Decentralized scenario-based plug and play MPC for linear systems with multiplicative uncertainties

To cite this article: Anna Nehring and Ping Zhang 2017 J. Phys.: Conf. Ser. 783 012023

View the article online for updates and enhancements.

Related content
- Stability of distributed MPC in an intersection scenario
  T Sprodowski and J Pannek
- Corrigendum: Environmental implications of large-scale adoption of wind power: a scenario-based life cycle assessment
  Anders Arvesen and Edgar G Hertwich
- Analysis of Sensitivity of Accelerometer Using Multilayer Piezoelectric Ceramics
  Ken Sato, Kohichi Okamoto, Yoshiaki Fuda et al.
Decentralized scenario-based plug and play MPC for linear systems with multiplicative uncertainties

Anna Nehring, Ping Zhang
Institute of Automatic Control, University of Kaiserslautern, Germany
E-mail: nehring@eit.uni-kl.de, pzhang@eit.uni-kl.de

Abstract. This paper proposes a novel approach to decentralized control with Plug and Play capabilities. Plug and Play allows a flexible addition and removal of subsystems to an existing plant. The proposed approach guarantees stability and robustness to a certain degree for systems with additive disturbances and multiplicative uncertainties. For the controller design, the plant is decomposed into subsystems. The algorithms for adding and removing subsystems are given. By applying the basic idea of scenario-based methods, the proposed approach is less conservative and can handle complex multiplicative uncertainties.

1. Introduction
Plug and Play applied to control systems is a principle where new subsystems can be added to or removed from a system without the need of redetermining all controller parameters for the system [1], [2]. This principle can be very useful in large-scale systems and distributed systems, as they usually consist of a number of subsystems. The class of studied systems varies, from cooperative systems, where all information is shared, to decentralized systems, where only static informations or no information is shared. Most approaches use a form of cooperative design. Some cooperative designs are given below.

In [3], a Plug and Play MPC scheme is introduced for distributed LTI systems. A Lyapunov approach is used, redefining the constraints of each subsystem on a Plug and Play operation, to guarantee boundedness of the states. To guarantee input-to-state stability, in [4] an approach is proposed where each subsystem communicates to other subsystem it’s desired trajectory and the guaranteed set it will lie in. In [5], a Plug and Play method is proposed using identification and parameter estimation for Plug and Play operations. When a system changes, the new system is identified and the controller is changed accordingly. In [6], a Plug and Play algorithm is presented, where network communication concepts are used to notify other subsystems of new players in the network. The overall cost function is minimized with only minimal communication between the subsystems. This method has been applied to microgrids. In [7], a Plug and Play procedure is proposed for adding and removing subsystems in a running system for an agent based architecture, where the Nyquist plots of the subsystems are used to prove the input-output stability of the whole system. A decentralized approach is proposed in [8], which utilizes only the information of constraints of neighbouring subsystems to guarantee the input-to-state stability.

To deal with additive disturbances and multiplicative uncertainties, robust MPC methods have been studied [9], [10], [11]. Robust MPC requires the state and input constraints to be satisfied for all realizations of the uncertainty and usually considers the worst case. Most of the
robust MPC approaches modify the constraint sets [12], by using offline information such as the bounds of the disturbance. One example for this is tube-based MPC which is referenced later in this paper [12]. Its basic idea is to restrict the state trajectories to a tube which always lies within the constraints regardless of a (bounded) disturbance, which makes the control scheme rather conservative in some cases. The $\mathcal{H}_\infty$-optimization based robust MPC approach takes the $\mathcal{H}_\infty$-norm as the cost function to be minimized under constraints [10]. Mainly used for nonlinear systems, the input-to-state stability methods [10] impose stricter conditions for the Lyapunov stability. This results in new terminal costs in the cost function and time-varying state constraints.

In order to reduce the conservatism, stochastic MPC has been developed, which does not aim at the satisfaction of all constraints [13], [14], [15], [16]. Basic ideas include the reformulation of deterministic constraints into chance constraints (which only have to be satisfied most of the time) paired with affine disturbance feedback [14] and the prediction of state trajectories via samples of the uncertainties [17], so called scenario-based approaches. For scenario-based approaches, two basic methods exist. One is based on random convex programming [18], which guarantees feasibility of the optimization problem and constraint satisfaction by requiring a minimum number of used scenarios [17], [19], [20]. The other method is based on stochastic programming [21], which utilizes a stochastic Lyapunov function to achieve feasibility and a scenario tree to generate and weight scenarios based on probability to achieve constraint satisfaction [22], [23], [24]. In [25] and [26], distributed cooperative scenario-based MPC algorithms are presented, which use information from other subsystems to determine their output sequence.

Recently, based on the tube-based MPC, a Plug and Play variant for decentralized systems has been developed [8]. In order to reduce the conservativeness of the design, we propose another way to take into account the knowledge about uncertainties and disturbances in the design. The scenario-based approach provides an interesting alternative [27]. We propose to apply the basic idea of the scenario-based approach, proposed in [27], to enhance the capabilities of the Plug and Play MPC controller to deal with uncertainties.

The main contribution of this paper is the enhancement of the decentralized control scheme proposed in [8] with the capabilities of scenario-based control as proposed in [27]. Compared with [8], the proposed approach is capable of considering complex model uncertainties and moreover, it reduces the conservatism of the solution.

This paper is organized as follows. First, in Section 2 a description of the uncertain system under consideration will be given and the basic idea of Plug and Play control will be explained. In Section 3, the decentralized scenario-based MPC will be introduced and the stability of the decentralized system will be discussed as well as Plug and Play operations. In Section 4, the proposed approach will be compared with the existing approach (i.e. the tube-based Plug and Play MPC approach) by means of an example.

2. Problem formulation

2.1. System description

In this paper, we consider the following linear system with additive disturbances and multiplicative uncertainties described by

$$x(k + 1) = A(\theta)x(k) + B(\theta)u(k) + B_\gamma(\theta)\gamma(k)$$

where $x \in \mathbb{X} \subset \mathbb{R}^n$ is the state vector, $u \in \mathbb{U} \subset \mathbb{R}^m$ is the control input vector, $\gamma \in \Gamma \subset \mathbb{R}^{m_\gamma} \subset \mathbb{R}^{\gamma}$ is the additive disturbance vector and $\theta \in \Theta \subset \mathbb{R}^\theta$ denotes the multiplicative uncertainties. The system matrices $A(\theta), B(\theta), B_\gamma(\theta)$ and the set $\Gamma$ are bounded. It is assumed that $(A(\theta), B(\theta))$ is stabilizable for all $\theta$ and

$$[A(\theta) \ B(\theta) \ B_\gamma(\theta)] = \sum_{\xi=1}^{r_\theta} \theta_\xi [A_\xi \ B_\xi \ B_\gamma_\xi], \quad 0 \leq \theta_\xi \leq 1, \quad \sum_{\xi=1}^{r_\theta} \theta_\xi = 1.$$
The additive disturbances \( \gamma \) may have time-varying distributions and may be correlated in time [28]. The constraints on \( x \) and \( u \) are characterized as zonotopes [29], i.e. centrally symmetric convex polytopes which are represented by

\[
X = \{ x \in \mathbb{R}^n | g_r x \leq 1, r = 1, \ldots, r_x \} = \{ x \in \mathbb{R}^n | G x \leq 1_{r_x} \} \tag{3}
\]

\[
U = \{ u \in \mathbb{R}^m | h_r u \leq 1, r = 1, \ldots, r_u \} = \{ u \in \mathbb{R}^m | H u \leq 1_{r_u} \} \tag{4}
\]

where \( r_x, r_u \) represent the number of constraints, \( g_r \in \mathbb{R}^{1 \times n} \), \( h_r \in \mathbb{R}^{1 \times m} \) are row vectors with only one non-zero entry, \( G = [g_1^T \cdots g_{r_x}^T]^T \in \mathbb{R}^{r_x \times n} \), \( H = [h_1^T \cdots h_{r_u}^T]^T \in \mathbb{R}^{r_u \times m} \), \( 1_{r_x} \) denotes a column vector of ones of length \( r_x \).

2.2. System decomposition

Assume that system (1) can be partitioned into \( M \) subsystems as described in [30] (see Chapter 3.3 therein). Partition the state vector and the control input vector as \( x = [x_1^T x_2^T \cdots x_M^T]^T \), \( u = [u_1^T u_2^T \cdots u_M^T]^T \), where \( x_i \in \mathbb{R}^{n_i}, u_i \in \mathbb{R}^{m_i}, i = 1, \ldots, M \), \( \sum_{i=1}^{M} n_i = n \), \( \sum_{i=1}^{M} m_i = m \). Correspondingly, partition the matrix \( A(\theta) \) into

\[
A(\theta) = \begin{bmatrix}
A_{11}(\theta) & \cdots & A_{1M}(\theta) \\
\vdots & \ddots & \vdots \\
A_{M1}(\theta) & \cdots & A_{MM}(\theta)
\end{bmatrix}
\]

where \( A_{ij} \in \mathbb{R}^{n_i \times n_j}, i = 1, \ldots, M, j = 1, \ldots, M \). Assume that the subsystems are input decoupled (i.e. the input \( u_i \) only affects subsystem \( S_i \)). Therefore, \( B(\theta) \) and \( B_i(\theta) \) can be partitioned into

\[
B(\theta) = \begin{bmatrix}
B_1(\theta) & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & B_M(\theta)
\end{bmatrix}, \quad B_i(\theta) = \begin{bmatrix}
B_{\gamma 1}(\theta) \\
\vdots \\
B_{\gamma M}(\theta)
\end{bmatrix}
\]

\[
B_1(\theta) \in \mathbb{R}^{n_1 \times m}, \quad B_{\gamma i}(\theta) \in \mathbb{R}^{r_i \times m}, \quad i = 1, \ldots, M \tag{6}
\]

The \( i \)-th subsystem \( S_i \), \( i = 1, \ldots, M \), is given by

\[
S_i : \quad x_i(k+1) = A_{ii}(\theta)x_i(k) + B_i(\theta)u_i(k) + w_i(k) \\
w_i(k) = B_{\gamma i}(\theta)\gamma(k) + \sum_{j \in \mathcal{N}_i} A_{ij}(\theta)x_j(k) \tag{7}
\]

where \( \mathcal{N}_i = \{ j | A_{ij}(\theta) \neq 0 \} \) is the set of neighbouring subsystems to the subsystem \( S_i \). Assume that if the subsystems are decoupled, they stay decoupled independent of the multiplicative uncertainties, i.e. if the \( i \)-th subsystem and the \( j \)-th subsystem are decoupled, then \( A_{ij}(\theta) = 0, \forall \theta \).

For the \( i \)-th subsystem, the coupling between it and other subsystems described by \( \sum_{j \in \mathcal{N}_i} A_{ij}(\theta)x_j \) is treated as an additional additive disturbance and added to the disturbance \( w_i \).

The constraint sets \( X \) and \( U \) are also partitioned into

\[
X_i = \{ x_i \in \mathbb{R}^{n_i} | g_r x_i \leq 1, r_i = 1, \ldots, r_{i_x} \} = \{ x_i \in \mathbb{R}^{n_i} | G_i x_i \leq 1_{r_{i_x}} \}, \quad \sum_{i=1}^{M} r_{i_x} = r_x \tag{8}
\]

\[
U_i = \{ u_i \in \mathbb{R}^{m_i} | h_r u_i \leq 1, r_i = 1, \ldots, r_{i_u} \} = \{ u_i \in \mathbb{R}^{m_i} | H_i u_i \leq 1_{r_{i_u}} \}, \quad \sum_{i=1}^{M} r_{i_u} = r_u \tag{9}
\]

where \( g_r \) are the vectors \( g_r \) corresponding to the constraints of \( x_i \) and \( h_r \) are the vectors \( h_r \) corresponding to the constraints of \( u_i \).
2.3. Control objective
The goal is to drive the state vector \( x \) to the target \( x_r \) with a small control effort \( u \) while being robust against additive disturbances and multiplicative uncertainties. Moreover, the computational effort should be kept relatively small for the control scheme to be applicable in real systems, which do not necessarily have much computing power. Plug and Play enables a flexible system structure allowing for changes in subsystems as well as in the number of subsystems. Changes such as the removal and the addition have to be executed with relatively easy operations. After a change in the system structure, the local controllers as well as the overall system should still be stable and perform well.

3. The new approach
In this section, an approach is proposed to combine scenario-based MPC with a Plug and Play scheme.

3.1. Controller structure
Each subsystem \( S_i, i = 1, \cdots, m \), is equipped with a controller

\( C_i : u_i(k) = K_ix_i(k) + v_i(k) \) (10)

where \( K_i \) is the local feedback gain matrix, \( v_i \) is a correction term that will be calculated later by the local MPC scheme. The main task of \( K_i \) is to stabilize the subsystem \( S_i \), while the correction term \( v_i \) is used to compensate for disturbances. The basic idea of the proposed approach is to use a Plug and Play framework to guarantee input-to-state stability and then ensure constraint satisfaction with a local scenario-based MPC scheme.

3.2. Design of the local feedback gain matrix \( K_i \)
Lemma 1 Given a linear system (1) that can be decomposed into \( M \) subsystems as (5)-(7) with the state constraints (8) and a controller given by (10). Assume that the local feedback gain matrices \( K_1, K_2, \cdots, K_M \) of the subsystems are selected so that the matrices \( F_i = A_{ii}(\theta) + B_i(\theta)K_i, i = 1, \cdots, M \) are Schur. If the following condition is fulfilled

\[
\alpha_i = \sum_{j \in \mathcal{N}_i} \sum_{k=0}^{\infty} \| G_i F_i^k A_{ij}^* G_j^* \|_\infty < 1, \quad \forall i = 1, \cdots, M
\] (11)

where \( \| \cdot \|_\infty \) denotes the largest absolute row sum of a matrix [31] and \( G_j^* \) is the pseudo-inverse of \( G_j \), then

(i) the matrix \( F = A(\theta) + B(\theta)K \) is Schur, where \( K = \text{diag}\{K_1, \cdots, K_M\} \).

(ii) the whole system (1) is asymptotically stable.

Lemma 1 tells us that if the gain of subsystem \( S_i \) caused by the other subsystems \( S_j, j \in \mathcal{N}_i \), is small enough and the disturbance \( \gamma \) is bounded, then the overall system (1) is asymptotic stable if (11) holds. In the control design, \( K_i \) is the only parameter which affects \( \alpha_i \), therefore this condition has to hold when designing \( K_i \). Lemma 1 can be proven in a way similar to [8] while taking into account the influence of multiplicative uncertainties. For the sake of clarity, the proof is put in the appendix.

The local feedback gain matrix \( K_i, i = 1, \cdots, M \), can be selected with the help of the LMI technique proposed by [32]. The state feedback matrix \( K_i \) is given by \( K_i = Y_i G_i^{-1} \), where the
square matrix $G_i$, the rectangular matrix $Y_i$ and the symmetric matrices $Q_{l,\xi}$, $Q_{2,\xi}$, $\cdots$, $Q_{r_\theta\xi}$ satisfy the following LMI
$$
\begin{bmatrix}
G_i + G_i^T - Q_{l,\xi} & G_i^T A_i \xi + Y_i^T B_i \xi \\
A_i G_i + B_i Y_i & Q_{l,\xi}
\end{bmatrix} > 0, \quad \forall \xi = 1, \cdots, r_\theta, \forall l = 1, \cdots, r_\theta.
$$
(12)
The solution of this LMI is not unique and has to satisfy (11). If no solution can be found, the subsystem cannot be plugged in using the proposed method. Equation (12) represents Lyapunov functions for the vertices of the uncertainty (2).

### 3.3. MPC scheme

Scenario-based MPC is used in this approach, as it is a way of dealing with additive and multiplicative uncertainties. The randomized finite horizon control problem is formulated here. The technical details of scenario-based MPC theory are largely omitted here due to limitations of space. For further informations, the reader is referred to [28], [33], [34] and the references therein. The reliability of the solution, the number of scenarios and the basic setup are discussed below.

For each subsystem a local MPC scheme is used. Let $N_i$ be the chosen prediction horizon length. In the proposed approach, the disturbance is dealt with by considering a discrete set of state trajectories $x_i$ obtained for a number $L_i$ of randomly extracted samples of $\theta, \gamma$ and $x_j$ at the sampling time $k$, so called scenarios. These random parameters are collected in $\delta = (\theta, \gamma, x_j)$. These samples can be generated based on the probability distribution if it is known. Otherwise, the samples can be obtained from empirical data or by sampling a stochastic model [28]. It is assumed that enough samples of $\delta$ can be generated. By substituting (10) into the system model (7), the dynamics of the subsystem $S_i$ is governed by
$$
x_i(k+1) = F_i(\theta)x_i(k) + B_i(\theta)v_i(k) + w_i(k, \delta)
$$
(13)
Based on the random selected samples $\delta^1, \delta^2, \cdots, \delta^{L_i}$, different state and input predictions over a prediction horizon of length $N_i$ can be obtained from (13) as
$$
x_i^l(0) = x_i(0),
$$
$$
u_i^l(k) = K_i x_i^l(k) + v_i(k), \quad k = 0, \cdots, N_i - 1
$$
(14)
Let the cost function be
$$
J_i(v_i) = \sum_{l=1}^{L_i} \sum_{k=0}^{N_i-1} \{(x_i^l(k) - x_{i,r}(k))^T Q_i (x_i^l(k) - x_{i,r}(k)) + v_i(k)^T R_i v_i(k)\}
$$
(15)
where $Q_i \in \mathbb{R}^{n_1 \times n_1} > 0$ and $R_i \in \mathbb{R}^{n_i \times n_i} > 0$ are weighting matrices chosen by the designer. The optimization problem is
$$
\min_{v_i} J_i(v_i)
$$
(16a)
subject to
$$
x_i^l(k+1) = F_i(\theta^l)x_i^l(k) + B_i(\theta^l)v_i(k) + w_i(k, \delta^l),
$$
$$
x_i^l(0) = x_i(0),
$$
$$
u_i^l(k) = v_i(k) + K_i x_i^l(k),
$$
$$
G_i x_i^l(k) \leq 1_{r_{x_i}},
$$
$$
H_i u_i^l(k) \leq 1_{r_{u_i}},
$$
$$
k = 0, \cdots, N_i - 1, \quad l = 1, \cdots, L_i
$$
(16b-16h)
The optimization problem (16a) can be solved with quadratic programming [35]. In comparison with [8], the scenario-based optimization can deal with the time varying parameters $A_i(\theta)$, $B_i(\theta)$ and $B_{x_i}(\theta)$ of the uncertain system (7).

Since scenario-based MPC does not aim at satisfying all constraints at all times, the hard constraints are transformed into chance constraints which only have to be satisfied with a probability $R_i$

$$R_i = \text{Prob}\{H_i u_i(k) \leq 1 \text{ and } G_i x_i(k) \leq 1\},$$

where $\text{Prob}$ refers to the probability of the event. The reliability of the optimal solution $v_i^*$ to the optimization problem (16a) is defined as the probability $R_i$ that the optimal solution satisfies the constraints under all possible realizations of the disturbances and uncertainties. Note that $R_i \in [0,1]$ is itself a random variable, since it depends on the extracted scenarios $\theta_i$, i.e. $R_i = R_i(\delta)$. In [36] a connection between the number of scenarios $L_i$ and the probability of constraint violations (17) for the optimal solution $v_i^*$ is given.

**Lemma 2 ([36])** Given a system described by (7) and assume that $(\theta^1, \gamma^1, x^1), \ldots, (\theta^L, \gamma^L, x^L)$ are $L_i$ samples of the multiplicative uncertainties $\theta$, the additive disturbance $\gamma$ and the states of the neighbouring system $x_j$, $j \in N_j$. Let $N_i$ be the chosen horizon length in an MPC scheme and $d_i = m_i N_i + q_i$ represent the decision complexity of the system, where $q_i$ is the number of soft constraints. Let $p_i$ be the acceptable reliability level for $R_i$. If the number of scenarios $L_i$ is chosen such that

$$\sum_{s=0}^{d_i-1} \binom{L_i}{s} (1 - p_i)^s p_i^{L_i-s} \leq \beta_i,$$

where $\binom{L_i}{s}$ is the binomial coefficient, then it holds that

$$\text{Prob}\{R_i(\delta) \geq p_i\} \geq 1 - \beta_i.$$

Also, the optimal solution $v_i^*$ of (16a) is a feasible solution with a confidence of at least $1 - \beta_i$.

Lemma 2 tells us that the probability that the reliability $R_i$ of the solution is higher than $p_i$ is at least $1 - \beta_i$. Note that $\beta_i$ can be chosen by the designer to be very small, for instance, in the order of $10^{-12}$. In [36] it has been proven that

$$L_i \geq \frac{2}{1-p_i} (\ln \beta_i^{-1} + d_i)$$

implies the fulfilment of condition (18).

As it can be seen in (18), the number of scenarios $L_i$ depends on the reliability of constraint fulfilment $p_i$, the confidence of a feasible solution $\beta_i$ and the decision complexity $d_i$. For larger systems this leads to a large number of scenarios that quickly becomes unmanageable due to high computational effort. In [17], scenario removal techniques are proposed to remove unlikely scenarios for a quicker optimization. However, a smaller number of scenarios $L_i$ than given in (20) leads to more constraint violations.

Each subsystem $S_i, i \in M$. may have its own reliability level $p_i$ and confidence level $\beta_i$. The subsystems $S_i$ and $S_j, j \neq i, j \in \{1, 2, \ldots, M\}$ are decoupled from each other as the states of the other systems are treated as an unknown disturbance. Therefore, the probabilities $\text{Prob}(R_i \geq p_i)$ and $\text{Prob}(R_j \geq p_j)$ are independent. The probability of constraint satisfaction for the whole system (1) is then [37]

$$\text{Prob}\left( \bigcap_{i=1}^{M} R_i \geq p_i \right) = \prod_{i=1}^{M} \text{Prob}(R_i \geq p_i).$$

(21)
3.4. Plug and Play operations

In this subsection, we shall give the details of the Plug and Play aspects of the proposed approach. The Plug and Play operations are done in offline mode.

3.4.1. Generating samples of neighbouring systems

To generate the samples of the subsystem $S_i$, a description of $S_i$ has to be provided to each coupled subsystem $S_j$, $j \in N_i$. We assume that no information is shared between the subsystems other than static constraints $G_i$, $H_i$ and system matrices $A_i(\theta)$, $B_i(\theta)$, $B_{\sigma_i}(\theta)$ during Plug and Play operations. Therefore no information about setpoints or actual values of the other subsystems can be used to generate these samples. For simplicity we generate these samples of $x_i$ as uniformly distributed in $X_i$, independent of the other subsystem $S_j$ [38].

3.4.2. Plug in operation

Let $S_{M+1}$ be a new subsystem described by $x_{M+1}(k + 1) = A_{M+1,M+1}(\theta)x_{M+1}(k) + B_{M+1}(\theta)u_{M+1}(k) + w_{M+1}(k)$ with the bounds $X_{M+1}$, $U_{M+1}$ and the neighbouring systems $N_{M+1} = \{ j | A_{M+1,j}(\theta) \neq 0, j = 1, \cdots, M \}$. To get the new local feedback gain matrix $K_{M+1}$, the LMI (12) has to be solved under condition (11) for $i = M + 1$. Since $S_{M+1}$ is now a new neighbour to the subsystems $S_j$, $j \in N_{M+1}$, the sets $N_j$ change accordingly and $\alpha_j$ may change. When $N_j$ grows in size, $\alpha_j$ can only increase and therefore the controller $K_j$, $j \in N_{M+1}$ may have to be redesigned with (12) so that condition (11) is still satisfied. The state of the new system $S_{M+1}$ has to be added to the disturbance vectors $w_j$ of the neighbouring subsystems $S_j$, $j \in N_{M+1}$ and the samples of $x_{M+1}$ can be drawn as described in Section 3.4.1. If no feedback gain matrix $K_i$ can be found satisfying (11) and (12), the new system $S_{M+1}$ can not be plugged in. This procedure is summarized in Algorithm 1.

**Algorithm 1:** Plug in operation of a new subsystem $S_{M+1}$

(i) Determine $A_{M+1,M+1}(\theta)$, $B_{M+1}(\theta)$, $B_{\sigma,M+1}(\theta)$ and the coupling terms $A_{M+1,j}(\theta)$ as well as the constraint sets $X_{M+1}$, $U_{M+1}$

(ii) Determine the neighbouring set $N_{M+1} = \{ j | A_{M+1,j}(\theta) \neq 0, j = 1, \cdots, M \}$

(iii) Add $M + 1$ to the neighbouring sets $N_j$ for all $j \in N_{M+1}$

(iv) Determine $K_{M+1}$ such that (11) and (12) hold

(v) For each $\alpha_j$, $j \in N_{M+1}$, check (11) and redesign $K_j$ if necessary

(vi) Add $x_{M+1}$ to the disturbance vectors of $S_j$ and $x_j$ to the disturbance vector of $S_{M+1}$ as described in Subsection 3.4.1

(vii) Choose $p_{M+1}$, $\beta_{M+1}$ (see Subsection 3.4.4)

(viii) Optional: change $p_i$, $i = 1, \cdots, M + 1$, so that the reliability level for the whole system $p$ holds its prespecified value (see Subsection 3.4.4)

---

3.4.3. Plug out operation

Let $S_{i,N}$ be a subsystem with a description $S_{i,N}$ that is to be used. The Plug out operation of subsystem $S_{i,N}$ is described as follows:

**Algorithm 2:** Plug out operation of subsystem $S_i$

(i) Remove $\lambda$ from $N_j$ for all $j \in N_{i,N}$

(ii) Remove $x_{\lambda}$ from the disturbance vectors of $S_j$, $j \in N_{i,N}$

(iii) Optional: redesign $K_j$, $j \in N_{i,N}$ for better performance

(iv) Optional: change $p_i$, $i = 1, \cdots, M$, such the reliability level for the whole system $p$ holds its prespecified value.
Algorithm 3: Design of a distributed scenario-based Plug and Play MPC controller for the system described by (1)

(i) Decompose the system (1) into $M$ subsystems in the form of (7)

(ii) For each subsystem $S_i$, $i = 1, \ldots, M$, execute the plug in operation detailed in Algorithm 1 and obtain the local feedback controller gain $K_i$

(iii) For each subsystem $S_i$, generate $L_i$ samples for $\gamma$, $\theta$ and each $x_j$, $j \in \mathcal{N}_i$ coupled to $S_i$

(iv) Solve the optimization problem (16a) for each subsystem, e.g. with quadratic programming

(v) For each subsystem $S_i$, apply the first element of $v_i(k)$ from the solution of (16a) to $u_i = K_i x_i + v_i$

(vi) To add a new subsystem $S_{M+1}$: execute Algorithm 1

(vii) To remove a subsystem $S_{\lambda}$: execute Algorithm 2

(viii) To change a subsystem $S_{\lambda}$: execute Algorithm 2 and then Algorithm 1

3.4.3. Plug out operation. The unplugging of subsystem $S_{\lambda}$, $\lambda \in \{1, \ldots, M\}$ is comparatively easy. In this case, the state $x_{\lambda}$ has to be removed from all neighbouring disturbance vectors $w_j$. Since for each neighbouring subsystem $S_j \in \mathcal{N}_{\lambda}$ the set $\mathcal{N}_j$ gets smaller with the disconnection of subsystem $S_{\lambda}$, $\alpha_j$ can only decrease and the inequality (11) still holds. Therefore, the controllers $K_j$, $j \in \mathcal{N}_{\lambda}$ do not have to be redesigned upon removal of $S_{\lambda}$. Algorithm 2 summarizes this procedure.

Remark 1 A redesign of $K_j$, $j \in \mathcal{N}_{\lambda}$, might improve the local control performance as it might be less conservative.

3.4.4. Reliability of the subsystems. The designer has to make a choice between keeping the reliability level of the whole system $p$ at a certain value and keeping the reliability level of the subsystems $p_i$ constant. In the first option, the reliability $p$ of the whole system is fixed. When a subsystem is plugged in, the reliability of all subsystems $p_i$ has to change according to (21) such that $p$ is still above the specified value. Here the number of scenarios changes with the number of subsystems and grows larger with each new subsystem. The second option fixes the reliability $p_i$ for each subsystem. Here the number of scenarios $L_i$ is constant, but the reliability of the whole system $p$ changes with the number of subsystems.

The design procedure for the proposed controller of (1) is summarized in Algorithm 3.

4. Example
In this section, an example of building climate control is given to illustrate the proposed approach. Consider a system described by

$$x(k+1) = A(\theta)x(k) + B(\theta)u(k) + B_\gamma(\theta)\gamma(k)$$

$$A = \begin{pmatrix}
0.200 + \theta & 0.001 + \theta & 0.001 + \theta \\
0.002 + \theta & 0.300 + \theta & 0 \\
0.002 + \theta & 0 & 0.400 + \theta
\end{pmatrix},$$

$$B = \begin{pmatrix}
0.010 + \theta & 0 & 0 \\
0 & 0.010 + \theta & 0 \\
0 & 0 & 0.010 + \theta
\end{pmatrix},$$

$$B_\gamma = \begin{pmatrix}
0.020 + \theta \\
0.020 + \theta \\
0.020 + \theta
\end{pmatrix}$$

(22)

where the uncertainty $\theta$ is truncated normally distributed with mean 0, variance 1 and a maximal magnitude of 0.03, i.e. $\theta$ varies between $-0.03$ and 0.03, the additive disturbance $\gamma$ is a sine signal.
with a amplitude of 10 and an unknown offset of maximal 10%. In the simulation, the samples are generated using random number generators for known distributions. The system matrices are indeed a simplified model of a three-room-building. In this context the states $x$ represent the room temperature, the uncertainty $\theta$ can be anything from open windows to modelling errors and $\gamma$ can be, for instance, the outside temperature of a winter day influencing the room temperature of the building. The input $u$ represents a climate control unit which can provide heating as well as cooling.

The initial state is $x(0) = [15, 15, 15]^T$ and the target constraints are $[21, 21, 21]^T \leq x \leq [25, 25, 25]^T$, $[-1500, -1500, -1500]^T \leq u \leq [1500, 1500, 1500]^T$ due to actuator saturation.

The goal of the controller design is to drive all states to $x_r = [23, 23, 23]^T$ while being robust to the external disturbance $\gamma$ as well as the uncertainty $\theta$.

The system can be decomposed into three subsystems. For instance, the first subsystem is

$$S_1 : x_1(k+1) = A_{11}(\theta)x_1(k) + B_1(\theta)u_1(k) + w_1(k)$$

where $A_{11}(\theta) = 0.2 + \theta$, $B_1(\theta) = 0.01 + \theta$ and $w_1 = (0.001 + \theta)x_2 + (0.001 + \theta)x_3 + (0.02 + \theta)\gamma$.

Since the constraints have to be zonotopes, the state $x_1$ has to be shifted to $x_1 - 23$. The $X_i$ can be written in the form given in (3):

$$X_1 = \left\{ x_1 \in \mathbb{R}^1 \left| \begin{bmatrix} -0.5 & 0.5 \end{bmatrix} x_1 \leq \begin{bmatrix} 1 & 1 \end{bmatrix} \right. \right\}$$

In Figure 1 the disturbance $\gamma$ (i.e. the outside temperature) is shown, which is simplified as a sine signal. In Figure 1(b) the changes of the system matrix $A_{11}(\theta)$ due to the existence of model uncertainty $\theta$ are shown over the time.

At first only the first subsystem is examined. The reliability level of subsystem $S_1$ is set to $p_1 = 0.9$ and the confidence level to $\beta_1 = 10^{-9}$. According to the formula (20), in the optimization 855 scenarios are used. By solving (12), it gives $Y_1 = 0.58$, $G_1 = 1$, $Q_{11} = 1$, $Q_{12} = 1$. As a result $K_1 = Y_1G_1^{-1} = 0.58$. As $N_1 = 0$, $a_1 = 0 < 1$. At each time instant, $v_1$ is determined by solving the optimization problem (16a). The controller designed with the proposed approach is compared with the Plug and Play tube-based MPC controller, which is constructed as shown in [8].

Figure 2(a) shows the state trajectory of $x_1$ with $\theta = 0$. It can be seen that both trajectories of the tube-based control and the new approach show little difference. In Figure 2(b) the state trajectories are shown with $|\theta| \leq 0.03$ as described above. In this simulation for the validation of the performance, the states of the other subsystems are assumed to stay at $x_j = 0$ so that they have no influence on $x_1$. In the following we compare $x_1$ with $x_{1r}$ to determine the robustness of the controller [30], [31]. The maximum control error is defined as $e = |x - x_r|$ and is $e_{x_{1,\text{tube}}} = 1.52$ and $e_{x_{1,\text{scenario}}} = 0.1$. Here the proposed approach clearly outperforms the tube-based MPC control, as the tube-based controlled state is rather erratic.
Figure 2. The controlled state trajectories $x_1$ and $x_2$ got by the tube-based Plug and Play MPC controller (dotted line) and the scenario-based Plug and Play MPC controller (solid line). In (a) and (b), only the first subsystem is considered, in (c) and (d) the first and the second subsystem are considered.

Now the second subsystem $S_2$ is added by applying Algorithm 1. The new local controller gain $K_2$ is $K_2 = 0.8$. As both $\alpha_1 = 0.16 < 1$ and $\alpha_2 = 0.25 < 1$, $K_1$ does not have to be redesigned. $x_1$ is added to $w_2$ and $x_2$ is added to $w_1$ to take the interaction of the subsystems into account. In Figure 2(c) the trajectories for $x_1$ and $x_2$ with no multiplicative uncertainties $\theta = 0$ are shown and in Figure 2(d) the trajectories are shown with an uncertainty $|\theta| \leq 0.03$ present. The maximum error with scenario-based Plug and Play MPC here is $e_{x_1,\text{scenario}} = 0.11$, $e_{x_2,\text{scenario}} = 0.11$. Note that with the second subsystem added, the maximum error $e_{x_1}$ is increased by 10%. For the Plug and Play tube-based MPC controller, the control error is $e_{x_1,\text{tube}} = 1.53$ and $e_{x_2,\text{tube}} = 1.54$. All state trajectories lie within the constraints.

The addition of the third subsystem would only affect subsystem $S_1$ as it is physically decoupled from subsystem $S_2$. Due to the limitations of space, the Plug in operation of the third subsystem is omitted here.

5. Conclusions
In this paper, a decentralized scenario-based Plug and Play MPC approach is proposed. In comparison with the existing tube-based Plug and Play MPC approach, the new approach leads to smaller control errors if there are uncertainties in the system matrices. The application of the approach is demonstrated by an example in the field of building climate control. In future work, the stochastic description of the neighbouring states $x_j$ as well as scenario reduction methods will be further investigated.

Appendix A. Proof of Lemma 1
First we prove that an expansion of the original system is input-to-state stable which implies asymptotic stability for linear systems [39]. Suppose that a linear transformation matrix $T$ satisfies the conditions in [30] (see Theorem 3.3 of [30]). Then the linear transformation of the original system with respect to $T$ results in an expansion of this system. Then, the inclusion principle given in [30] (see Theorem 3.4 of [30]) is utilized. It shows that if the expansion of a system is asymptotically stable, the original system is also asymptotically stable and vice versa. Using $G_i$ as the transformation matrix and denoting $\tilde{x}_i(k) = G_i x_i(k)$, we can define the expanded system

$$\tilde{x}_i(k + 1) = (\tilde{A}_{ii}(\theta) + \tilde{B}_i(\theta)K_i)\tilde{x}_i(k) + \tilde{B}_i(\theta)v_i(k) + \sum_{j \in \mathcal{N}_j} \tilde{A}_{ij}(\theta)\tilde{x}_j(k) + \tilde{B}_{\gamma,i}(\theta)\gamma_i(k) \quad (A.1)$$

$$\tilde{A}_{ii}(\theta) = G_i A_{ii}(\theta) G_i^+, \quad \tilde{B}_i(\theta) = G_i B_i(\theta), \quad \tilde{K}_i = K_i G_i^+, \quad \tilde{A}_{ij}(\theta) = G_i A_{ij}(\theta) G_j^+, \quad \tilde{B}_{\gamma,i}(\theta) = G_i B_{\gamma,i}(\theta).$$
where $\tilde{G}_i^+$ denotes the pseudo-inverse of $G_i$.

Let $\tilde{F}_i(\theta) = \tilde{A}_i(\theta) + \tilde{B}_i(\theta)\tilde{K}_i$. The state equation can be rewritten as

$$\tilde{x}_i(k) = \tilde{F}_i^k(\theta)\tilde{x}_i(0) + \sum_{t=0}^{k-1} \tilde{F}_i^t(\theta) \sum_{j \in N_i} \tilde{A}_{ij}(\theta) \tilde{x}_j(k - t - 1) + \sum_{t=0}^{k-1} \tilde{F}_i^t(\theta) [\tilde{B}_i(\theta) \tilde{B}_{\gamma,i}(\theta)] \left[ v_i(k - t - 1) \right]$$

which satisfies the following:

$$\|G_i x_i(k)\|_\infty \leq \|G_i F_i^k(\theta) \tilde{G}_i^+\|_\infty \|G_i x_i(0)\|_\infty + \sum_{j \in N_i} \sum_{t=0}^{\infty} \sum_{t=0}^{\infty} \|G_i F_i^t(\theta) A_{ij}(\theta) \tilde{G}_j^+\|_\infty \max_{t \leq k} \|G_j x_j(k)\|_\infty$$

$$+ \sum_{t=0}^{\infty} \|G_i F_i^t(\theta) [B_i(\theta) B_{\gamma,i}(\theta)]\|_\infty \max_{t \leq k} \left\| \left[ v_i \right] \right\|_\infty$$

(A.2)

Corollary 16 in [40] states that the overall system $\tilde{x}(k+1) = (\tilde{A} + \tilde{B}\tilde{K})\tilde{x}(k) + \tilde{B}v(k) + \tilde{B}_\gamma \gamma(k)$ with $\tilde{x}(k) = \tilde{G}x(k)$, $\tilde{G} = \text{diag}(\tilde{G}_1, \cdots, \tilde{G}_M)$, is input-to-state stable as long as the spectral radius of the gain matrix $\rho(\Psi) < 1$ with $\psi_{ij}$ being the entries of $\Psi$ and

$$\psi_{ij} = \begin{cases} 0, & \text{if } i = j \\ \sum_{t=0}^{\infty} \|G_i F_i^t(\theta) A_{ij}(\theta) \tilde{G}_j^+\|_\infty, & \text{if } i \neq j. \end{cases}$$

(A.3)

Since $A(\theta), B(\theta), B_{\gamma}(\theta)$ are bounded the norm can be found via algebraic means in spite of the uncertainty.

According to Gershgorin circle theorem [41], all eigenvalues of $\Psi$ lie within a circle centred at the origin with a radius equal to the largest row sum of $\Psi$. Since (11) represents the largest row sum of $\Psi$, if (11) holds, all eigenvalues of $\Psi$ lie within the unit circle. According to Corollary 16 in [40], the overall system is input-to-state stable. Hence, the expanded system (A.1) is asymptotically stable. Based on the inclusion principle, it can be concluded that the original system (1) is asymptotically stable.

References

13th European Workshop on Advanced Control and Diagnosis (ACD 2016)  
IOP Publishing  
doi:10.1088/1742-6596/783/1/012023

[20] Zhang X, Schildbach G, Sturzenegger D and Morari M 2013 Scenario-based mpc for energy-efficient building scenario-based mpc for energy-efficient building climate control under weather and occupancy uncertainty In *Proc. of the 2013 European Control Conference*
[23] Bernardini D and Bemporad A Scenario-based model predictive control of stochastic constrained linear systems 2009 In *Proc. of the 2009 Conference on Decision and Control* 6333–6338
[32] Mao W J Robust stabilization of uncertain time-varying discrete systems and comments on “an improved approach for constrained robust model predictive control” 2003 *Automatica* **39** 1109–1112
[34] Campi M C and Garatti S The exact feasibility of randomized solutions of uncertain convex programs 2008 *SIAM Journal on Optimization* **19** 1211–1230
[41] Varga R S 2010 *Gersgorin and his circles* (Springer Science & Business Media)