RPN TUNING STRATEGY FOR MODEL PREDICTIVE CONTROL

J. O. Trierweiler#, L. A. Farina, and R. G. Duraiski

Laboratory of Process Control and Integration (LACIP)
Department of Chemical Engineering, Federal University of Rio Grande do Sul (UFRGS)
Rua Marechal Floriano, 501, CEP: 90020-061 - Porto Alegre - RS - BRAZIL, E-MAIL: jorge@enq.ufrgs.br

Abstract: A novel tuning strategy based on RPN for MIMO MPC is presented. The RPN indicates how potentially difficult it is for a given system to achieve the desired performance robustly. It reflects both the attainable performance of a system and its degree of directionality. These system's properties are the basis of the proposed RPN-MPC tuning strategy, which is applied in the controller design of an air separation plant. Although it was only used a linear nominal model, the results can also be applied at least at some extent for nonlinear systems with uncertainties.

Keywords: - model predictive control, controllability measure, tuning strategy, RPN

1 INTRODUCTION

Model Predictive Control (MPC) is a discrete-time technique in which the control action is obtained by solving open loop optimization problems at each time step. The flexibility of this type of implementation has been useful in addressing various implementation issues that traditionally have been problematic. From a practical viewpoint, an attractive feature of MPC is its ability to naturally and explicitly handle both multivariable input and output constraints by direct incorporation into the optimization. The MPC strategy was first exploited and successfully employed on linear plants, especially in the process industries, where relatively slow sample times made extensive on-line intersample computation feasible. Recent improvements in computer power have made MPC a viable alternative approach in a variety of additional applications as well.

Dynamic matrix control (DMC) (Cutler and Raemaker, 1980) is considered the most popular MPC algorithm currently used in the chemical process industry. It is not surprising why DMC, one of the earliest formulations of MPC, represents the industry standard today. A large part of DMC appeal is drawn from an intuitive use of a finite step response model of the process, a quadratic performances objective over a finite prediction horizon, and optimal manipulated input moves computed as the solution to a least squares problem. Another form of MPC that has rapidly gained acceptance in the control community is Generalized Predictive Control (GPC) (Clarke et al. 1987). It differs from DMC in that it employs a controlled autoregressive and integrated moving average (CARIMA) model of the process which allows a rigorous mathematical treatment of the predictive control paradigm. The GPC performance objective is very similar to that of DMC. Nevertheless, GPC reduces to the DMC algorithm when the weighting polynomial that modifies the predicted output trajectory is assumed to be unity (Camacho and Bordons, 1995). The tuning strategy proposed in this paper is directly applicable to DMC and GPC. Moreover, it can be easily extend to include all other MPC control strategies.

The full dimension of the control design task consists of two parts: control structure design and controller

Author to whom the correspondence should be addressed.

design. In (Trierweiler, 1997) and (Trierweiler and Engell, 1997a) a new index, the Robust Performance Number (RPN), was introduced. RPN is a controllability measure that can be used both for control structure design and for controller design. In this paper, we present a tuning strategy for MPC algorithm based on RPN. The main idea of the proposed tuning strategy is scaling the system and weighting matrices correctly. To do that, we applied a scaling procedure based on RPN.

The paper is structured as follows: in section 2, the necessary background about MPC is presented. In section 3, the RPN is shortly discussed. Section 4 presents the RPN-tuning strategy for MPC. In section 5, the RPN-MPC tuning strategy is analyzed using an air separation plant as example.

2 BACKGROUND

A MPC algorithm employs a distinctly identifiable model of the process to predict its future behavior over an extended prediction horizon. A performance objective to be minimized is defined over the prediction horizon, usually a sum of quadratic set point tracking error and control effort terms. This cost function is minimized by evaluating a profile of manipulated input moves to be implemented at successive sampling instants over the control horizon. The feedback behavior is achieved by implementing only the first manipulated input move and repeating the complete sequence of steps at the subsequent sample time.

The various MPC algorithms propose different cost functions for obtaining the control law. A quite general expression for the objective function is:

$$J = \sum_{j=P_0}^{P} \left\| \hat{y}(t+j \mid t) - r(t+j) \right\|_Q^2 + \sum_{j=1}^{M} \left\| \Delta u(t+j-1) \right\|_W^2 \tag{1}$$

where $\hat{y}(t+j \mid t)$ is the predicted output j steps into the future based upon information available at time t, $r(t+j)$ is the reference signal j steps into the future, $\Delta u(t) = (1-z^{-1})\, u(t) = u(t)-u(t-1)$, and $\|x\|_W^2$ is the weighted Euclidean norm of $x \in \Re^n$ defined as $\|x\|_W^2 = x^T W x$ with $W \in \Re^{n\times n}$ positive definite.

The tuning parameters are the minimum costing horizon (P_0), the maximum costing horizon (P), the control horizon (M), the sampling time (t_S), the controlled variable weight (Q), and the move suppression weight (W). The weighting matrices Q and W can be chosen as time-varying (i.e., functions of j). Here, for simplicity, they are assumed to be time-invariant.

Each of the above parameters has a specific role in tuning of MPC algorithm. Using the accumulated experience of applying predictive algorithms, a number of engineering rules have been identified to obtain appropriate values of the parameters for good performance in different applications, such as:

***P_0* and *P*.** The meaning of P_0 and P is rather intuitive. They mark the limits of the time in which it is desirable for the output to follow the reference. Thus, if a high value of P_0 is taken, it is because it is unimportant if there are errors in the first instants which will provoke a smooth response of the process. Note that process with dead time there is no reason for P_0 to be less than this time delay, since the output will only begin to evolve passed this time. Equivalently, process with inverse response, the output will only to go to the final response direction after passed inverse response effect. Of course, the corresponding inverse response time can also be used to set P_0. The maximum costing horizon P should be equal or smaller than the open-loop settling time of the process in samples, since nothing is gained by costing future error in (1) that cannot be influenced by future control actions.

M. The control horizon M should be $M \le P$. The integer M specifies the degrees of freedom in selecting future controls, so that, after M future sampling intervals, the control increments are assumed to be zero, giving a constant control signal. A basic rule for selecting M in GPC algorithms is to set it at least equal to the number of unstable poles in the plant (Rawlings and Muske, 1993). It means that for stable systems, $M = 1$ can be used. Although this is a feasible choice, it should be avoided, since the controller will usually present a very poor performance. On the other side, special care must be taken if M=P and W=0, since in this case the MPC reduces to a Minimum Variance controller (Grimble, 1992), which is known to be unstable on nonminimum phase processes.

t_S. Of course, the choice of prediction horizon P cannot be made independent of the sampling time t_S. It is appropriate to relate the sampling rate to the closed-loop bandwidth of the feedback system, f_c, since f_c is related to the speed at which the feedback system should track the command input. Also, the bandwidth f_c is related to the amount of attenuation the feedback system must provide in face of plant disturbances. As a general rule of thumb, the sampling period t_S should be chosen in the range (Santina et al. 1996)

$$\frac{1}{30 f_c} < t_S < \frac{1}{5 f_c}\,. \tag{2}$$

Note that the parameters P_0, P, and M have a direct influence on the size of matrices required to compute the optimal control, and thus on the amount of computation involved.

Usually the weighting matrices Q and W are diagonal matrices, whose the elements are tuned to achieve the desired performance in closed loop and to scale the inputs and outputs making the units of measurements and manipulated variables comparable. Additionally, they are also used for:

Q. It is possible to achieve tighter control of a particular measured output by selectively increasing the relative weighting element.

W. The role of W is to penalize excessive incremental control actions. The larger the value of W, the more sluggish the control will become.

The parameters W and M are strongly related to each other. Regardless, M must be in the range *Number of RHP-polos* $\leq M \leq P$, it continues to be an important tuning parameter. For stable process, M=1 and W=0 can produce acceptable result, but quite better performance can be achieve with $M > 1$ and $W > 0$.

Q, W, and M can be seen as the main parameters to be manipulated to improve the control performance. Of course, the other parameters are also important, but they are well determined by the process dynamics. In section 4, it is shown how the MPC tuning parameters should be set to achieve a given attainable performance. The tuning methodology is based on the Robust Performance Number (RPN) which will now be introduced.

3 RPN – ROBUST PERFORMANCE NUMBER

The Robust Performance Number (RPN) was introduced in (Trierweiler 1997; Trierweiler and Engell, 1997a) as a measure to characterize the controllability of a system. The RPN indicates how potentially difficult it is for a given system to achieve the desired performance robustly. The RPN is influenced both by the desired performance of a system and its degree of directionality.

3.1 Definitions

The Robust Performance Number (RPN, Γ) of a multivariable plant with transfer matrix $G(s)$ is defined as

$$\text{RPN} \overset{\Delta}{=} \Gamma_{\sup}(G,T,\omega) = \sup_{\omega \in \mathbb{R}} \{\Gamma(G,T)\} \tag{3a}$$

$$\Gamma(G,T) \overset{\Delta}{=} \sqrt{\bar{\sigma}\left(\left[I - T(j\omega)\right]T(j\omega)\right)\left(\gamma^*\left(G(j\omega)\right) + \frac{1}{\gamma^*\left(G(j\omega)\right)}\right)} \tag{3b}$$

where $\gamma^*(G(j\omega))$ is the *minimized* condition number of $G(j\omega)$ and $\bar{\sigma}([I\text{-}T]\ T)$ is the maximal singular value of the transfer function matrix $[I\text{-}T]\ T$, being T the (attainable) desired output complementary sensitivity matrix, which is determined for the nominal model $G(s)$.

3.2 Attainable Performance

In this section, it is discussed how the attainable closed loop performance can be characterized for systems with RHP-zeros.

Specification of the desired performance

We specify the desired performance by the (output) complementary sensitivity function T which relates the reference signal r and the output signal y in the 1 degree of freedom (DOF) control configuration (see fig. 1). For the SISO case, specifications as settling time, rise time, maximal overshoot, and steady-state error can be mapped into the choice of a transfer function of the form

$$T_d \overset{\Delta}{=} \frac{1-\varepsilon_\infty}{\left(\dfrac{s}{\omega_n}\right)^2 + 2\zeta\dfrac{s}{\omega_n} + 1} \tag{4}$$

where ε_∞ is the tolerated offset (steady-state error). The parameters ω_n (undamped natural frequency) and ζ (damping ratio) of (4) can be easily calculated from the time domain specifications, as it can be seen in Trierweiler (1997).

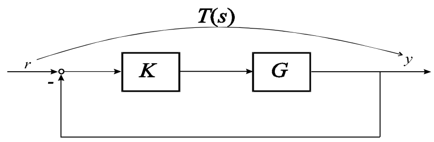

Fig. 1: Standard feedback configuration

For the MIMO case, a straightforward extension of such a specification is to prescribe a decoupled or almost decoupled response, with possibly different parameters for each output, i.e., $T_d = diag(T_{d,1},...,T_{d,no})$, where each $T_{d,i}$ corresponds to a SISO time domain specification.

Factorization of systems with RHP-zeros and RHP-poles

To satisfy the RHP-zeros and –poles constrains it is possible to make use of the Blaschke input and output factorization (for the definition of the factorization and an algorithm to calculate it, see, e.g., Havre and Skogestad (1996) or Trierweiler (1997)).

The attainable performance for the case when the plant $G(s)$ has both RHP-zeros and –poles can easily be obtained (Trierweiler, 1997) and is given by

$$T(s) = B_{O,z}(s)\, B_{O,z}^{\dagger}(0) \left[I - \left(I - T_d(s)\right) B_{I,p}^{\dagger}(0)\, B_{I,p}(s)\right]. \tag{5}$$

where $B_{O,z}(s)$ and $B_{I,p}(s)$ are the zero output and pole input Blaschke factorization, respectively, and the operator $B^{\dagger}$ denotes the pseudo-inverse of B, in such way that $B(0)\,B^{\dagger}_{r}(0) = I$.

$T(s)$ is different from the original desired transfer function $T_d(s)$, but has exactly the same singular values, being also the specified robustness properties preserved at the plant output. The factors $B_{O,z}^{\dagger}(0)$ and $B_{I,p}^{\dagger}(0)$ ensures that $T(0) = T_d(0)$ so that the steady-state characteristics (usually $T_d(0) = I$) are preserved.

3.3 RPN-Scaling Procedure

The scaling of the transfer matrix is very important for the correct analysis of the controllability of a system and for controller design. In the definition of $\gamma^*(G(j\omega))$, L and R are frequency dependent, however, in the design L and R usually are constant. The following procedure based on the RPN is recommended to be used to optimally scale a system G.

RPN-scaling procedure:

1. Determination of the frequency $\omega_{\sup}$ where $\Gamma(G,T,\omega)$ achieves its maximal value.
2. Calculate the scaling matrices L_S and R_S, such that $\gamma(L_S G(j\omega_{\sup})R_S)$ achieves its minimal value $\gamma^*(G(j\omega_{\sup}))$.
3. Scale the system with the scaling matrices L_S and R_S, i.e., $G_S(s) = L_S\, G(s)\, R_S$

The analysis and controller design should then be performed with the scaled system G_S.

4 RPN-TUNING STRATEGY

Before starting to design the controller, it is necessary to determine how difficult the control problem is. For it, we calculate the RPN for the system using an attainable desired performance T, which is a function both of nonminimum-phase behavior of the system and of the closed loop desired performance. The RPN is a measure of how potentially difficult it is for a given system to achieve the desired performance robustly. The easiest way to design a controller is to use the process inverse. An inverse-based controller will have potentially good performance robustness only when the RPN is small. Then, the first step of our tuning strategy, which is summarized in Table 1, consists of specifying the desired performance. The second step is the factorization of $G(s)$ and determination of an attainable performance T. The third step corresponds to the application of RPN-scaling procedure to calculate the scaling matrices L_S and R_S.

The forth step is the choice of sampling time t_S. Based on (2) and on the desired performance (4), The sampling time, t_S, can be expressed as a function of the rise time τ_r as follows (Santina et al., 1996): $0.06\tau_r < t_S < 0.4\tau_r$.

Table 1: MPC tuning procedure for MIMO-systems based on RPN

1. Specification of the desired performance T_d
2. Determination of an attainable performance by the factorization of the nominal model
3. RPN-scaling procedure where the scaling matrices L_s and R_s are determined as shown in section 3.3
4. Sampling time: $0.06\tau_r < t_S < 0.4\tau_r$
5. Two possibilities for costing horizon:

   $$(\mathrm{A})\, P \approx \frac{t_{80\%}}{t_S}, P_0 = 0, \text{ and } r_s(s) = L_s T(s) r_d(s)$$

   $$(\mathrm{B})\, P \approx \frac{t_{90\%}}{t_S}, P_0 \approx \frac{t_{10\%}}{t_S}, \text{ and } r_s(s) = L_s r_d(s)$$

6. Control horizon: $M \approx P/4$
7. The control action u_S should be calculated using the following scaled objective function:

   $$J_s = \sum_{j=P_0}^{P} \left\| \hat{y}_s(t+j\,|\,t) - r_s(t+j) \right\|_Q^2 + \sum_{j=1}^{M} \left\| \Delta u_s(t+j-1) \right\|_W^2$$

   where $\hat{y}_s(t+j\,|\,t)$ is the predicted output j steps into the future based upon the scaled model $G_S(s)$, i.e., $G_S(s) = L_S\, G(s)\, R_S$.
8. Input and output weighting matrices:

   $$Q = \frac{1}{\sqrt{1+y_{Z,s}}} \text{ and}$$

   $$W = \sqrt{(1+u_{Z,s})\log_{10}(RPN+1)}\; mean\left(\left| g_s^{i,j}(\omega_{\sup}) \right|\right)$$

   where y_Z and u_Z are respectively the output and input zero directions of the RHP-zero closest to the origin.
9. Back to the original units of the manipulated variables, i.e., $u = R_S \cdot u_S$

In step 5, P_0, P, and r are determined. The minimum (P_0) and maximum (P) costing horizons are related to closed loop system's dynamic. For stable process, it can be expressed as a function of the open loop system's dynamic. As already mentioned, the maximum costing horizon P should be equal or smaller than the open-loop settling time of the process in samples, since nothing is gained by costing future error. If the reference signal r is based on the attainable performance (i.e., $r_S = L_S\, T(s) r_d(s)$, where r_d is the desired reference signal), P_0 can be made 0, since the nonminimum-phase behavior is considered in r_S automatically. In this case, $P \approx t_{80\%}/t_S$ will usually give very good results. $t_{80\%}$ corresponds to the time when the response of $G(s)$ to an input step reaches 80% of its final value. For multivariable systems the maximal value of $t_{80\%}$ for all outputs and inputs should be used. When r_S is not an attainable performance, P_0 and P should approximately be equal to $t_{10\%}/t_S$ and $t_{90\%}/t_S$, respectively, where $t_{10\%}$ and $t_{90\%}$ correspond to the time when the response of $G(s)$ to an input step reaches 10% and 90% of its final value.

In step 6, the control horizon M is chosen approximately equal to $P/4$. This value is a good compromise between performance and stability. Step 7 calculates the control action u_S using the scaled objective function:

$$J_s = \sum_{j=P_0}^{P} \left\| \hat{y}_s(t+j \mid t) - r_s(t+j) \right\|_Q^2 + \sum_{j=1}^{M} \left\| \Delta u_s(t+j-1) \right\|_W^2 \tag{6}$$

where $\hat{y}_s(t+j \mid t)$ is the predicted output j steps into the future based upon the scaled model $G_S(s)$, i.e., $G_S(s) = L_S\, G(s)\, R_S$, using

$$Q = \frac{1}{\sqrt{1 + y_{Z,s}}} \text{ and}$$

$$W = \sqrt{\left(1 + u_{Z,s}\right)\log_{10}(RPN+1)}\; mean\left(\left| g_s^{i,j}(\omega_{\sup}) \right|\right) \tag{7}$$

as weighting matrices. Here y_Z and u_Z are the output and input zero directions of the RHP-zero closest to the origin calculated for the scaled system $G_S(s)$. These directions give an idea how the RHP-zero effect is distributed on the output (y_Z) and input (u_Z), respectively. The idea is to apply in less extension the manipulated variables where the RHP-zero effect is concentrated. In principle, if we consider y_Z and u_Z equal to 1, it will produce almost the same result. Therefore, we recommend the users to use y_Z and u_Z just as guideline. The RPN factor is included in W to penalize excessive incremental control actions. The larger the value of RPN, the more sluggish the control will become. The factor $mean\left(\left| g_s^{i,j}(\omega_{\sup}) \right|\right)$ is included to make the second term independent of the scaling matrices L_S and R_S.

Finally, to apply the control action to the system, the scaled control action u_S should be restored to the original units, i.e., $u = R_S \cdot u_S$.

5 A HEAT INTEGRATED AIR SEPARATION PLANT

Here, it is analyzed the control structures ST_632 for air separation unit studied in (Trierweiler and Engell, 2000). This control structure has several RHP-zero close to the origin and RPN=6.3. Since this control structure is a difficult control problem and, therefore, more indicate to show the benefits of the proposed MPC tuning procedure.

Here, all simulations were made using the functions MPCCON and MPCSIM of the MATLAB MPC-toolbox (Morari and Ricker,1994). These functions use a model in step format as the DMC algorithm. Similar results are also obtained by the corresponding state-space functions, i.e., SMPCCON and SMPCSIM. The desired performance used in the simulations was: y1-u6 (5 min rise time and 1% overshoot), y2-u3 (30 min rise time and 10% overshoot), and y3-u2 (30 min rise time and 10 % overshoot). The attainable performance T is considerable different from T_d for ST_632, since this control structure has RHP-zeros close to origin.

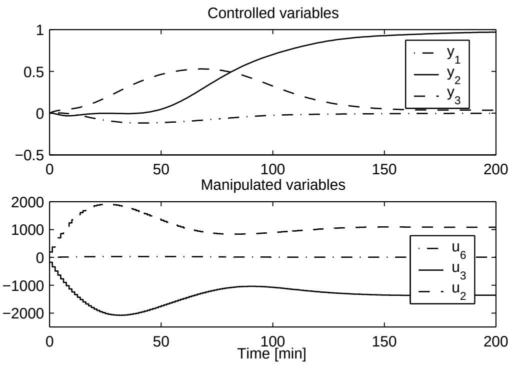

Fig. 2: Setpoint step in y_2, P=100, M=26, Q and W as in Table 1, and r_S=$L_S T(s) r_d(s)$ applied to $G_S(s)$

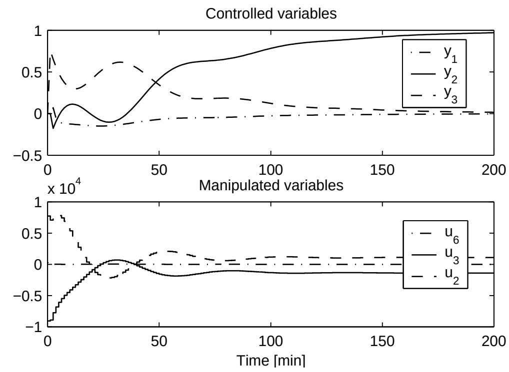

Fig. 3: Setpoint step in y_2 using controller parameters P=100, M=26, Q=$\underline{I}$, W=$\underline{0}$, and G(s)

Fig. 2 shows the simulation of a MPC controller with the tuning parameters based on RPN-MPC-tuning procedure (see Table 1) to a setpoint step in y_2 and t_S=1.25 min. The calculated controller parameters was $P = 100$, $M = 26$, scaling matrices $L_S = diag([240, 147, 82])$ and $R_S = diag([1, 8.6, 8.1])$ and weighting matrices $Q = diag([0.97, 0.77, 0.77])$ and $W = diag([1.51, 1.95, 1.97])$. Fig. 3 shows the same simulation for Q=$\underline{I}$ and W=$\underline{0}$, and the unscaled system. The MPC tuned by RPN-tuning procedure has a considerably better performance. Moreover, the control action is much smaller than for the untuned case (cf. Figs 2 and 3).

Of course, the quite better performance of RPN tuned MPC is strongly related to the reference signal *T* which includes the nonminimum-phase behavior of ST_632 automatically. In Fig. 4, we can see the closed loop performance for the case when *T* is not used as reference signal. Observe that already the correct scaling can considerably improve the nominal system performance.

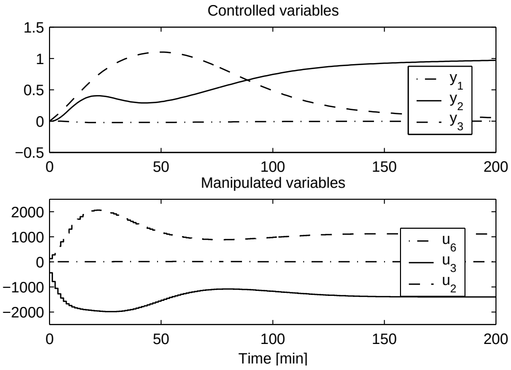

Fig. 4: Setpoint step in y_2 tuned as in Fig. 2, but without reference trajectory.

5.1 Influence of some tuning parameters

Based on simulation results, which are not shown here, we can conclude the following points:

- **Control horizon *M*:**

The control performance for M=P for the RPN tuned MPC presents the same performance as before (shown in Fig. 2), while the untuned MPC with M=P reduces to a Minimum Variance controller producing, therefore, unlimited and unrealistic control action on nonminimum phase processes.

- **Prediction horizon *P*:**

The RPN tuned MPC has almost the same performance as shown in Fig. 2 for P=50 and M=12, while the untuned MPC has decreased its performance taking more time to reach the setpoint.

In general, we can conclude that the RPN-MPC tuning strategy is much less sensible to the parameter changing. In all analyzed cases, the closed loop performance was almost the same. Moreover, the RPN tuned MPC has always produced smooth and smaller control action than the untuned MPC, indicating that the unconstrained RPN-MPC tuning procedure will also produce good results for the constrained control problem.

5.2 Applying to a commercial DMC

To apply the proposed methodology to a commercial MPC algorithm, it is just necessary to include the scaling matrices into the weighting matrices as follows: $W_{U,i} = W_i / R_{S,\,i}$ and $Q_{U,i} = Q_i \times L_{S,\,i}$
Here, W_U and Q_U are the weighting matrices for the unscaled system $G(s)$ and the subindexes i is the element (i,i) of each matrix. Note that step 9 of Table 1 should not be applied in this case.

6 CONCLUSIONS

The paper presented a novel tuning strategy for MIMO MPC. The proposed MPC tuning procedure is based on Robust Performance Number (RPN), which can measure how difficult the control problem is. The RPN, because of the dependency on the *attainable* closed-loop performance, takes the effect of nonminimum-phase behavior and the desired closed loop performance into account. In addition, the frequency dependent directionality of the system is quantified correctly and is therefore used to scale the system allowing a more robust and efficient tuning.

The performance of the RPN-MPC tuning strategy was demonstrated for a wrong selected control structure of an air separation plant. The controller design was performed using a linear nominal model, but can be extend to include nonlinearities and uncertainties in the same way as done for the controllability analysis (see, e.g., Trierweiler and Engell, 1997b and Trierweiler, 1997).

The RPN-MPC tuning procedure was also successfully applied to other examples, which are available at http://www.enq.ufrgs.br/rpn.

REFERENCES

Camacho, E.F., Bordons, B.(1995) Model Predictive Control in the Process Industry, Springer-Verlag.

Clarke, D.W., Mohtadi, C., Tuffs, P.S. (1987) Generalized Predictive Control - Part I. The Basic Algorithm, Automatica, 23(2), pp.137-148.

Cutler, C.R., Raemaker, B.L. (1980) Dynamic Matrix Control - A computer control algorithm, Proc. Automatic Control Conference, San Francisco, CA.

Grimble, M. J. (1992) Generalised predictive control: An introduction to the advantages and limitations, Int. J. Syst. Sci., 23, 85,

Havre, K, Skogestad, S (1996). Effect of RHP Zeros and Poles on Performance in CONTROL'96, Exeter, UK, pp. 930-935.

Morari, M; Ricker, N.L.(1994) Model Predictive Control Toolbox- for use with Matlab - V1.0.

Rawlings, J. B. and Muske, K. (1993). The Stability of Constrained Receding Horizon Control. IEEE Trans. Autom. Control, 38(10), pp.1512-1516.

Rohde, W. (1994) Production of pure argon from air, Reports on Science and Tech. 54, Linde A.G., pp.3-7

Santina, M.S., Stubberud, A.R., Hostetter, G.H., (1996) Sample-Rate Selection, in the Control Handbook, CRC press, pp.313-321.

Skogestad, S. and Postlethwaite, I.(1996). Multivariable Feedback Control - Analysis and Design, John Wiley

Trierweiler, J.O. and Engell, S. (1997a). The Robust Performance Number: A New Tool for Control Structure Design, Comp. chem. Eng., (21), Suppl. (PSESCAPE-7), pp.S409-414.

Trierweiler, J.O. and Engell, S. (1997b). Controllability Analysis via the Robust Performance Number for a CSTR with Van de Vusse Reaction, Proc. ECC '97, TU-A-H3.

Trierweiler, J.O. and Engell, S. (2000). A Case Study for Control Structure Selection: Air Separation plant, *Journal of Process Control, pp.* 237-243.

Trierweiler, J.O. (1997). A Systematic Approach to Control Structure Design, Ph.D. Thesis, University of Dortmund.