1. Introduction

Due to the ability to handle control and state constraints, MPC has become quite popular recently. In order to guarantee the stability of MPC, a terminal constraint and a terminal cost are added to the on-line optimization problem such that the terminal region is a positively invariant set for the system and the terminal cost is an associated Lyapunov function [1, 9]. As we know, the domain of attraction of MPC can be enlarged by increasing the prediction horizon, but it is at the expense of a greater computational burden. In [2], a prediction horizon larger than the control horizon was considered and the domain of attraction was enlarged. On the other hand, the domain of attraction can be enlarged by enlarging the terminal region. In [3], an ellipsoidal set included in the stabilizable region of using linear feedback controller served as the terminal region. In [4], a polytopic set was adopted. In [5], a saturated local control law was used to enlarge the terminal region. In [6], SVM was employed to estimate the stabilizable region of using linear feedback controller and the estimated stabilizable region was used as the terminal region. The method in [6] enlarged the terminal region dramatically. In [7], it was proved that, for the MPC without terminal constraint, the terminal region can be enlarged by weighting the terminal cost. In [8], the enlargement of the domain of attraction was obtained by employing a contractive terminal constraint. In [9], the domain of attraction was enlarged by the inclusion of an appropriate set of slack terminal constraints into the control problem.

In this paper, the domain of attraction is enlarged by enlarging the terminal region. A novel method is proposed to achieve a large terminal region. First, the sufficient conditions to guarantee the stability of MPC are presented and the maximal terminal region satisfying these conditions is defined. Then, given the terminal cost and an initial subset of the maximal terminal region, a subsets sequence is obtained by using one-step set expansion iteratively. It is proved that, when the iteration time goes to infinity, this subsets sequence will converge to the maximal terminal region. Finally, the subsets in this sequence are separated from the state space one by one by exploiting SVM classifier (see [10,11] for details of SVM).

2. Model predictive control

Consider the discrete-time system as follows
where \( x_k \in \mathbb{R}^n, \ u_k \in \mathbb{R}^m \) are the state and the input of the system at the sampling time \( k \) respectively. \( x_{k+1} \in \mathbb{R}^n \) is the successor state and the mapping \( f : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^n \) satisfying \( f(0,0) = 0 \) is known. The system is subject to constraints on both state and control action. They are given by \( x_k \in X, \ u_k \in U \), where \( X \) is a closed and bounded set, \( U \) is a compact set. Both of them contain the origin.

The on-line optimization problem of MPC at the sample time \( k \), denoted by \( P_N(x_k) \), is stated as

\[
\min_{u(i,x) \in U} J_N(u,x) = \sum_{i=0}^{N-1} q(x(i,x_k),u(i,x_k)) + F(x(N,x_k)) \\
\text{s.t. } x(i+1,x_k) = f(x(i,x_k),u(i,x_k)) \\
x(i+1,x_k) \in X, u(i,x_k) \in U, x(N,x_k) \in X_f
\]

where \( x(0,x_k) = x_k \) is the state at the sample time \( k \), \( q(x,u) \) denotes the stage cost and it is positive definite, \( N \) is the prediction horizon, \( X_f \) denotes the terminal region and it is closed and satisfies \( 0 \in X_f \subseteq X \), \( F(\cdot) \) satisfying \( F(0) = 0 \) is the terminal cost and it is continuous and positive definite.

Consider an assumption as follows.

**Assumption 1.** For the terminal region and the terminal cost, the following two conditions are satisfied [1]:

1. \( F(\cdot) \) is a Lyapunov function. For any \( x \in X_f \), there exists
   \[
   F(x) \geq \min_{u \in U} \{q(x,u) + F(f(x,u))\}.
   \]

2. \( X_f \) is a positively invariant set. For any \( x \in X_f \), by using the optimal control resulting from the minimization problem showed in (C1), denoted by \( u_{opt} \), we have \( f(x,u_{opt}) \in X_f \).

Let \( J_N^*(x_k) \) be the minimum of \( P_N(x_k) \) and \( u_N^*(x_k) = \{u_N^*(0,x_k), \cdots, u_N^*(N-1,x_k)\} \) be the optimal control trajectory. The control strategy of MPC is that, at the sample time \( k \), \( u_N^*(0,x_k) \) is inputted into the real system and at the sample time \( k+1 \), the control inputted into the system is not \( u_N^*(1,x_k) \) but the first element of the optimal control trajectory resulting from the similar on-line optimization problem. At the sample time \( k+1 \), the state is \( x_{k+1} = f(x_k, u_N^*(0,x_k)) \) and the on-line optimization problem, denoted by \( P_N(x_{k+1}) \), is same as (2) except that \( x_k \) is replaced by \( x_{k+1} \). Similarly, let \( J_N^*(x_{k+1}) \) be the minimum of \( P_N(x_{k+1}) \) and \( u_N^*(x_{k+1}) = \{u_N^*(0,x_{k+1}), \cdots, u_N^*(N-1,x_{k+1})\} \) be the optimal control trajectory. The control inputted into the system at the sample time \( k+1 \) is \( u_N^*(0,x_{k+1}) \). So, the control law of MPC can be stated as \( u_{RH}(x_k) = u_N^*(0,x_k) \), \( k = 0, 1, 2, \cdots, \infty \).

The closed-loop stability of the controlled system is showed in lemma 1.

**Lemma 1.** For any \( x_0 \in X \), if \( x_0 \) satisfies \( x_N^*(N,x_0) \) \( \in X_f \) and assumption 1 is satisfied, it is guaranteed that, \( x_0 \) will be steered to \( 0 \) by using the control law of MPC. The proof can be found in [1].
Proof. The proof of lemma 1 is composed of two parts: the existence of feasible solution; the monotonicity of $J_N^*(\cdot)$.

Part 1. At the sample time 1, $x_1 = x^*(1,x_0) = f(x_0,u^*(0,x_0))$ is obtained by inputting $u^*(0,x_0)$ into the system, where $u^*(0,x_0)$ denotes the first element of the optimal solution of $P_N(x_0)$. It is obvious that, $u(x_1)=[u^*(1,x_0),u^*(N−1,x_0),u_{OPT}(x^*(N,x_0))]$ is a feasible solution of $P_N(x_1)$ since $x^*(N,x_0) \in X_f$ and $f(x^*(N,x_0),u_{OPT}(x^*(N,x_0))) \in X_f$ as assumption 1 shows.

Part 2. When $u(x_1)$ is used, we have

$$J_N(u(x_1),x_1) - J_N^*(x_0)$$

$$= q(x^*(N,x_0),u_{OPT}(x^*(N,x_0)))+F\{f(x^*(N,x_0),u_{OPT}(x^*(N,x_0)))\}$$

$$- q(x^*(0,x_0),u^*(0,x_0))-F(x^*(N,x_0))$$

$$\leq -q(x^*(0,x_0),u^*(0,x_0))$$

$$\leq 0$$

Since $J_N^*(x_1) \leq J_N(u(x_1),x_1)$, it follows that, $J_N^*(x_1) - J_N^*(x_0) \leq 0$.

Endproof.

3. Using subsets sequence to approach the maximal terminal region

Using SVM classifier to estimate the terminal region is not a new technology. In [6], a large terminal region was achieved by using SVM classifier. However, the method in [6] is somewhat conservative. The reason is that, the obtained terminal region actually is the stabilizable region of using a predetermined linear feedback controller.

In this section, a novel method of computing a terminal region is proposed. Given the terminal cost and a subset of the maximal terminal region, a subsets sequence is constructed by using one-step set expansion iteratively and SVM is employed to estimate each subset in this sequence. When some conditions are satisfied, the iteration ends and the last subset is adopted to serve as the terminal region.

3.1 The construction of subsets sequence

Consider an assumption as follows.

Assumption 2. A terminal cost is known.

If the stage cost is a quadratic function as $q(x,u)=x^T Q x + u^T Ru$ in which $Q$, $R$ are positive definite, a method of computing a terminal cost for continuous-time system can be found in [3]. In this paper, the method in [3] is extended to discrete-time system. Consider the linearization of the system (1) at the origin

$$x_{k+1} = A_d x_k + B_d u_k$$

with $A_d = (\partial f / \partial x)(0,0)$ and $B_d = (\partial f / \partial u)(0,0)$.

A terminal cost can be obtained through the following procedure:

Step 1. Solving the Riccati equation to get $G_0$.
Step 2. Getting a locally stabilizing linear state feedback gain $K$,

$$K = -\left(B_d^T G_0 B_d + R\right)^{-1} \left(B_d^T G_0 A_d\right)$$

Step 3. Computing $G_K$ by solving the following Riccati equation,

$$(\alpha A_K)^T G_K (\alpha A_K) - G_K = -Q_K$$

where $A_K = A_d + B_d K$, $Q_K = Q + K^T R K$, and $\alpha \in [1, +\infty)$ is an adjustable parameter satisfying $\alpha \lambda_{\text{max}}(A_K) < 1$. Then, $F(x) = x^T G_K x$ can serve as a terminal cost.

Given $F(\cdot)$ and from conditions (C1,C2), the terminal region $X_f$ can be defined as

$$X_f := \left\{ x \in X \mid F(x) \geq F^*_{X_f}(x) \right\}$$

where $F^*_{X_f}(x)$ is the minimum of the following optimization problem

$$\min_{u \in U} F_{X_f}(x) = q(x, u) + F(f(x, u))$$

s.t. $f(x, u) \in X_f$, $f(x, u) \geq q(x, u) + F(f(x, u))$ (4)

Remark 1. The construction of $X_f$ has two meanings: (I) the optimization problem (4) has feasible solution, that is to say, $\exists u \in U$, s.t. $f(x, u) \in X_f$; (II) the minimum of the optimization problem satisfies that $F^*_{X_f}(x) \leq F(x)$.

Remark 2. From the definition of $X_f$, it is obvious that, the terminal region is essentially a positively invariant set of using the optimal control resulting from the optimization problem (4) when $F(\cdot)$ is given.

Remark 3. In [3,4,6], the linear feedback control is attached to the construction of $X_f$ and $X_f$ is the stabilizable region of using the linear feedback controller. In [5], a saturated local control law was used. But, in this paper, there is no explicit control attached to the definition of $X_f$. So, the requirement on $X_f$ is lower than that in [3-6] while guaranting the stability of the controlled system.

From the definition of $X_f$, it can not be determined whether a state point belongs to $X_f$. The difficulty lies in that, the $X_f$ itself acts as the constraint in the optimization problem (4).

To avoid this problem, the method of using one-step set expansion iteratively is adopted. Define $X_{f, \text{max}}$ as the largest terminal region and consider an assumption.

Assumption 2. A subset of $X_{f, \text{max}}$, denoted by $X_f^0$ and containing the origin, is known.

Assumption 3. $X_f^0$ is a positively invariant set, that is to say, for any $x \in X_f^0$, $\exists u \in U$, s.t. $F(x) \geq q(x, u) + F(f(x, u))$ and $f(x, u) \in X_f^0$.

Given $X_f^0$, another subset of $X_{f, \text{max}}$, denoted by $X_f^1$, can be constructed as

$$G_0 = A_d^T G_0 A_d - \left(A_d^T G_0 B_d\right) \left(B_d^T G_0 B_d + R\right)^{-1} \left(B_d^T G_0 A_d\right) + Q$$
Using Subsets Sequence to Approach the Maximal Terminal Region for MPC

\[ X^1_j := \left\{ x \in X \mid F(x) \geq F^*_{X^j} (x) \right\} \]  \hspace{1cm} (5)

where \( F^*_{X^j} (x) \) is the minimum of

\[
\min_{u \in U} F_{X^j} (x) = q(x,u) + F(f(x,u)) \\
\text{s.t.} \quad f(x,u) \in X^0_f
\]  \hspace{1cm} (6)

As mentioned in remark 1, the construction of \( X^1_f \) contains two meanings: (I) for any \( x \in X^1_f \), \( \exists u \in U \), s.t. \( f(x,u) \in X^0_f \); (II) the minimum of (6) satisfies \( F^*_{X^j} (x) \leq F(x) \). The constructions of \( X^j_f \) in sequel have the similar meanings.

**Lemma 2.** If assumption 3 is satisfied, there is \( X^0_f \subseteq X^1_f \).

Proof. If assumption 3 is satisfied, it is obvious that, for any \( x \in X^0_f \), \( \exists u \in U \), s.t. \( F(x) \geq q(x,u) + F(f(x,u)) \) and \( f(x,u) \in X^0_f \). It follows that, \( F(x) \geq F^*_{X^j} (x) \). From the construction of \( X^1_f \), we can know \( x \in X^1_f \), namely, \( X^0_f \subseteq X^1_f \).

Endproof.

**Remark 4.** From the construction of \( X^j_f \), it is obvious that, if assumption 3 is satisfied, \( X^j_f \) is a positively invariant set. We know that, for any \( x \in X^j_f \), \( \exists u \in U \), s.t. \( F(x) \geq q(x,u) + F(f(x,u)) \) and \( f(x,u) \in X^0_f \). Because of \( X^0_f \subseteq X^1_f \) as showed in lemma 2, we have \( f(x,u) \in X^1_f \).

Similarly, by replacing \( X^0_f \) with \( X^j_f \) in the constraint of (6), another subset, denoted by \( X^2_f \), can be obtained as follows

\[ X^2_f := \left\{ x \in X \mid F(x) \geq F^*_{X^j} (x) \right\} \]  \hspace{1cm} (7)

where \( F^*_{X^j} (x) \) is the minimum of

\[
\min_{u \in U} F_{X^j} (x) = q(x,u) + F(f(x,u)) \\
\text{s.t.} \quad f(x,u) \in X^1_f
\]  \hspace{1cm} (8)

Repeatedly, \( X^j_f, j = 3, 4, \cdots, \infty \) can be constructed as

\[ X^j_f := \left\{ x \in X \mid F(x) \geq F^*_{X^{j-1}} (x) \right\} \]  \hspace{1cm} (9)

where \( F^*_{X^{j-1}} (x) \) is the minimum of

\[
\min_{u \in U} F_{X^{j-1}} (x) = q(x,u) + F(f(x,u)) \\
\text{s.t.} \quad f(x,u) \in X^{j-1}_f
\]  \hspace{1cm} (10)
This method of constructing \( X_j \) given \( X_{j-1} \) is defined as one-step set expansion in this paper. By employing it iteratively, a subsets sequence of largest terminal region, denoted by \( \{ X_j \}, \ j = 1, 2, \cdots \ , \) can be achieved.

**Remark 5.** Similar with lemma 2 and remark 4, any subset in this sequence is positively invariant and any two neighbouring subsets satisfy \( X_{j-1} \subseteq X_j \).

As \( j \) increases, \( \{ X_j \} \) will converge to a set, denoted by \( X_{\infty} \). Theorem 1 will show that, \( X_{\infty} \) is equal to the largest terminal region.

**Theorem 1.** If assumption 2 and assumption 3 are satisfied, for \( X_j \) constructed in (9) and (10), when \( j \) goes to infinity, \( \{ X_j \} \) will converge to \( X_{f,max} \).

Proof. This theorem is proved by contradiction.

(A) Assume that, there exists a set which is denoted by \( X_{spo} \) satisfying \( X_{spo} \subseteq X_{f,max} \) and \( X_j \rightarrow X_{spo} \) when \( j \rightarrow \infty \). From remark 5, we can know \( X_j^0 \subseteq X_{spo} \). It is obvious that \( \theta \in X_{spo} \) because of \( \theta \in X_j^0 \) as showed in assumption 2. It follows that, \( \theta \in X_{f,max} \setminus X_{spo} \) and for any \( x \in X_{f,max} \setminus X_{spo} \), we have \( F(x) > 0 \) since \( F(\cdot) \) is positive definite. Define \( \xi \) as the infimum of \( \{ F(x) | x \in X_{f,max} \setminus X_{spo} \} \), it is satisfied that, \( \xi > 0 \).

From the construction of \( X_j \), we know that, for any \( x_0 \in X_{f,max} \setminus X_{spo} \), there exists no such a \( u \in U \) satisfying \( F(x_0) \geq q(x_0,u) + \int \{ f(x_0,u) \} \) and \( (x_0,u) \in X_{spo} \) because of \( X_{spo} \subseteq X_{f,max} \). However, from (C1) and (C2), we know that, \( \exists u(x_0) \in U \), s.t. \( F(x_0) \geq q(x_0,u(x_0)) + F(x_1) \) and \( x_1 \in X_{f,max} \), where \( x_1 = f(x_0,u(x_0)) \). It is obvious that, \( x_1 \notin X_{spo} \). So we have, \( x_1 \in X_{f,max} \setminus X_{spo} \). Similarly, we can know, \( \exists u(x_1) \in U \), s.t. \( F(x_1) \geq q(x_1,u(x_1)) + F(x_2) \) and \( x_2 \in X_{f,max} \setminus X_{spo} \), where \( x_2 = f(x_1,u(x_1)) \), since \( x_1 \notin X_{f,max} \setminus X_{spo} \).

Repeately, for \( x_i \in X_{f,max} \setminus X_{spo}, \exists u(x_i) \in U \), s.t. \( F(x_i) \geq q(x_i,u(x_i)) + F(x_{i+1}) \) and \( x_{i+1} \in X_{f,max} \setminus X_{spo}, \) where \( x_{i+1} = f(x_i,u(x_i)) \), \( i = 2, 3, \cdots \). It is clear that, \( F(x_i) \rightarrow 0 \) when \( i \rightarrow \infty \). We know that, for the infimum of \( \{ F(x) | x \in X_{f,max} \setminus X_{spo} \} \), defined as \( \xi \), there is a positive real number \( \delta \) satisfying \( 0 < \delta < \xi \). Since \( F(x_i) \rightarrow 0 \) when \( i \rightarrow \infty \), \( \exists x_{i+1} \in X_{spo} \) s.t. for any \( x_{i+1} \), we have \( F(x_{i+1}) < \delta \). Obviously, this is contradicted with that \( \xi \) is the infimum of \( \{ F(x) | x \in X_{f,max} \setminus X_{spo} \} \). (B) Similarly, assume that, there exists a \( X_{spo} \) satisfying \( X_{spo} \supseteq X_{f,max} \) and \( X_j \rightarrow X_{spo} \) when \( j \rightarrow \infty \). For any \( x \in X_{spo} \), we have that \( F(x) \geq \min \{ q(x,u) + F(x,u) \} \) \( x,u \in X_{spo} \). Obviously, this is contradicted with that \( X_{f,max} \) is the largest one satisfying (C1) and (C2).

Endproof.

**Remark 6.** In this paper, the largest terminal region means the positively invariant set satisfying conditions (C1) and (C2). But, (C1) and (C2) are sufficient conditions to guarantee the stability of the controlled system, not the necessary conditions. There may be a set larger than \( X_{f,max} \) and the stability of the controlled system can be guaranteed by using this set as the terminal region.

**Remark 7.** In the calculation of \( X_{f,max} \), it is impossible to keep iteration computation until \( j \rightarrow \infty \). When the iteration time goes to \( j = E \) (\( E \) is a positive integer), if \( X_j^E \) is equal to \( X_j^{E-1} \) in principle, it can be deemed that \( \{ X_j \} \) converges to \( X_j^E \) in rough. Hence, \( X_j^E \) can be taken as the terminal region and it is a good approximation to \( X_{f,max} \).

**Remark 8.** If the iteration time does not go to infinity, the obtained set may be just a large positively invariant subset of \( X_{f,max} \). This has no effect on the stability of the controlled
system. The only negative influence is that its corresponding domain of attraction is smaller than that corresponding to $X_{f, \text{max}}$.

Until now, it seems that we can choose any $X^j_f$ in the subsets sequence as the terminal region. This is infeasible. Since $X^j_f$ is not described in explicit expression, it can not serve as the terminal constraint in the optimization problem (2) directly. Then, an estimated one described in explicit expression is needed. Due to the strong optimizing ability of SVM, SVM is exploited to separate each $X^j_f$ from the state space.

### 3.2 Support vector machine

SVM is the youngest part in the statistical learning theory. It is an effective approach for pattern recognition. In SVM approach, the main aim is to obtain a function, which determines the decision boundary or hyperplane. This hyperplane optimally separates two classes of input data points.

Take the example of separating $X$ into $A$ and $X \setminus A$. For each $x_i \in A$, an additional variable $y_i = +1$ is introduced. Similarly, for each $x_i \in X \setminus A$, $y_i = -1$ is introduced. Define $I^+ := \{i : y_i = +1\}$ and $I^- := \{i : y_i = -1\}$, SVM will find a separating hyperplane, denoted by $O(x) := w \cdot \phi(x) + b = 0$, between $A$ and $X \setminus A$. Therefore, $A$ can be estimated as $\hat{A} = \{x \in X | O(x) \geq 0\}$, where $O(x)$ is determined by solving the following problem:

$$
\min_{\alpha} \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \ker(x_i, x_j) - \sum_i \alpha_i
$$

s.t. 
$$
\sum_i \alpha_i y_i = 0 \\
0 \leq \alpha_i \leq C, \quad \forall i \in I^+; \quad \alpha_i \geq 0, \forall i \in I^-
$$

where $\ker(\cdot, \cdot)$ denotes the kernel function and the Gaussian kernel as follows is adopted in this paper:

$$
\ker(x, x_j) = \exp \left( -\frac{||x - x_j||^2}{2\sigma^2} \right)
$$

with $\sigma$ being the positive Gaussian kernel width.

When $\{\alpha_i\}$ are computed out, some support vectors are chosen from $\{x_i\}$ and the optimal hyperplane can be determined with these support vectors and their relevant weights. Denote $P_s$ as the number of support vectors and $X_s$ as the support vectors set, the optimal hyperplane is described as:

$$
O(x) = \sum_{i=1}^{P_s} w_i \cdot \ker(x_i, x) + b
$$

where $x_i \in X_s$ is the support vector and $w_i = \alpha_i y_i$ satisfying $\sum_{i=1}^{P_s} w_i = 0$ is the relevant weight. There are many software packages of SVM available on internet. They can be downloaded and used directly. To save space, it is not introduced in detail in this paper. For more details, please refer to [10] and [11].
3.3 Estimating the subset by employing SVM

From subsection 3.2, we know that, SVM find a separating hyperplane between \( \{ x_i | i \in I^+ \} \) and \( \{ x_i | i \in I^- \} \). This hyperplane is used to separate \( X \) into \( A \) and \( X \setminus A \). All of \( \{ x_i \} \) and their relevant \( \{ y_i \} \) compose a set, named the training points set. This subsection will show how to achieve the training points set when estimating \( X^j \) and how to determine \( X^j \) when the separating hyperplane is known.

Firstly, choose arbitrary points \( x_i \in X, \ i = 1, 2, \ldots, P \ (P \ is \ the \ number \ of \ training \ points) \); then, assign \( y_i \) to each \( x_i \) by implementing the following procedure:

IF (I) the following optimization problem has feasible solution

\[
\min_{u \in U} J_{x_i}^j (x_i) = q(x_i, u) + F(f(x_i, u))
\]

s.t. \( f(x_i, u) \in \hat{X}_j^{i-1} \).

(When \( j = 1, \hat{X}_j^0 = X^0 \).)

(II) its minimum satisfies

\[ F(x_i) \geq F_{x_i}^j (x_i). \]

THEN \( y_i = +1 \)
ELSE \( y_i = -1 \)
ENDIF.

By implementing this procedure for every \( x_i \), each \( y_i \) is known. Input \( \{ x_i \} \) and \( \{ y_i \} \) into SVM classifier, an optimal hyperplane \( O^j (x) = 0 \) will be obtained. Therefore, the estimated set of \( X^j \) can be achieved as \( \hat{X}_j = \{ x \in X | O^j (x) \geq 0 \} \).

When \( \hat{X}_j \) is known, the training points for separating \( X^j+1 \) from \( X \) can be computed by the similar procedure. By inputting them into SVM classifier, a hyperplane \( O^{j+1} (x) = 0 \) and an estimated set of \( X^j+1 \), denoted by \( \hat{X}_j^{j+1} = \{ x \in X | O^{j+1} (x) \geq 0 \} \) will be obtained. Repeatedly, \( \{ O^j (x) \}, j = 1, 2, \ldots, \infty \) and \( \{ \hat{X}_j \} \) can be be achieved by the similar technology.

4. Estimating the terminal region

Section 3 showed how to achieve the subsets sequence by employing SVM. Theoretically, the larger the iteration time \( j \), the higher the precision of \( \hat{X}_j \) approaching to \( X_{j, \text{max}} \). But, it is impossible to keep computation until \( j \to +\infty \). To avoid this problem, the iteration should be ended when some conditions are satisfied.

When \( j = E \), if it is satisfied that, for \( x_i \in X_{s,E-1}, \ i = 1, 2, \ldots, P_{s,E-1} \), there exists

\[
\sum_{i=1}^{P_{s,E-1}} \| O^E (x_i) - O^{E-1} (x_i) \| \leq e P_{s,E-1},
\]
it can be deemed that $\hat{X}_J^E$ is equal to $\hat{X}_J^{E-1}$ in principle and $\hat{X}_J^E$ converges to $\hat{X}_J^E$. In (14), $X_{n,E-1}$ is the support vectors set at $j = E - 1$, $P_{n,E-1}$ is the number of support vectors and $\varepsilon$ is a tunable threshold. The smaller $\varepsilon$ is, the higher the precision of $\hat{X}_J^E$ approximating to $X_{f,\text{max}}$ is. Finally, $\hat{X}_J^E$ is used to serve as the terminal region.

Remark 9. Here, we used the information that, in SVM classifier, the hyperplanes are only determined on the support vectors. Now, the concrete algorithm of estimating the largest terminal region is displayed as follows.

**Step 4.** Set the number of training points $P$ used in SVM and the tunable threshold $\varepsilon$.

**Step 5.** For $j = 1, 2, \cdots, \infty$, use SVM to achieve the optimal hyperplane $O^j(x) = 0$ and the estimated set of $X_j^j$, denoted by $\hat{X}_J^E$.

**Substep 1.** Choose arbitrary points $x_i \in X$, $i = 1, 2, \cdots, P$.

**Substep 2.** Assign $y_i$ to each $x_i$ by implementing the procedure in subsection 3.3.

**Substep 3.** Input $\{x_i, y_i\}$ into the SVM. An optimal hyperplane $O^j(x) = 0$ will be obtained and $X_j^j$ can be approximated by $\hat{X}_J^E = \{x \in X | O^j(x) \geq 0\}$, where

$$O^j(x) = \sum_{i=1}^{P_{s,j}} w_i \cdot ker(x_i, x) + b_j$$

with $P_{s,j}$ denoting the number of support vectors, $x_i$ being the support vector, $w_i$ denoting its relevant weight and $b_j$ denoting the classifier threshold.

**Step 6.** Check the iteration status. When $j = E$, if inequality (14) is satisfied, end iteration and take $\hat{X}_J^E$ as the largest terminal region.

Remark 10. It is obvious that, $\hat{X}_J^E$ is achieved one by one. Namely, $\hat{X}_J^E$ can only be achieved when $\hat{X}_J^{E-1}$ is known.

5. Simulation experiment

The model is a discrete-time realization of the continuous-time system used in [3, 6]:

$$\begin{bmatrix} x_1(k + 1) \\ x_2(k + 1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ T & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} T \mu \\ T \mu \end{bmatrix} u(k) + \begin{bmatrix} T(1 - \mu) & 0 \\ 0 & -4T(1 - \mu) \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} u(k)$$

where $\mu = 0.5$, $T = 0.1s$, and the state constraint and control constraint are $X = \{x | \|x\|_1 \leq 4\}$, $U = \{u | \|u\| \leq 2\}$, respectively.

The stage cost is $q(x, u) = x^T Q x + u^T R u$ where $Q = 0.5I$ and $R = 1$. The terminal cost is chosen as $F(x) = x^T G x$ where $G = [1107.356 \ 857.231; 857.231 \ 1107.356]$ and $X_f^0$ is given as the terminal region in [3] which is
\[ X^0_f = \left\{ x \in X \mid x^T \begin{bmatrix} 16.5926 & 11.5926 \\ 11.5926 & 16.5926 \end{bmatrix} x \leq 0.7 \right\}. \]

To estimate each \( X^j_f \), 4000 training points are generated. Set \( \varepsilon = 2.5 \), when \( j = 22 \), there exists

\[
\sum_{i=1}^{P_{s,21}} \| O^{22} (x_i) - O^{21} (x_i) \| \leq \varepsilon P_{s,21},
\]

where \( x_i \in X_{s,21} \), \( X_{s,21} \) is the support vectors set and \( P_{s,21} \) is the number of support vectors at \( j = 21 \). Then, it is deemed that, \( \hat{X}^{22}_f \) is equal to \( \hat{X}^{21}_f \) in principle and \( \hat{X}^{22}_f \) can be taken as the final estimation of \( X_{f,\text{max}} \). Figure 1 shows the approximation process of \( X_{f,\text{max}} \).

In figure 1, the blue ellipsoid is the terminal region in [3], which serves as \( X^0_f \) in the estimation of \( X_{f,\text{max}} \) in this paper. The regions surrounded by black solid lines are \( \{ \hat{X}^j_f \}, j = 1, 2, \cdots, 22 \) in which the smallest one is \( \hat{X}^1_f \), the largest one is \( \hat{X}^{22}_f \) and the regions between them are \( \{ \hat{X}^j_f \}, j = 2, 3, \cdots, 21 \) satisfying \( \hat{X}^{j-1}_f \subseteq \hat{X}^j_f \). The time cost of employing SVM to estimate each \( \hat{X}^j_f \) is about 44 minutes and the total time cost of computing the final estimation of \( X_{f,\text{max}} \), namely, \( \hat{X}^{22}_f \) is about 16 hours.

![Fig. 1. The approximation process](www.intechopen.com)
Set the prediction horizon as $N = 3$, some points in the region of attraction (this example is very exceptional, the region of attraction is coincident with the terminal region in rough. Therefore, these points are selected from the terminal region) are selected and their closed-loop trajectories are showed in Figure 2.

Fig. 2. The closed-loop trajectories of states

In figure 2, the blue ellipsoid is the terminal region in [3] and the region encompassed by black dash lines is the result in [6]. The region encompassed by black solid lines is the terminal region in this paper. We can see, the terminal region in this paper contain the result in [3], but not contain the result in [6] although it is much larger than that in [6]. The reason is that, the terminal region in this paper is the largest one satisfying conditions (C1) and (C2). However, (C1) and (C2) are just the sufficient conditions to guarantee the stability of the controlled system, not the necessary conditions as showed in remark 6. The red solid lines denote the closed-loop trajectories of the selected points. Note that, with the same sampling interval and prediction horizon as those in this paper, these points are not in the regions of attraction of MPC in [3] and [6]. But, they can be leaded to the orgin by using the control law of MPC in this paper.

6. Conclusion

Given the terminal cost, a sequence of subsets of the maximal terminal region are extracted from state space one by one by employing SVM classifier. When one of them is equal to its
succesive one in principle, it is used to serve as the terminal region and it is a good approximation to the maximal terminal region.

7. References


Model Predictive Control (MPC) refers to a class of control algorithms in which a dynamic process model is used to predict and optimize process performance. From lower request of modeling accuracy and robustness to complicated process plants, MPC has been widely accepted in many practical fields. As the guide for researchers and engineers all over the world concerned with the latest developments of MPC, the purpose of "Advanced Model Predictive Control" is to show the readers the recent achievements in this area. The first part of this exciting book will help you comprehend the frontiers in theoretical research of MPC, such as Fast MPC, Nonlinear MPC, Distributed MPC, Multi-Dimensional MPC and Fuzzy-Neural MPC. In the second part, several excellent applications of MPC in modern industry are proposed and efficient commercial software for MPC is introduced. Because of its special industrial origin, we believe that MPC will remain energetic in the future.

**How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:
