



# 实现系统解耦的预补偿阵设计新方法

何建平

(太原钢铁公司自动化研究所, 030003)

金萍

曾建平

(太原大学机电系, 030012) (太原机械学院自动控制系, 030051)

## 摘 要

本文提出一种实现多变量系统解耦的预补偿器常数阵的设计方法, 即从传递函数阵  $G(s)$  导出一个复常阵  $C$ , 通过  $C$  求得预补偿阵  $K_p$ . 文中还论述了这种方法的原理, 同时给出了应用实例, 结果令人满意.

**关键词:** 解耦控制, 补偿器, 多变量系统.

## 一、引 言

解耦控制是自动控制理论领域的一个重要分支, 解耦方法有状态反馈法、Rosenbrock 对角优势解耦方法和 Shinskey 的相对放大系数法等. 近十多年来, 不断有人提出新的解耦理论与方法. 状态反馈方法不易在工程上实现, 频域法直接对  $G(s)$  运用“伪对角化法”, 需要进行多项式矩阵求逆的复杂运算和绘制比较复杂的逆乃氏阵列图和盖氏图.

本文从传递函数阵  $G(s)$  导出一个复常阵  $C$ , 对  $C$  运用“伪对角化法”求得预补偿阵  $K_p$ , 设计过程简化, 对工程控制有比较重要的意义.

## 二、解 耦 原 理

### 1. 耦合度

设被控对象的传递函数为

$$G(s) = \begin{bmatrix} g_{11}(s) \cdots g_{1j}(s) \cdots g_{1m}(s) \\ \cdots \cdots \cdots \cdots \cdots \cdots \\ g_{i1}(s) \cdots g_{ij}(s) \cdots g_{im}(s) \\ \cdots \cdots \cdots \cdots \cdots \cdots \\ g_{m1}(s) \cdots g_{mj}(s) \cdots g_{mm}(s) \end{bmatrix}_{m \times m}, \quad (1)$$

式中

$$g_{ij}(s) = \frac{K_{ij} \prod_{k=1}^{m_{ij}} (s + z_k)}{\prod_{k=1}^{m_{ij}} (s + p_k)} = \frac{k_{ij} \prod_{k=1}^{m_{ij}} (s + z_k)}{\prod_{k=1}^{n_{ij(1)}} (s + d_k) \prod_{k=1}^{n_{ij(2)}} (s + a_k + jb_k)(s + a_k - jb_k)} \quad (2)$$

耦合度的度量方法很多，现代频域法是以传递函数阵的各行(列)的非对角元素模的和与对角元素模之比<sup>[1,2]</sup>来衡量系统的耦合度，即

$$d_i(s) = \left[ \sum_{\substack{j=1 \\ (i \neq j)}}^m |g_{ij}(s)| \right] / |g_{ii}(s)|, d'_j(s) = \left[ \sum_{\substack{i=1 \\ (i \neq j)}}^m |g_{ij}(s)| \right] / |g_{jj}(s)|. \quad (3)$$

将 (2) 式部分分式展开得

$$g_{ij}(s) = \sum_{k=1}^{n_{ij(1)}} \frac{r_k}{s + d_k} + \sum_{k=1}^{n_{ij(2)}} \left[ \frac{r_k^+}{s + a_k + jb_k} + \frac{r_k^-}{s + a_k - jb_k} \right], \quad (4)$$

式中  $r_k^+$  与  $r_k^-$  共轭且  $r_k^+ = R_k e^{j\theta_k}$ .

根据时域的脉冲响应与频域的传递函数是拉氏变换与拉氏反变换的关系，SISO 系统时域的脉冲响应与频域的传递函数代表的系统性能是一致的，MIMO 系统时域的脉冲响应矩阵与频域的传递函数阵反映的系统性能也是一致的。因此，传递函数矩阵  $G(s)$  可以用如下脉冲响应矩阵  $G(t)$  表示：

$$G(t) = \begin{Bmatrix} g_{11}(t) \cdots g_{1j}(t) \cdots g_{1m}(t) \\ \cdots \cdots \cdots \\ g_{i1}(t) \cdots g_{ij}(t) \cdots g_{im}(t) \\ \cdots \cdots \cdots \\ g_{m1}(t) \cdots g_{mj}(t) \cdots g_{mm}(t) \end{Bmatrix}_{m \times m} \quad (5)$$

式中 
$$g_{ij}(t) = \sum_{k=1}^{n_{ij(1)}} r_k e^{-d_k t} + \sum_{k=1}^{n_{ij(2)}} 2R_k e^{-a_k t} \cos(b_k t - \theta_k), \quad (6)$$

因此， $G(s)$  包含的耦合度  $d_i(s), d'_j(s)$  可以用  $G(t)$  包含的耦合度  $d_i(t), d'_j(t)$  表示，即

$$d_i(t) = \left[ \sum_{\substack{j=1 \\ (i \neq j)}}^m |g_{ij}(t)| \right] / |g_{ii}(t)|, \quad (7)$$

$$d'_j(t) = \left[ \sum_{\substack{i=1 \\ (i \neq j)}}^m |g_{ij}(t)| \right] / |g_{jj}(t)|,$$

(3) 式表示复平面上  $s$  点处的耦合度，(7) 式表示  $t$  时刻第  $i$  行 ( $j$  列) 的耦合度。对应的时域矩阵  $C(t)$  为

$$C(t) = \begin{Bmatrix} c_{11}(t) \cdots c_{1j}(t) \cdots c_{1m}(t) \\ \cdots \cdots \cdots \\ c_{i1}(t) \cdots c_{ij}(t) \cdots c_{im}(t) \\ \cdots \cdots \cdots \\ c_{m1}(t) \cdots c_{mj}(t) \cdots c_{mm}(t) \end{Bmatrix}_{m \times m} \quad (8)$$

式中

$$C_{ij}(t) = \sum_{k=1}^{n_{ij(1)}} \int_0^t r_k e^{-d_k t} dt + \sum_{k=1}^{n_{ij(2)}} \int_0^t 2R_k e^{-a_k t} \cos(b_k t - \theta_k) dt, \quad (9)$$

要实现系统为对角优势, 用如下常数阵  $C$  表示:

$$C = \begin{bmatrix} c_{11} \cdots c_{1j} \cdots c_{1m} \\ \cdots \cdots \cdots \\ c_{i1} \cdots c_{ij} \cdots c_{im} \\ \cdots \cdots \cdots \\ c_{m1} \cdots c_{mj} \cdots c_{mm} \end{bmatrix}_{m \times m}, \quad (10)$$

$$\begin{aligned} \text{其中 } c_{ij} &= \int_0^{\infty} g_{ij}(t) dt \\ &= \sum_{k=1}^{n_{ij(1)}} \int_0^{\infty} r_k e^{-d_k t} dt + \sum_{k=1}^{n_{ij(2)}} \int_0^{\infty} 2R_k e^{-a_k t} \cos(b_k t - \theta_k) dt \\ &= \sum_{k=1}^{n_{ij(1)}} \frac{r_k}{d_k} + \sum_{k=1}^{n_{ij(2)}} 2R_k \left[ \frac{a_k}{a_k^2 + b_k^2} \cos(\theta_k) + \frac{b_k}{a_k^2 + b_k^2} \sin(\theta_k) \right] \\ &= \sum_{k=1}^{n_{ij(1)}} \frac{r_k}{d_k} + \sum_{k=1}^{n_{ij(2)}} 2R_k \cos(\varphi_k - \theta_k), \end{aligned} \quad (11)$$

$$\text{式中 } \varphi_k = \operatorname{tg}^{-1} \frac{b_k}{a_k}.$$

这样, 系统的耦合度可以用下列常数表示:

$$d_i = \left[ \sum_{\substack{j=1 \\ (i \neq j)}}^m |c_{ij}| \right] / |c_{ii}|, \quad d_j = \left[ \sum_{\substack{i=1 \\ (i \neq j)}}^m |c_{ij}| \right] / |c_{jj}|. \quad (12)$$

**定理.** 根据  $c_{ij}$  求补偿阵  $K_p$  实现解耦, 不一定实现全频解耦, 总能实现某段时间后解耦, 即  $\exists t_1, s.t. t > t_1$  时,

$$|c_{ii}(t)| > d \sum_{\substack{j=1 \\ (i \neq j)}}^m |c_{ij}(t)|, \quad d \geq 1 \quad (13)$$

成立.

$$\text{证明. 记 } \sum_{\substack{j=1 \\ (j \neq i)}}^m |c_{ij}(t)| = s(t), \quad f(t) = |c_{ii}(t)| - d \sum_{\substack{j=1 \\ (i \neq j)}}^m |c_{ij}(t)| = |c_{ii}(t)| - d \cdot s(t),$$

则  $f(+\infty) = |c_{ii}| - d \cdot s(+\infty) = |c_{ii}| - d \sum_{\substack{j=1 \\ (i \neq j)}}^m |c_{ij}| > 0$ , 即  $\lim_{t \rightarrow +\infty} f(t) > 0$ . 由  $f(t)$

的连续性,  $\exists t_1, s.t. t > t_1$  时,  $f(t) > 0$ , 亦即  $|c_{ii}(t)| > d \sum_{\substack{j=1 \\ (i \neq j)}}^m |c_{ij}(t)| > \sum_{\substack{j=1 \\ (i \neq j)}}^m |c_{ij}(t)|$ , 表

明  $t > t_1$  时实现了对角优势(解耦).

证毕.

## 2. 解耦算法

如图 1 所示, 在被控对象  $G(s)$  的前面加入预补偿阵  $K_p$ , 使得  $G(s)K_p$  解耦或弱解

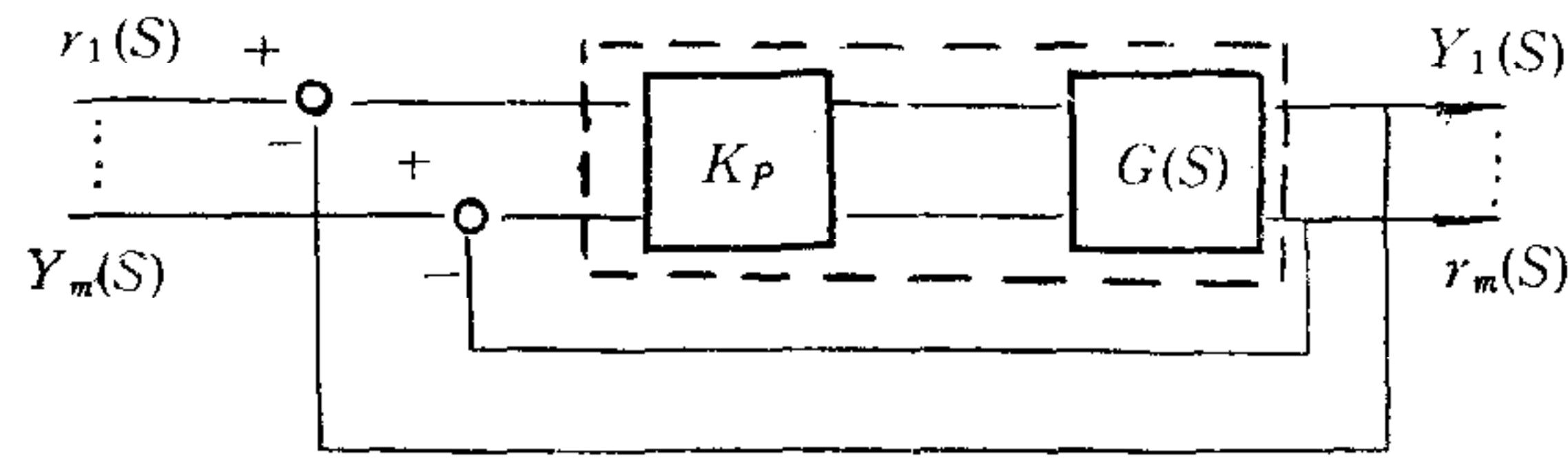


图 1 多输入多输出系统框图

耦。逆乃氏阵列法的基本思想<sup>[1,2]</sup>是：通过使  $G(j\omega_i)K_{p_i}$  的非对角元素模的平方和尽可能小，而对角元素模的平方均等于常数导出计算  $K_{p_i}$  的公式，但  $\omega_i$  值不同，得到的  $K_{p_i}$  也不同。本文的基本思想是：通过使  $CK_p$  的非对角元素模的平方和尽可能小，而对角元素模的平方均等于常数（可以取任意常数，文中取 1）导出计算  $K_p$  的公式，由于  $C$  是复常阵，因此，得到的  $K_p$  是唯一的。方法如下<sup>1)</sup>：

设  $\hat{K}_r$  为  $K_p^T$  的第  $r$  行，并定义  $W_{m \times 2m} \triangleq [R_r C^T : I_m C^T]$ ,

$$E_{r(2m \times 2)} \triangleq [e_1, e_2, \dots, e_r, \dots, e_{r+m}, \dots, e_{2m}]^T, \quad (14)$$

式中  $e_r = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ ;  $e_{r+m} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ , 其余  $e_j = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ , ( $j \neq r, j \neq r+m, j = 1, 2, \dots, 2m$ )。

系统解耦就是求取  $\hat{K}_r (r = 1, 2, \dots, m)$  使得目标函数

$$J_r = \hat{K}_r (A - B_r) \hat{K}_r^T \quad (15)$$

$$\text{在约束条件} \quad \hat{K}_r B_r \hat{K}_r^T = 1 \quad (16)$$

之下为最小<sup>1)</sup>。其中  $A = WW^T, B_r = WE_r E_r^T W^T$ 。

此条件极值问题可转换为无条件极值问题，即使下列目标函数极小：

$$\phi_r = J_r + \lambda_r (1 - \hat{K}_r B_r \hat{K}_r^T). \quad (17)$$

由此可得<sup>1)</sup>

$$M_r \hat{K}_r^T = \mu_r \hat{K}_r^T, J_r = \frac{1}{\mu_r} - 1 = \lambda_r, \quad (18), (19)$$

式中  $M_r = A^{-1} B_r, \mu_r = \frac{1}{1 + \lambda_r}$ 。

要使  $J_r$  为最小，则  $\lambda_r$  应取最小，对应的  $\mu_r$  应取最大。这样待求的  $K_p^T$  就是 (18) 式  $M_r$  的最大特征值  $\mu_r$  对应的特征向量。为减少计算量，经变换得

$$N_r \phi_r = \mu_r \phi_r. \quad (20)$$

式中  $N_r = E_r^T W^T (WW^T)^{-1} W E_r, \phi_r = E_r^T W^T \hat{K}_r^T$ ,

$N_r$  是一个  $2 \times 2$  维矩阵， $\phi_r$  是一个 2 维向量，这样，求解  $\hat{K}_r^T$  最后归结为求解  $N_r$  的最大特征值  $\hat{\mu}_r$  和对应的特征向量  $\hat{\phi}_r$ ，于是

$$\hat{K}_r^T = \frac{1}{\hat{\mu}_r} (WW^T)^{-1} W E_r \hat{\phi}_r, (r = 1, 2, \dots, m). \quad (21)$$

当  $\hat{\mu}_r \leq 0.5$  时，不能用常数阵解耦，这时预补偿阵  $K_p$  应改用矩阵多项式。

1) 何建平，控制系统复域计算机辅助设计，硕士学位论文，1986。

### 三、应用举例

一个四输入四输出的系统，其开环传递函数矩阵如下：

$$G(s) = \frac{1}{s^5 + 2s^4 + 16.93s^3 + 14.42s^2 + 4.48s + 0.32} \times \begin{bmatrix} s^2 + 4s + 3 & s + 5 & 2s + 2 & s - 4 \\ s + 3.5 & s^2 + 3s + 2 & s - 2 & s^2 - 4s - 12 \\ s + 1.8 & s + 6 & s + 7 & s - 8 \\ s + 2.4 & s + 4 & s + 3 & s + 7 \end{bmatrix}$$

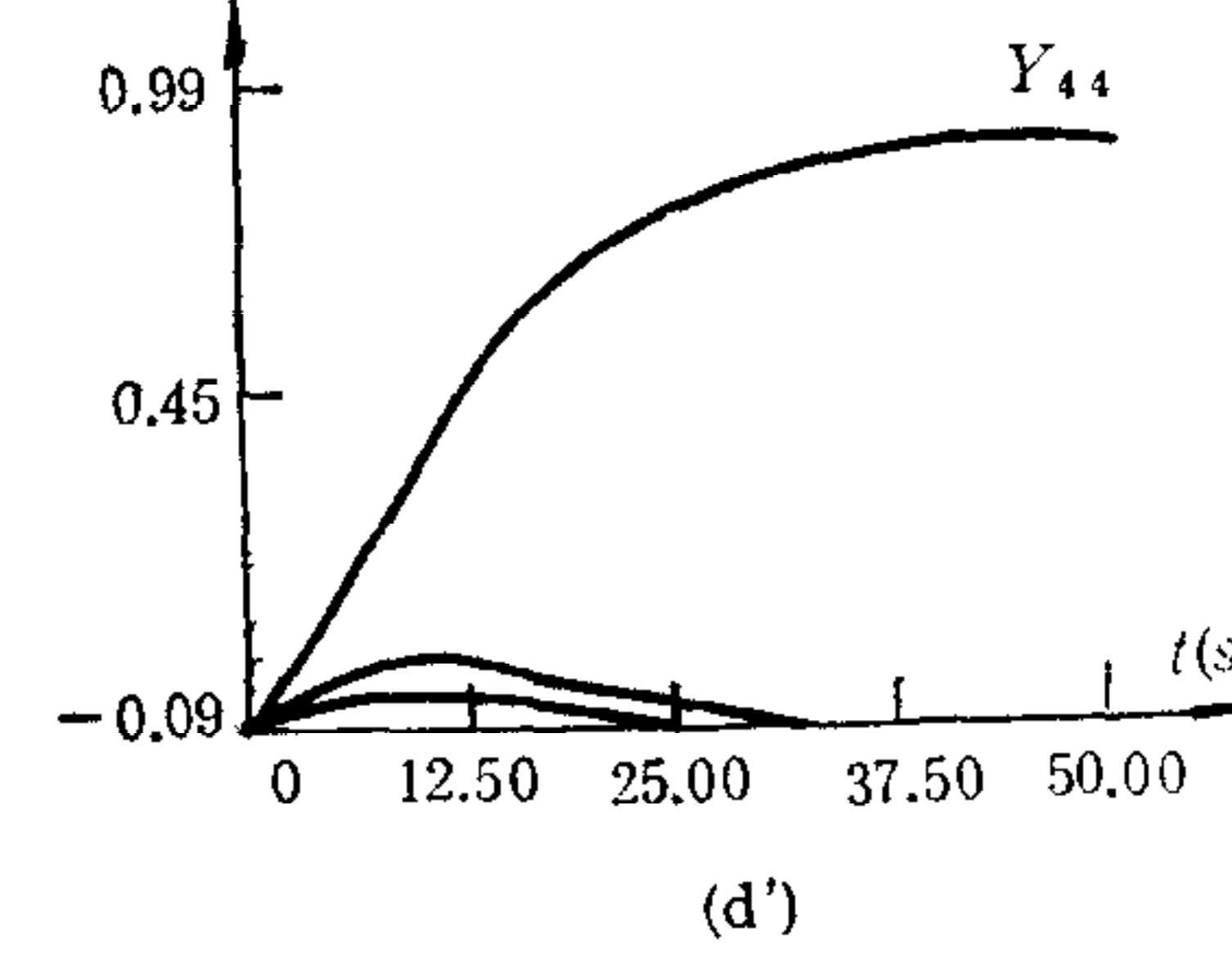
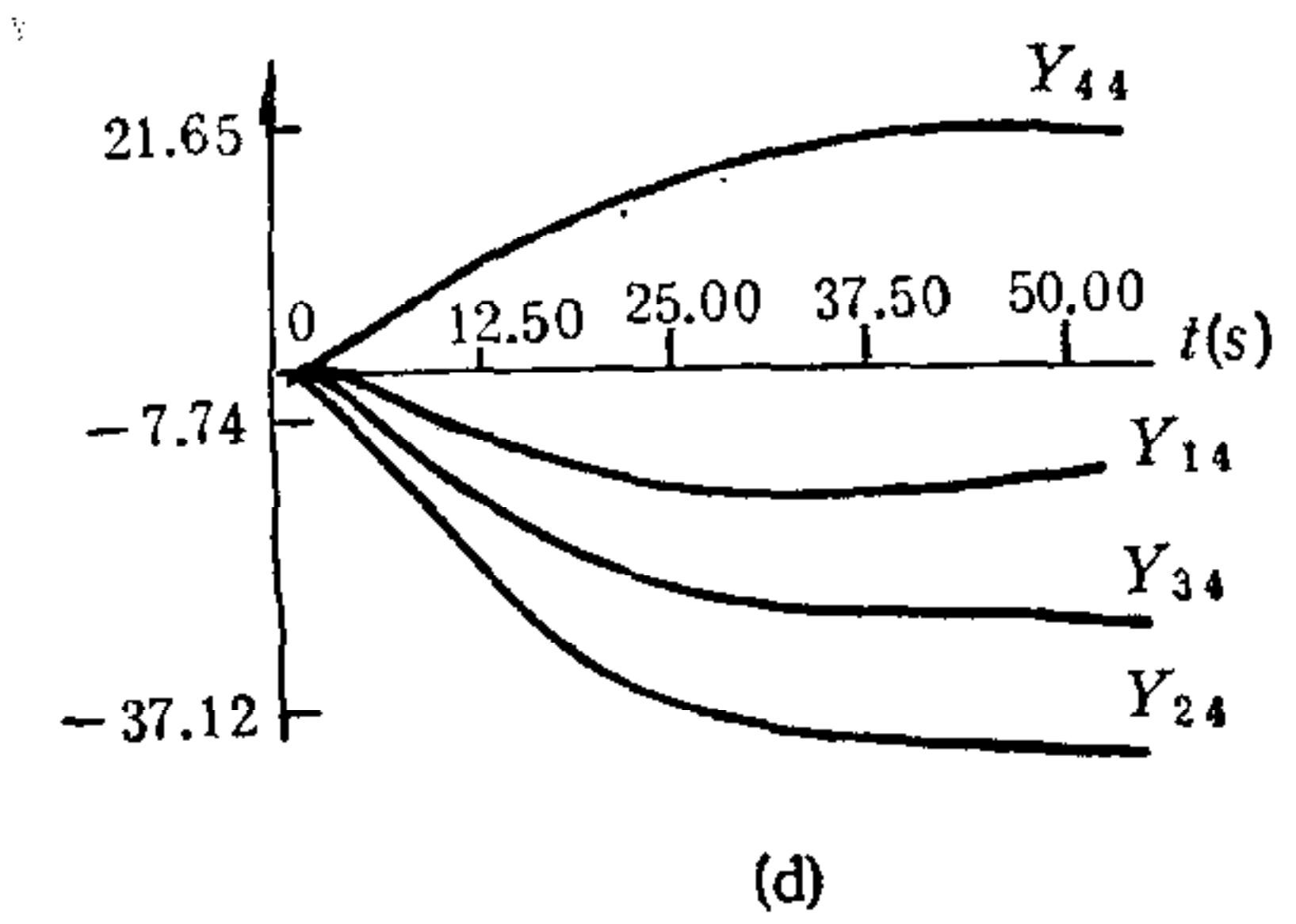
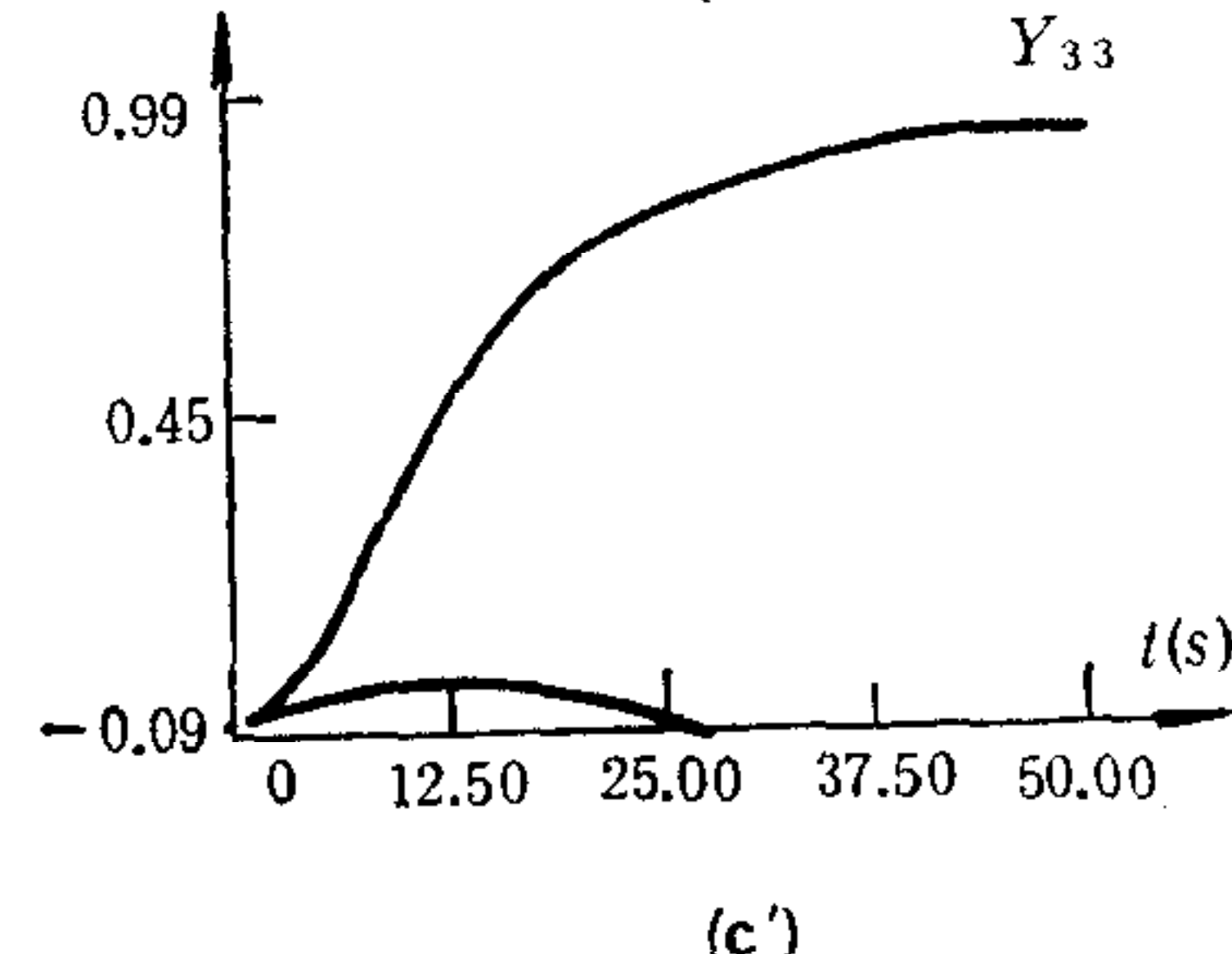
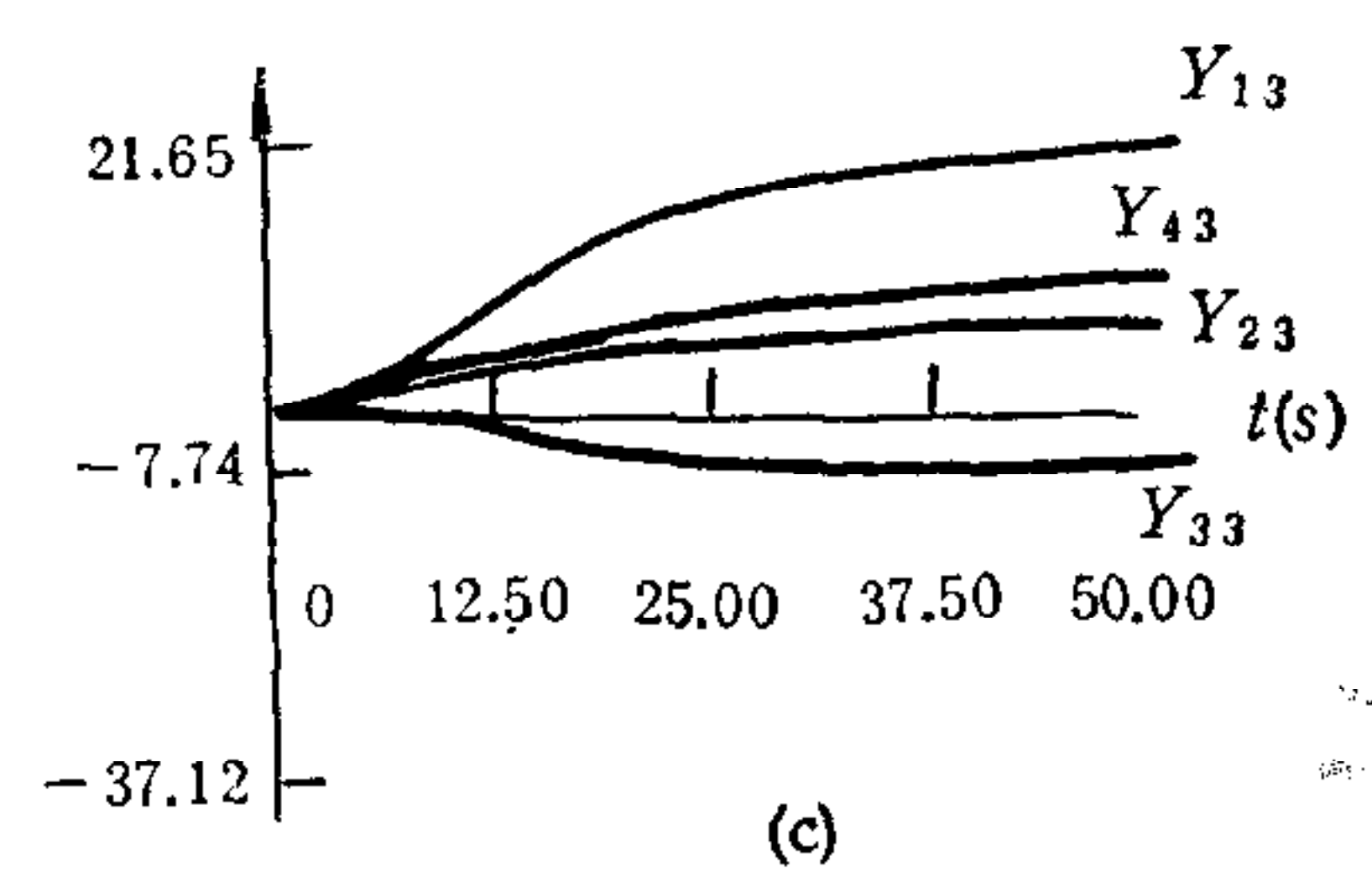
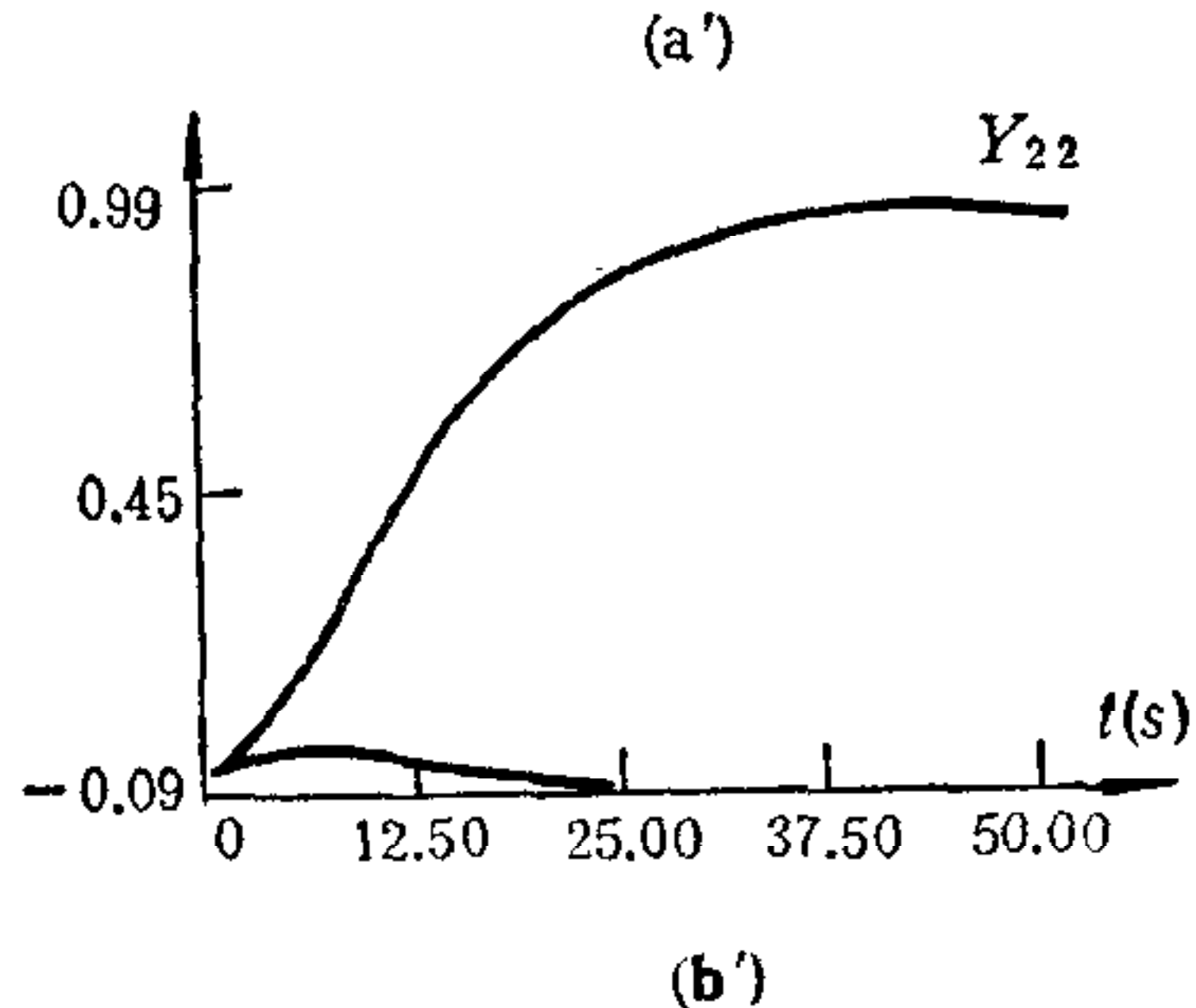
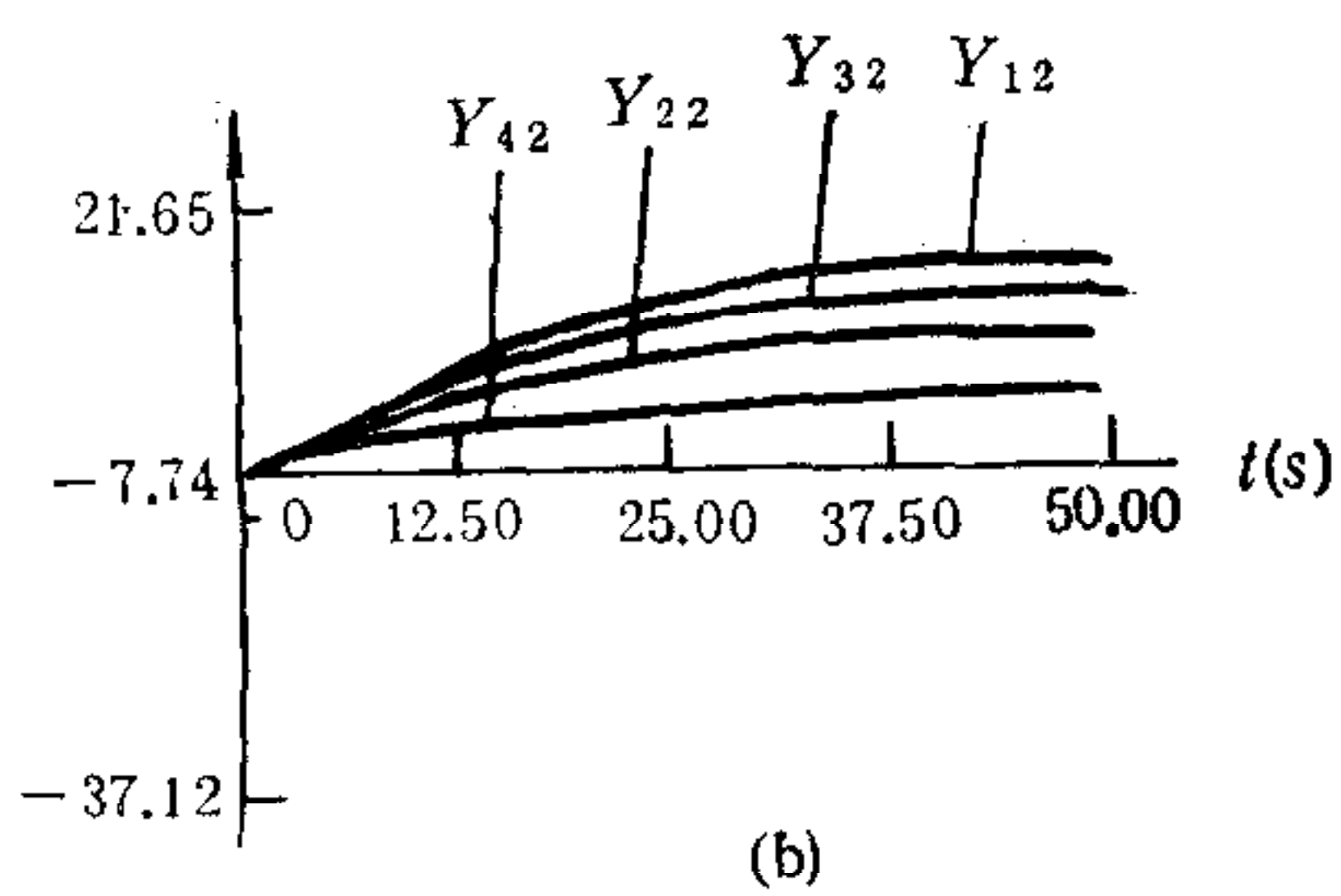
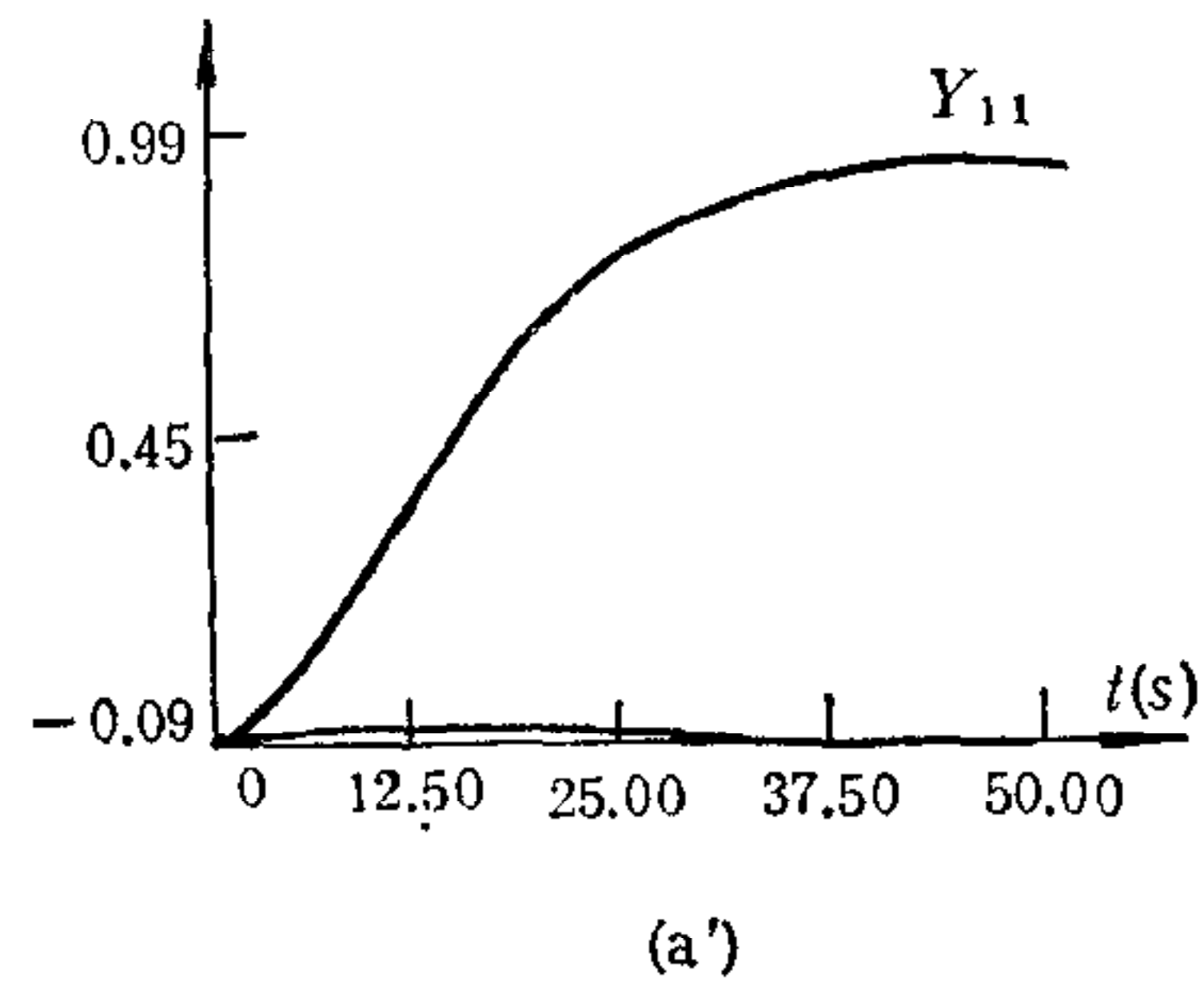
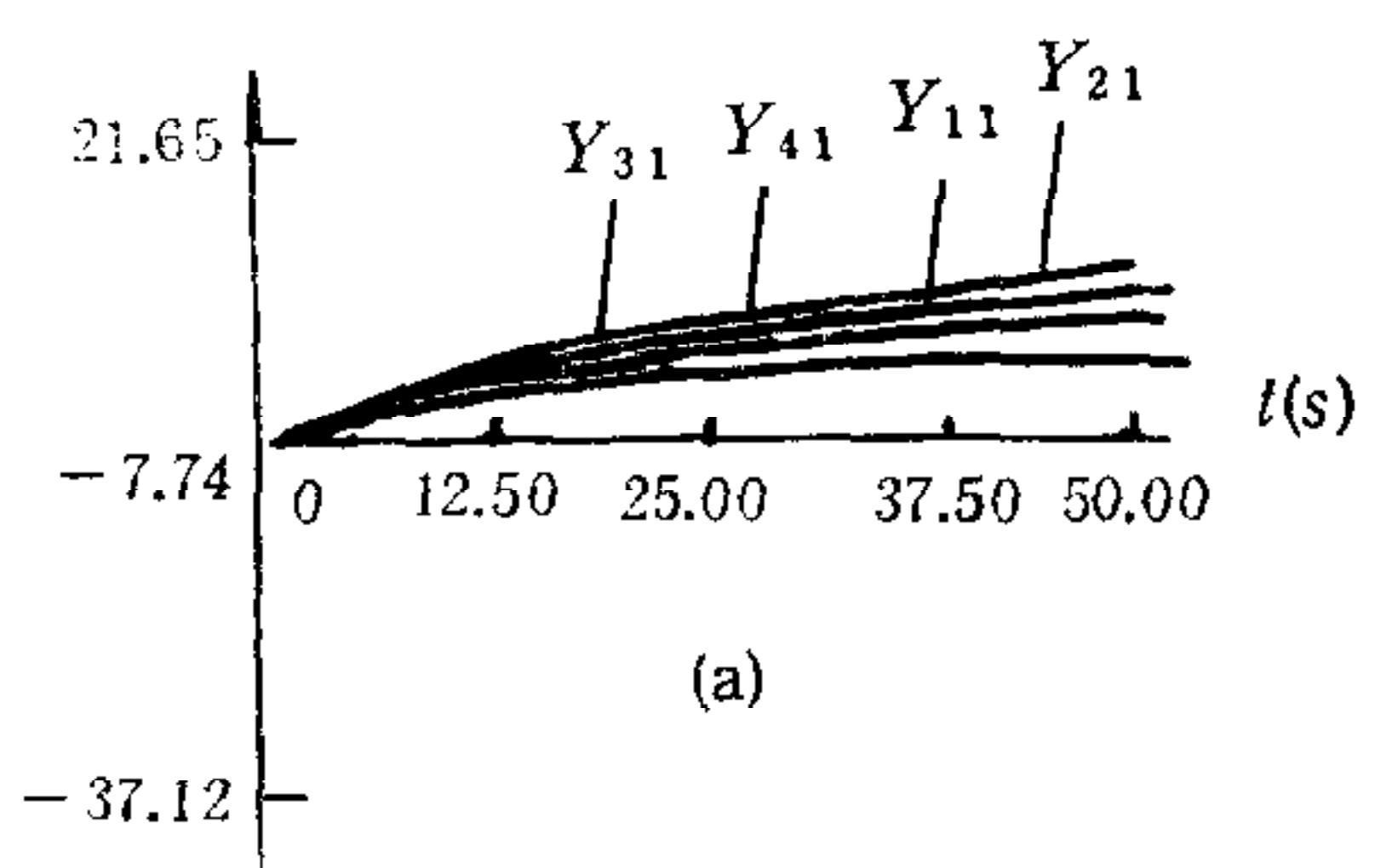


图 2  $G(s)$  的开环阶跃响应

图 3  $G(s)K_p$  的开环阶跃响应

$G(s)$  的开环阶跃响应如图 2 所示,由图 2 可知,系统具有非常严重的耦合作用.用此法在 IBM-PC 机上通过辅助设计软件求得预补偿阵  $K_p$  为

$$K_p = \begin{bmatrix} -0.2809824 & 0.1750178 & 0.0471618 & 0.1939547 \\ 0.3106385 & -0.1368756 & -0.0692559 & -0.1367445 \\ -0.1925242 & 0.06252357 & 0.08035195 & 0.08921003 \\ 1.299031E-3 & -8.569497E-3 & -0.01102258 & 0.01914028 \end{bmatrix}$$

加入预补偿阵  $K_p$  后,开环  $G(s)K_p$  的阶跃响应如图 3 所示,由图 3 可知,系统的耦合作用基本消除,几乎实现了全解耦.这表明,本文提出的实现系统解耦的预补偿阵  $K_p$  的设计方法是可行且有效的.

致谢. 在研究过程中,得到了陆载德教授的指导,在此表示感谢.

### 参 考 文 献

- [1] 白方周等,多变量系统频域理论与设计技术,国防工业出版社,1986.  
[2] Rosenbrock, H. H., Computer-aided Control System Design, Academic Press, London, 1974.

## A NEW DESIGN METHOD OF PRECOMPENSATING MATRIX FOR REALIZING SYSTEM DECOUPLING

HE JIANPING

(Institute of Automation, Taiyuan Iron and Steel Corporation, 030003)

JIN PING

(Taiyuan University, 030012)

ZENG JIANPING

(Dept. of Automatic Control, Taiyuan Institute of Mechanics, 030051)

### ABSTRACT

This paper presents a new design method for implementing the pre-compensator constant matrix of decoupling of a multivariable system. The method evolves a complex constant matrix  $C$  from the transfer function matrix  $G(s)$ , then it calculates precompensating matrix  $K_P$  by matrix  $C$ . The basic principle is discussed and an example is illustrated. The results are satisfactory.

**Key words :** Decoupling control; compensator; multivariable systems.