第 22 卷 第 8 期 控 制 与 决 策 2007 年 8 月
Vol. 22 No. 8 Control and Decision Aug. 2007

文章编号: 1001-0920(2007)08-0946-05

基于蚁群算法的 MAS 多目标协调优化

郭红霞, 吴 捷, 黄飞龙
(华南理工大学 电力学院, 广州 510640)

摘 要: 利用蚁群算法的群体搜索策略,研究了基于蚁群算法的 MAS 多目标协调优化机制. 对每个 Agent 的目标函数分配一群蚂蚁,使之在问题空间寻优,并对所有的优化解采用谈判机制进行协调,以产生多目标优化问题的 Pareto 折衷解. 采用"误差率"和"空间矩阵"方法对算法的性能指标进行度量. 用该方法求解两个典型的多目标优化测试函数,仿真结果表明所提出的方法可成功地解决 MAS 的多个目标函数的优化问题,收敛速度较快.

关键词: 蚁群算法; 多目标协调优化; 谈判机制
中图分类号: TP301 **文献标识码**: A

Multi-objective coordinated optimal of MAS based on ant system

GUO Hong-xia, WU Jie, HUANG Fei-long
(College of Electrical Power, South China University of Technology, Guangzhou 510640, China. Correspondent: GUO Hong-xia, E-mail: guohx@scut.edu.cn)

Abstract: A mechanism of multiple objective coordinated optimization based on ant system for MAS is proposed by using colony searching strategy. A family of ants is assigned for each objective of each agent, by which the optimal solution is searched in solution space. Negotiation mechanism is applied to coordinate all the solutions. Performance is measured by usihg "error ratio" and "spacing" metrics. Multiple objective coordinated optimization based on ant system is applied to two typical multiple objective test functions. Simulation results show that this algorithm is able to solve the multiple objective optimization problems successfully and has fast convergence speed.

Key words: Ant system; Multiple-objective coordinated optimization; Negotiation mechanism

1 引 言

具有不同目标的多个 Agent 之间的协调优化问题是 MAS 研究的一个重点. 其主要特点是:由于各 Agent 目标之间的矛盾性,实际上不可能存在一个绝对的最优解,而只能存在 Pareto 最优解集. 因此,多目标协调优化问题的主要目标是寻求 Pareto 解集中的一个或多个满意解. 解决多目标协调优化问题通常有两种比较常用的方法:一种是通过将各目标进行线性加权,将多目标优化问题转化为单目标优化问题. 这种方法的最大优点是许多用于单目标协调优化的求解方法都可用于多目标优化问题,但它存在如下的问题:权系数的合理选择;在物理意义不明确的情况下,各个目标难以用同一尺度衡量;在不满足凸性假设时,存在丢失部分非劣解的问题. 另一种常用的方法是采用群体搜索策略,例如遗传算法等并行进化计算的方法. 典型的有 Fonseca 和 Fleming 提出的多目标遗传算法(MOGA), Srinivas 和 Deb 提出的非劣分类遗传算法(Non-dominated sorting genetic algorithm),Lino 提出的多目标进化策略(MOES)等. 这些算法的优点是无需非凸性假设,能有效阻止种群分化的发生,但存在的问题是需要采用合适的参数编码以及如何确定遗传算法中的有关参数.

蚁群算法作为一种群体策略的并行进化计算方法,运用了正反馈原理、分布式计算及贪婪式启发搜索等方法,在一定程度上可加快进化过程. 通过个体之间进行信息交流和传递,有利于发现较好的解. 蚁群算法也被认为是一种特殊的强化学习算法[1],其优点在于不需进行合适的参数编码,运用正反馈原理加快了进化过程.

将蚁群算法引入连续空间的函数寻优,通过将传统蚁群算法中的"信息量留存"过程拓展为连续空

收稿日期: 2006-03-21; **修回日期**: 2006-07-07.
基金项目: 国家自然科学基金重点项目(60534040).
作者简介: 郭红霞(1971—),女,河南偃师人,讲师,博士,从事多 Agent 系统、强化学习及其在电力系统中的应用研究; 吴捷(1937-),男,吉林省吉林市人,教授,博士生导师,从事电力系统自动化等研究.

间中的"信息量分布函数",使蚂蚁在问题空间中作地毯式搜索.这样,可减少算法的迭代次数,加快收敛时间,表明了蚁群算法在连续空间优化问题中具有很好的应用前景[2,3].将蚁群算法用于连续空间的多目标协调优化问题,对每个 Agent 的目标函数分配一群蚂蚁,利用蚂蚁个体之间的信息交换和传递来加快优化的收敛速度.这种方法,迄今国内外还鲜有文章发表.因此,本文将它运用于 MAS 的多目标优化问题,并采用谈判机制进行多个目标解的协调.

2　基于蚁群算法的多目标协调优化

蚁群算法(AS)是受自然界中真实蚁群集体行为的启发而提出的一种基于种群的模拟进化算法,属于带有构造性特征的随机搜索算法.该算法根据蚁群从蚁穴到食源搜索最短路径的过程与 TSP 问题之间的相似性,通过模拟人工蚂蚁搜索食物的过程来求解具有 NP-难的 TSP 问题,并得到了最优解.该算法还被应用于求解二次指派(QAP)问题、job-shop 调度问题等,取得了较好的实验结果.这些结果已经充分显示了蚁群算法在求解复杂优化问题(特别是离散优化问题)的优越性.

2.1　连续空间的蚁群算法

连续空间的函数优化问题与离散的 TSP 问题有明显的区别[3].当用蚁群算法解决 TSP 问题时,如果蚂蚁处于某一城市,则可直接通过计算选择概率,确定下一个城市.而在连续空间的函数优化中,解空间是以区域性方式表示.因此,用于连续空间寻优的蚁群算法与离散空间的蚁群算法之间在蚁群信息量留存方式、蚁群在解空间的寻优方式以及蚁群行进策略等方面有所不同.

以一维空间函数的寻优为例,考察它的搜索情况.设优化函数为 $\min Z = f(x), x \in [a,b]$.为了实现蚁群算法的群体搜索策略,构造蚂蚁的转移概率准则如下:

令 m 只人工蚂蚁,初始位置随机位于解空间 $[a,b]$ 的 n 个等分区域的某些位置,各区域蚂蚁的状态转移概率定义为

$$p_{ij} = \begin{cases} \dfrac{\tau_j^{\alpha}\eta_{ij}^{\beta}}{\sum\limits_{j=1}^{m}(\tau_j)^{\alpha}(\eta_{ij})^{\beta}}, & \eta_{ij} > 0; \\ 0, & \eta_{ij} \le 0. \end{cases} \tag{1}$$

式中:τ_j 为区域 j 的吸引强度(相当于 TSP 问题中的路径信息素强度);期望值 η_{ij}(相当于 TSP 问题中的启发信息)可定义为 $\eta_{ij} = f_{i\min} - f_{j\min}$,即蚁群在区域 j 与区域 i 中目前已经搜索到的某处目标函数最小值的差值;参数 $\alpha, \beta > 0$ 为启发式因子.

区域 j 吸引强度的更新公式为

$$\tau_j(t+1) = \rho\tau_j(t) + \sum_{k=1}^{m}\Delta\tau_j^k,\ j = 1,2,\cdots,n; \tag{2}$$

$$\Delta\tau_j^k = \begin{cases} QL_j^k, & L_j^k > 0, \\ 0, & L_j^k \le 0, \end{cases}\quad j = 1,2,\cdots,n. \tag{3}$$

式中:$\Delta\tau_j^k$ 反映本次循环中蚂蚁 k 在区域 j 的局部搜索中吸引强度的增加;L_j^k 表示在本次循环中蚂蚁 k 在区域 j 的局部搜索中目标函数的变化量,定义为 $L_j^k = f(x_j^k) - f(x_{j_0}^k)$,$x_j^k$ 为本次循环中蚂蚁 k 在区域 j 的局部搜索中的当前位置,$x_{j_0}^k$ 为本次循环中蚂蚁 k 在区域 j 的局部随机搜索中的初始位置;$0 \le \rho \le 1$,它体现各个区域吸引强度的持久性;给定参数 Q 为蚂蚁释放的信息素密度;$\tau_j(0) = c, \Delta\tau_j(0) = 0$.因此,函数 $f(x)$ 的寻优问题可借助于 m 只蚂蚁在 $x \in [a,b]$ 的 n 个等分区域间的不断移动,以及一些区域内的局部搜索来进行.处于区域 i 中的第 k 只蚂蚁的转移及其搜索规则为

$$j = \begin{cases} \arg\max\limits_{j\in\{1,2,\cdots,n\}}\{p_{ij}\},\ \text{进入第 } j \text{ 个区域搜索}, & p_{ij} > 0; \\ i,\ \text{在本区域内搜索}, & p_{ij} = 0. \end{cases}$$

2.2　多目标协调优化算法

多目标协调优化问题可定义为在一组约束条件下,极大化(或极小化)多个 Agent 的目标函数.本文将连续空间的蚁群算法用于 MAS 中的各 Agent 的多个目标协调优化中,利用蚁群算法的群体搜索策略,对每个 Agent 的目标函数分配一群蚂蚁,使之在问题空间中作地毯式搜索.当每个 Agent 完成一定的步骤或到达它的目标时,完成一次迭代,找到它当前的优化解.当所有的 Agent 都完成一个迭代时,所有 Agent 的优化解采用谈判机制进行协调,产生多目标优化问题的 Pareto 折衷解,同时更新每个 Agent 中各区域的吸引强度,进行下一次的迭代.

令第 i 个 Agent 的目标函数为

$$\begin{aligned} &\min\ Z_i = f_i(X), \\ &\text{s.t.}\ g_j(X) \le 0,\ j = 1,2,\cdots,p. \end{aligned} \tag{4}$$

其中:$X = (x_1, x_2, \cdots, x_n)$ 为 n 维向量,x_i 的解空间为 $[x_{i_0}, x_{i_f}]$;$g_j(X) \le 0$ 为 Agent 的约束条件,包括不等式约束和等式约束.取 m 组蚂蚁,每组的人工蚂蚁数与向量的维数相同,即每组取 n 个人工蚂蚁.对每个 x_i 的解空间进行 N_i 等分,每组蚂蚁的初始位置随机位于解空间的 $N_1 \quad N_2 \quad \cdots \quad N_n$ 个等分区域的某些位置处.在每次循环中,各区域蚂蚁的状态转移概率按式(1)计算,各区域吸引强度的更新按式(2)和(3)计算.

采用如下方法处理约束条件:自变量的取值自动适应其定义域范围. 采用乘子罚函数法[4] 将其他等式和不等式约束转化为无约束问题.

当每个 Agent 都采用这种方法找到本次迭代的最优解时,采用谈判机制对所有 Agent 的当前最优解进行协调.

本文采用文献[1] 提出的谈判机制来产生满足所有目标的折衷解. 这种谈判机制已在离散优化问题的 Ant-Q 算法中得到验证[5].

在谈判机制中,判断第 i 个 Agent 的解是否同时也是其他 Agent 当前的最优解,如果是,则它就是目前的折衷解(Compromise_solution,C_S) 中的一个候选解;如果目前有多个候选解,则取其中最好的那个解作为折衷解. 折衷解采用非劣标准构成 Pareto 前端,并保存在 Perato 前端中. 如果没有一个满足所有目标的最优解,则在当前所有 Agent 的最优解中任取一个解作为折衷解. 基于蚁群算法的多目标函数优化算法及其谈判机制描述如下:

1) 基于蚁群算法的多目标函数优化算法.

```
初始化
  将每个 Agent 中的各 Δτj 及 τj 赋初值
    Repeat (对每个 Agent)
      将 m 个蚂蚁置于各自的初始邻域中
        Repeat (对每个 Agent 中的每个蚂蚁)
         按概率 ρij 移动或作邻域搜索
           计算各蚂蚁的目标函数 Zk(k = 1, …, m),
             记录当前的最好解
         按更新公式修正每个 Agent 的吸引强度
C_S = negotiation(m solution)
        If (C_S 是非劣解)
        Pareto = Pareto + C_S
      count ←count + 1
  如果 count < 预定的迭代次数,则返回到第 2 步
输出 Pareto 解集
```

2) 基于蚁群算法的谈判机制(针对 m 个解进行折衷).

```
初始化
  C_S(折衷解) = 0
    Repeat (对每个解 i)
        Repeat (对每个目标函数 j)
          If 第 j 个目标函数的第 i 个解满足所有
             的目标函数
           则 C_S = best_of ( solution_i ,C_S)
          else if
              (C_S 是空集)
           则 C_U 任取 m 个解中的一个
        Return C_U
```

3 仿真实验与分析

为了测试本文提出的用于多目标函数优化的蚁群算法的性能,采用以下两个度量标准来衡量测试问题的结果:一个是 Error Ratio 标准[6],一个是 Spacing 标准[7]. 描述如下:

Error Ratio 的数学表达式为

$$E \triangleq \frac{\sum_{i=1}^{n} e_i}{n}. \tag{5}$$

其中:n 为 PF_{Ant} 中向量的个数,PF_{Ant} 为采用本文所提出的算法得到的解;e_i 为

$$e_i = \begin{cases} 0, & i \in PF_{true}; \\ 1, & \text{otherwise}. \end{cases} \tag{6}$$

如果 $E = 0$,则表示由本文算法得到的 PF_{Ant} 中的解向量属于 PF_{true};如果 $E = 1$,则不属于 PF_{true}. 其中 PF_{true} 为测试问题实际的 Pareto 非劣解.

为了度量 PF_{Ant} 中向量的分布,采用 Spacing 标准. 此标准用于测量 PF_{Ant} 中相邻两个解之间的距离,其数学描述为

$$S \triangleq \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\bar{d} - d_i)^2}. \tag{7}$$

其中:$d_i = \min_j(|f_1^i(\vec{x}) - f_1^j(\vec{x})| + |f_2^i(\vec{x}) - f_2^j(\vec{x})|)$, $i, j = 1, 2, \cdots, n$; $\vec{d}$ 为所有 d_i 的平均值;n 为 PF_{Ant} 中向量的个数;$S = 0$ 表示 PF_{Ant} 中的解向量是等距离分布的.

3.1 仿真实验

本文通过求解两个典型的多目标优化测试函数来验证算法的有效性.

第 1 个是文献[8] 中提到的用于测试多目标优化算法的测试函数. 其目标函数和约束条件为

$$\begin{aligned} &\min f_1(x, y) = x, \\ &\min f_2(x, y) = \\ &(1 + 10y)\left[1 - \left(\frac{x}{1 + 10y}\right)^{\alpha} - \frac{x}{1 + 10y}\sin(2\pi q x)\right]. \\ &\text{s.t. } 0 \le x, y \le 1, q = 4, \alpha = 2. \end{aligned} \tag{8}$$

第 2 个是文献[9] 中提到的,其目标函数和约束条件为

$$\begin{aligned} &\min f_1(x, y) = (x - 2)^2 + (y - 1)^2 + 2, \\ &\min f_2(x, y) = 9x - (y - 1)^2, \\ &\text{s.t. } -20 \le x, y \le 20, 0 \ge x^2 + y^2 - 255, \\ &\qquad 0 \ge x - 3y + 10. \end{aligned} \tag{9}$$

采用本文提出的蚁群算法对以上两个问题进行仿真实验. 算法的参数取值分别为:蚂蚁在各区搜索中释放的信息素密度 $Q = 1$,吸引强度启发式因

子 $\alpha = 1$,期望值的启发因子 $\beta = 1.5$,$\rho = 0.7$. 每个仿真迭代次数为 20 次,每次迭代循环 100 次,函数解空间的分区数为 $NL = 10 \times 10$,蚂蚁组数取 9 组. 由于每个测试问题的每个问题为两维函数,每组取 2 只蚂蚁.

仿真结果如图 1～图 4 所示. 图 1 和图 3 为采用蚂蚁算法求解时解的搜索过程;图 2 和图 4 为仿真得到的优化问题的非劣解前沿. 两个问题所得到的性能度量结果如表 1 所示,其中多目标遗传算法(MOMGA)是文献[10]所采用的算法针对同样的问题得到的结果.

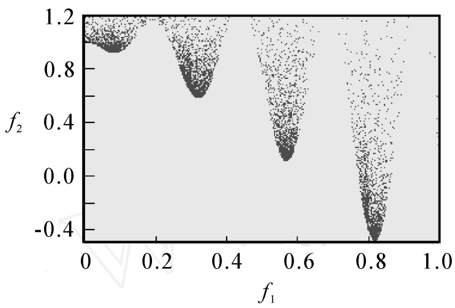

图 1　问题 1 的解搜索过程

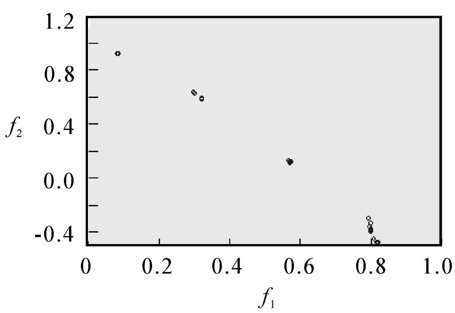

图 2　问题 1 的 Pareto 前沿

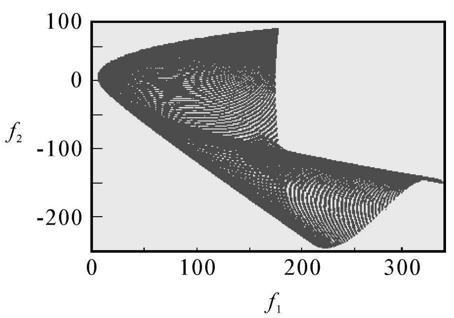

图 3　问题 2 的解搜索过程

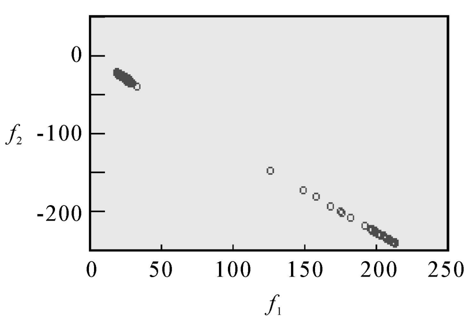

图 4　问题 2 的 Pareto 前沿

表 1　性能指标的度量比较

问题	Ant 算法		MOMGA	
	E	S	E	S
1	0	0.00～0.05	20～60	0.00～0.03
2	0.076～0.508 2	9.644 6～9.669 2	NA	NA

3.2　仿真结果分析

从图 1～图 4 和表 1 的仿真结果可以看出,将蚂蚁算法应用于多目标函数优化,并采用谈判机制对每个 Agent 的优化解进行协调,总体而言可以较成功地解决 MAS 的多个目标函数的优化问题. 迭代到第 7 次便能找到非劣解,收敛速度较快,与文献[10]的 MOMGA 算法相比,对第 1 个问题采用蚁群算法所得到的非劣解几乎全部属于问题实际的非劣解,解的分布也较为均匀. 对于第 2 个问题,结果没有第 1 个好,但与 MOMGA 相比,结果还是比较令人满意的. 在最差的情况下,也有 50 %非劣解不属于实际的 Pareto 前沿.

4　结　　语

本文将蚁群算法应用于连续空间的多目标协调优化问题,对每个 Agent 的目标函数分配一群蚂蚁,使之在各自的问题空间中寻优. 此算法对多个目标函数采用并行进化计算的方法,利用蚂蚁个体之间的信息交换和传递来加快优化的收敛速度. 采用谈判机制对所有 Agent 的优化解进行协调,以产生多目标优化问题的 Pareto 折衷解. 最后通过求解两个典型的多目标优化测试函数,并采用性能度量标准进行测试. 从仿真结果可以看出,迭代到第 7 次便能找到非劣解. 因此,采用蚁群算法解决多目标优化问题的收敛速度较快,可用于求解一类连续空间的多目标协调优化问题.

参考文献(References)

[1] Mariano C E, Morales E F. Distributed reinforcement learning for multiple objective optimization problems [C]. Evolutionary Computation 2000, Proc of 2000 Congress. California, 2000, 1: 188-195.

[2] 汪镭, 吴启迪. 蚁群算法在连续空间寻优问题求解中的应用[J]. 控制与决策, 2003,18(1): 45-48.
(Wang Lei, Wu Qi-di. An application of ant system algorithm in continuous space optimization[J]. Control and Decision, 2003,18(1): 45-48.)

[3] 詹士昌. 蚁群算法及其在连续性空间优化问题中的应用[D]. 杭州: 浙江大学, 2002.
(Zhan Shi-chang. Ant system algorithm and its application in continuous space optimization [D]. Hangzhou: Zhejiang University,2002.)

[4] 柳焯. 最优化原理及其在电力系统中的应用[M]. 哈尔

滨:哈尔滨工业大学出版社,1988.
(Liu Zhuo. Optimization theory and its application in power system [M]. Harbin: Harbin Institute of Technology Press, 1988.)
[5] Mariano C E, Morales E F. (1999a) MOAQ an ant-Q algorithm for multiple objective optimization problems [C]. Proc of the Genetic and Evolutionary Computation Conf. San Fancisco, 1999: 894-901.
[6] Van Veldhuizen D, Lamont G. Multiobjective evolutionary algorithms test suites [C]. Proc of Symposium Applied Computing. San Antonio, 1999: 351-357.
[7] Schott J. Fault tolerant design using simple and multicriteria genetic algorithms [D]. Cambridge: Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1995.
[8] Deb K. Multiobjective genetic algorithms: Problem difficulties and construction of test problems [R]. Dortmund: University of Dortmund, Department of Computer Science/XI, 1998.
[9] Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms [J]. Evolutionary Computation, 1994, 2(3): 221-248.
[10] Van Veldhuizen D. Evolutionary algorithms: Classification, analysis and new innovations[D]. Ohio: Air Force Institute of Technology, 1999.

(上接第 942 页)

[4] 魏蛟龙,张驰. Internet 拥塞控制和资源分配中的对策论分析框架[J]. 电子学报, 2003, 31(10): 1452-1455.
(Wei Jiao-long, Zhang Chi. A game theoretical framework for congestion control and resource allocation in the internet[J]. Acta Electronica Sinica, 2003, 31(10): 1452-1455.)
[5] 陈惠民,卢欣,王普,等. 对策论方法在信息网络资源分配中的应用[J]. 通信学报, 1999, 20(8): 63-68.
(Chen Hui-min, Lu Xin, Wang Pu, et al. Using game theoretic approaches for resource allocation in information network [J]. J of China Institute of Communications, 1999, 20(8): 63-68.)
[6] 井元伟,杨开阳,金福德,等. 具有多优先级多服务网络的激励价格控制[J]. 控制与决策, 2001, 16(4): 425-429.
(Jing Yuan-wei, Yang Kai-yang, Jin Fu-de, et al. Inventive pricing problem of multi-service networks with multi-priority- level[J]. Control and Decision, 2001, 16(4): 425-429.)
[7] Shenker S. Fundamental design issues for the future Internet [J]. IEEE J on Selected Areas in Communications, 1995, 13(7): 1176-1188.
[8] Ho Y C, Luh P B, Olsder GJ. A control theoretic view on incentive[J]. Automatica, 1982, 18(2): 167-179.
[9] Varian H R. Microeconomic analysis[M]. 3rd ed. New York: Norton, 1992.
[10] Stalling W. TCP/IP and ATM design principles[M]. New Jersey: Prentice Hall, 1998.
[11] Odlyzko A. The economics of the Internet: Utility, utilization, and quality of service[EB/OL]. (1998-07-07). AT&T Labs-Research, http://www.research.att.com/~amo.

(上接第 945 页)

[3] Zhang Ling-bo, Mao Jian-qin. An approach for selecting weighting matrices of LQ of optimal controller design based on genetic algorithms[C]. Proc of IEEE Tencon'02. 2002: 1331-1334.
[4] Broussard J R. A quadratic weight selection algorithm [J]. IEEE Trans on Automatic Control, 1982 27(4): 945-947.
[5] Harvey C A, Stein G. Quadratic weights for asymptotic regulator properties [J]. IEEE Trans on Automatic Control, 1978, 23(3): 378-387.
[6] Arar A S, Sawan M E, Rob R A. The inverse output feedback LQ problem[C]. Proc of American Control Conf. Ballimore Maryland, 1994: 2736-2737.
[7] 王耀青. LQ 最优控制系统加权矩阵 Q 的一种数值算法[J]. 控制与决策, 2000, 15(5): 513-517.
(Wang Yao-qing. A numerical algorithm for the weighting matrix Q of LQ optimal control systems[J]. Control and Decision, 2000, 15(5): 513-517.)
[8] 郑大钟. 线性系统理论[M]. 第 2 版. 北京:清华大学出版社, 2002: 197.
(Zheng Da-zhong. Linear system theory[M]. 2nd ed. Beijing: Tsinghua University Press, 2002: 197.)