
隐蔽化、多维化、开放化

——论当今信息学竞赛中数学建模的灵活性

杭州外国语学校 石润婷

【关键字】 数学建模 隐蔽化 多维化 开放化

【摘要】

数学建模是信息学奥林匹克竞赛的有机组成部分。当今信息学竞赛越来越追求数学建模的灵活性。其表现大致有模型的隐蔽化、多维化和开放化三条。本文通过对这“三化”的含义及表现的探讨，研究相应的解题策略。

一、引子

数学建模作为信息学奥林匹克竞赛的一个不可或缺的组成部分，自该竞赛诞生以来，一直在进化，在完善，在发展。当今信息学竞赛越来越追求数学建模的灵活性。也正是这种灵活性，使数学建模的魅力毕现，从而赋予信息学竞赛以无限的生命力和广阔的发展前景。

通过对一系列新兴竞赛题的考察和研究，我发现当今信息学竞赛中数学建模的灵活性可以概括为模型的隐蔽化、多维化和开放化这三条。下面，让我们通过对这“三化”的含义及其表现的探讨，以获得相应的解题策略。

二、主体

(一) 隐蔽化

1、定义

“隐蔽”的本意是“借旁的事物来遮掩”。而具体落实到信息学竞赛中，“旁的事物”和被“遮掩”的对象便有了特定的指代。显然，从我们的论题便可一目了然：被“遮掩”的对象即数学模型；而“旁的事物”在这里指的是扑朔迷离的现实情景。

这样，信息学竞赛中数学模型隐蔽化的定义便显而易见了，即借扑朔迷离的现实情景来遮掩数学模型。

2、表现

隐蔽化在信息学竞赛中的一大表现就是“老模型，新面孔”也就是说，沿用我们都熟悉的模型，而制造出全新的场景来容纳此模型，从而给原本赤裸裸的模型披上了新装，将它“掩护”起来。因而相同的模型，在不同竞赛题中的表现往往变幻莫测，如《最佳旅行路线》(NOI97)和《新型导弹》两题，就是典型的例子。题目请参阅附录一、二。

这两题前者描述的是一个由“林荫道”、“旅游街”组成的街道网格，而后者描述却的是一个“导弹爆炸”问题。因此，单从表面看，两者应该是风马牛不相及的然而，两种截然不同的表面现象背后，恰恰蕴藏着相同的原型——求一维数列中“最大”(元素和最大)连续子序列的问题，即已知数列

$$A = \{a_1, a_2, a_3, \dots, a_m\};$$

$$\text{求 } \overline{Maxsum} = \max \left\{ \sum_{k=i}^j a_k \right\}, 1 \leq i \leq j \leq N;$$

这两题有一个共同点，即题目本身并没有直截了当地将数学模型展现出来，而是通过对复杂的实际情景的具体描绘，要求选手自己从实际情景中归纳、抽象出数学模型。因而，它们充分地体现了信息学竞赛中数学模型的隐蔽化特点。

3、策略——“拨开迷雾”法

虽然扑朔迷离的现实情景往往给观者以一种雾里看花的朦胧感，但是，只要我们能以敏锐的目光，透过这纷繁复杂的表面现象去观察并很好地把握模型的实质，问题往往就能迎刃而解。

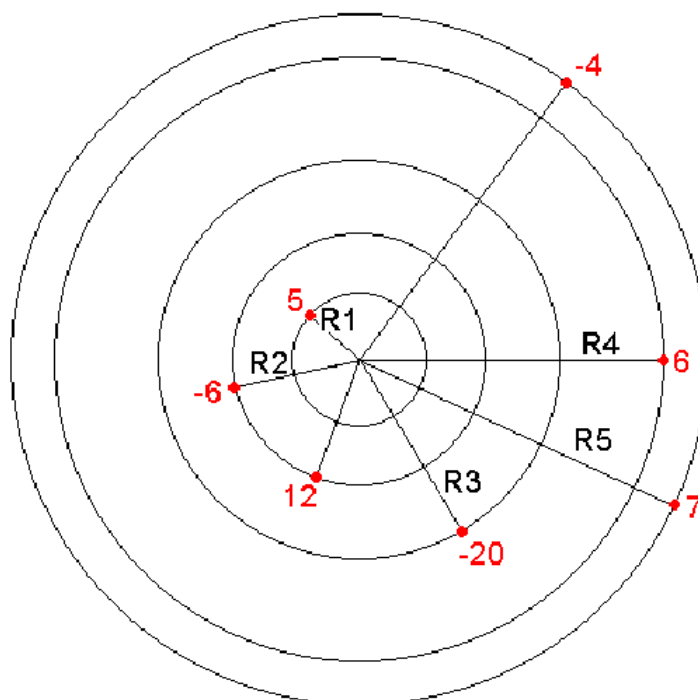
下面，让我们通过对《新型导弹》一题的分析，具体看一看我们拨开迷雾、挖掘问题本质的思维历程。我们首先不能被所谓的“屏蔽半径”和“攻击半径”、“居民点”和“碉堡”这些表象所迷惑。我们应该注意到以下事实：

① 可以将居民点的分值修改为它的相反数，则爆炸总利益 = 所有居民点的分值和 + 所有碉堡的分值和；于是我们就能将居民点和碉堡统一起来看待。

② 一旦确定了“屏蔽半径”和“攻击半径”后，某一个建筑物是否被炸毁只与它与圆心（爆炸点）之间的距离有关，而与其在平面内（战场上）的具体位置无关。因此，我们在读入数据的时候，就可以只储存各点与圆心之间的距离，而摒弃具体的 x, y 坐标。

③ 可以将这些点按照离圆心由近到远的顺序排序，同时将与圆心等距的点合并成一个代表点，其分值为这些点的总分值。

如图：



5	6	-20	6	3
---	---	-----	---	---

于是，我们的任务就变成了求上图所示的一维表的最大子序列问题，即模型的实质。（附程序《新型导弹》missile.pas）

总之，我们在拿到题目时，不能急于动手编程，而首先应该冷静地去思考分析问题，并从与之关联的各种信息中，正确地“过滤”掉迷惑人的无用的部分，“提炼”出关键的部分，从而很好地把握这“新面孔”背后隐藏的“老模型”。俗话说，磨刀不误砍柴功。只有经过了周密深入的思考，我们才有可能透过现象洞察到问题的本质。而此时再着手编程，就能胸有成竹，事半功倍。

（二）多维化

模型的多维化是信息学竞赛中数学建模灵活性的另一个体现。多维化大致可分为“实”的和“虚”的两类。

1、“实”的多维化

(1) 含义

所谓“实”的多维化，顾名思义，就是指实实在在的，“看得见，摸得着”的多维化。这是它的内涵。而它的外延就是模型由线向面扩展，由面向空间扩展。

我们来看如下两个模型，从中来领略一下“由线向面向空间”的具体含义：

模型 1: 已知平面内的若干个点的，求覆盖这些点的最小圆。

模型 2: 已知空间内的若干个点, 求覆盖这些点的最小球。

我们可以看出, 模型 2 是在模型 1 的基础上进行多维化而得到的产物。

事实上, 这类多维化模型已屡次在竞赛题和练习题中出现。上述模型 2 只是其中小小的一例。

(2) 策略——“降维”法

这种“实”的多维化趋势增添了我们所要考虑的空间因素, 进而加大了模型求解的难度。

解这类题目时, 我们往往需要追寻出题者的思路, 也来一个循序渐进, 先从低维的问题出发, 在找到低维问题的合适解法后, 再加以引申和推广, 从而得到相应多维问题的解法。这实际上就是一种“降维”的思想, 其优点在于它先化繁为简, 有利于我们找出分析思考的着手点。而在我们把握了低维模型的解法后再由简返难, 就如囊中探物般地获得多维问题的解法。

“降维”思想在不同的题目中有着不同的运用方式。

i. 类比法

让我们以 NOI97《卫星覆盖》一题为例 (请参阅附录三)。此题在分析过程中的第一大障碍也许要数空间难度了。但是此时, 如果摆在面前的不是一个三维情景, 而是简单的二维情景, 我们也许就会信心百倍了。那么, 何不先尝试着去求解此二维模型, 或许还能从中获得一点启示。事实上, 该题的二维模型, 即求若干个可能有重叠的共面矩形所覆盖的总面积的问题, 在第五届 IOI 中《求图形面积》一题 (请参阅附录四) 已经出现过, 因此为我们所熟悉。其算法描述如下:

第一步, 预处理。删去所有被包含矩形。

第二步, 平面离散化。

第三步, 统计所有被覆盖离散平面格的总面积。

然后, 通过类比, 我们顺利地推出相应的三维模型的求解方法:

第一步, 预处理。删去所有被包含立方体。

第二步, 空间离散化。

第三步, 统计所有被“覆盖”离散立方格的总体积。

该三维模型求解方案与二维模型求解方案大同小异, 只是考虑到效率问题, 在统计的时候还需要做一些优化工作。

就这样, 通过运用“降维”的思想, 我们在相应二维模型求解方案的启示下, 圆满地完成了三维模型的求解。(附程序《卫星覆盖》cover.pas)

通过上面的例子可以看到, 多维模型从低维模型中诞生, 因而难免“遗传”了低维模型的某些特征, 使得我们有可能通过类比法直接套用低维模型的求解模式, 来为较为复杂的多维问题“接生”。

ii. 落到实处的“降维”法

从上述类比法中, 我们看到, “降维”思想的整个运用过程其实是在我们的脑海中完成的, 而程序的实际操作对象始终都是多维模型。因此, “降维”并没有真正在我们的程序中体现, 即没有落到实处。

虽然有不少“多维化”竞赛题可以运用类比法直接套用现成的低维求解模式, 但是我们也应该看到, 这一招并不是时时处处都能够左右逢源的。我们来看

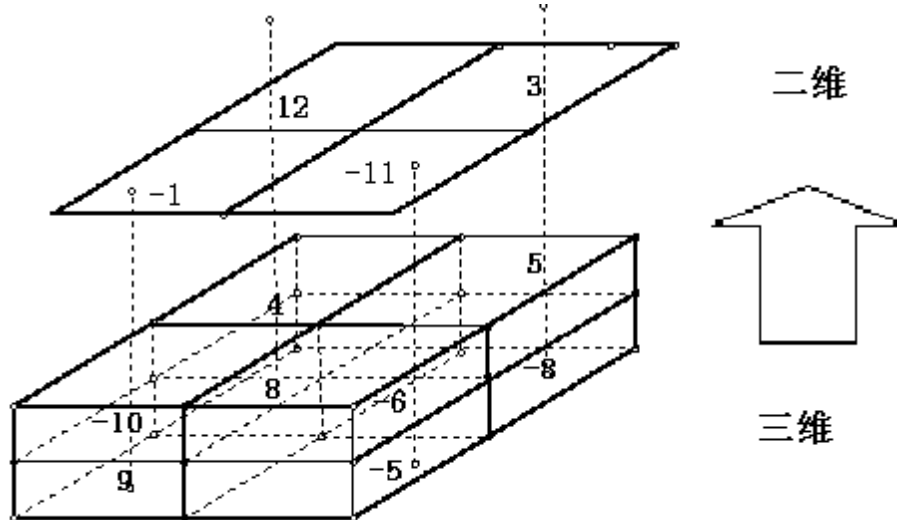
《宇宙探险》一题（请参阅附录五）。

该题的一维模型是上面已经提到过的求一维数列中最大（元素和最大）连续子序列的问题。但是，该一维模型的求解模式（一重循环法）并无法直接套用到三维空间来。

对于此题，比较容易想到的就是穷举法，但是其效率奇低。因为为了确定一个子长方体，共需要6个变量，即长方体的左下前角坐标（ a_1, b_1, c_1 ），以及长方体的右上后角坐标（ a_2, b_2, c_2 ）。因而需要六重循环。而即使不考虑这六重循环内为统计该长方体分值和而带来的新的循环，算法的时间复杂度也已经达到了 N^6 （ $N \leq 50$ ）规模。这显然是个庞大的数字。因此，这样的算法是不可取的。

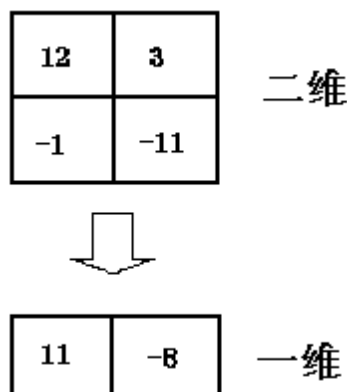
这里，我们可以采用一种落到实处的“降维”方法来解该题。

具体的方法是：假设我们当前所搜索的长方体是自上往下第 a_1 -- a_2 层的，为了找出夹在 a_1 -- a_2 层之间分值最高的长方体，即进一步确定 b_1, b_2, c_1, c_2 ，我们可以先将该 a_1 -- a_2 层从三维压缩到二维的，然后再加以讨论。如图：



现在我们已经将问题转化为了一个二维模型——如何在一张二维表内找出一个分值最大的矩形。我们很容易发现，这个“最大”矩形还原到三维，就是夹在 a_1 -- a_2 层间的“最大”长方体。

为了求解上述二维模型，我们可以用同样的方法先确定 b_1, b_2 ，然后将 b_1 -- b_2 层从二维压缩到一维，然后，就可以运用已知的一维模型求解模式进



行最终求解。如图。

这样，我们不仅在最后求解一维模型时，利用了现成的优秀算法而节省了一重循环，从而将算法的时间复杂度降低到 N^5 ，更重要的是，我们可以利用“降维”过程减少重复的求和工作。当我们已经完成对 $a_1 \rightarrow a_2$ 层的搜索，而着手进行对 $a_1 \rightarrow a_{2+1}$ 层压缩时，我们可以把累加工作建立在已有的 $a_1 \rightarrow a_2$ 层压缩表上，即只需把 a_{2+1} 层对应地往 $a_1 \rightarrow a_2$ 层压缩表上累加。这样，我们就有效地利用了已经获得的信息而避免了大量的重复计算；二维到一维的压缩过程类似。与穷举法相比，该算法的统计工作是分散地进行的，即分布在各层循环之间进行，而非嵌套在最后一重循环内部。这样，就有效地控制了算法复杂度的急剧增长趋势。（附程序《宇宙探险》`explore.pas`）

在此题的求解过程中，我们将“降维”过程物理地落实到了程序中去。这就是我们所说的“落到实处”的降维。通过与穷举法的对比，可以看到，这种“落到实处”的降维，不仅为思考分析问题提供了清晰的思路，而且往往还能收到一些意想不到的奇妙效果。

由于多维化题自身的多样性，“降维”思想的运用方式还有很多。关键是要针对每题的独特性灵活运用。由于篇幅关系，在此只介绍较为常见的两种运用方式，希望能起到抛砖引玉的作用。

2、“虚”的多维化

(1) 定义

在上述“实”的多维化中，“维”沿用了它的本义，即构成“构成空间的因素”。而在“虚”的多维化中，“维”的含义可以引申为广义的“构成数学模型的因素”。

由此，“虚”的多维化的含义就是“构成数学模型的因素”的增加。

(2) 表现和策略

区别于“实”的多维化，“虚”的多维化的是以增加阶段参数或状态参数等形式体现在我们的数学模型和程序当中的。这既是“虚”的多维化的表现形式，也是相应的解题策略。

其中最为典型的的就是动态规划模型的多维化。（图论中的标号法可以看成是动态规划的优化，因此，这里把标号法也归入动态规划。）

我们知道，经典的动态规划是由阶段、状态、决策三重循环构成的。但是，如今的动态规划，常常不再局限于这陈旧的三重循环模式，而是借助于阶段、状态变量的增加，将模型建筑到了多重循环之上。具有代表性的试题有第七届 IOI《商店购物》、NOI97《积木游戏》以及 IOI98 中国队组队赛《罗杰游戏》等题。题目请参阅附录六、七、八。

为了说明问题，我们以《罗杰游戏》一题为例来具体看一看“虚”的多维化的表现形式。在看该题之前，我们首先来看一个熟悉的模型，以区别比较：

有一个 $A \times B$ 的二维表格，表中每一格都被赋予一个整数值 -1 或 $0 \sim 255$ 。现在，我们要从表中 (x_1, y_1) 格出发，走到 (x_2, y_2) ，途中不能经过 -1 格。把沿途经过的所有格中的数累加起来，称为该路线的费用。求所有可能路线的最小费用。

众所周知，该模型是一个经典的动态规划模型。其动态规划方程可以表示为

$$\begin{aligned}
 \text{Minimize } f(x) &= \begin{cases} 0, & (\text{当 } x_1, x_2 \in \{1, 2, 3, 4, 5, 6\} \text{ 时}) \\ \infty, & (\text{否则}) \end{cases} \\
 \text{Minimize } f(x) &= \begin{cases} \sum_{i=1}^6 \sum_{j=1}^6 |x_{ij} - x_{i+1,j}|, & (\text{当 } x_{ij} \in \{1, 2, 3, 4, 5, 6\} \text{ 时}) \\ \infty, & (\text{否则}) \end{cases} \\
 \text{Minimize } f(x) &= \sum_{i=1}^6 \sum_{j=1}^6 |x_{ij} - x_{i+1,j}| + \sum_{i=1}^6 \sum_{j=1}^6 |x_{ij} - x_{i,j+1}| \\
 \text{其中 } x_{ij} &= \begin{cases} 1, & (\text{当 } x_{ij} \in \{1, 2, 3, 4, 5, 6\} \text{ 时}) \\ 0, & (\text{否则}) \end{cases}
 \end{aligned}$$

现在，我们来看一下《罗杰游戏》一题是怎样在上述模型的基础上进行“多维化”的。

我们看到，同样是从 (x_1, y_1) 格出发，走到 (x_2, y_2) ，与第一个模型相比，《罗》一题由于“罗杰”自身六个面上的数字位置不同而引进了成千上万种不同状态。因此，为了体现这些状态间的区别，新的动态规划方程势必要引入新的参量即新的“维”：

$$\begin{aligned}
 \text{Minimize } f(x) &= \begin{cases} 0, & (\text{当 } x_1, x_2 \in \{1, 2, 3, 4, 5, 6\} \text{ 时}) \\ \infty, & (\text{否则}) \end{cases} \\
 \text{Minimize } f(x) &= \begin{cases} \sum_{i=1}^6 \sum_{j=1}^6 |x_{ij} - x_{i+1,j}|, & (\text{当 } x_{ij} \in \{1, 2, 3, 4, 5, 6\} \text{ 时}) \\ \infty, & (\text{否则}) \end{cases} \\
 \text{Minimize } f(x) &= \sum_{i=1}^6 \sum_{j=1}^6 |x_{ij} - x_{i+1,j}| + \sum_{i=1}^6 \sum_{j=1}^6 |x_{ij} - x_{i,j+1}| \\
 \text{其中 } x_{ij} &= \begin{cases} 1, & (\text{当 } x_{ij} \in \{1, 2, 3, 4, 5, 6\} \text{ 时}) \\ 0, & (\text{否则}) \end{cases}
 \end{aligned}$$

(附程序《罗杰游戏》roger.pas)

3、小结

无论是“实”的多维化，还是“虚”的多维化，都着重考查了选手的思维素质，和知识的引申、迁移能力。因此我们在平时的练习中，特别要注意联想与比较，学会用发展的眼光来看信息学竞赛。不要满足于对某个具体问题的练习，而要在解决该问题后，注意引申和拓宽，去主动地开展多维化，而不是被动地应付多维化。只有这样，才能更上一层楼。

(三) 开放化

1、含义

“开放”的本义是“打破禁令、封锁、束缚”，在信息学竞赛中，我们将它引申为“打破原有的、经典的模型的束缚”。

如果说上文所述的隐蔽化、多维化过程是建立在现有模型的基础上的，那么开放化就是一个彻底的推陈出新，因为它完全打破了信息学竞赛中数学模型的

陈旧模式。

2、表现

竞赛中不断涌现出来的开放化模型，往往给人以一种焕然一新的感觉；同时，也常常使得那些“有名有姓”的算法无用武之地。由于失去了经典模型的借鉴，解题者往往难以找到入手点。这就充分地考验了选手们的创造力。它不但要求我们能够灵活运用已知的模型来解各种题目，更要求我们在遇到超越了经典模型的能力范围的新题时，学会创造性地分析新情况，解决新问题。因此，它是一类颇具挑战性的题目。

比如说 IOI98 中国队组队赛中《电阻网络》一题，就是典型的一例。题目请参阅附录九。

这是一个来源于物理电学的题目。它出现在信息学竞赛中，从头到尾都是崭新的，没有任何的经验模型可以借鉴。如果说该题有什么独特的算法的话，那就是利用简单串并联电路的几个基本物理公式，按部就班地两两合并能够合并的电阻，直至最后只剩下唯一的电阻，即外电路总电阻——这是“顺推”；然后，我们再沿着“顺推”的脚步逐渐回溯，直到将该总电阻再扩展回到原来的电阻网络，并在回溯过程中沿途求得所有电阻上的电流和电压——这是逆推。但是，要实现这“两步曲”，并不是件容易的事。最大的障碍就是没有专门用来描述电路的现成的数据结构可以依赖。因此，此题的关键就在于如何利用有限的程序语言来储存、表示以及操作这张纵横交错的电阻网络，才能既方便快捷又保证不产生歧义。而如何设计出一套漂亮的数据结构模型，也为我们提供了发挥自己的创造与想象能力的广阔天地。而考查选手们运用知识的灵活程度和创造性思维，恐怕也正是该题命题者的意图所在吧。

3、策略——对症下药

由于开放化所引进的模型是千变万化的，没有什么固定的解题模式可寻，因此，只有随机应变，对症下药，才是解决这类题唯一通用的方法。（附程序《电阻网络》resistor.pas）

4、小结

开放化模型在竞赛中层出不穷，是信息学与实践日益结合的必然产物。客观世界的丰富多采决定了我们在通过程序设计以解决实际问题的工作中，所遇到的问题是各种各样的。因此，从没有什么“万能”的模型可以屡试不爽。我们也只有从特定问题的特殊性出发，具体情况具体分析，才能够探求到该问题独一无二的最优解法。

三、总结

模型的隐蔽化、多维化以及开放化，是信息学竞赛中数学建模灵活性的三大体现。我们应该看到，这“三化”并不是彼此独立地存在的，而是水乳交融地贯穿于竞赛题当中的。譬如我们在论述隐蔽化的过程中提到过的《求最佳旅行路线》一题。命题者在隐蔽模型的过程中，实际上将一维的模型用了二维的现象来描述这虽然不同于前文所述的多维化，因为此“维”事实上是非真实的，但是，我们应该承认，这当中同样寄寓着多维的思想。因此，我们应当用联系的而非隔离的目光来看待三者。往往也正是三者的珠联璧合，为我们带来了更新、更妙、更富

有创意的竞赛题。

而作为参赛者，信息学竞赛中数学建模的灵活性无疑对我们的全面素质提出了更高的要求。当然，我们应该意识到，这远不止是竞赛对我们的要求，更是时代对我们青少年一代的殷切期望和热切召唤。参加信息学竞赛，积极主动地培养自己各方面的能力与素质，尤其是数学建模能力和程序设计能力，也正是我们不断追求自我发展与完善，努力为未来做准备的重要实际行动。

【参考书目】

- 1、国际国内青少年信息学（计算机）竞赛试题解析（1994～1995）
- 2、国际国内青少年信息学（计算机）竞赛试题解析（1992～1993）