# Parameter identification in NMPC algorithm with application to waypoint following by autonomous vehicle

Esten Ingar Grøtli

Humberto Gonzales, Jonathan Sprinkle

# Outline

# Background: DARPA Grand Challenge

- Two previous challenges: 2004 and 2005
- In 2004 none of the competitors finished the race
- In 2005 five teams completed the race with The Stanford Racing Team as the winner
- The two previous races have been through the desert
- This years competition is in an urban environment

# Problem Formulation

## Optimization Problem

$$\min J = \min \sum_{j=0}^{N_x} K_1 J_x + K_2 J_u + K_3 J_d + K_4 J_{obs}$$

$$\text{s.t} \;\; x_{k+1} = f(x_k, u_k(x_k, \theta_k))$$

with $f(x_k, u_k(x_k, \theta_k))$ defined as follows:

$$f_k(x_k, u_k^*(x_k, \theta)) = \begin{bmatrix} x_{1,k} + p_3 u_{1,k} \cos(x_{3,k} + p_1 u_{2,k}) \\ x_{2,k} + p_3 u_{1,k} \sin(x_{3,k} + p_1 u_{2,k}) \\ x_{3,k} + p_3 u_{1,k} \frac{1}{p_2} \sin(p_1 u_{2,k}) \end{bmatrix}$$

## Obstacle Cost

$$J_{obs} = \sum_{o=1}^{n} \delta_2 \left( \frac{1}{\delta_1 + \left|(\hat{x}_o)_{k|j}\right|} - \frac{1}{\delta_1 + \delta_3} \right), \quad \left|(\hat{x}_o)_{k|j}\right| \leq \delta_3, \delta_1, \delta_2 \in \mathbb{R}$$

## Directional Cost

$$J_d = \left| b^T \tilde{x}_{k,j} b - \tilde{x}_{k,j} \right|$$

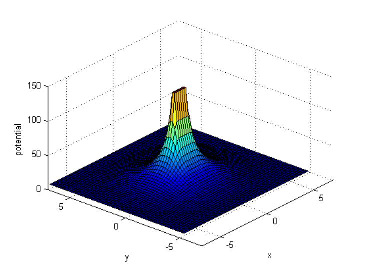
## Position Cost

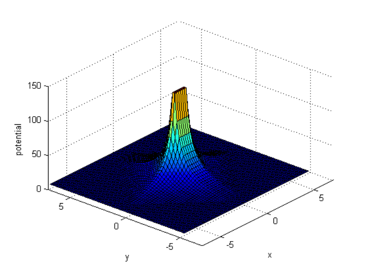
$$J_x = \bar{x}_{k,j}^T Q \bar{x}_{k,j}$$

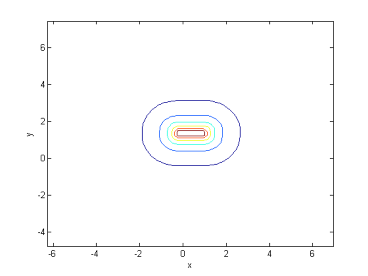
## Input Cost

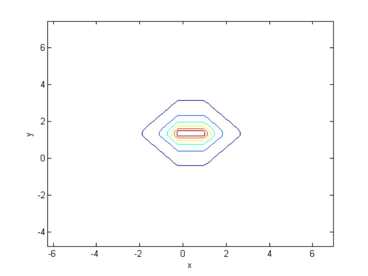
$$J_u = \bar{u}_{k,j-1}^T R \bar{u}_{k,j-1}$$

# Obstacle Potential



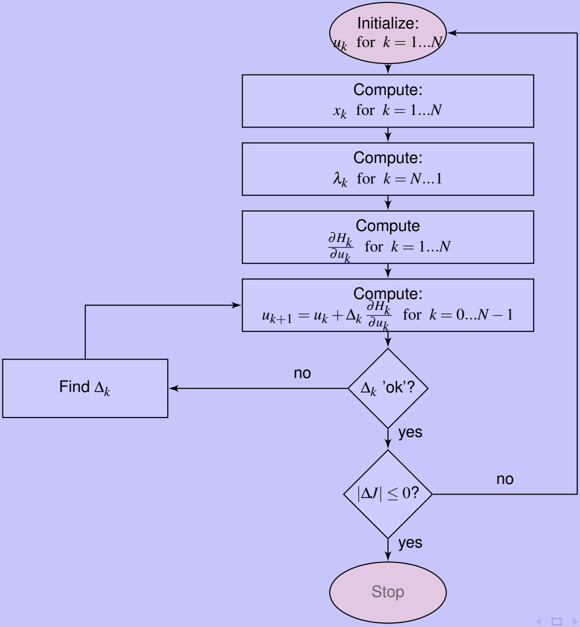



# Videos of NMPC algorithm

| Waypoints | Obstacles1 | Obstacles2 |
| --- | --- | --- |


Initialize:
$u_k$ for $k = 1...N$
Compute:
$x_k$ for $k = 1...N$
Compute:
$\lambda_k$ for $k = N...1$
Compute
$\frac{\partial H_k}{\partial u_k}$ for $k = 1...N$
Compute:
$u_{k+1} = u_k + \Delta_k \frac{\partial H_k}{\partial u_k}$ for $k = 0...N-1$
Find $\Delta_k$
no
$\Delta_k$ 'ok'?
yes
$|\Delta J| \leq 0$?
no
yes
Stop

# Problem Formulation

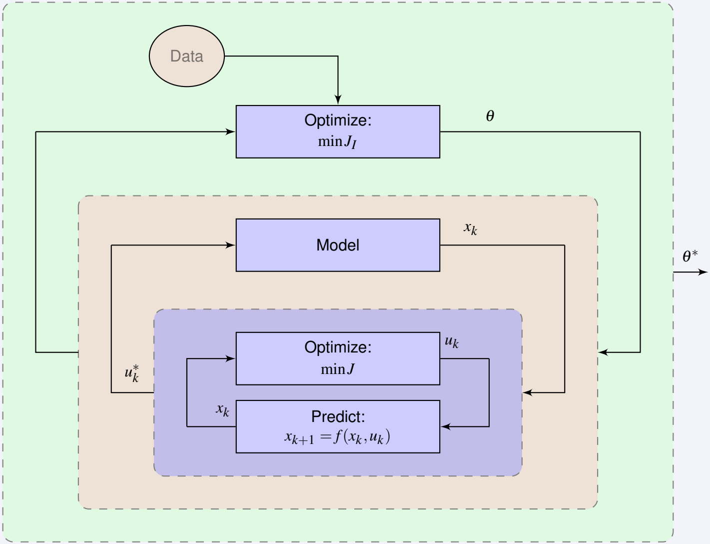
## Optimization Problem

$$\min J_I = \min \sum_{k=0}^{n_h-1} |x_k - x_k^o|^2$$

$$\text{s.t.} \;\; x_{k+1} = f(x_k, u_k^*(x_k, \theta_k))$$

$$f_k(x_k, u_k^*(x_k, \theta)) = \begin{bmatrix} x_{1,k} + p_3 u_{1,k}^* \cos(x_{3,k} + p_1 u_{2,k}^*) \\ x_{2,k} + p_3 u_{1,k}^* \sin(x_{3,k} + p_1 u_{2,k}^*) \\ x_{3,k} + p_3 u_{1,k}^* \frac{1}{p_2} \sin(p_1 u_{2,k}^*) \end{bmatrix}$$

# Overview of algorithm



# Adjoint State and Gradient

## Adjoint State

$$y_k = -\frac{\partial \left|x_k - x_{obs,k}\right|^2}{\partial x_k}^T - \frac{\partial}{\partial x_k} f_k(x_k, u_k^*(x_k, \theta))^T y_{k+1}$$

$$y_{n_H} = 0$$

## Gradient

$$\nabla J_I = \sum_{k=0}^{n_H - 1} \left( \frac{\partial \left|x_k - x_{obs,k}\right|^2}{\partial \theta} + y_{k+1}^T \frac{\partial}{\partial \theta} f_k(x_k, u_k^*(x_k, \theta)) \right)$$

# Future Work

- Check correctness of the algorithm/Debugging
- Finish the implementation of the algorithm in Matlab
- Do testing with made-up data
- Extend algorithm to account for other parameters in the costfunction
- If identification of parameters seems successful, the code will be ported to C, and ORCA
- Apply the algorithm on data from the real vehicle?!