric Nyiri and Stéphane Thiery

LSIS - Laboratoire des Sciences de l'Information et des Systèmes
Arts et Métiers ParisTech

Industrial Electronics Conference 2016, Firenze

Plan

1. Introduction
   - Long term objective
   - Application description
2. Contribution and validation
   - iLQR with learnt dynamics and cost
   - Parameters tuning
   - Experimental validation
3. Conclusion

Self-programming robots in industry

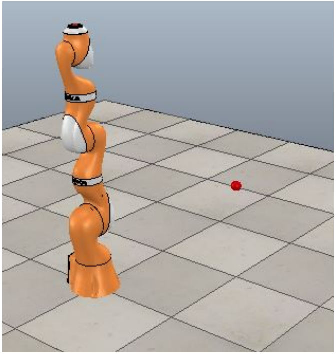
- Easy robot programming for unqualified worker
  - Task description : Simple cost function
  - No tool characterization
  - No calibration
- Industrial context
  - High precision tasks

This presentation : first step towards this goal

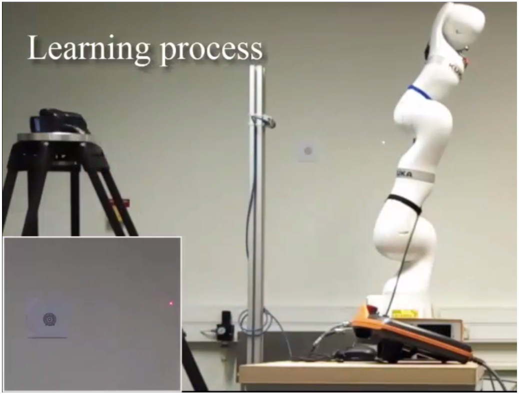
Application description

Precise Cartesian positioning under joint position control

- Angular joint position control
- Cartesian tool positioning



Practical example



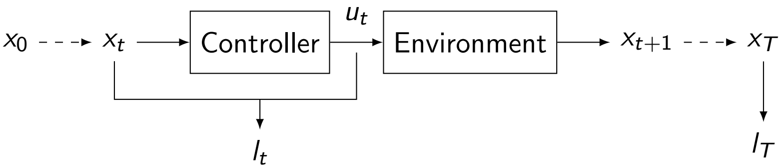
Independence of :

- Robot model
- Tool orientation
- Robot location

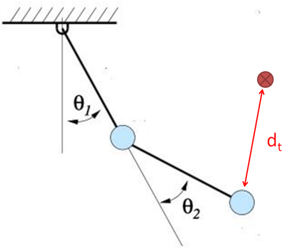
Plan

Plan

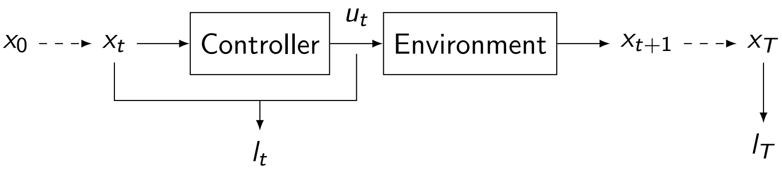
Optimal control problem



- x_t : State vector of the system
- u_t : Control vector
- l_t : Cost



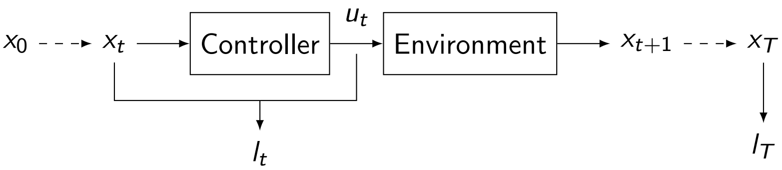
Linear-quadratic control



$$x_{t+1} = Ax_t + Bu_t$$

$$l_t = x_t^T Q\, x_t + u_t^T R\, u_t + 2x_t^T N\, u_t$$

iterative Linear-Quadratic Regulator

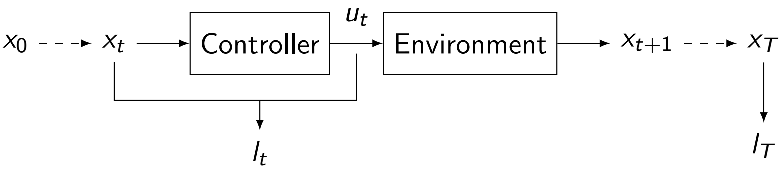


$$x_{t+1} = F_t(x_t, u_t)$$

$$l_t = L_t(x_t, u_t)$$

[Li et al., 2004]

iterative Linear-Quadratic Regulator

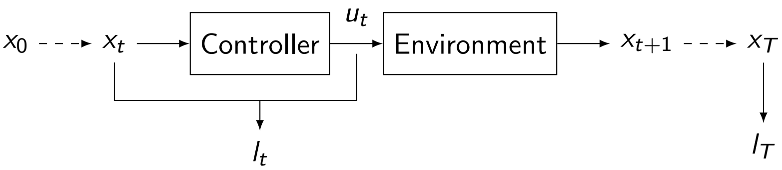


Nominal trajectory :

$$\{\bar{x}_0, \bar{u}_0, \ldots, \bar{x}_T\}$$

[Li et al., 2004]

iterative Linear-Quadratic Regulator



$$F_t(\bar{x}_t + \delta x_t, \bar{u}_t + \delta u_t) = \bar{x}_{t+1} + F_{x_t}\delta x_t + F_{u_t}\delta u_t$$

$$L_t(\bar{x}_t + \delta x_t, \bar{u}_t + \delta u_t) = \bar{l}_t + L_{x_t}\delta x_t + \frac{1}{2}\delta x_t^T L_{x,x_t}\delta x_t + L_{u_t}\delta u_t +$$
$$\frac{1}{2}\delta u_t^T L_{u,u_t}\delta u_t + \delta x_t^T L_{x,u_t}\delta u_t$$

[Li et al., 2004]

iLQR and environment learning

$$\cancel{F_t(\bar{x}_t + \delta x_t, \bar{u}_t + \delta u_t) = \bar{x}_{t+1} + F_{x_t}\delta x_t + F_{u_t}\delta u_t}$$

Exploration :

$$\{\delta x_t,\ \delta u_t\} \rightarrow \delta x_{t+1}$$

Linear regression :

$$F_t(\bar{x}_t + \delta x_t, \bar{u}_t + \delta u_t) = \bar{x}_{t+1} + C_1\delta x_t + C_2\delta u_t$$

[Mitrovic et al., 2010], [Levine et al., 2014]

Quadratic cost function

Most optimal control problems :

Define the cost as a quadratic function of x_t and u_t :

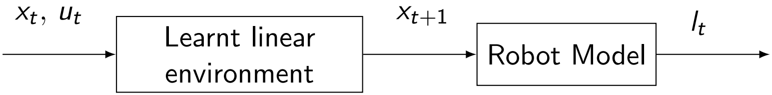
$x_t,\ u_t \longrightarrow$ Quadratic cost function $\longrightarrow L(x_t,\ u_t)$

Issue : not appropriate to all problems

Quadratic cost function

[Levine et al., 2015] :

Use a geometric model of the robot :



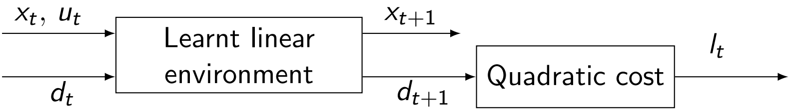
Issues :

- Not independent of robot model, tool, ...
- Calibration required

Quadratic cost function

[Levine et al., 2014] :

Include distance in state vector :

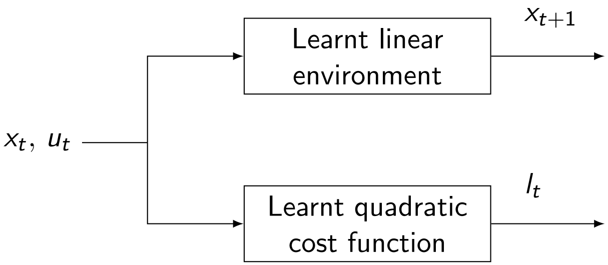


Issue : Quadratic expansion of a quantity approximated linearly

Quadratic cost function

Proposed approach :

Learn the cost with polynomial regression :



- Independent of robot model
- Local cost known quadratically

Local optimal control problem

$$\underset{u_t}{\text{Minimize}} \quad \sum_{t=1}^{T} l_t$$
$$\text{subject to} \quad D_{KL}(\Pi_{new}(\tau)||\Pi_{old}(\tau)) \leq \epsilon$$

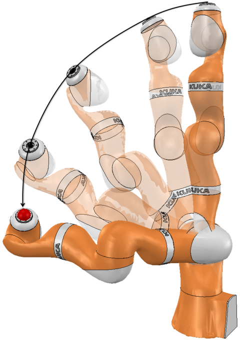
[Peters et al., 2008], [Levine et al., 2014]

Plan

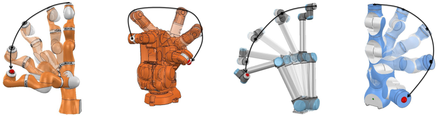
Several learning parameters

- Number of samples during exploration
- Initial variance during exploration
- Cost function : $l_t = d^2 + v \log(d^2 + \alpha)$; *[Levine et al., 2015]*
- Maximum deviation from nominal during update

Use of V-REP simulation software



- Many configurations to test
- Avoid irrelevant real robot sampling
- "Low cost" robot model independence validation *[Guérin et al., 2016]*



Results

$N_{ech} = 40$; *[Guérin et al., 2016]*

$cov_{ini} = 1$

v	ϵ_{ini}	α: 10^{-3}	10^{-5}	10^{-7}
0.1	100	11	16	13
	1000	0.25	12	10
	10000	13	0.27	8
1	100	0.11	14	16
	1000	10	12	10
	10000	0.10	1.69	0.24
10	100	0.11	0.22	0.84
	1000	0.13	12	0.20
	10000	13	0.23	15

$cov_{ini} = 10$

v	ϵ_{ini}	α: 10^{-3}	10^{-5}	10^{-7}
0.1	100	0.32	0.15	0.39
	1000	0.45	0.28	0.22
	10000	0.30	0.29	0.31
1	100	0.14	0.32	0.32
	1000	14	1.93	1.70
	10000	1.82	0.99	0.11
10	100	0.34	0.38	0.39
	1000	0.71	0.29	0.53
	10000	0.70	0.14	2.31

$cov_{ini} = 100$

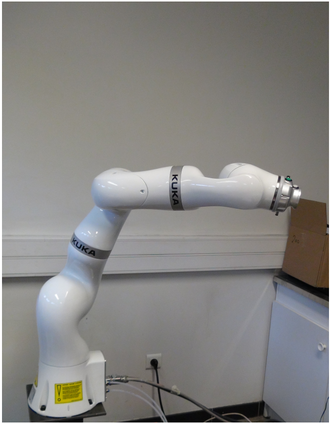
v	ϵ_{ini}	α: 10^{-3}	10^{-5}	10^{-7}
0.1	100	12.79	12.42	17.83
	1000	4.42	0.30	3.50
	10000	2.88	10.93	2.60
1	100	24.37	15.75	10.13
	1000	7.66	6.32	1.87
	10000	2.67	8.37	6.44
10	100	1.93	8.93	10.11
	1000	8.03	2.23	3.50
	10000	2.70	4.83	2.60

Choices for validation :
$cov_{ini} = 1$, $\epsilon_{ini} = 10000$, $v = 0.1$ and $\alpha = 10^{-7}$

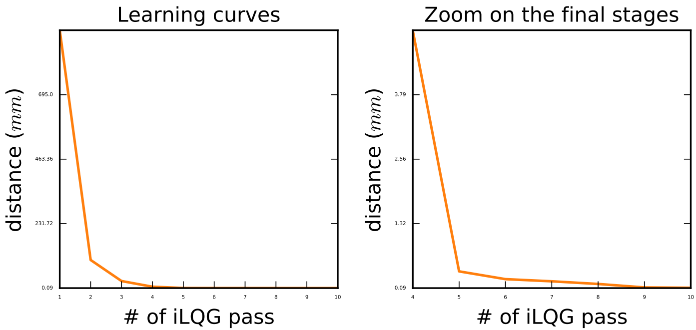
Plan

1. Introduction
   - Long term objective
   - Application description
2. Contribution and validation
   - iLQR with learnt dynamics and cost
   - Parameters tuning
   - Experimental validation
3. Conclusion

Experimental validation



Experimental validation



- Video

Plan

Conclusion

- Approach shown to work on real robots

Perspectives :

- Improve sample efficiency ; *[Levine et al., 2014]*
- Lower level controllers for manipulation task
  - Grasping
  - Assembly
  - ...