《智能优化理论与方法》

学生姓名 黄盛

学生学号 20191003504

学生班级 193191

日期: 2022.5.17

目录

[1 实验介绍 2](#_Toc103711931)

[1.1 实验目的 2](#_Toc103711932)

[1.2 仿真数据集 3](#_Toc103711933)

[1.3 算法设计 3](#_Toc103711934)

[1.4 遗传算法控制参数的选择 5](#_Toc103711935)

[1.5 实验小结 5](#_Toc103711936)

# 实验介绍

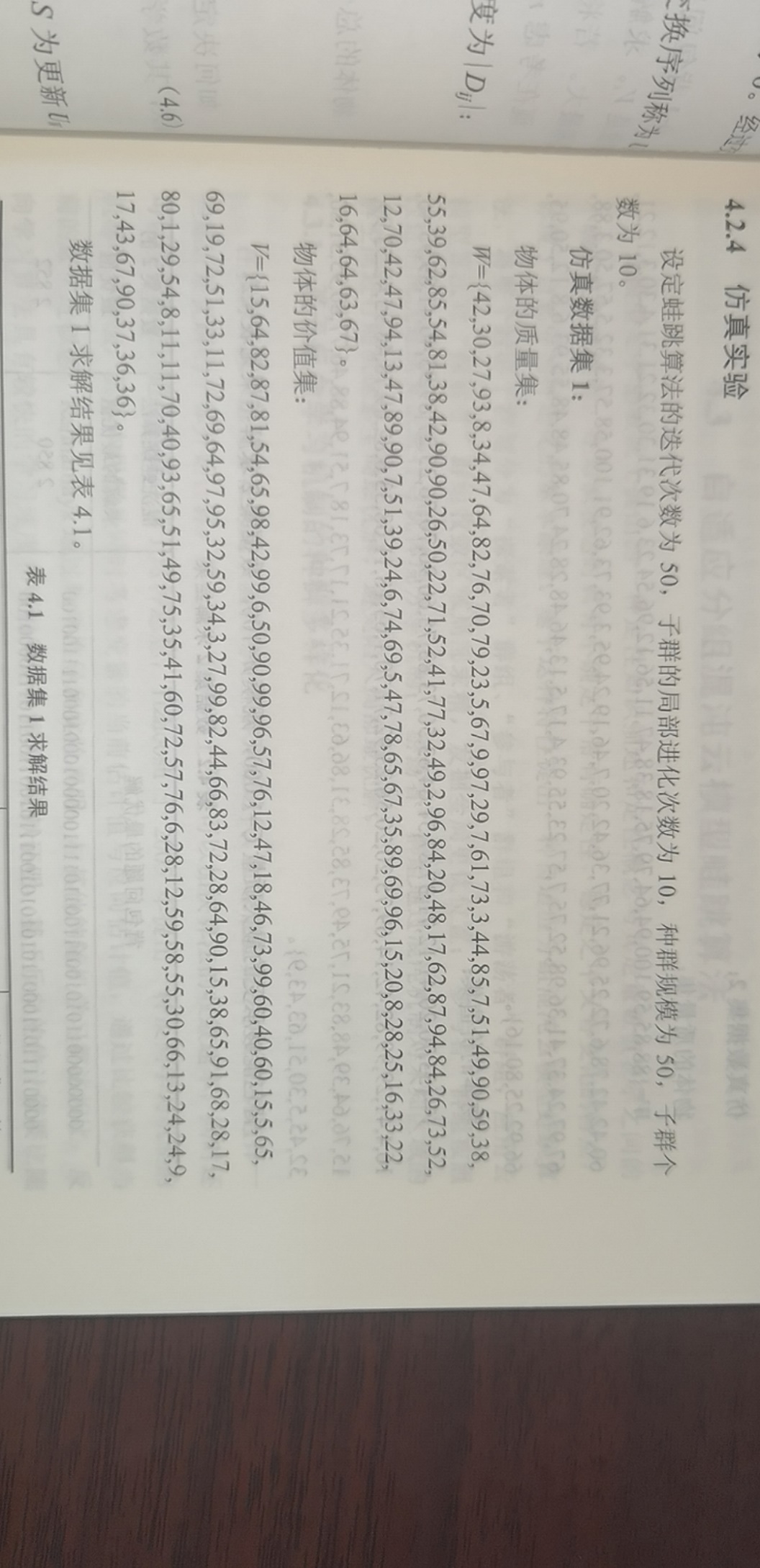
0-1背包问题是易于理解且较为容易的组合优化问题。假设现有若干个物品，它们的质量和价值都是已知的。此外，还有一个有承重质量限制的背包，则0-1背包问题可以简单地描述为：如何把这些物品放入这个有承重质量限制的背包中，在不超出背包最大承重限制的前提下，使得放入背包中的物品总价值最大。

遗传算法（Genetic Algorithm, GA）作为一种经典的智能优化算法，已被广泛应用于组合优化问题中。因此，本次上机内容是使用GA算法来求解0-1背包问题。

## 实验目的

本章实验的主要目的是掌握遗传算法相关基础知识点，了解算法中的选择操作、交叉操作、变异操作等。掌握遗传算法求解0-1背包问题的一般流程。要求使用C++编程语言来实现算法具体步骤。

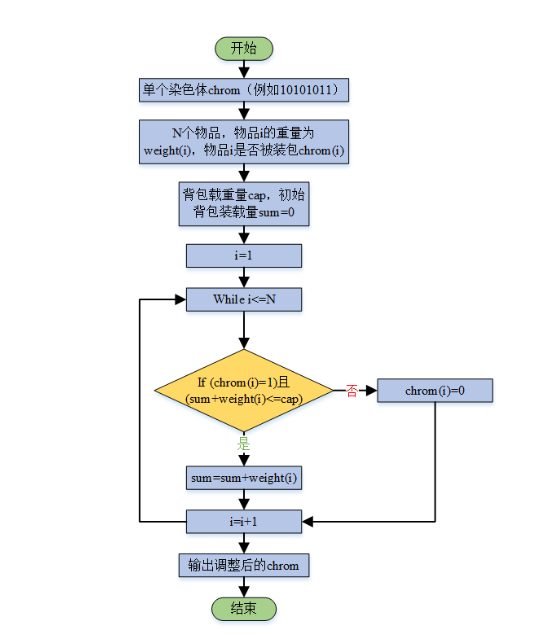
## 仿真数据集



背包能够承受的最大质量C=1000kg.

## 算法设计

1. 编码采用01编码的形式，但如果仅采用简单的随机生成不一定能满足背包载重量的限制。下面举一个简单的例子：有5个物品，每个物品的重量为[3,2,6,1,5]，每个物品的价值为[12,6,20,5,15]，背包最大载重量为10。假设一个随机的染色体为11110，这个染色体表示为第1、2、3、4个物品装入背包，此时背包载重量为12（>10），不满足背包最大载重量约束，这里我们需要对这个染色体进行调整，使其满足载重量约束，具体的调整步骤如下图所示。



按照上述步骤调整后，原始11110染色体调整为11010.

（2）适应度函数适应度函数为物品价值之和，即，适应度值越大的染色体越优，反之越劣质。

（3）选择操作采用轮盘赌策略，个体适应度值越大，被选中的概率越大。

（4）交叉操作假定物品个数为6，产生[1,6]区间的随机整数a1和a2，然后截取这两个位置中间那段染色体进行交换，如a1=2，a2=4。（PS，交叉部分的代码要比这个解释稍微复杂一些，不过没有什么影响）1 | 0 1 0 | 1 00 | 1 0 1 | 0 1交叉为1 | 1 0 1 | 1 00 | 0 1 0 | 0 1

（5）变异操作变异操作就是随机选择染色体上两个位置点进行交换1 0 1 0 1 0变异为1 1 1 0 0 003 |

## 遗传算法控制参数的选择

定义种群规模是500

static const int PS = 100;

交叉率

static double PC = 0.8;

变异率

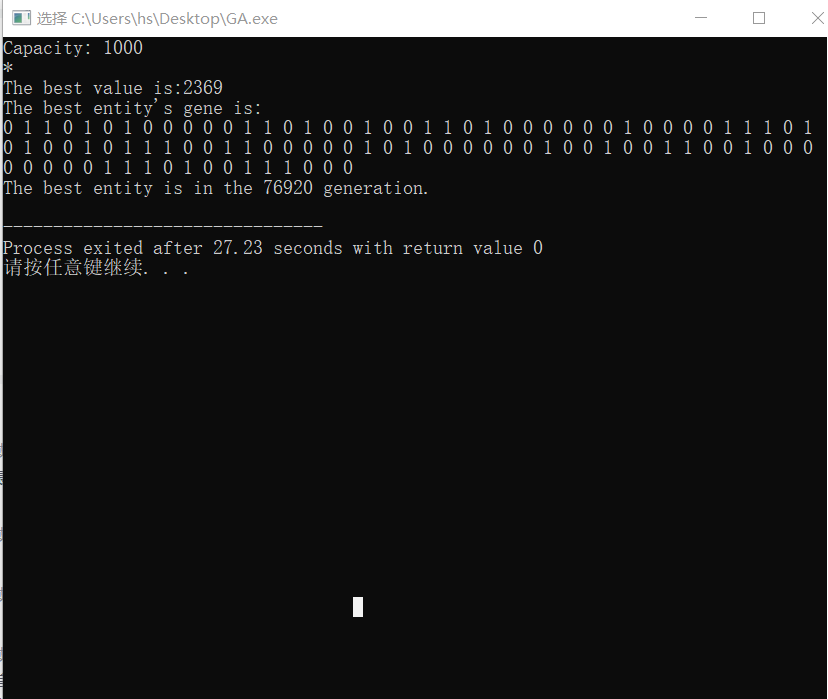
static double PV = 0.1;

遗传最大代数

static int MAX\_GENERATION = 100000;

## 实验小结

实现效果



只是简单地只使用遗传算法，导致算法性能不是很好。代码改进是可以综合使用其他算法，提升算法性能。

若同时求解多个背包，只需在代码基础上稍作修改。可以从个体（内含基因[数组](https://so.csdn.net/so/search?q=%E6%95%B0%E7%BB%84&spm=1001.2101.3001.7020)）初始化下手，也可以简单的写多个个体类。