

基于混合遗传算法的模拟电路自动设计研究

焦素敏, 王学梅, 王彩红

(河南工业大学 信息科学与工程学院, 河南 郑州 450001)

摘 要: 模拟电路在电子系统设计中具有重要意义。模拟电路设计包括电路拓扑结构设计和元件值设计, 这两个方面对计算机辅助模拟电路设计都很重要。但是, 传统的遗传算法在不断优化电路元件值时效率低。因此, 本文提出一种具有超突变和精英策略的混合遗传算法 HME-GA。该方法不仅可以用来设计电路的拓扑结构, 而且可用于设计电路的元件值。实验结果表明, HME-GA 算法优于简单遗传算法 GA, 可作为计算机辅助模拟电路设计的有效方法, 在模拟电路自动化设计中具有非常重要的意义和巨大的应用潜力。

关键词: 超突变; 精英; 遗传算法; 模拟电路设计; 计算机辅助设计

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2012)05-0129-05

A Hybrid Genetic Algorithm with Hyper-mutation and Elitist Strategies for Automated Analog Circuit Design

JIAO Su-min, WANG Xue-mei, WANG Cai-hong

(School of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China)

Abstract: Analog circuits are of great importance in electronic system design. Considering the lack of simple GA on evolving the analog circuit's components values, this paper proposes a hybrid genetic algorithm with hyper-mutation and elitist strategy. Experimental results show that, the proposed algorithm performs much better than simple GA. This paper proposes an efficient way of automated analog circuit design. It is obvious that, evolutionary methods have significant importance and large potential in analog circuit design automation.

Key words: hyper mutation; elitist; GA; Analog Circuit Design; Computer Aided Design

1 引言

电子设计自动化(EDA)技术是现代电子系统设计的核心, 数字电路的自动化设计技术已相对成熟, 而模拟电路由于线路复杂、工艺难度大, 其 EDA 方法与工具的研究和开发落后于数字电路 EDA。这就使得模拟电路成为系统整体性能的限制因素和最难设计的部分。因此, 模拟电路自动化设计技术, 已成为一个亟待解决的难题。在过去的几十年中, 虽然研究者曾提出多种模拟电路的自动化设计方法, 但它们并不尽善尽美^[1-2]。

研究表明, 遗传算法 GA 是模拟电路自动化设计的一种有效方式。GA 算法包括交叉和变异, 在基于 GA 的模拟电路自动设计中, 交叉用于进化设计

电路的拓扑结构, 变异则用来进化计算电路的元件值。而 GA 具有高交叉率(0.6~1.0)和低突变率(0.001~0.1)的固有属性^[3]。因此, 简单的 GA 算法在设计模拟电路的元件参数值时效果不佳。在模拟电路自动化设计中如何弥补简单 GA 的不足就显得非常重要。

本文针对简单 GA 在自动模拟电路设计中的不足, 提出一种具有超突变和精英策略的新型混合遗传算法: HME-GA (Genetic Algorithm with Hyper Mutation and Elitist strategies)。其中, 超突变策略用来增强电路元件值的不断优化能力, 精英策略则有助于引导超突变朝着更富有成效的解空间发展从而得到更优解^[4]。实验表明, 该算法比简单的遗传算法 GA 具有更高的效率。

2 具有超突变和精英策略的混合遗传算法

2.1 超突变策略

由于模拟电路自动化设计包括电路拓扑设计和元件值设计. 因此, 交叉机制和突变机制都是电路设计的代表方法. 本文的超突变策略就是在交叉和变异操作之后增加一个高频变异程序, 然后不断提高突变率, 这种方法称为超突变. 事实上, 有关 GA 的突变率和突变策略已经有很多研究. 文献[5]采用把突变率从 0.1% 提高到 50.0% 的方法保持了结果的多样性. 文献[6]把超突变引入到克隆选择算法中, 克服了在迭代优化阶段的早熟和停滞现象.

本文使用超突变策略的主要目的是提高电路元件值的不断进化能力, 它只影响电路的元件值. 超突变操作是通过随机选择一些个体(cc-bot 指令集)并随机改变这些个体的元件值来实现的. 为了提高效率, 超突变操作只能选择由精英策略所提供的个体. 用 p_{hm} 表示超突变率, N 代表算法中的种群数量. 则超突变数量的估计值 $E[n(p_{hm})]$ 可用方程(1)表示

$$E[n(p_{hm})] = p_{hm} \cdot N \quad (1)$$

每个超突变操作都紧跟一个进化结果评估操作, 进化结果的估计值 $E[n_e]$ 可用公式(2)定义

$$\begin{aligned} E[n_e] &= E[n(p_c)] + 2E[n(p_m)] + E[n(p_{hm})] \\ &= E[n(p_c)] + 2E[n(p_m)] + p_{hm} \cdot N \end{aligned} \quad (2)$$

式中, $E[n(p_c)]$ 是交叉数量的估计值, $E[n(p_m)]$ 是简单突变的估计值. 从公式(2)可以看出, 进化结果的估计值随超突变率 p_{hm} 的增大而增加.

2.2 精英策略

精英策略可用来自引导随机搜索, 使超突变策略能在较好的解决方案(精英集)中进行搜索. 文献[7]从一对双亲和其后代中使用局部精英策略, 增加了遗传算法的收敛性. 文献[8]使用非支配精英排序遗传算法优化了射频集成电路. 以往的研究表明, 精英策略可以提高进化算法的收敛性.

为了使模拟电路符合一定的设计要求, 合适的拓扑结构是必要的. 因此用更好的拓扑结构来发展电路的元件值将比在种群中发展所有电路更有效. 精英策略正是用来引导超突变在种群中搜索更优的电路拓扑结构.

要执行精英策略, 我们需要建立一个精英集, 拥有更好拓扑结构的模拟电路更容易获得较好的适应度, 因此, 精英集由种群中适应度好, 元件少的个体组成. 为设计算法中的精英策略, 我们提出以下假设, 在随机产生的元件参数值下, 电路具有较好的拓

扑结构将比较差的拓扑结构更容易获得较好的适应度, 精英机制正是基于以上假设. 具有较差拓扑结构的电路相对于较好的拓扑结构也有机会获得较好的适应度, 所以精英集的大小应该认真选择. 一方面, 为了得到足够好的电路拓扑结构, 精英集应该足够大, 另一方面, 精英集越大, 它提供的引导超突变操作的能力就越弱.

用 S_e 代表种群的精英集, P_e 代表电路中属于 S_e 的百分比, 则精英集的大小 $N(S_e)$ 等于 $n \cdot P_e$, n 是人口数量. 所以精英集 S_e 是由 $N(S_e)$ 最靠前的电路组成. 具有较好适应度和较少元件数量的个体有较好的名次. 用 Pop 表示所有个体的种群, $individual_i$ 表示第 i 个个体, $individual_i_rank$ 表示第 i 个个体的排名, 那么选择精英集的算法伪码如图 1 所示.

```

for (i=1; i≤n; i++)
    individual_i_rank = i;
end for;
for (i=1; i≤n; i++)
    if ( fitness(individual_i) > fitness(individual_j) )
        if ( individual_i_rank > individual_j_rank )
            change_rank((individual_i, individual_j);
        end if;
    else if ( fitness(individual_i) = fitness(individual_j) )
        if ( individual_i_rank > individual_j_rank )
            and component_num(individual_i) <
            component_num(individual_j)
            change_rank((individual_i, individual_j);
        end if;
    end if;
end for;
for (i=1; i≤n; i++)
    if ( individual_i_rank < N )
        S_e = S_e ∪ individual_i;
    end if;
end for;

```

图 1 选择精英集的算法伪码

2.3 混合遗传算法 HME-GA

混合遗传算法 HME-GA 的具体过程描述如下:

(1) 初始化: 初始化过程需要随机产生 n 个初始个体并对每个个体的适应度进行评估.

(2) 交叉变异: 交叉操作是从上一代中随机选择两个双亲并产生两个后代, 执行交叉的次数称之为交叉率; 变异操作是随机选择一个双亲并产生一个后代, 执行变异的次数称之为突变率.

(3) 精英集选择: 交叉变异操作之后, 就会选择

精英集,其算法如图 1 所示。

(4) 超突变:从最优解中随机选择一个双亲并产生一个后代,超突变的次数称之为超突变率。

(5) 终止判断:如果算法满足终止条件,进程就结束,否则进入第(2)步继续。

本文用 Lohn 等人提出的电路构造指令集(CCsets)表示模拟电路.用 N 表示种群大小, gen 表示进化代数, P_c 表示交叉率, P_m 表示简单的突变率, P_{hm} 表示超突变率, S_e 表示精英集, P_e 表示属于 S_e 的百分比.则算法伪码截图如图 2 所示.如果 $P_e=0$,超突变和精英策略将不起作用,如果 $P_e=1$,精英策略将不起作用而超突变将在整个种群中起作用。

```
CCsets = initialize_pop(N);
for (i=1; i<=gen; i++)
    intermediate_pop = CCsets ∪ crossover(Cossets, p_c);
    CCsets = select(intermediate_pop, N);
    CCsets = order_by_fitness(CCsets, down);
    S_e = Φ;
    for (i=1; j<=N * p_e; j++)
        if (rand(0,1) <= p_hm)
            ccset = randomly_select(S_e);
            ccset = hyper_mutation(S_e);
            intermediate_pop = intermediate_pop ∪ ccset;
        end if;
    end for;
    CCsets = select(intermediate_pop, N);
end for;
```

图 2 混合遗传算法 HME-GA 的程序伪码截图

3 实验和结果分析

为了验证所提出的混合遗传算法 HME-GA 的性能,本文选用典型的模拟低通滤波器作为实验对象进行自动化设计.要求设计的模拟低通滤波器的技术参数如表 1 所示。

表 1 模拟低通滤波器的技术参数

通带上限频率 f_p (Hz)	阻带截止频率 f_s (Hz)	通带增益 K_p (dB)	阻带增益 K_s (dB)
1000	2000	-3.0	-60.0

实验对简单 GA 和混合算法 HME-GA 同时进行测试.两种算法的种群大小都为 100,交叉率为 0.8,突变率为 0.1. HME-GA 算法中超突变率按 0.1 幅度从 0.1 增加到 1.0,并分别在不使用精英策略(记为 HM-GA)和精英率为 20%(记为 HME_{0.2}-GA)两种情况下分别测试。

实验得到的简单 GA 和 HM-GA 的平均适应度曲线如图 3 和图 4 所示.图 3 的超突变率为 0.1~0.5,图 4 的超突变率为 0.5~1.0。

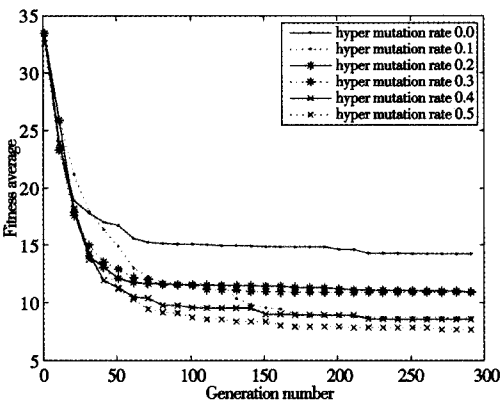


图 3 简单 GA 和 HM-GA 的平均适应度曲线(超突变率为 0.1~0.5)

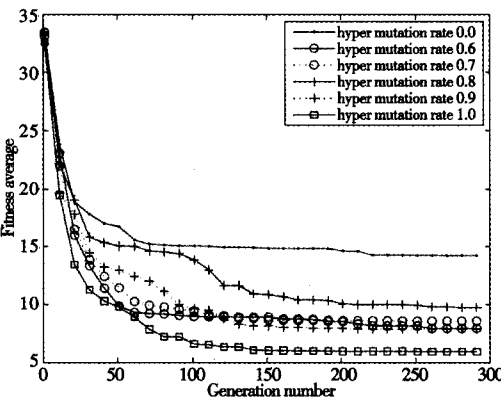


图 4 简单 GA 和 HM-GA 的平均适应度曲线(超突变率为 0.5~1.0)

可以看出,超突变策略使 GA 算法在模拟电路自动设计中的性能优于简单的 GA 算法.最佳适应度出现在所提算法的超突变率为 1.0 时.总的来说,该算法的优越性随超突变率的增大而提高。

图 5 和图 6 是简单 GA 和 HME_{0.2}-GA 的平均适应度曲线.图 5 中,当超突变率从 0.1 变到 0.5, HME_{0.2}-GA 算法性能都比简单 GA 好,最好结果出现在超突变率为 0.4 时.图 6 中,当超突变率从 0.6 变到 1.0,除 0.8 外,算法的适应度依然比简单 GA 理想.由此可见,当使用精英策略时, HME-GA 算法的性能并不总是随着超突变率的增加而提高,在超突变率增加到 0.4 时,性能达到最优.接下来随着超突变率的提高,算法的执行性能显得并不那么活跃.但这也并不意味着精英策略不起作用。

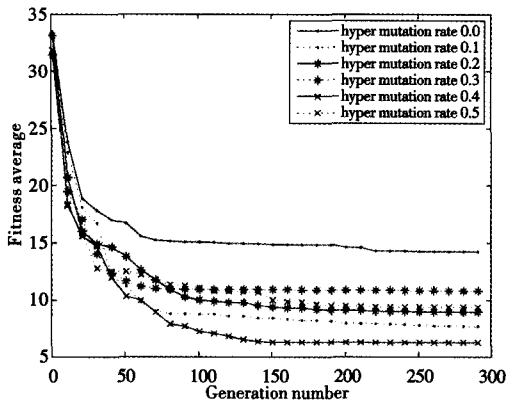


图5 简单 GA 和 HME_{0.2}-GA 的平均适应度曲线(超突变率为 0.1~0.5)

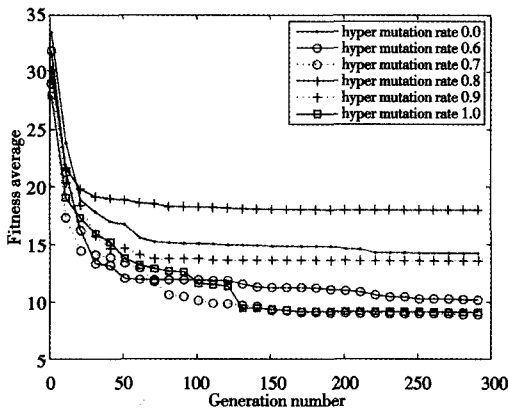


图6 简单 GA 和 HME_{0.2}-GA 的平均适应度曲线(超突变率为 0.6~1.0)

表 2 给出了用简单 GA、HME_{0.2}-GA 和 HM-GA 进化低通滤波器时得到的适应度平均值、最好值和最坏值数据。适应度最好值是在 HME_{0.2}-GA 算法中突变率为 1.0 时产生的。相对来说,三种算法的最好值在多数情况下都是以 HME_{0.2}-GA 算法最突出。只有两种情况 HME_{0.2}-GA 算法不如 HM-GA 算法:突变率为 0.3 和 0.8。这说明 HME_{0.2}-GA 算法比 HM-GA 算法在寻找最佳模拟电路方面更熟练。同时也反映出精英策略在 HME-GA 算法中的有效性。超突变策略使 HME-GA 在模拟电路设计时的收敛速度比简单 GA 更快。总的来说,HME-GA 和 HM-GA 都比简单 GA 优越。

根据表 2 数据经过估算, HM-GA 算法适应度的最好值和最坏值的平均差值是 4.614,而使用精英策略的 HME_{0.2}-GA 算法适应度的最好值和最坏值的平均差值是 10.823,这表明使用精英策略似乎更容易进入本地最佳解区域。如果没有进入局部最优,它也能找到很好的结果,如果进入局部最优,它

将很难得到任何好的结果。然而并不是在所有的突变率下都能表现出这种性能。

表 2 比较简单 GA、超突变精英比率为 20% 的 GA 超突变无精英设置之间的平均值最好和最坏的适应度

算法	超突变率	适应度		
		平均值	最好值	最坏值
简单 GA	0.0	14.244	12.207	15.806
	0.1	7.659	4.328	13.805
	0.2	8.920	7.322	9.737
	0.3	10.786	7.317	17.569
	0.4	6.270	3.771	9.116
	0.5	9.351	4.288	17.572
	0.6	10.147	6.420	17.583
	0.7	8.873	2.783	13.446
	0.8	18.008	8.864	25.527
	0.9	13.599	5.484	27.527
HME _{0.2} -GA	1.0	9.089	2.739	17.568
	0.1	8.694	6.372	11.284
	0.2	10.879	10.359	11.841
	0.3	10.880	5.966	15.856
	0.4	8.550	7.178	10.467
	0.5	7.670	6.096	8.883
	0.6	7.910	7.167	8.914
	0.7	8.581	6.703	9.553
	0.8	9.704	5.242	13.151
	0.9	7.890	5.541	12.501
HM-GA	1.0	5.937	4.135	7.885

另外,在突变率为 0.1~0.4 时, HM-GA 算法适应度的最好值和最坏值的平均差值是 4.893,而使用精英策略的 HME_{0.2}-GA 算法适应度的最好值和最坏值的平均差值是 6.872,而在突变率为 0.5~1.0 时, HM-GA 算法适应度的最好值和最坏值的平均差值是 4.334, HME_{0.2}-GA 算法适应度的最好值和最坏值的平均差值是 14.774。所以超突变率 0.4 就成了一个转折点。在超突变率低于 0.4 时, HME-GA 算法不容易进入局部最优解空间,但当超突变率大于 0.4 时,就比较容易进入局部最优解空间。该结果说明,当使用精英策略,最好选择突变率为 0.4。

图7给出了用HME-GA算法所设计的低通滤波器的幅频特性曲线。

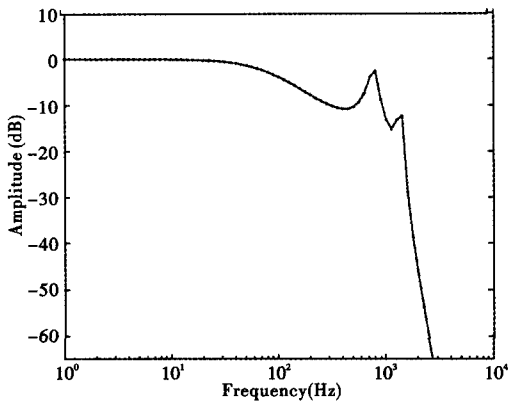


图7 设计出的低通滤波器的幅频特性

4 结束语

本文提出的混合基因遗传算法包含超突变和精英两种策略,超突变用来提高遗传算法GA对模拟电路元件值的进化能力,精英策略用来引导超突变使之工作更有效。实验结果表明,混合基因遗传算法HME-GA和HM-GA在模拟电路自动设计中都比简单GA要好,而前者比后者在获得高质量结果时更有效,这说明精英策略增强了超突变的有效性。因此本文提出的HME-GA算法不仅能进化出适当的电路拓扑结构,而且能兼顾到电路元件值的选取,是一种有效的模拟电路设计方法。

参考文献:

- [1] Giuseppe N, Salvatore R, Eva S. An evolutionary algorithm-based approach to robust analog circuit design using constrained multiobjective optimization [J].

Knowledge-Based Systems, 2008, 21(3):175-183.

- [2] Das A, Vemuri R. An Automated passive analog circuit synthesis framework using genetic algorithms[J]. IEEE Computer Society Annual Symposium on VLSI, 2007:145-152.
- [3] Srinivas M, Patnaik L M. Genetic algorithms: a survey[J]. Computer, 1994,27(6): 17-26.
- [4] Venkatraman, Sangameswar Yen, Gary G. A simple elitist genetic algorithm for constrained optimization [J]. Congress on Evolutionary Computation, 2004: 288-295.
- [5] Cobb H G. An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, Time-Dependent Nonstationary Environments[R]. Naval Research Laboratory Memorandum Report 6760, 1990.
- [6] Z X Jin, G Y Liu, W H Wen. Clonal selection algorithm with hyper mutation and spatial clone extension [J]. Computational Intelligence and Security, 2006 International Conference, 2006(1):402-405.
- [7] Venkatraman, Sangameswar Yen, Gary G. A simple elitist genetic algorithm for constrained optimization [J]. Proceedings of the 2004 Congress on Evolutionary Computation, 2004: 288-295.
- [8] Chu M, Allstot D J. Elitist nondominated sorting genetic algorithm based RF IC optimizer [J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2005, 52(3):535-545.

作者简介:

焦素敏 女,(1965—),副教授。研究方向为电子电路与系统设计。