

算法作曲中软计算方法的研究

朱青

新疆博乐市高级中学华中师大一附中博乐分校

【摘要】算法作曲也被叫做自动化作曲，是尝试利用一种特定形式的过程，作曲创作人员在充分利用计算机开展音乐创作环节时，减低自身接入度。随着社会的不断发展，创作方法的不断创新和完善，软计算方法是当下作曲创作环节中作曲家应用较为普遍的方法。本文主要就算法作曲中人工神经网络算法、基于模糊逻辑的算法以及遗传算法的软计算方法进行分析和研究。

【关键词】算法作曲；软计算方法；分析研究

【DOI】10.12252/j.issn.2096-627X.2021.12.1393

算法作曲是希望通过采用某种形式化的方法，使作曲家在利用计算机进行音乐创作时参与的程度最小化的研究。算法作曲研究存在的主要问题和重点及难点，希望通过对软件计算方法技术的全面讨论阐述当今算法作曲这一领域的研究和进展，为下一步工作打下基础。

一、人工神经网络

人工神经网络是指通过模拟人脑的工作方式对输入的信息进行整合，然后对整合以后的信息进行加工处理，最后输出处理后的结果。这种网络可以从实例中学习，从中获取必要的知识，其具有强大的自组织、自学习的能力；由于神经网络本身具有并行性，因此便于硬件的实现，而且由于网络的鲁棒性强，因此具有很强的噪声抑制能力和容错能力。

人工神经网络在感知和认知方面的研究已取得相当大的成功，也开始被应用在音乐研究领域，并取得非常大的结果，由于人工神经网络能够从一个样板实例集合中学习以避免传统方法所必须的对规则的形式化，因此目前人工神经网络已经应用于算法作曲中，并已经有许多成果。例如Mozer使用递归神经网络技术构造出CONCERT系统，CONCERT系统采用反向传播学习算法进行训练。CONCERT系统通过生成一个音符接一个音符的方式来创作出乐曲旋律，然而由此获得的旋律却缺乏音乐全局的连贯性，这是因为虽然递归神经网络可以经过音乐的表层结构获取的知识为基础，产生出新的旋律，但是其和乐段或调性功能相关的信息却丢失了，这和递归神经网络中学习梯度的逐渐消失有关。对此通过一种被称为长的短时记忆的递归神经网络可以解决这一问题。

人工神经网络能模拟人脑的思维方式进行处理，它为算法作曲提供了一种新的选择。但从目前的应用效果来看并不十分有效，其主要存在的问题如下：

(一) 神经网络算法目前还未建立完整的理论体系，对于神经网络算法的研究还处在起步阶段。目前一般应用的网络模型都是通过节点以及节点间的互联构成的有向拓扑网，节点间的互联强度可以用一个权值矩阵来表示，这个权值矩阵可以采用某种学习策略来确立。然而对于具体神经网络的构建却是一个经验探索的过程，从而对最终的结果不可预期，系统的性能也很难保证。

(二) 对于特定旋律的设定，在对一个人工神经网络进

行训练时，必须事先收集大量类似旋律风格的作品，并对其进行处理以便作为实例输入给人工神经网络，然而其结果是网络的学习周期过长，所设计的神经网络收敛速度也很慢。

(三) 由于神经网络算法实质上是一个近似算法，因此其所搜索到的解为局部最优解。因此在应用的过程中需多次重复运行，从而保证解得最优性。

由此，人工神经网络技术更适合分析音乐作品，对于乐曲创作则不一定适合。相对于知识库方法，人工神经网络所能处理的音乐创作问题相对而言比较简单，因此作曲大师Cope曾经对神经网络是否能成功应用于音乐的创造存有疑问。由此，人工神经网络技术更适合分析音乐作品，对于乐曲创作则不一定适合。相对于知识库方法，人工神经网络所能处理的音乐创作问题相对而言比较简单，因此作曲大师Cope曾经对神经网络是否能成功应用于音乐的创造存有疑问。

二、基于模糊逻辑的算法作曲

模糊逻辑是以模糊集合理论作为基础来研究模糊思维的推理和模糊语言的形式及规律的科学，它是一个多值逻辑，不同于传统的二值逻辑，对二值逻辑中的矛盾律和排中律提出了挑战。

对于音乐可以从多种角度进行解释，而如何表示这方面的不同，模糊逻辑就显示出其优越性，对于音乐中的和弦模糊逻辑可以很好的定义。而利用模糊逻辑多值性的特点可以保证用此方法生成的音乐具有更多的创新性，其更加符合作曲家的创作思路。然而虽然模糊逻辑有这些优点，然而如何在计算机上实现它确实一个比较困难的问题，因此目前基于模糊逻辑的算法作曲系统非常少，现有的系统大多只是对一些简单音乐模式的模仿。这些系统的创建说明模糊逻辑应用于算法作曲是可行的，然而其有效性却还需要等待技术的进一步应用，还需要更多的尝试。

三、遗传算法

遗传算法是采用模拟生物自然进化的方式进行全局最优解查找的算法，从理论上来说基本是一个概率算法，由于其借鉴的是生物进化的方式因此也需相应设计一些对应的处理方法，其中关于个体的表示、种群的设计和个体优劣的确定（适应值函数）都需要针对具体的问题进行设计。在

将遗传算法应用于音乐创作的过程中，需要考虑如何如何表示一个乐曲中各个因素（音高、音程、时值、音长等），同时如何整体评价一首乐曲的好坏以及乐曲旋律问题等等都需要考虑。遗传算法的思想实质是对整个问题空间的一个搜索过程，通过个体适应值的比较和遗传操作决定搜索的方向。即将遗传算法应用于音乐创作实质上就是对乐曲因素组合所有可能进行的搜索，从中找到作曲家认为可以称为音乐的组合。下面就遗传算法中在音乐创作中涉及的一些概念先做个简单介绍。

染色体（个体）的定义：每个染色体中主要包含乐曲中音乐信息的描述，如音高、音程、时值、音长等；还隐含有乐曲的旋律信息等等。

选择过程：适应值函数通过判断种群中每个染色体的适应值来决定每个染色体的后代繁殖机会，如何定义一个音乐片断比另一个音乐片断好是个比较困难的事情，即适应值函数的选取是个关键的问题；

繁殖操作：繁殖操作包括交叉、变异等，种群的搜索方向通过繁殖操作来实现，从而实现对整个状态空间的搜索，这里需要考虑的是繁殖操作实现的方式（一些参数的设置，如交叉点和变异率及交叉率等）。

对于适应值函数的设计时有时还需要考虑到染色体中不同特征对染色体适应值的影响是不同的，因此需用对不同的特征分别予以不同的权值，然后通过加权求和的方式决定适应值函数的计算模型如何选取。

现在已经有不少作曲系统通过选择出合适的适应值函数来指导染色体（乐曲片断）进行进化，这方面的已经取得不少成果：1994年，McIntyre利用遗传算法生成出具有巴洛克风格的4声部的和声曲目；1995年，Homer和Ayers利用遗传算法进行旋律的和声连接；1998年，PaPadopoulos和Wiggins则利用遗传算法演化爵士音乐，最有趣的是其设计的遗传操作采用的是可变长的染色体；1999年，Phon Amnuaisuk等人则试图依靠拥有大量知识的系统使遗传算法变得清晰而有效。

综上所述，使用遗传算法进行算法作曲的研究工作时，对于适应值函数的确定是整个算法的重点和难点，而针对算法作曲中所采用的适应值函数的不同，可以将基于遗传算法的作曲系统分为以下五类：

（一）交互式系统

交互式系统一般需要作曲家的参与，它实质是用作曲家来进行对染色体的评价，这样就可以避免前面提到的适应值函数的设计和选择问题，然而由于人在这种系统中扮演着适应值函数的角色，因此需要作曲家从始到终对种群中的所有曲目进行艺术评价进而引导乐曲的进化过程，系统把这些评价作为产生下一代乐曲的标准，评价者（作曲家）可以是一个人或者一个团队（最好由一个乐队组成，避免个人的偏见）。这种系统是能够克服普通遗传算法作曲盲目性和随机性的弊端。

交互式系统的代表性系统和研究工作有如下这些：1994年Biles设计了一个称为“GenJam”的交互式即兴演奏系统，通过它可以生成爵士乐曲；2001年Unemi设计了一个称为“SBEAT”的交互式音乐作曲支持系统，该系统初始生成几种单小节音乐片段然后基于作曲家的评估来进行演化，我们可以将这些小节片段连接起来形成一个长的乐段；2003年，Unhera M和Onisawa T也设计了一个交互式音乐作曲支持系统，该系统支持毫无音乐知识技能的人可以用该系统创作出自己喜欢的音乐，而且生成的旋律长度可以达到16个小节。

由于引入人的主观因素，因此交互式遗传算法作曲系统存在以下缺点：

- 主观性：这是由于人的主观性（偏好性）会导致评估结果受作曲家的主观性左右，使得整个进化过程趋向于符合作曲家的个人喜好；

- 费时性：由于每代进化过程中都需要人工对群体中的每个个体均进行人工评价（为保证评价前后的一致性，评价者需要隔段时间休息一会），因此耗时较多，整个进化速度非常慢；

- 非有效性：只有当用户听到所有可能潜在的解才能作出具有普遍性的评估结果，然而由于遗传算法是一个概率算法，因此并不能保证最优解每次都可以被找到。

这些缺点一定方面限制了交互式系统的发展和应用，为了克服这些缺点，一些设计者采用了一些减少人工参与程度的方法来避免这些，如采用简单的模式匹配的思想进行染色体的判定，这样不需用人工的参与最终的乐曲也符合设计的要求或具备最初设定的乐曲风格。总之交互式系统主要的解决方向是尽量减少人工的参与程度，实现系统的自动演化进化。

（二）基于知识库的系统

基于知识库的系统其中关于适应值函数的确定是通过对于音乐知识的表达来确定的，这需要建立大量的音乐编码知识以及知识规则的建立，从而保证在每次进化和交叉操作之后，可以对其后代个体直接进行评价。对于具有已知的音乐模式和清晰的结构或规则进行模仿时，用一个知识库系统则是一种最自然的选择，该系统的优点是可以对其选择行为作出合理的解释，而且其具有清晰的推理过程，然而这种系统也存在着以下这些不足：

- 将乐曲知识从乐曲库中提取出来并组成乐曲知识库是一项很艰巨的工作，不仅需要大量的时间，还需要合适的方法来处理如何表示这些知识，尤其是象音乐这类主观性较强的领域；

- 知识库系统运行时将按照指定的规则进行，其运行的效果依赖于所设计的专家系统的能力，为此我们必须寻求一种便捷的知识表示方法，然而系统最终如何运行我们仍然无法确定，因此有时会产生一些意想不到的结果。对此可以通过添加规则或前提到知识库中的方法解决，然而这有可能需

要将所有可能的规则或前提都加进知识库中，这时由于知识库过于复杂导致根本无法运用。

（三）基于实例的系统

基于实例的系统通过采取将一部分实例保存在一个样本实例子系统中来记忆用户对所生成乐曲的特殊要求，这样系统就能够在既保证作曲系统的学习效率又能克服交互式系统中存在的进化太慢的问题以及生成的曲子过于主观性的缺陷等问题。样本实例子系统通常采用人工神经网络进行训练从而确保生成的乐曲具有指定的旋律或风格。

这种系统的主要缺点是：

- 和认知活动相比音乐作曲是一个要复杂得多、也高级得多的智能过程。即使经过人工神经网络训练后所输出的音符序列能符合人们的需求，也会有其自身的局限性，在整体旋律和局部细节方面都会存在许多不足，这是因为无法明确乐曲的风格的表现以及整体的评价。
- 在节拍的表现上若采用人工神经网络的方法有时根本无法对其进行有效的处理。

（四）自发式系统

一般进化算法中，适应值函数一开始就是确定的，这样可以保证整个种群随着时间的推移向着整个种群的适应值加强的方向进化。然而自发式系统中的适应值函数是可变的（即和种群一起进化）。对于自发式系统而言，适应值函数的确定完全由系统自己定义，这样的系统避免了传统进化算法中后期种群进化缓慢的缺点，而且由于适应值函数和种群一起进化，这两部分一起协调共同完成整个乐曲的进化过程。它具有很多优点，如适应值函数与种群同时进化可以减少初始种群欺骗适应值函数的概率，避免一开始的种群就具有较高的适应值；由于共同进化可以保证任何时候种群都具有多样性。

这种系统实际上是对大自然的最直接描述，正如自然界里的物种进化一样，单个种群内实现不断的优胜劣汰，而多个种群之间则形成共同的进化，都保证每个种群中的最优个体不断地繁殖出更好的后代，最终促进整个自然界生物种群的演化，物竞天择这一自然现象正好可以解释自发式系统的运行原理。

该系统存在的主要问题是设计最初的适应值函数并对其进行进化，这点目前还没有很好的方法，还需要进一步的研究才能将其应用到乐曲的创作中。

（五）综合性系统

由于单个适应值函数会出现初始欺骗问题以及演化后期进化速度变慢等，因此有的遗传算法作曲系统中采用多个适应值函数的方法来解决这些问题，这样系统就可以最终生成符合人们欣赏习惯的乐曲，我们称这样的作曲系统为综合性系统。有的综合性系统通过事先定义一系列的规则对要进化的乐曲加以限制，从而减少乐曲的搜索空间、提高整个搜索的效率。还有一类综合性系统是将前面提到的四种系统组合

而成的，这类系统吸取了四种系统的各自特长，通过各个子系统之间的合作与竞争，使最终产生的乐曲更能满足人们的需要。

虽然遗传算法作曲系统取得很大的成果，并广泛应用于作曲系统。然而遗传算法作曲系统的效果在很大程度上依赖于系统对于音乐知识的表达方式和程度，系统对作曲的规则表达得越充分，则其生成的作品也会越能符合作曲家的需求。然而这些系统生成的乐曲从传统的美学角度来看还不够完美，这主要是因为遗传算法应用于算法作曲时还受到以下几个方面的限制：

- 遗传算法的编码有时显得不规范对于一些信息的表达不够准确，从而导致算法运行时对音乐的运行方向无法精确控制，这也是由于音乐所包含的信息过于复杂无法用简单的编码来进行表示。
- 遗传算法本身是一种随机的、启发式的搜索方法，因此它不能保证每次运行都可以找到最优解，所以即使所求解的问题存在一个理想解也很难保证在一次进化过程中找到理想解，因此有时需要多次重复运行，或采用随机数的方式使得系统能从不同的初始点开始搜寻最优解。

尽管遗传算法是依据设定的音乐知识进行乐曲创作的，但是由于动机创作并不完全拘泥于规则，因此若完全按照设定的规则进行创作所得到的结果相反会出现曲调显得单调呆板。

目前设计的许多系统在推理结构的设计上还存在着许多不足，和作曲家的创作思维相比还差得很远，这些系统有时创作出的曲目中会存在无意义乐曲片段、噪声片段等。

遗传算法作曲系统的设计应该根据系统自身的适用范围以及所要生成曲目的自身特点选择相应的进化系统，这样才能保证最终的乐曲符合作曲家的要求。

遗传算法的研究目前理论基础基本成熟，关键是应用过程中存在一些难点，将其应用于算法作曲中时，乐曲旋律的生成通过整个种群的进化来实现。虽然作曲家可以确定遗传算法中的适应值函数等，然而这些参数并不能确定乐曲创作过程中动机创作，结构的组织这些东西的控制，最终结果就是生成的乐曲过分杂乱无章或者过分单调。因此如何在算法演化中嵌入这些思想，还需要进一步的研究。

尽管利用遗传算法创作时存在上述的各种不足之处，但是它为我们进行算法作曲的研究提供了有益的尝试，而且实践也表明其完全可以胜任算法作曲的工作，只是需要在以后的研究过程中考虑如何将一些情感之类的信息嵌入到种群的演化过程中去。

参考文献

- [1] 冯寅, 周昌乐. 算法作曲的研究进展[J]. 软件学报, 2006, 17(2): 7.
- [2] 李璐璐. 基于算法的频谱作曲策略研究[J]. 音乐时空, 2015(22): 1.