

设计“解的编码”，不同的编码方案可以有不同的求解策略。

【视频 9.7 怎样用遗传算法求解具体的应用问题(III)—策略选择的多样性】

算法研究很重要的就是要有想法，要有发散性的思维，思维不能僵化；要能从平凡中看出不平凡的东西。遗传算法的重要步骤设计就是“交叉”策略的设计，怎样交叉，有多少种交叉策略，如何针对具体问题选择交叉策略，本段视频通过展现不同的“交叉”策略，试图展现算法设计中策略选择的多样性，……。

【视频 9.8 怎样用遗传算法求解具体的应用问题(IV)—策略选择的随机性】

算法研究要针对算法的本质，即前述的遗传算法为什么可以求解 NPC 问题，本质是“随机”。随机，一要考虑在什么环节什么步骤引入随机，二要考虑怎样的概率模型。本段视频着重给大家展现了在什么环节什么步骤引入随机。具体概率模型，则未做细致介绍，因为那是《概率论和数理统计》课程的学习任务。

【视频 9.9 怎样用遗传算法求解具体的应用问题(V)—其他方面】

对遗传算法的设计，除前述步骤设计外，还包括其他步骤的设计。限于时间，其它步骤的设计不能一一讲解，本段视频对其仅做一概要性的介绍，未做详细探究，目的是告诉读者遗传算法还有许多要解决的问题，可以通过自我学习和研究来探索。

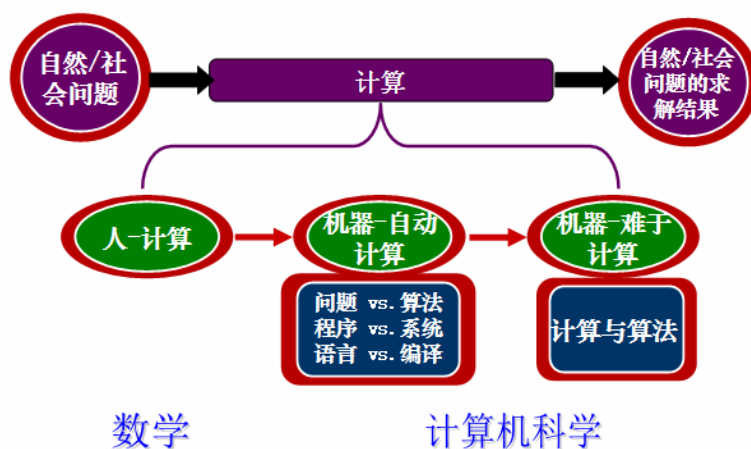
【视频 9.10 怎样研究算法暨遗传算法的总结】

本段视频对遗传算法做了一个总结，进而对怎样研究算法做了总结。试图提升大家的认识高度。

2、学习要点指南

2.1 要点一：理解难计算问题的“计算”与“算法”

本讲内容是贯穿计算思维非常重要的一讲，如下图(首次出现在第 1 讲的导读文档中)。机器自动计算问题已由第二讲至第八讲解决，而机器-难于计算问题应由本讲来解决。本讲内容是算法思维三部分内容之一。第八讲给大家展现了对社会/自然问题，可设计算法进行求解，如 PageRank 算法、排序算法等，算法是求解具体问题的方法和步骤。本讲中，需要使大家理解什么是计算——为求得精确解不得不完成的计算量，什么是算法——为完成求解而寻求降低计算量的算法。本讲的遗传算法，目的是要使大家理解什么是计算，什么是算法，怎样研究算法；当然，遗传算法本身的学习也是很有意思的。



2.2 要点二：下列概念对理解算法、理解怎样研究算法非常关键

为理解计算和算法，下图所示概念的辨析很重要。

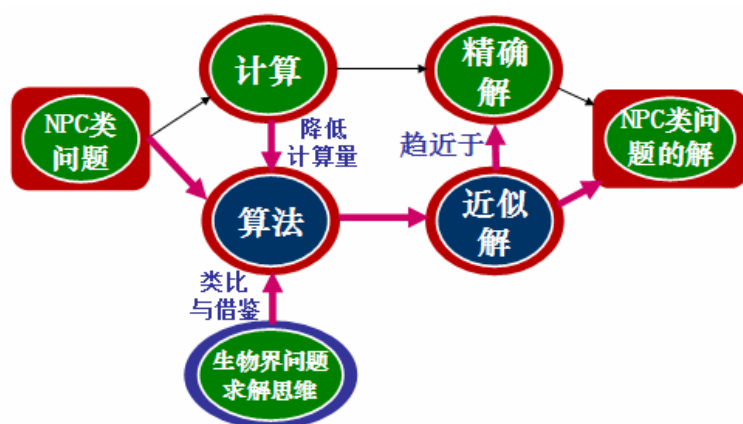


问题可被分为可求解问题、难求解问题和不可计算求解问题。若要区分这些概念，就需要区分算法复杂性和计算复杂性，所谓计算复杂性是问题本身所固有的复杂性，而算法复杂性则是求解问题的某一个算法的复杂性。

P 类问题是可求解问题，NP 类问题是可验证问题，所有解都可验证的问题是 NP-完全问题 (NPC)，难求解问题中的一类是 NP 难求解问题，也是 NPC 问题。P 类问题包含在 NP 类问题中。详细的解释参见课程视频。

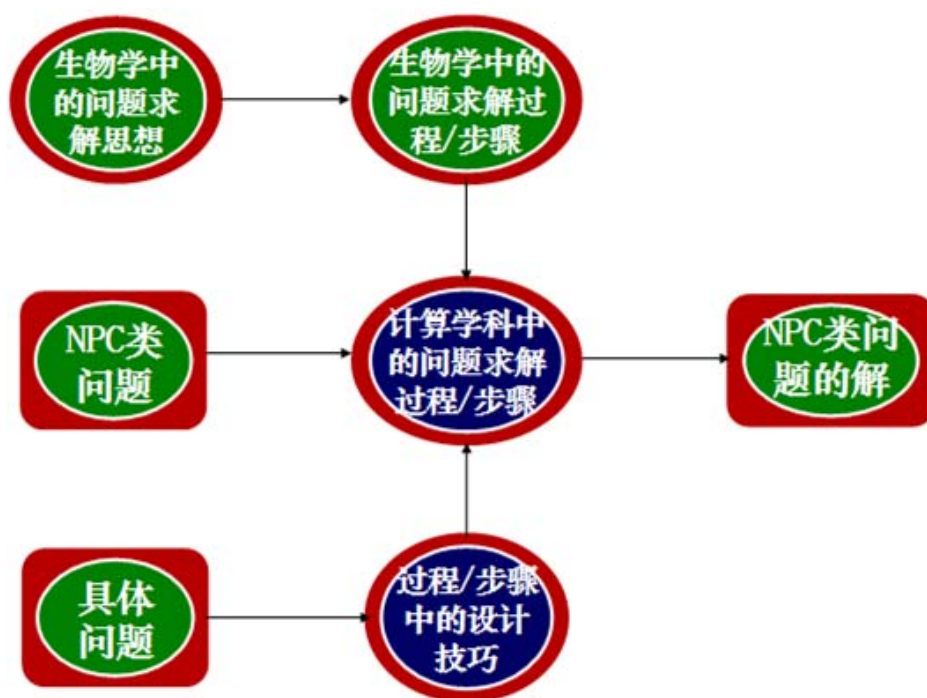
2.3 要点三：理解难求解问题的基本思想

如下图所示，要理解难求解问题的基本思想。可以通过类比自然界的生物求解思维，来设计降低计算量的算法。而降低计算量的算法求得的就是近似解，而算法的目标是如何在降低计算量的情况下，努力使近似解更接近精确解。这种思想可通过遗传算法的学习来获得体验。



2.4 要点四：如何通过类比进行算法的研究

若要通过类比的方法进行算法的研究，则需要注意将自然界的生物求解思维过程化，通过理解其求解的步骤和过程，理解生物求解思维。然后依据所理解的生物求解的步骤和过程，设计计算学科算法的求解步骤和过程。完成了此方面内容以后，则结合具体问题进行算法设计时，即是结合具体问题研究和设计算法中的每个步骤，并体现具体问题的设计技巧。不同问题有不同的设计技巧，虽然其框架是一致的。这一点，需要通过遗传算法的学习深入理解之。



2.4 要点五：怎样理解本讲的遗传算法

本讲遗传算法的学习，主要是让大家理解下图中被放在衬底中的四个子图。这四个子图反映了本讲希望传达给学生的主要思想。每一个子图体现了遗传算法学习的一个方面：左上子图以过程的形式展现了“遗传”与“优胜劣汰”，及其相关的概念；右上子图以数据模拟的形式展现了计算学科遗传算法求解的基本过程，反映了由社会/自然生活到计算与计算系统的宽度学习内容。

进一步，左下子图反映了遗传算法为什么可以求解 NPC 问题，即算法的本质研究；右下子图反映了结合具体问题如何设计遗传算法，即算法的应用。这两部分都属于深度学习的内容。真正掌握一个算法，就是能应用其解决具体的问题。

因此，对遗传算法的理解可以分几个层面来理解

(1)宽度学习层面

一般，在这一层面大家都能理解。即理解左上子图和右上子图。理解什么是遗传算法，遗传算法的基本求解框架。

(2)深度学习层面之本质探究

在这一层面希望大家都能理解，耐心的观看视频大家也确实能够理解，即有关可求解难求解问题的一些概念，以及 NPC 问题求解的基本思想，遗传算法为什么可以求解 NPC 问题，即左下子图的理解。

(3)深度学习层面之应用探究

在这一层面大家理解起来可能有些难，主要是对待求解问题背景的理解，短时间内难以消化，但大家多看几遍视频，也能理解。



3、常见问题

附：宽度教学与深度教学相结合—计算思维教学方法浅谈

下面所附的是从教学角度谈如何学习计算思维的一篇文章，其中是以“遗传算法”的教学为例。阅读一下该文，能够对“遗传算法”的学习，以及“由遗传算法到一般性算法”的学习都有帮助，尤其是文中提到，通过学习要对计算有一个较为深入的认识并建立较为科学的研究习惯，很有启示。

宽度教学与深度教学相结合—计算思维教学方法浅谈

战德臣，聂兰顺，徐晓飞

哈尔滨工业大学计算机科学与技术学院

from 《工业与信息化教育》2013.06

摘要：大学计算机应该讲授计算思维已经形成普遍的共识，人们也从各种角度各个层面来探索大学计算机的计算思维教学方法。本文在多年的教学实践基础上，以遗传与遗传算法的教学为例，提出计算思维教学要采取宽度教学与深度教学相结合的方法，能够由自然/社会生活中的计算讲解到计算技术与计算系统中的计算，更进一步由计算技术与计算系统中浅层次的计算讲解到深层次的计算，不仅使学生了解和接受“计算”，而且能够使学生对计算有一个较为深入的认识并建立较为科学的研究习惯。

关键词：计算思维；教育；教学方法；遗传算法

1、大学计算机的计算思维教育

关于计算思维的讨论已经持续了很长时间，计算思维是运用计算科学的基础概念进行问题求解、系统设计、以及人类行为理解等涵盖计算科学之广度的一系列思维活动^[1]，计算思维是除理论思维、实验思维外的第三大思维^[2]，大学计算机教学应克服“狭义工具论”^[2]，计算思维能力的培养是大学计算机教学的核心任务^[3]，大学计算机应是计算思维通识教育课程^[4]等观点已经被普遍接受。

人们在讨论计算思维概念的同时，也在从各种角度各个层面来探索大学计算机的计算思维教学内容与教学方法。笔者提出了“计算之树”来概括大学计算机的计算思维教育空间^[5]，同时依据计算之树编著了《大学计算机-计算思维导论》^[6]教材，试图能为大学计算机课程面向计算思维的教学改革提供内容上和方法上的支持和借鉴。

如何进行计算思维的教学？计算思维的教学目的是使学生能够像计算机科学家一样思考，像计算机科学家一样进行问题求解。对大学目前的计算机基础教学的一种批评声音是“计算机基础

课程深度不够”，为此很多教师积极探索增加课程的理论和数学深度，例如在教学中引入计算理论、数理逻辑、集合论与图论、计算算法等内容，然而受限于学时数、学生接受程度的限制，引入哪些内容以及如何引入、引入多少等则困惑着许多教师：引入少，则深度不够；引入多，则学时不够。另一方面的探索是在教学中引入跨学科元素，例如北大李晓明教授引领的以社会网络为对象的网络计算课程，作为一门选修课程是非常好的，对培养学生的科学素养非常有益，但它与大学计算机以计算技术和计算系统为中心的主题是否一致呢，也困惑着许多教师。

如何解决上述问题？笔者在多年的教学实践基础上提出：计算思维教学要采取宽度教学与深度教学相结合的方法，能够由自然/社会生活中的计算**讲解到**计算技术与计算系统中的计算，更进一步由计算技术与计算系统中浅层次的计算**讲解到**深层次的计算，不仅使学生了解和接受“计算”，而且能够使学生对计算有一个较为深入的认识并建立起较为科学的研究习惯和研究兴趣。下文以遗传与遗传算法的教学为例来简要探讨这一宽度与深度相结合的计算思维教学方法。

2、宽度与深度相结合的计算思维教学方法

通识教育课程应着重在培养学生的思维，思维的特性决定了它能给人以启迪，给人创造想象的空间，并可能潜移默化地被融入到未来的创新活动中。如何使学生更好地理解思维—宽度教学与深度教学相结合的教学是必要的，如下图1所示。

针对计算思维通识教育，**宽度教学**是指能够从社会/自然生活中的广义计算讲起，进一步讲解到计算技术与计算系统中的计算，要能够将社会/自然生活中的概念/原理映射到计算技术与计算系统中的概念/原理，通过这种映射，一方面使学生易于理解和接受计算学科的概念和原理(通过对比社会/自然生活中的概念和原理来理解)，另一方面可使学生理解计算机科学家是如何借鉴现实中的概念进行抽象的，进而还可强化学生通过类比跨学科事物进行抽象的能力培养。

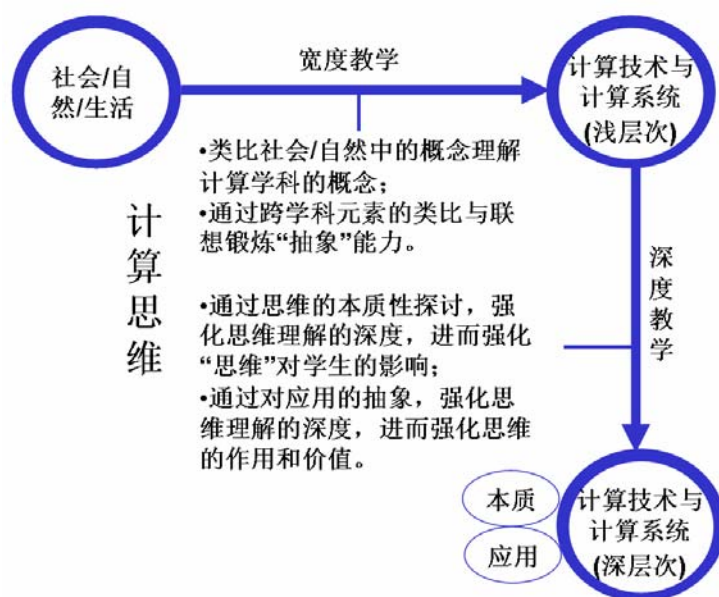


图 1. 宽度与深度相结合的计算思维教学方法示意

如果仅仅停留在宽度教学的层次，那也只是使学生基本理解了思维相关的概念，并不能使学生对相应的思维有深入的理解和体验，即所谓的计算思维的浅层次理解。为强化计算思维的深层次理解，则还需要深度教学。

所谓**深度教学**，是指在学生理解了思维相关的基本概念后，能够进一步探讨思维的一些深层次内容，如思维的本质和思维的应用：通过思维的本质性探讨，强化思维理解的深度，进而强化思维对学生的影响；通过对应用的抽象，强化思维理解的深度，进而强化思维的作用和价值。

计算思维的教学应能从宽度教学延展到深度教学，把计算思维讲清楚、讲透彻，对学生的进一步学习和未来的工作奠定一个坚实的基础。

3、计算思维的宽度教学示例--由自然/社会的广义计算到计算学科的浅层次计算

下面以遗传算法的讲授技巧介绍如何进行宽度教学。遗传算法代表了计算学科中一类典型的算法思维，在很多方面有重要应用，理解遗传算法对于学生深入理解算法思维，进而理解问题求解的基本思维模式有重要意义。那么如何讲解遗传算法呢？

遗传算法是计算机科学家借鉴了生物学领域的遗传和自然界的优胜劣汰规律而提出的一类问题求解算法。因此，我们可从生物学中的遗传与自然界的优胜劣汰规律讲起，如下图 2(a)示意，在这一部分讲解中有两个重点，一是完整的遗传与进化过程：由种群到个体，由个体到个体的染色体，进一步到个体交配、个体染色体中的基因交叉与突变，产生新个体，形成新的种群，再到新种群对环境的适应能力和优胜劣汰，产生新一代种群，不断繁衍、不断淘汰与进化。二是这一过程中的重要概念及其含义，如种群、适应能力、交叉与突变等。

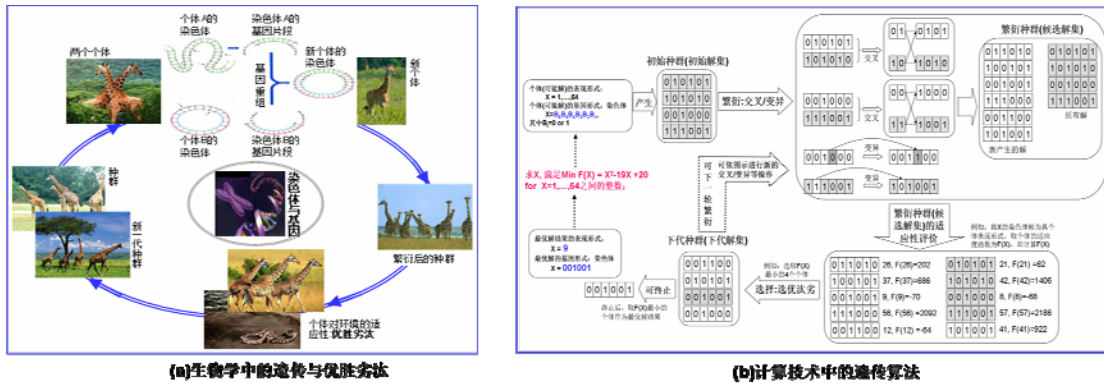


图 2. 遗传算法的宽度教学示意：由生物学的遗传讲解到计算技术中的遗传算法

在生物学的遗传原理介绍清楚后，进一步则以一个规模较小的问题求解示例来展现计算学科如何应用遗传原理进行问题求解的过程，如图 2(b)示意，在这一部分讲解中有三个重点，一是完整的遗传算法求解过程：由问题理解到问题的可能解的表达与可能解的解空间，初始解集，交叉与变异产生新的可能解，形成新的可能解集，如何判断解的好坏，可能解的选择，形成新的解集等，不断迭代直到满意解出现；二是这一过程中的概念及其与生物遗传中的概念的映射，例如“种

群”对应“解集”，个体“染色体”对应到编码形式的“可能解”，“基因”对应可能解的一个或若干编码位，“适应度”对应解的好坏的判定函数等；三是在前述基础上给出算法设计的框架和步骤描述，详细内容可参阅笔者编著的教材^[6]。

4、计算思维的深度教学示例--由计算学科的浅层次计算到深层次计算

仅仅讲解前节介绍的内容，笔者认为并不能使学生对遗传算法有一个较为深入的认识或者说并不能使学生对遗传算法产生深刻的印象，因此还需要由前述的宽度教学延展到本节的深度教学。

我们可从两个方面使学生对遗传算法产生印象和兴趣。一是探讨遗传算法的本质：即遗传算法能够求解什么类别的问题，为什么它能够进行求解？二是探讨针对实际的应用问题，如何应用遗传算法进行求解。

4.1 从本质性揭示角度加深对计算思维的理解

怎么揭示遗传算法的本质呢？如下图 3(a)示意，我们可按照如下的逻辑来引导学生思考：

(1)问题或计算的复杂性。我们可以通过一个具体问题来讨论问题规模与求解算法的计算量之间的关系，使学生认识到多项式时间的求解算法和非多项式时间的求解算法的效率差异，进而引出问题复杂性的概念。

(2)用前述的问题复杂性概念，对问题进行分类：P 类问题、NP 类问题和 NP-Complete 类问题：一个问题如果存在多项式时间计算复杂性的求解算法，则其是可解的 P 类问题；而一个问题如果没有多项式时间计算复杂性的求解算法，则被称为是难解性问题。在多项式时间内可以由一个算法验证一个解是否正确的计算问题，是 NP 类问题；而如果这个问题的所有可能解都可以在多项式时间内进行正确与否的验证，则为 NP-Complete 问题。

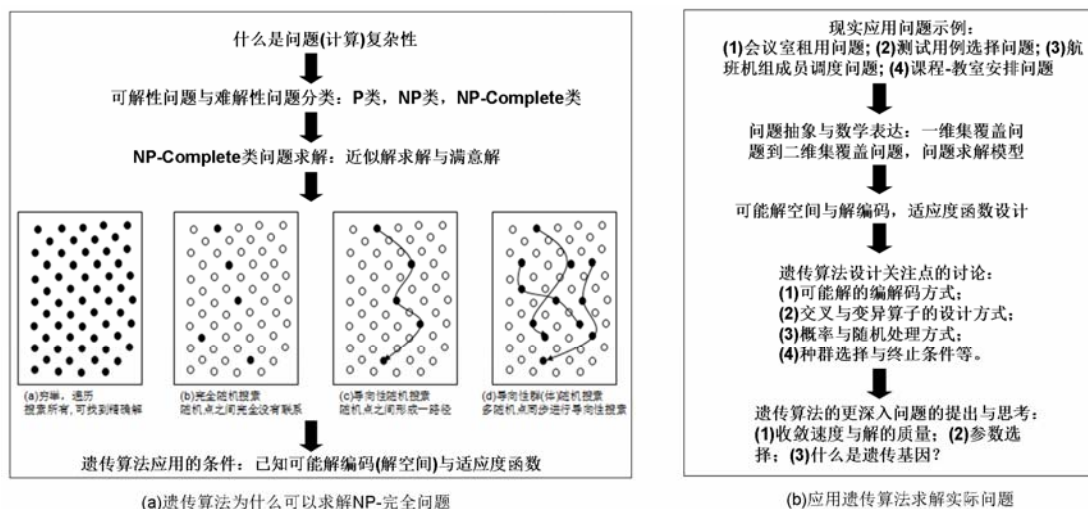


图 3. 遗传算法的深度教学示意：从思维的本质性和思维的应用性角度深入理解遗传算法

(3)进一步,对 NP-Complete 问题如何进行求解呢?如果要求出精确解,则需对可能解空间要完整地遍历,即验证其每一个可能解是否是精确解,而前面介绍,这种遍历在多项式时间内是完不成的。那么是否一定要求精确解呢?进而引出求近似解与满意解的思路。

(4)再进一步,如何求满意的近似解呢?既然遍历做不到,那么可以随机的选择一些可能解,在这些可能解中求最优解,引出随机搜索的思想,引导学生思考怎样随机选择可能解集合使之能覆盖最优解呢?更进一步,可以采取导向性的随机搜索的办法,即在选择可能解时依据先前的解的特点,选择那些偏向满意解的可能解,即在选择过程中有一定的导向性。那么再进一步思考,导向就一定能导向到最优解吗,万一最初的解选择有问题怎么办,此时是否可以采取多条导向路径进行选择,此即是遗传算法的本质---从具有一定规模的种群开始,同时产生下一个种群,即相当于多条导向搜索同时进行。

(5)有了以上的认识,我们就可探讨什么情况下适合用遗传算法进行求解,或者说应用遗传算法进行求解的前提是什么呢?--已知可能解的编码(可能解的解空间)以及已知适应度函数(可以判定一个解是偏离最优解还是靠近最优解)。

前述思维本质性的揭示有助于学生深入理解遗传算法,有助于深入理解算法思维。通过一环扣一环的问题引导,可使学生对遗传算法产生兴趣、产生进一步思考的冲动,例如如何导向呢?(提示:给定一组解,通过交叉、变异与基于适应度的选择产生新的一组解)。

4.2 从应用性角度加深对计算思维的理解

如果能够使算法和实际问题求解关联起来,即通过社会/自然中的问题及其求解,如何应用遗传算法进行求解,则将使学生对遗传算法的理解更为深入。怎么通过应用来理解遗传算法呢?如下图 3(b)示意,我们可按照如下的逻辑来引导学生思考:

(1)首先给出几个需要求解的现实应用问题。例如会议室的租用问题、软件测试用例的选择问题、航班机组成员调度问题。通过讲解使学生理解需要求解的问题是什么?有什么约束?目标是什么?

(2)进一步,引导学生对这些问题进行抽象并用数学进行表达。例如前面三个问题是典型的一维的集覆盖问题,即从数学角度来看这三个问题是同一个问题,只是应用的语义不同,表达出来如下所示:

$$\min \quad z(x) = \sum_{j=1}^n c_j x_j \dots\dots\dots(1)$$

$$\text{s.t.} \quad \sum_{j=1}^n a_{ij} x_j \geq 1, \quad i = 1, 2, \dots, m \dots\dots\dots(2)$$

$$x_j \in \{0, 1\}, \quad j = 1, 2, \dots, n \dots\dots\dots(3)$$

通过如此的训练可使学生掌握利用数学来理解应用问题的技巧和习惯,并能注意问题分析过程中的严谨性以及不同问题的求解特点。

例如会议室的租用问题和课程-教室的安排问题有什么差异?(提示:前者是一维集覆盖问题,只要租用的会议室覆盖了所有的讲座即可,并未要求给出哪一个讲座安排在哪一个会议室;后者则是二维集覆盖问题,即不仅要能够使选择出的教室覆盖所有的课程,同时还要给出哪一门课程

安排在哪一个教室), 表达这两个问题的数学公式是有差别的, 详见教材^[6]。

(3)再进一步, 分析可能解空间与解的编码、适应度函数。会议室租用问题的解是 $x = \langle x_1, x_2, \dots, x_n \rangle$, 其中 $x_j=0$ 表示第 j 个会议室未被选择, $x_j=1$ 表示第 j 个会议室被选择, 只要使被选择的 x_j 能覆盖所有的讲座并满足相应的约束条件即可, 即会议室租用问题的可能解是一组 n 位的二进制编码。适应度函数可以使用(1)式, 即给定两个可能解, 在满足约束的前提下, 代入(1)式计算得到值, 值越小的越偏向于最优解。有了可能解的编码和适应度函数, 便可采用遗传算法进行求解: 随机选择初始种群, 通过交叉与变异产生新个体、新种群, 通过计算适应度值并进行选择得到进化的新一代种群, 迭代直到求出满意解。

(4)更进一步, 可引导学生探讨遗传算法设计的关注点, 如可能解的编码方式、交叉与变异算子的设计方式、概率化随机处理方式, 以及种群选择与终止条件等。为使讨论具有多样性, 此时可使用课程-教室问题(二维集覆盖问题)来进行探讨: 编码设计的多样性、交叉算子设计的多样性、随机化处理的多样性, 训练学生的发散式思维, 使同学理解: 问题的特点不同、设计出的遗传算法也是不同的, 使同学能够体验到不同的遗传算法, 在哪些方面是不同的, 加深同学对遗传算法的理解。

如果上述过程被成功地实现, 则可进一步通过作业或课后思考的形式引导学生进一步思考: 遗传算法的收敛速度和解的质量有什么关系呢? 遗传算法的各项参数对算法收敛速度和解的质量有什么影响? 染色体编码中什么是“遗传基因”, 怎样发现种群的遗传基因, 怎样使遗传基因被遗传被继承? 更进一步, 生物遗传基因和遗传算法中的遗传基因有什么异同呢? 怎样发掘生物遗传基因呢?

如此, 建立在学生理解基础上的算法思维以及算法的深入探讨将会极大提高学生的学习兴趣, 如果学生对这些问题能够进行系统化的思考, 则有助其养成科学的研究习惯。

结论

通过前面的介绍, 计算思维通识教育宜采用**宽度教学**与**深度教学**相结合的方法。宽度教学虽能使学生理解和接受一种思维, 然这种接受可能是在浅层次上, 即学生只是“知其然但不知其所以然”, 对如何应用或者说如何研究仍旧是不清晰的。如果进一步扩展到深度教学, 则将激发学生的好奇心, 提升其主动研究的动力, 进而提升其对计算思维、对计算学科的兴趣, 有了兴趣, 则一切问题都将变得更为容易!

计算思维是每个大学生都应具备的一种思维能力, 如何培养学生的计算思维是摆在每一位从事大学计算机教育的教师面前的一个重要课题, 不仅要解决大学计算思维的教育空间问题(笔者提出的计算之树^[5]可能是一种解决方案), 而且要解决大学计算思维的教学方法(本文提出的宽度教学与深度教学相结合的教学方法可能是一种解决方案)。因此, 笔者认为, 怎样加强计算机基础课程的深度, 既不能简单地增加理论和数学深度, 也不能简单的抛离计算技术与计算系统这一主题, 而应将他们结合起来。另一方面, 笔者认为, 学生的“数学”可能不是问题, 学生对“应用”的理解也可能不是问题, 而如何建立由“应用”到“数学”, 由跨学科问题到计算学科问题这种转换

可能是个问题。面对这些问题，采取宽度教学与深度教学相结合的方式，强化打通数学与应用之间的连接瓶颈，打通跨学科问题与计算学科问题的映射瓶颈将是大学计算教育值得推广的方法。

参考文献：

- [1] Jeannette M. Wing. Computational Thinking[J]. Communications of ACM, 2006, 49(3): 33-35 .
- [2] 九校联盟（C9）计算机基础教学发展战略联合声明.中国大学教学，2010.
- [3] 陈国良等，计算思维与大学计算机基础教育，中国大学教学, 2011（1）.
- [4] 战德臣，聂兰顺，徐晓飞.《大学计算机》--所有大学生都应学习的一门计算思维通识教育课程，中国大学教学, 2011（4）.
- [5] 战德臣，聂兰顺. 计算思维与大学计算机课程改革的基本思路，中国大学教学, 2013（2）.
- [6] 战德臣，聂兰顺.《大学计算机--计算思维导论》，电子工业出版社, 2013.