Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

State Estimation Based Model Predictive Control for LHD Vehicles ⋆

Felipe Núñez * **Sergio Navarro** * **Alberto Aguado** ** **Aldo Cipriano** *

* *Department of Electrical Engineering, Pontificia Universidad Católica de Chile, Santiago, Chile (e-mail: fenunez@ing.puc.cl, sanavarm@puc.cl, aciprian@ing.puc.cl)*
** *Cybernetics, Mathematics and Physics Institute, La Habana, Cuba (e-mail: aguado@icmf.inf.cu)*

Abstract: LHD (load-haul-dump) vehicles are used extensively in underground mining operations for ore transporting, primarily in tunnels where access is difficult or dangerous. To ensure underground efficient and safe LHD's performance, a robust feedback control strategy is needed. A state estimation based MPC scheme was designed for control purposes, and evaluated by simulation. The state estimator was developed by testing four approaches, in order to select the optimal one: the extended Kalman Filter, Particle Filter, Moving Horizon Estimator and a genetic algorithm based Moving Horizon Estimatior. The simulation shows that non-linear MPC performs better than linear MPC for path tracking.

Keywords: Model Predictive Control; Nonlinear system control; Nonlinear observer and filter design; Monte Carlo methods; Genetic algorithms.

1. INTRODUCTION

LHD (load-haul-dump) vehicles are used extensively in underground mining operation for ore transporting, primarily in tunnels where access is difficult or dangerous. To ensure underground efficient and safe LHD's performance, a robust feedback control strategy is needed.

Model predictive control (MPC) is a widely used control concept, specially in the process industry. Most popular MPC applications use black-box linear models obtained from process data with the disadvantage that these models do not consider non-linear dynamics. State space based MPC seems to be an interesting LHD control alternative, since it can deal with multivariable non-linear dynamics and explicit constraints. However, the use of a state space model brings a new problem, the nonlinear state estimation, that must be successfully solved in a previous stage.

The most universally accepted state estimator is the Kalman Filter, due to its proved qualities as an optimal estimator and predictor for linear systems (Welch and Bishop, 2004). However, the corresponding nonlinear state estimator, the Extended Kalman Filter (EKF), can not claim a comparable success, due mainly to the fact that it is based on a linear approximation of the system model.

Recently, interest for state estimators based on Monte Carlo simulation as the so-called Particle Filters (PF) has reemerged. Also, the Moving Horizon Estimation (MHE) method has been successfully compared with Extended Kalman Filtering (Haseltine and Rawlings, 2005) in the case of chemical reactors state estimation. Since nonlinear optimization is required the use of Genetic Algorithms (GA) represents an attractive option for on-line MHE implementation.

The purpose of this paper is to design a state space based model predictive control strategy for particular mining vehicles known as Load-Haul-Dump (LHD). Two MPC schemes, a linear and a non-linear MPC, have been compared in order to test their potentials on path tracking. Since a state estimator is needed, a comparison of four estimation methods was realized, in order to choose the optimal one for MPC design.

2. LHD VEHICLES

A LHD (Load-Haul-Dump) is a centrally articulated, four wheeled vehicle, steered by controlling the angle between its two constituent sections (Altafini, 1999), (Ridley and Corke, 2001). Each of the two sections has it own axle, and the wheels are nor steerable. Thus, an LHD's control and path tracking characteristics differ significantly from those of conventional vehicles such as automobiles.

Figure 1 shows a scheme where the LHD main variables are specified. A simplified nonlinear model of a LHD, which only takes into account the cinematic relations among variables, can be formulated in the state space (Altafini, 1999) as follows:

$$\dot{x_1} = u_1 \cos(\theta_1)$$

⋆ This work was supported by Fondecyt Project No. 1050684 and the Pontificia Universidad Católica de Chile

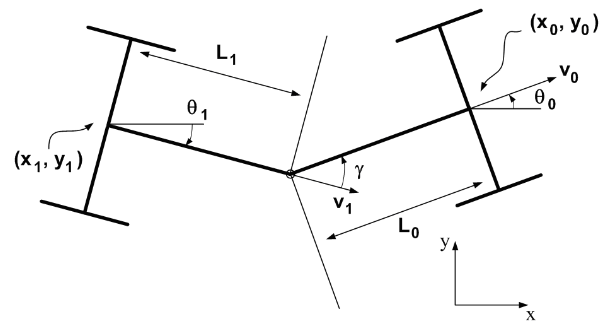


Fig. 1. Simplified scheme of a LHD

$$\begin{aligned} \dot{y}_1 &= u_1 \sin(\theta_1) \\ \dot{\theta}_1 &= \frac{u_1 \sin(\gamma) - L_0 \Gamma}{L_0 + L_1 \cos(\gamma)} \\ \dot{\gamma} &= \Gamma \end{aligned} \tag{1}$$

In this model, the variables usually measured are the angles θ_1 and γ. The manipulable variables are velocity u_1 and Γ , that is proportional to the applied torque.

3. STATE ESTIMATION

3.1 Extended Kalman Filter

The Extended Kalman Filter (EKF) is a suboptimal filter that minimizes the mean error of the state estimation of the LHD non-linear system given by the followings equations:

$$x_{k+1} = f(x_k, u_k, w_k) \tag{2}$$

$$y_k = C x_k + v_k \tag{3}$$

In these equations, x_k is the state vector of the system, u_k is the vector of the manipulated variables, y_k is the measured output vector, $f(\cdot)$ is the non-linear function that define the LHD dynamics, C is the output matrix and w_k and v_k are the process and measurement disturbances which we assume characterized by white noise covariance matrices Q and R, respectively.

The EKF algorithm first estimates the state and measurement vectors at time k using the previous information and the system equations. Along with this, the covariance matrix P_k is also estimated:

$$x^-_{k+1} = f(\hat{x}_k, u_k, 0) \tag{4}$$

$$y^-_{k+1} = C\hat{x}_k \tag{5}$$

$$P^-_{k+1} = A_{k+1} \hat{P}_k A^T_{k+1} + Q \tag{6}$$

Using the covariance matrix R and the residual covariance matrix S_k, the Kalman gain K_k is obtained:

$$S_{k+1} = C P^-_{k+1} C^T + R \tag{7}$$

$$K_{k+1} = P^-_{k+1} C^T S^{-1}_{k+1} \tag{8}$$

Finally, the estimations are corrected using the Kalman gain and the difference between the estimation and the measurement:

$$\hat{x}_{k+1} = x^-_{k+1} + K_{k+1}(y_{k+1} - y^-_{k+1}) \tag{9}$$

$$\hat{y}_{k+1} = C\hat{x}_{k+1} \tag{10}$$

$$\hat{P}_{k+1} = (I - K_{k+1}C) P^-_{k+1} \tag{11}$$

In the preceding equations, A_k is the linearization of the non-linear function $f(\cdot)$ around an operating point at time k:

$$A_{k+1}(i,j) = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_k, u_k, 0) \tag{12}$$

3.2 Particle Filter

Particle Filters (PF) are also suboptimal filters. They perform sequential Monte Carlo estimation based on point mass or particle representation of probability densities (Arulampalam et al., 2002). The most common method is Sequential Importance Sampling (SIS), which attempts to approximate the posterior probability density function (pdf) of the state $p(x_k/y_{1:k})$ with a set of realizations x^i_k, $i = 1 \ldots N_P$, known as particles. $y_{1:k}$ represents the histogram of y from 1 to k. An approximation of the distribution is given by:

$$\hat{p}(x_k/y_{1:k}) = \frac{1}{N_p} \sum_{i=1}^{N_p} \delta(x^i_k) \tag{13}$$

Here $\delta(x^i_k)$ is a Dirac delta in (x^i_k). This approximation can be used to obtain the mean value of the distribution:

$$E(x_k/y_{1:k}) = \int x_k p(x_k/y_{1:k}) dx_k \approx \frac{1}{N_p} \sum_{i=1}^{N_p} x^i_k \tag{14}$$

However, it is not possible to sample directly from the posterior distribution $p(x_k/y_{1:k})$. To generate samples, a known auxiliary distribution $q(x_k/y_{1:k})$ referred to as an Importance Function must be use, whose only constraint is that its support include the support of the real distribution. To represent the desired distribution, the samples must be weighted. The weights $\bar{w}(x_k)$ used to correct the samples are given by:

$$\bar{w}(x_k) = \frac{p(x_k/y_{1:k})}{q(x_k/y_{1:k})} \tag{15}$$

Solving for the distribution $p(\cdot)$ in (15) and then substitute it into (14) yields:

$$\begin{aligned} E(x_k/y_{1:k}) &= \int x_k \bar{w}(x_k) q(x_k/y_{1:k}) dx_k \approx \frac{1}{N_p} \sum_{i=1}^{N_p} x^i_k \bar{w}^i_k \\ \bar{w}^i_k &= \bar{w}(x^i_k) \end{aligned} \tag{16}$$

The samples x^i_k are now obtained from the importance function. If the prior state distribution is used as the importance function, then the weights satisfy the relation (Arulampalam et al., 2002):

$$\bar{w}^i_k = \bar{w}^i_{k-1} p(y_k/x^i_k) \tag{17}$$

3.3 Moving Horizon Estimation

Consider the system evolution as a Markov process, the quantity of interest in state estimation becomes the conditional probability density of the state evolution x_k, given the measurements y_k:

$$p\left(x_0, x_1, \cdots, x_T | y_0, y_1, \cdots, y_{T-1}\right) \tag{18}$$

The optimal state estimation is then a functional $L(\cdot)$ of the conditional probability density function (18). Typically $L(\cdot)$ is chosen as the maximum a posteriori Bayesian (MAP) estimation.

Using the Markov property and assuming normal distributions for the measurement noise, the disturbances and the initial state, the probabilistic expression could be transform into a mathematical programming problem.

Under the precedent assumptions, state estimation becomes the following optimization problem (Rao, 2000):

$$\min_{\hat{x}_0, \{\hat{w}_k\}_{k=0}^{T-1}} \Phi_T\left(\hat{x}_0, \{\hat{w}_k\}\right) \tag{19}$$

Where:

$$\Phi_T\left(\hat{x}_0, \{\hat{w}_k\}\right) = \sum_{k=0}^{T-1} \{(y_k - C\hat{x}_k)^T R^{-1}(y_k - C\hat{x}_k) + \hat{w}_k^T Q^{-1}\hat{w}_k\} + (x_0 - \hat{x}_0)^T \Pi^{-1}(x_0 - \hat{x}_0) \tag{20}$$

subject to:

$$\hat{x}_{k+1} = f(\hat{x}_k, u_k, \hat{w}_k) \tag{21}$$

This is known as the full information problem where x_0 is an estimation of the initial state. The solution of the optimization problem gives $\hat{x}_0$ and the model disturbance sequence $\{\hat{w}_k\}$. The actual state is found by iterate in the state equation (21). In the full information problem the horizon length increase with time k, this makes impracticable an on-line implementation.

Considering the objective function $\Phi_T(\cdot)$, we may rearrange the terms by breaking the time interval into two parts as follows:

$$\Phi_T\left(\hat{x}_0, \{\hat{w}_k\}\right) = \sum_{k=T-N}^{T-1} \{(y_k - C\hat{x}_k)^T R^{-1}(y_k - C\hat{x}_k) + \hat{w}_k^T Q^{-1}\hat{w}_k\} + \sum_{k=0}^{T-N-1} \{(y_k - C\hat{x}_k)^T R^{-1}(y_k - C\hat{x}_k) + \hat{w}_k^T Q^{-1}\hat{w}_k\} + (x_0 - \hat{x}_0)^T \Pi^{-1}(x_0 - \hat{x}_0) \tag{22}$$

The first term on the right depends only on the variables values at time $T - N$ so we can reformulate the problem as follows:

$$\min_{x_{T-N}, \{\hat{w}_k\}_{k=T-N}^{T-1}} \sum_{k=T-N}^{T-1} \{(y_k - C\hat{x}_k)^T R^{-1}(y_k - C\hat{x}_k) + \hat{w}_k^T Q^{-1}\hat{w}_k\} + Z_{T-N}(x_{T-N}) \tag{23}$$

subject to equation (21), where N is known as the fixed horizon length.

The function $Z(\cdot)$ is known as the arrival cost function and represent the effect of the previous data on the state x_{T-N}. The formulation (23) is the basis of MHE.

Selecting the arrival cost penalty function properly MHE can approximate the full information problem, with processing time advantages.

Unfortunately an algebraic expression for arrival cost does not exist when the system is non linear or when constraints are present. One strategy for approximating the arrival cost function is to take a first order Taylor approximation, yielding in an EKF covariance update formula (Rao, 2000).

Other strategies for approximating the arrival cost function have been studied recently (Tenny, 2002), (Haseltine, 2005).

3.4 Genetic Algorithm based Moving Horizon Estimation

Genetic algorithms are a family of computational models inspired by evolution. These algorithms encode a potential solution to a specific problem on a simple chromosome-like data structure and apply recombination operators to these structures (Michalewicz, 1996).

The GA chosen for the present application has the following characteristics:

(a) Individuals or chromosomes are encoded with real numbers (Michalewicz, 1996).
(b) A ranking operation (Baker, 1987) assigns each individual z a natural number in accordance with its minimization criteria value $J(z)$. Thus each individual has a new criteria value $J_1(z)$ corresponding to its order number.
(c) A selection operation known as Stochastic Universal Sampling (SUS) (Baker, 1987) is carried out that ensures the survival probability $p(z_i)$ of an individual z_i will be given by the equation:

$$p(z_i) = \frac{J_1(z_i)}{\sum_{j=1}^{N_{ind}} J_1(z_j)} \tag{24}$$

where N_{ind} is the number of individuals.
(d) Chromosome crossover is performed, with probability pc, using an intermediate recombination operator defined by the following transformation:

$$\begin{aligned} z_1' &= \alpha_1 z_1 + (1 - \alpha_1) z_2 \\ z_2' &= \alpha_2 z_2 + (1 - \alpha_2) z_1 \\ \alpha_1 &,\ \alpha_2 \in [0, 1] \end{aligned} \tag{25}$$

(e) The mutation of a chromosome randomly selected with probability pm is obtained by changing a maximum of 20% of its genes.

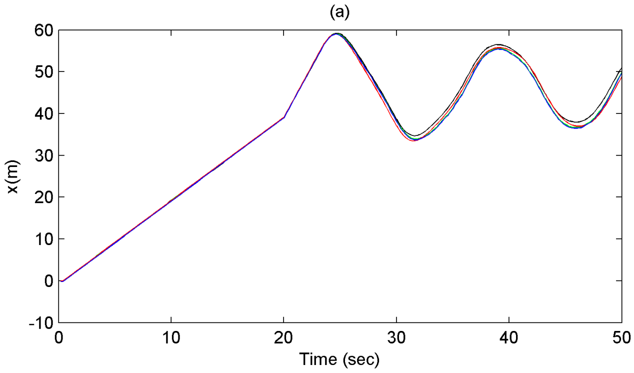
3.5 Comparative Evaluation of State Estimators

The comparison of the estimators involves a trajectory composed of a straight-line segment and a curved segment in different scenarios. For a simulation horizon of $t = 50\ s$ in a MATLAB environment.

Each scenario is characterized by variations in the following parameters with respect to the nominal situation:

(a) Covariance of process noise (Q_P) and measurement noise (R_P) for the model generating the data. The nominal values are given in the following matrices:

$$Q_P = \begin{pmatrix} 10^{-3} & 0 & 0 & 0 \\ 0 & 10^{-3} & 0 & 0 \\ 0 & 0 & 10^{-5} & 0 \\ 0 & 0 & 0 & 10^{-5} \end{pmatrix}$$



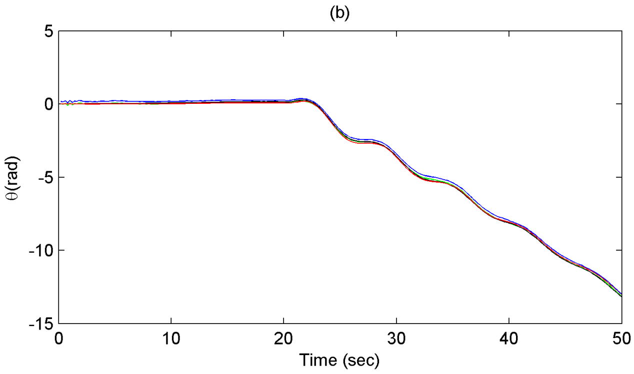


Fig. 2. State estimation, black: real state, green: EKF estimation, red: MHE estimation, blue: PF estimation; (a): non-direct observable state x; (b): direct observable state θ.

$$R_P = \begin{pmatrix} 10^{-1} & 0 \\ & 0 \; 10^{-3} \end{pmatrix}$$

(b) To represent the effects of changes in the dispersion of process and measurement noise, the nominal covariance matrices are scaled by the parameter α, which may take the following values: 0.1, 1 or 10.
(c) For EKF and MHE design in dispersion test, we assume that the values of matrices Q and R are known and equal to those for the model generating the data.
(d) For sensitive analysis the values of matrices Q and R for the design of EKF and MHE are modified as follows $Q = \alpha_Q Q_P$ and $R = \alpha_R R_P$. As in the previous point, the parameters α_Q and α_R may take the following values: 0.1, 1 or 10.
(e) Number of particles in the Particle Filter. For this parameter the following value were used: $N_P = 1200$.
(f) Initial distribution of the filter particles. Normal distribution: mean 0, standard deviation 0.2.
(g) Horizon length. A fixed horizon of $N = 10$ were used for moving horizon estimation.
(h) Genetic algorithm parameters. The following parameters were chosen using trial and error for the genetic simulation: $N_{ind} = 20$, $pc = 0.8$, $pm = 0.4$, $\delta = 0.1$.

The quantitative indicator used to compare the estimators is the root mean square error (RMSE).

Figure 2 shows the estimation of non-direct observable state x and the direct observable state θ when $\alpha = 1$.

Table 1 shows that EKF performs a better estimation for the non direct observable state x in all covariance scenarios. The four estimators perform very similar results for the estimation of θ when the parameter α goes from 1 to 10. The estimation of PF, MHE and GA based MHE for the position x was poor at the same situation. However, for the direct observable state θ and α=1, MHE performs a better estimation.

Table 1: State estimation errors depending on covariances.

	α	$x(m)$ RMSE	$\theta(rad)$ RMSE
EKF	0.1	0.072	0.010
	1	0.200	0.042
	10	1.320	0.090
PF	0.1	0.440	0.024
	1	0.520	0.042
	10	2.630	0.110
MHE	0.1	0.161	0.022
	1	0.290	0.037
	10	2.120	0.120
MHE-GA	0.1	0.168	0.022
	1	0.360	0.042
	10	2.550	0.140

Tables 2, 3, 4 present the results of the sensibility test. Both estimators perform worser when parameters α_Q and α_R takes non nominal values. EKF performs, generally, better than MHE especially for the non direct observable state x. However, MHE performs better than EKF when $\alpha_Q = \alpha_R$.

Table 2: State estimation errors for the sensibility analysis with $\alpha_Q = 0.1$.

	α_R	$x(m)$ RMSE	$\theta(rad)$ RMSE
EKF	0.1	0.341	0.047
	1	0.415	0.045
	10	0.448	0.066
MHE	0.1	0.307	0.044
	1	0.965	0.126
	10	0.741	0.059

Table 3: State estimation errors for the sensibility analysis with $\alpha_Q = 1$.

	α_R	$x(m)$ RMSE	$\theta(rad)$ RMSE
EKF	0.1	0.545	0.053
	1	0.200	0.042
	10	0.363	0.037
MHE	0.1	0.505	0.059
	1	0.290	0.037
	10	1.457	0.123

Table 4: State estimation errors for the sensibility analysis with $\alpha_Q = 10$.

	α_R	$x(m)$ RMSE	$\theta(rad)$ RMSE
EKF	0.1	0.236	0.062
	1	0.204	0.047
	10	0.405	0.042
MHE	0.1	0.298	0.071
	1	0.225	0.050
	10	0.353	0.037

4. MODEL PREDICTIVE CONTROL

MPC is designed as solving on-line a finite horizon open-loop optimization problem subject to system dynamics and constraints involving states and controls.

The LHD dynamic model in the state space is given by:

$$x_{k+1} = f(x_k, u_k, 0) \quad (26)$$

$$y_k = Cx_k \quad (27)$$

MPC searches for the optimal control sequence that is solution of the following optimization problem (Camacho and Bordons, 2004):

$$\min_{\{u_k\}_{k=1}^{N_u}} \sum_{j=N_1}^{N_2} \delta(j)\left[\hat{y}_{t+j} - yref_{t+j}\right]^2 + \quad (28)$$

$$\sum_{j=1}^{N_u} \lambda_\Delta(j)\left[\Delta u_{t+j-1}\right]^2 + \sum_{j=1}^{N_u} \lambda_u(j)\left[u_{t+j-1} - uref_{t+j-1}\right]^2$$

subject to equations (26), (27), where N_u is the control horizon, N_1 and N_2 are the prediction horizon limits, $\delta(j)$ is the reference deviation cost at time $t+j$, λ_u is the control deviation cost and $\lambda_\Delta(j)$ is the incremental control cost at time $t+j-1$.

4.1 Linear MPC

Consider the non-linear model (26), (27) and the reference trajectory given by:

$$x^*_{k+1} = f(x^*_k, u^*_k, 0) \quad (29)$$

$$y^*_{k+1} = Cx^*_k \quad (30)$$

x^*, u^* are the knowns optimal states and optimal control, that generates the trajectory (29), (30). Let we consider:

$$x = x^* + x^1 \; and \; u = u^* + u^1 \quad (31)$$

Where x^1 and u^1 represents deviations from the optimal values x^* and u^*. Then taking a first order Taylor expansion of (29) we obtain a time variant linear model of the form:

$$x^1_{k+1} = A_k x^1_k + B_k u^1_k \quad (32)$$

$$y_k = Cx^1_k \quad (33)$$

The particular cost function is given by:

$$J(N_2, N_u) = \sum_{j=1}^{N_2} \delta\left[\hat{y}_{k+j} - yref_{k+j}\right]^2 + \sum_{j=1}^{N_u} \lambda\left[\Delta u_{k+j-1}\right]^2 \quad (34)$$

MPC outputs are obtained by minimizing the deviations x^1 with minimum extra control effort u^1.

The system outputs can be predicted as follows:

$$\begin{aligned} y_{k+1} &= C(A_k x_k + B_k u_{k-1} + B_k \Delta u_k) \\ y_{k+2} &= C(A_{k+1}A_k x_k + (A_{k+1}B_k + B_{k+1})u_{k-1} \\ &\quad + (A_{k+1}B_k + B_{k+1})\Delta u_k + B_{k+1}\Delta u_{k+1}) \quad (35) \\ &\vdots \quad \vdots \end{aligned}$$

We could write (35) in matrix form:

$$Y_{k+1} = \Psi x_k + \Phi u_{k-1} + \Theta \Delta u_k \quad (36)$$

Where:

$$\Theta = \begin{pmatrix} CB_k & \cdots & \cdots & 0 \\ C(A_kB_k + B_{k+1}) & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \sum_{i=k}^{k+N_2-1} C \prod_{j=k+1}^{i} A_j B_{2k+N_2-1-i} & \cdots & \cdots & \sum_{i=k}^{k+N_2-N_u} C \prod_{j=k+1}^{k+i} A_j B_{2k+N_2-1-i} \end{pmatrix} \quad (37)$$

$$\Psi = \begin{pmatrix} CA_k \\ C\prod_{j=k}^{k+1} A_j \\ \vdots \\ C\prod_{j=k}^{k+N_2-1} A_j \end{pmatrix} \quad (38)$$

$$\Phi = \begin{pmatrix} CB_k \\ \sum_{i=k}^{k+1} C \prod_{j=k+1}^{i} A_j B_{2k+1-i} \\ \vdots \\ \sum_{i=k}^{k+N_2-1} C \prod_{j=k+1}^{i} A_j B_{2k+N_2-1-i} \end{pmatrix} \quad (39)$$

In the absence of constraints we can use the Maciejowsky equation (Camacho and Bordons, 2004), for time-invariant linear systems written in matrix form, to find a close solution for the optimal control sequence:

$$\Delta u_k = (\Theta^T\Theta + \lambda I)^{-1}\Theta^T(yref_k - \Psi\hat{x}_k - \Phi u_{k-1}) \quad (40)$$

4.2 Non-Linear MPC

Non-linear MPC is giving by solving on-line the minimization problem given by (26), (27), (28). Generally, this results in a non-convex non-linear optimization, which is difficult to solve because of the excessive computational requirements and multiple suboptimal local minima.

Non-linear optimization techniques or evolution based algorithms could be use to find the non-linear MPC outputs (Michalewicz, 1996).

4.3 Comparative Evaluation of State Estimation based Predictive Controllers

Both controller strategies were programmed in Simulink using an EKF as state estimator due his good results in the state estimation test. All simulations were executed in Simulink for a simulation horizon of $t = 35\ s$.

Figure 3 shows the obtained LHD path tracking using the receding horizon control principle in three different scenarios: open loop, linear MPC and non-linear MPC, for following control and predictive horizons: $N_u = N_2 = 5$.

Table 5 shows the root mean square error in the coordinates x, y and the mean euclidean norm of the error vector.

Clearly, non-linear controller performs better than the linear MPC. However, the computational cost is higher, indeed, the non-linear MPC takes 12 times more time to run the simulation.

Table 5: Path tracking error for MPC schemes.

	$x(m)$ RMSE	$y(m)$ RMSE	Norm(m) Mean
Open Loop	2.73	2.33	3.25
Linear MPC	2.08	1.77	2.44
Non-Linear MPC	0.93	0.24	0.85

5. CONCLUSIONS AND FUTURE RESEARCH

In our study EKF performs, generally, better than MHE, PF and GA based MHE, especially for estimating the non direct observable state x. PF produces the worst results in estimation. However, it should be noted that we assumed that the disturbances are normally distributed, which represent an advantage for EKF and MHE.

Sensitivity test shows that EKF and MHE perform depends on the knowledge of the covariance matrices, which should not be the case in a real application.

GA seems to be an attractive alternative for on-line state estimation, since the results were very similar to the obtained with classical optimization methods.

Referring to the control schemes, linear MPC responds in between the non-linear control and the open loop scenario, with shorter processing time. However, the non-linear MPC presents smaller errors.

In the next future, we will test the four estimators using a more realistic LHD model, including explicit constraints, and non-Gaussian disturbances.

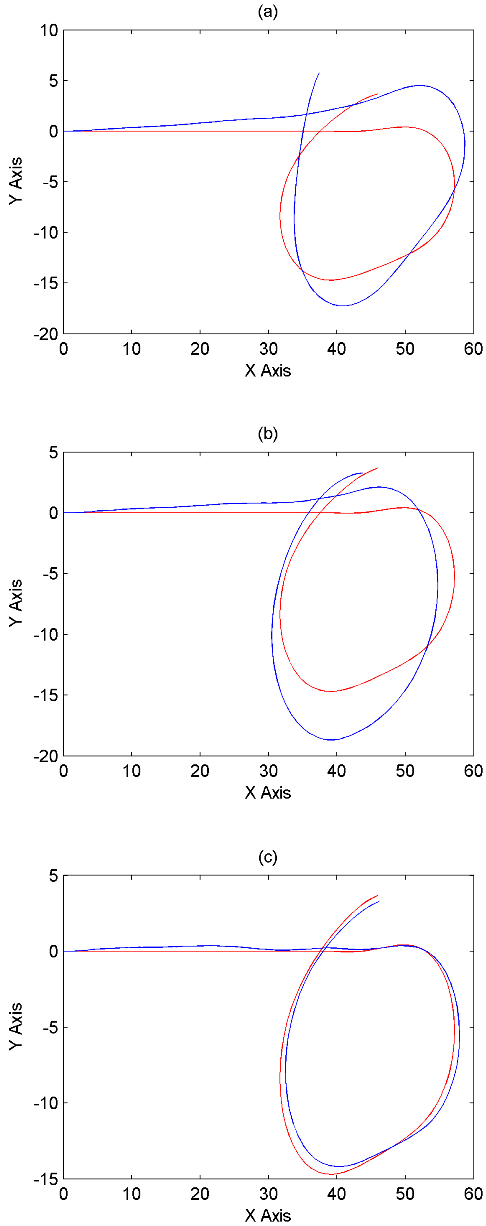


Fig. 3. Result of the simulations, red: reference trajectory, blue: LHD's real response; (a): open loop response; (b): linear MPC response; (c): non-linear MPC response.

REFERENCES

C. Altafini. A path tracking criterion for an LHD articulated vehicle. *Optimization and Systems Theory.* Royal Institute of Technology, Stockholm, Sweden, 1999.

S. Arulampalam, S. Maskell, N. Gordon and T. Clapp. A tutorial on particle filters for online non-linear/non-gaussian bayesian tracking. *IEEE Tansactions on Signal Processing.*, volume 50, pages 174–188, 2002.

E. Baker. Reducing bias and inefficiency in the selection algorithms. *Proceedings of the Second International Conference on Genetic Algorithms*, Grefenstette, Volume 1, pages 14–21, 1987

E.F. Camacho and C. Bordons. *Model Predictive Control.* Springer, 2004.

E.L. Haseltine. System analysis of stochastic and population balance models for chemically reacting systems. PhD Thesis. University of Wisconsin-Madison, 2005.

E.L. Haseltine and J.B. Rawlings. Critical Evaluation of Extended Kalman Filtering and Moving Horizon Estimation. *Ind. Chem. Res.*, volume 44, pages 2451–2460, 2005.

Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Program.* Springer series Artificial Intelligence, 3rd edition, 1996.

C.V. Rao. Moving horizon strategies for the constrained monitoring and control of nonlinear discrete-time systems. PhD Thesis. University of Wisconsin-Madison, 2000.

P. Ridley and P. Corke. Autonomous control of an underground mining vehicle. *Proceedings of the Australian Conference on Robotics and Automation*, Sydney, pages 26–31, 2001.

M.J. Tenny. Computational strategies for nonlinear model predictive control. PhD Thesis. University of Wisconsin-Madison, 2002.

G. Welch and G. Bishop. An introduction to the Kalman filter. University of North Carolina, Chapel Hill, NC, Tech. Rep. TR-95-041, 2004.