

拘束システムに対するリファレンスガバナの実現

M2004MM025 小島 伸也

指導教員 高見 勳

1 はじめに

現実の制御系設計を考えると、アクチュエータ、制御対象共に限界入力値が設定されている。もし、これらの拘束を考慮にいれなかった場合には、ウィンドアップ現象などによる制御系の不安定化や、ハードウェアの破壊を招く可能性がある。よって、制御系設計では、良好な過渡応答を達成することのみならず、上記の拘束条件を満足することが要求される。これらの要求を同時に満たすための手法の一つとして、制御系への目標値信号を修正することにより入力や状態制約を満足させる、リファレンスガバナが最近注目されている。

過去のリファレンスガバナの設計に関する研究は、オンライン設計 [1] とオフライン設計 [2] の 2 つに大きく分けられることができる。それぞれの特徴は以下の通りである。

〈オンライン設計〉

- フィードバック制御が可能となる。
- 閉ループ系の状態量の情報を常に必要とする。
- 制約条件を満たす際の保守性がオフライン設計よりも高い。
- サンプルングタイムの短いシステムに対しては、計算時間の影響によって実時間での実装は困難となる場合がある。

〈オフライン設計〉

- フィードバック制御ができないので、未知の入力に対して非常に弱い。
- 計算をオフラインで行うため、追従特性を考慮することができ、より積極的な参照入力を生成できる。
- オンライン設計と比べて、比較の実装が容易であり、実用性が高い。

ここで、特に、オフライン設計に注目し、現実問題を考えてみる。制御系を設計するにあたって、目標値をステップ関数に限定しても実用上大きな制約になることはない。また、目標値があらかじめ与えられる場合も多く、そのときにはオンラインで目標値を修正する必然性もあまりない。これらは、メカトロニクス系に代表される多くの位置決め系にあてはまり、まさにリファレンスガバナのオフライン設計に対して最適な制御対象であるといえる。

しかし、実装にあたって、リファレンスガバナは計算量という大きな問題を有する。これは、サンプルングタイムが小さくなればなるほど、線形行列不等式 (Linear Matrix Inequalities : LMI) で解くべき変数が増加し、数値的に LMI を解くことが困難になる設計時の問題と、変数の増加

による影響でメモリ不足に陥る問題がある。1[ms] 程度のサンプルングタイムにおいてもリファレンスガバナが設計できないという問題は、致命的であるといえる。

そこで本論文では、サンプルングの間引き [3][4][5] を行うことで計算量とメモリ容量の低減化をし、拘束条件を満足するための LMI を定式化する。さらに、それを三慣性システムに適用させてみることで、提案法の有効性を考察、検証する。

2 リファレンスガバナ設計

有限時間で入力される参照入力 $r(t)$ に対し、新たな修正目標値を生成する機構であるリファレンスガバナを付加した閉ループシステムを図 1 示す。ここで、 Σ_p は制御対象である。また、 Σ_c はフィードバック補償器であり、拘束条件を考慮して設計されているものではないとする。 Σ_p 、 Σ_c から成る閉ループシステム Σ_{cl} を考える。このとき、閉ループシステム Σ_{cl} は離散時間システムとして扱い、その状態方程式は次式のように記述されるものとする。

$$\begin{aligned} x(t+1) &= A_{cl}x(t) + B_{cl}g(t) \\ \Sigma_{cl}: \quad y(t) &= C_{ycl}x(t) + D_{ycl}g(t) \\ z(t) &= C_{zcl}x(t) + D_{zcl}g(t) \end{aligned} \quad (1)$$

ここで、 $x(t) \in \mathcal{R}^n$ は閉ループ系の状態、 $y(t) \in \mathcal{R}^m$ は出力、 $g(t) \in \mathcal{R}^m$ は設計すべき修正目標値、 $z(t) \in \mathcal{R}^l$ は拘束条件を表すためのベクトルである。なお、初期状態は $x(0) = 0$ である。拘束条件 $z(t)$ は以下の不等式で記述されるものとする。

$$|z_j(t)| \leq 1 \quad \forall j \in [1, l], \quad \forall t \quad (2)$$

ここで、 z_j は z の j 番目の要素を表す。

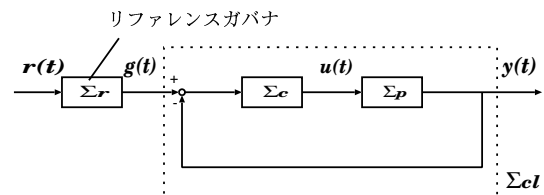


図 1 リファレンスガバナを付加した閉ループシステム

リファレンスガバナとは、(2) 式を満たしながら、出力 $y(t)$ が与えられたステップ目標値定数 $r(t) = r_0$ にできるだけ速く追従するような修正目標値列 $g(0), g(1), g(2), \dots$ を生成することにある [6]。

本論文では、リファレンスガバナを導出するにあたって

最小とすべき目的関数 J を (3) 式に定義する.

$$J = \sum_{t=0}^N \|r_0 - y(t)\|^2 + \sum_{t=0}^N w(t) \|r_0 - g(t)\|^2 \quad (3)$$

ここで, $w(t)$ は重み関数, N は総サンプリング数である. $w(t)$ を t の単調増加関数として与えることにより, t が小さい時点では $g(t)$ の変動を大きく, t が大きくなれば $g(t)$ が定数 r_0 に一致するといったように修正目標値 $g(t)$ の動きを大まかに決定することができる. この評価関数を最小化する修正目標値列 $g(t)$ の設計を LMI に帰着させて解くことで, 最適なりファレンスガバナを設計できる.

3 サンプリングの間引き

本節では, より実用的なりファレンスガバナ設計を考える. 従来の手法では, 修正目標値 $g(t)$ の生成の際, サンプリングタイム 0.01[s] で 1[s] 間のガバナを設計すると, $N=100$, サンプリングタイム 0.001[s] なら, $N=1000$ と LMI 変数の数は, サンプリングタイムと制御時間に比例して大きくなる. そこで, 各サンプリングからの情報を間引いても, 拘束条件を常に満たすような LMI 最適化問題を再度設計する. つまり, $g(0) = g(1), g(2) = g(3), \dots$ とすることで, $g(0)$ が決まれば次は $g(2)$ といったようにサンプリングの間隔を 2 倍にして扱うことにする. このとき, (1) 式は以下のような 2 つの状態 $\tilde{x}_1(t), \tilde{x}_2(t)$ に分けることができる. まず, 時刻 $2t$ のときの状態として, $\tilde{x}_1(t) = x(2t)$, $\tilde{g}(t) = g(2t)$ と考えることで,

$$\tilde{x}_1(t) = \sum_{j=0}^{t-1} (A_{cl}^{2(t-j)-1} B_{cl} + A_{cl}^{2(t-j)-2} B_{cl}) \tilde{g}(j) \quad (4)$$

$[t = 0, 1, \dots, N/2]$

と表せる. 次に, 時刻 $2t + 1$ の場合は, $\tilde{x}_2(t) = x(2t + 1)$, $\tilde{g}(t) = g(2t)$ と考えることで,

$$\tilde{x}_2(t) = \sum_{j=0}^{t-1} (A_{cl}^{2(t-j)} B_{cl} + A_{cl}^{2(t-j)-1} B_{cl}) \tilde{g}(j) \quad (5)$$

$[t = 0, 1, \dots, N/2]$

と表せる. このとき, 出力 $y(t)$ は状態 $\tilde{x}_1(t)$ のときだけ注目すれば良いとする. しかし, 拘束条件 $z(t)$ は, 全てのサンプリングにおいて満たされなければならないので全ての状態 $\tilde{x}_1(t), \tilde{x}_2(t)$ を考慮すると

$$\tilde{z}_1(t) = \sum_{j=0}^{t-1} C_{zcl} (A_{cl}^{2(t-j)-1} B_{cl} + A_{cl}^{2(t-j)-2} B_{cl}) \tilde{g}(j) + D_{zcl} \tilde{g}(t) \quad (6)$$

$$\tilde{z}_2(t) = \sum_{j=0}^{t-1} C_{zcl} (A_{cl}^{2(t-j)} B_{cl} + A_{cl}^{2(t-j)-1} B_{cl}) \tilde{g}(j) + (C_{zcl} B_{cl} + D_{zcl}) \tilde{g}(t) \quad (7)$$

$$\tilde{y}(t) = \sum_{j=0}^{t-1} C_{ycl} (A_{cl}^{2(t-j)-1} B_{cl} + A_{cl}^{2(t-j)-2} B_{cl}) \tilde{g}(j) + D_{ycl} \tilde{g}(t) \quad (8)$$

$[t = 0, 1, \dots, N/2]$

と表すことができる. この考えを使って, 間隔を n 倍にしたときの $z(t), y(t)$ を考える. すると,

$$\tilde{z}_1(t) = \sum_{j=0}^{t-1} C_{zcl} \left(\sum_{k=1}^n A_{cl}^{n(t-j)-k} B_{cl} \right) \tilde{g}(j) + D_{zcl} \tilde{g}(t) \quad (9)$$

$$\tilde{z}_2(t) = \sum_{j=0}^{t-1} C_{zcl} \left(\sum_{k=1}^n A_{cl}^{n(t-j)-k+1} B_{cl} \right) \tilde{g}(j) + (C_{zcl} B_{cl} + D_{zcl}) \tilde{g}(t) \quad (10)$$

\vdots

$$\tilde{z}_n(t) = \sum_{j=0}^{t-1} C_{zcl} \left(\sum_{k=1}^n A_{cl}^{n(t-j)-k+n-1} B_{cl} \right) \tilde{g}(j) + (C_{zcl} \left(\sum_{k=1}^{n-1} A_{cl}^{k-1} \right) B_{cl} + D_{zcl}) \tilde{g}(t) \quad (11)$$

$$\tilde{y}(t) = \sum_{j=0}^{t-1} C_{ycl} \left(\sum_{k=1}^n A_{cl}^{n(t-j)-k} B_{cl} \right) \tilde{g}(j) + D_{ycl} \tilde{g}(t) \quad (12)$$

$[t = 0, 1, \dots, N/n]$

となる. ベクトル $\tilde{Z}_1, \tilde{Z}_2, \dots, \tilde{Z}_n, \tilde{Y}, \tilde{G}$ を

$$\begin{aligned} \tilde{Z}_1 &= [\tilde{z}_1^T(0) \quad \tilde{z}_1^T(1) \quad \dots \quad \tilde{z}_1^T(N/n)]^T \\ \tilde{Z}_2 &= [\tilde{z}_2^T(0) \quad \tilde{z}_2^T(1) \quad \dots \quad \tilde{z}_2^T(N/n)]^T \\ &\vdots \\ \tilde{Z}_n &= [\tilde{z}_n^T(0) \quad \tilde{z}_n^T(1) \quad \dots \quad \tilde{z}_n^T(N/n)]^T \\ \tilde{Y} &= [\tilde{y}^T(0) \quad \tilde{y}^T(1) \quad \dots \quad \tilde{y}^T(N/n)]^T \\ \tilde{G} &= [\tilde{g}^T(0) \quad \tilde{g}^T(1) \quad \dots \quad \tilde{g}^T(N/n)]^T \end{aligned}$$

と定義すると (9), (10), (11), (12) 式は $\tilde{Z}_1 = \tilde{M}_{z_1} \tilde{G}$, $\tilde{Z}_2 = \tilde{M}_{z_2} \tilde{G}$, \dots , $\tilde{Z}_n = \tilde{M}_{z_n} \tilde{G}$, $\tilde{Y} = \tilde{M}_y \tilde{G}$ と表すことができる. ここで, $\tilde{M}_{z_1}, \tilde{M}_{z_2}, \dots, \tilde{M}_{z_n}$ は $(N/n \times N/n)$ 行列, \tilde{M}_y は $(N/n \times N/n)$ 行列である. まず, 評価関数 J に Schur Complement を適用すると

$$F_1(\gamma, \tilde{G}) = \begin{bmatrix} \tilde{\Theta}_1 & \tilde{G} \\ \tilde{G}^T & \gamma - \tilde{\Theta}_2 + 2\tilde{\Theta}_3 \tilde{G} \end{bmatrix} > 0 \quad (13)$$

が導かれる. なお, $\tilde{R}_0 = [r_0^T(0) \quad r_0^T(1) \quad \dots \quad r_0^T(N/n)]^T$, $\tilde{W} = \text{diag}(w(0), w(1), \dots, w(N/n))$, $\tilde{\Theta}_1 = (\tilde{M}_y^T \tilde{M}_y + \tilde{W})^{-1}$, $\tilde{\Theta}_2 = \tilde{R}_0^T (I + \tilde{W}) \tilde{R}_0$, $\tilde{\Theta}_3 = (\tilde{R}_0^T \tilde{M}_y + \tilde{R}_0^T \tilde{W})$ であり, γ は J の上界である. 制約条件 (2) 式も LMI となる. 実際, $\tilde{Z}_1 = \tilde{M}_{z_1} \tilde{G}$, $\tilde{Z}_2 = \tilde{M}_{z_2} \tilde{G}$, \dots , $\tilde{Z}_n = \tilde{M}_{z_n} \tilde{G}$ より,

$$\tilde{Z}_{1i}^2 = (\tilde{M}_{z_1} \tilde{G})_i^T (\tilde{M}_{z_1} \tilde{G})_i < 1 \quad (14)$$

$$\tilde{Z}_{2i}^2 = (\tilde{M}_{z_2} \tilde{G})_i^T (\tilde{M}_{z_2} \tilde{G})_i < 1 \quad (15)$$

\vdots

$$\tilde{Z}_{ni}^2 = (\tilde{M}_{z_n} \tilde{G})_i^T (\tilde{M}_{z_n} \tilde{G})_i < 1 \quad (16)$$

$[\text{for } i = 1, \dots, (N/n + 1)l]$

が成り立てば (2) 式は満たされる. ここで添字 i はベクトルの i 番目の要素であることを示す. 上式に Schur

Complement を適用し

$$F_2^i(\tilde{G}) = \begin{bmatrix} 1 & (\tilde{M}_{z_1} \tilde{G})_i \\ (\tilde{M}_{z_1} \tilde{G})_i^T & 1 \end{bmatrix} > 0 \quad (17)$$

$$F_3^i(\tilde{G}) = \begin{bmatrix} 1 & (\tilde{M}_{z_2} \tilde{G})_i \\ (\tilde{M}_{z_2} \tilde{G})_i^T & 1 \end{bmatrix} > 0 \quad (18)$$

$$\vdots$$

$$F_{n+1}^i(\tilde{G}) = \begin{bmatrix} 1 & (\tilde{M}_{z_n} \tilde{G})_i \\ (\tilde{M}_{z_n} \tilde{G})_i^T & 1 \end{bmatrix} > 0 \quad (19)$$

$[\forall i \in [1, (N/n + 1)l]]$

を得る. よって, 有限時間区間でリファレンスガバナを設計するには, (13), (17), (18), (19) 式を満たすような γ と \tilde{G} に関する LMI 最適化問題を解けばよい. この手法を導入すると, 解くべき LMI の変数 \tilde{G} は間引きするごとに $N/2, N/3, \dots$ と減少していく. その結果, n 回に 1 回修正目標値を生成したとき, LMI 行列のサイズは $(\frac{N}{n} + 2) + 2n(\frac{N}{n} + 1)l$, 変数の数は $\frac{N}{n} + 2$ に減少させることができる. これにより, 従来のリファレンスガバナでは設計できなかったサンプリングタイムが小さく, 制御時間が大きな LMI 最適化問題も解けるようになる.

4 実験例と検証

制御対象として三慣性システムを考える. この制御対象の状態ベクトル x_p を $[\theta_1 \ \theta_1 \ \theta_2 \ \theta_2 \ \theta_3 \ \theta_3]^T$, $u = V$ とする Σ_p の状態空間モデルは

$$\dot{x}_p = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{k_1}{J_1} & -\frac{c_1}{J_1} & \frac{k_1}{J_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{k_1}{J_2} & 0 & -\frac{(k_1+k_2)}{J_2} & -\frac{c_2}{J_2} & \frac{k_2}{J_2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{k_2}{J_3} & 0 & -\frac{k_2}{J_3} & -\frac{c_3}{J_3} \end{bmatrix} x_p + \begin{bmatrix} 0 \\ \frac{k_1 k_t k_p}{J_1} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u, \quad \begin{pmatrix} J_1 = 0.0025[kgm^2] & c_1 = 0.007[Nm/(rad/s)] \\ J_2 = 0.0018[kgm^2] & c_2 = 0.001[Nm/(rad/s)] \\ J_3 = 0.0018[kgm^2] & c_3 = 0.001[Nm/(rad/s)] \\ k_1 = 2.7[Nm/rad] & k_a = 2[A/V] \\ k_2 = 2.6[Nm/rad] & k_t = 0.1[Nm@M/A] \\ & k_p = 3[Nm@D/Nm@M] \end{pmatrix}$$

$$y = [0 \ 0 \ 0 \ 0 \ 1 \ 0] x_p$$

で与えられる. この制御対象には安全性の観点から, 入力電圧 $U[V]$ には飽和のために $|U| \leq 4.884$ なる制約があるものとする. なお, この三慣性システムは一定以上の入力飽和があると安全のため自動的に制御を中断する.

次に, 制御対象をサンプリングタイム 0.005[s] の零次ホールドで離散化して, 補償器を設計する. シミュレーションでは, 参照入力を $r_0 = 5000[\text{count}]$, 制御時間を 1[s] と与える ($N = 200$). まず, リファレンスガバナを適用しないシミュレーション結果を図 2 に, リファレンスガバナを適用した場合のシミュレーション結果を図 3 に示す.

図 2, 3 より, リファレンスガバナを付加しない場合は, 入力電圧が拘束条件を完全に満たしていないことがわかる. 実際, 実機に適用して実験してみると入力飽和の影響より三慣性システムは制御を中断してしまうといった結果になった. この点において, リファレンスガバナの有効性は十分に確認できる.

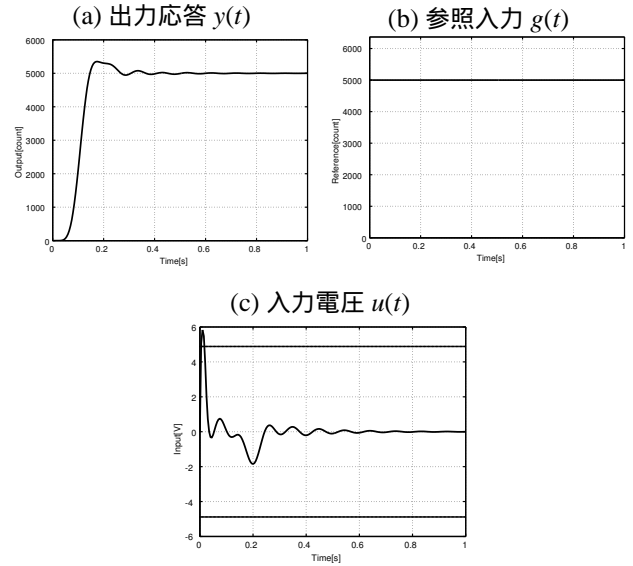


図 2 リファレンスガバナなしのシステム応答

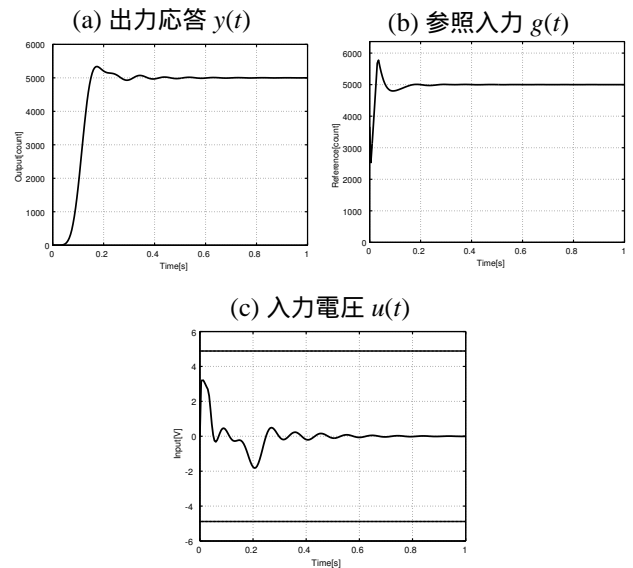


図 3 リファレンスガバナを付加したシステム応答

しかし, このリファレンスガバナを実制御対象に実装するにあたって, 各サンプリング周期ごとに修正目標値 $g(t)$ を生成する従来の手法では, 大規模な制御設計に対して, 実装のしやすさやデータ量といった問題が発生してくる. そこで, 本論文で提案するサンプリングタイムを間引きした設計を考える.

この手法を取り入れたことによって, 改善された結果を表 1 に示し, 参考としてサンプリングタイムを 10 倍に間引いたときのシミュレーション結果を図 4 に示す. シミュレーションでは, 参照入力を $r_0 = 5000[\text{count}]$, 制御時間を 1[s] と与える ($N = 20$).

表 1 間引きと計算時間の関係

間引き	なし	2 倍	4 倍	10 倍
行列サイズ	604	506	460	442
変数の数	202	102	52	22
計算時間	71[s]	19[s]	13[s]	11[s]

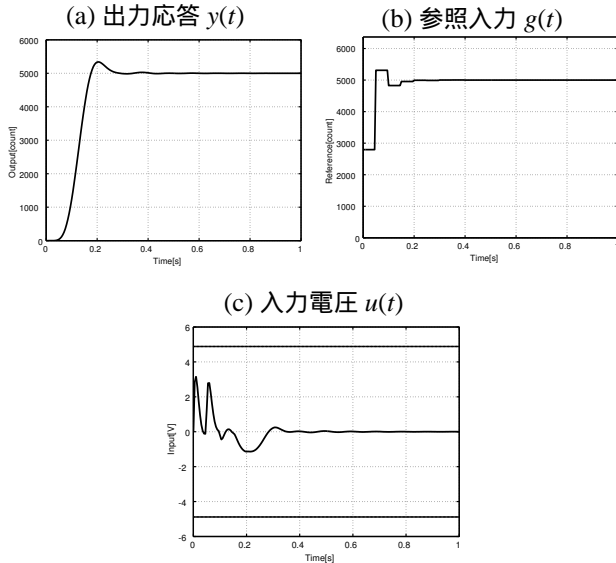


図 4 リファレンスガバナを付加したシステム応答 (10 倍)

図 4 より、サンプリングタイムを間引きしたとしても、拘束条件を破ることのない良好な結果を得ることを確認した。表 1 より注目すべき点は、各間引きに対する計算時間の違いで、変数の数が半分になっただけで計算時間が 1/4 ほど短縮されているのがわかる。この結果より、修正目標値のサンプリング時間を長くし、LMI 変数の数を減少するように提案した手法は、効果的であったといえる。

最後に、サンプリングタイムを 10 倍に間引いて設計したリファレンスガバナを三慣性システムに適用した際の出力結果を図 5 に示す。

5 おわりに

本論文では、サンプリングタイムを間引きすることで計算時間の大幅な短縮に成功し、より現実の制御対象に実装可能なリファレンスガバナを生成できた。この間引きの大きな利点は、データの軽減化に成功しながらも拘束条件は常に満たしているところにある。よって、単純な参照入力 (リファレンスガバナ) を閉ループ系に付加するだけで、大幅な改善が期待されると考えられる。実際、今回提案したリファレンスガバナを三慣性システムに適用した場合、リファレンスガバナなしでは制御できなかったコントローラでも、リファレンスガバナを付加することで制御可能となることが確認できた。

今後の課題は、オフライン設計の場合、モデル化誤差の影響が顕著に現れるので、シミュレーションではうまく

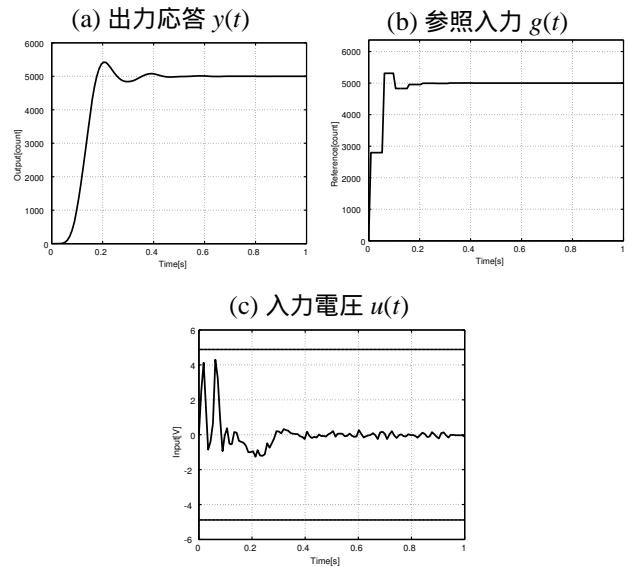


図 5 実制御対象のシステム応答 (10 倍)

いっても、実制御対象では拘束条件を破ってしまう点である。そこで、今後はこの不確かな拘束条件の見積もり法を考える必要がある。

参考文献

- [1] 大原, 平田: 参照入力集合のパラメトリゼーションにもとづく拘束システムの制御, 計測自動制御学会論文集, 40-8, 796/805 (2004)
- [2] 畑中, 鷹羽: 外乱を受ける拘束システムに対するリファレンスガバナの設計法, 計測自動制御学会論文集, 40-8, 806/814 (2004)
- [3] 平田, 小木曾: 状態アフィン関数表現によるリファレンスガバナの構成法と制御検証実験, 計測自動制御学会論文集, 40-2, 247/256 (2004)
- [4] 平田, 小木曾: 実装データの軽減化を考慮したリファレンスガバナの実験的検証および考察, 第 4 回制御部門大会資料, 193/196 (2004)
- [5] 平田, 小木曾: 変数のブロッキングを用いて表現される拘束システムに対するリファレンスガバナ, 計測自動制御学会論文集, 41-1, 88/90 (2005)
- [6] 杉江, 山本: 状態および入力の制約を考慮した閉ループ系の目標値生成, 計測自動制御学会論文集, 37-9, 849/855 (2001)