

多重积分 Fortran 子程序-----VEGAS.F 流程图

说明：这里只是对子程序的算法和如何分层做详细介绍，对打印输出和其他一些东西都做略过。还有这个子程序中涉及的变量很多，而且变量的意义也变化，所以下面涉及到的变量我都会在画流程图前加以适当的说明。

1. 子程序所涉及的变量及其的意义：

有用输入参数：

FXN：被积函数。

ACC：所要求的精度。

NDIM：被积函数的变量的个数。

NCALL：投点的总数。

ITMX：要求的叠代的次数。

NPRN：打印的要求（具体要求略过）。

子程序内部的参数及其意义：

TI (AVGI)：数值积分的结果。

IT：叠代的次数。

X (I, J)：各随机变量的分布，J 表示各个变量。

ERR (SD)：数值积分的误差。

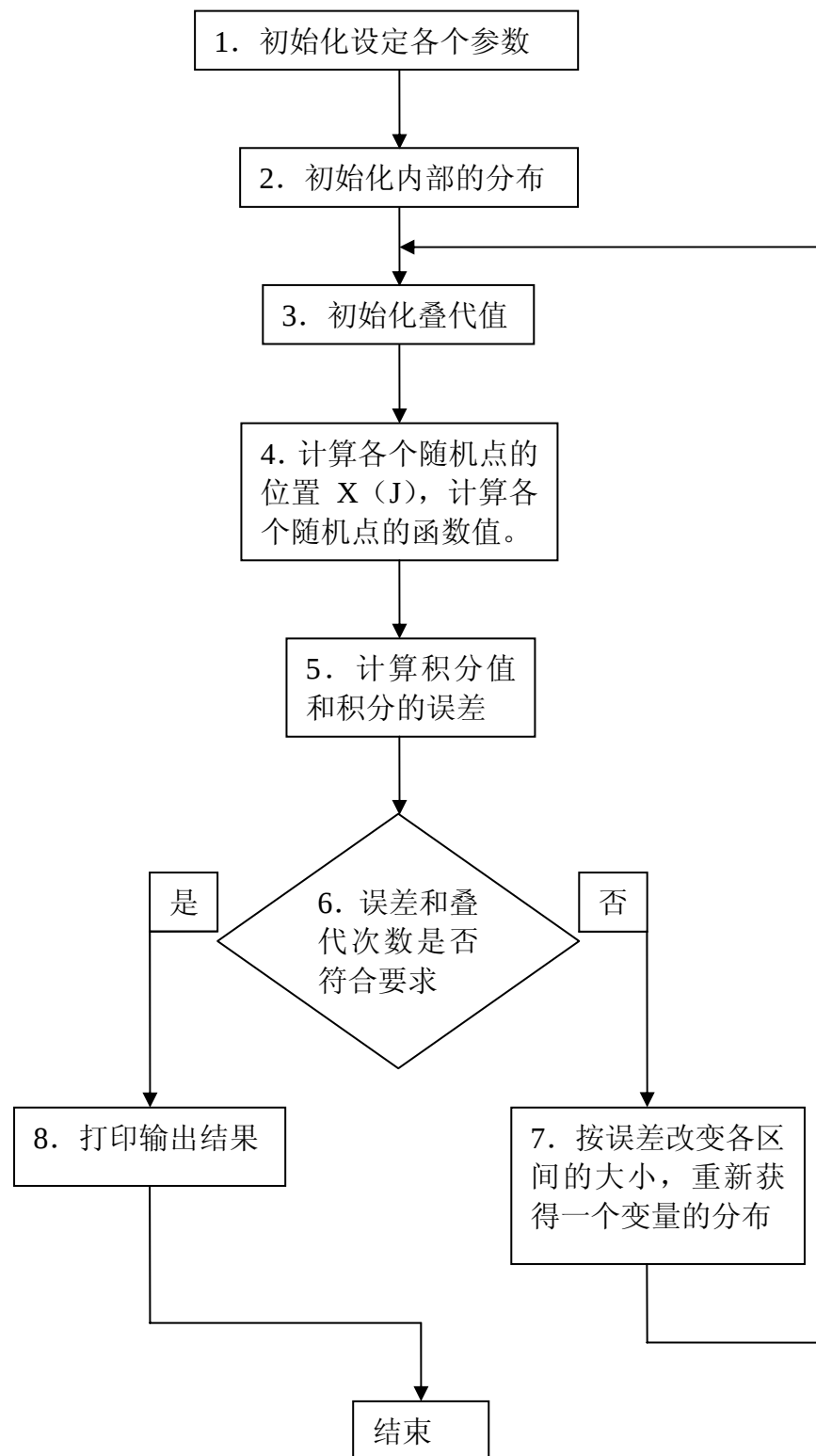
ALPH0：变化区间的参数。

RAND (20)：[0, 1]的随机数列。

X (J)：用于积分的坐标等。

2. 主要的程序模块和各模块的功能

总的流程图：



3. 各模块的流程图

1) 初始化设定各个参数

该部分的主要工作就是给各参数输入初始值，下面简述各个初始化的参数以及各参数的意义或在以下何模块里要用到。

ALPH=1.5; 这是在 7 中用到，作为变换分布的参数。

CALLS=NCALL; 投点的总数。

XND=ND=50; 区间分为 50 份。

NDM=ND-1=49; 为以下处理方便。

IT=0; 叠代初始为零。

SI=0;

SI2=0;

SWGT=0;

SCHI=0;

SCALLS=0; 以上五个变量均是叠代参数，在 5 中用。

2) 初始化内部分布：

变量的意义：

RC: 分层的间隔。

NDIM: 变量的个数。

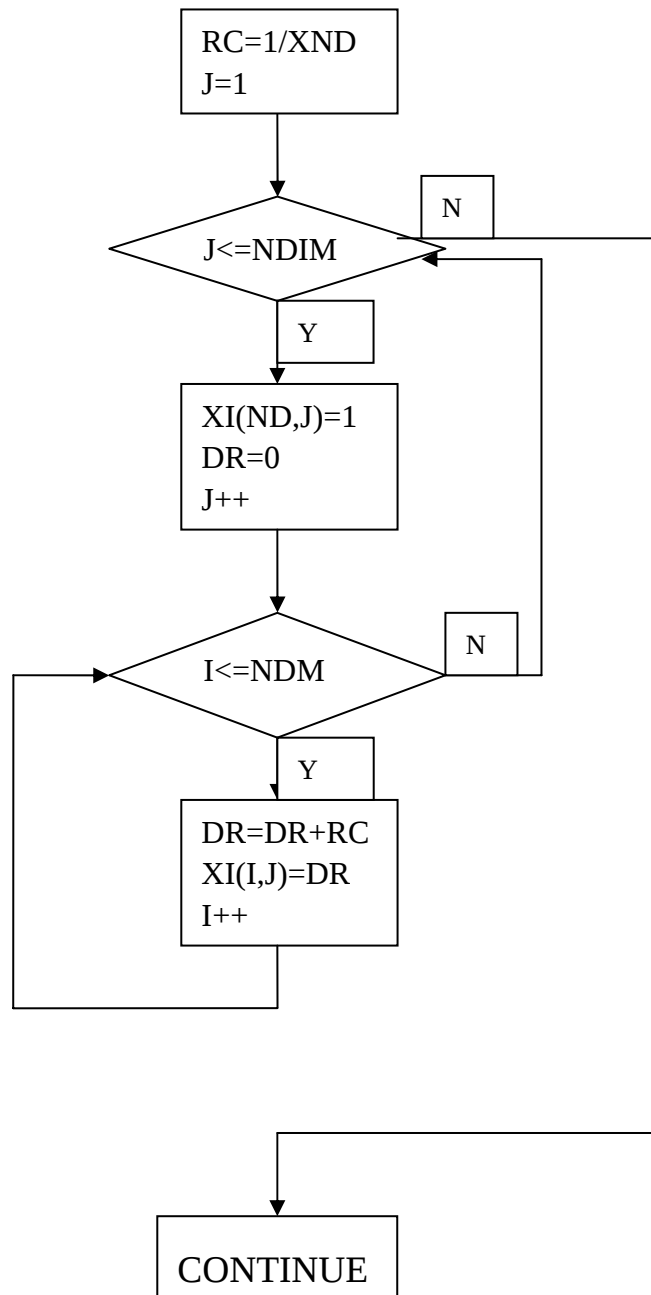
XI (I, J): 变量的区间的分布。

XND: 区间分层数。

XNDM=XND-1。

DR: 求和号。

初始化的分布是一个均匀分布，实现的流程图如下：



3) 初始叠代参数:

这不仅仅是初始化的过程了，而是大的循环的一部分，每改变一次变量的分布，就需要初始化叠代参数一次。下面简述各个初始化的参数以及各参数的意义或在以下何模块里要用到。

IT=IT+1; 叠代的次数。

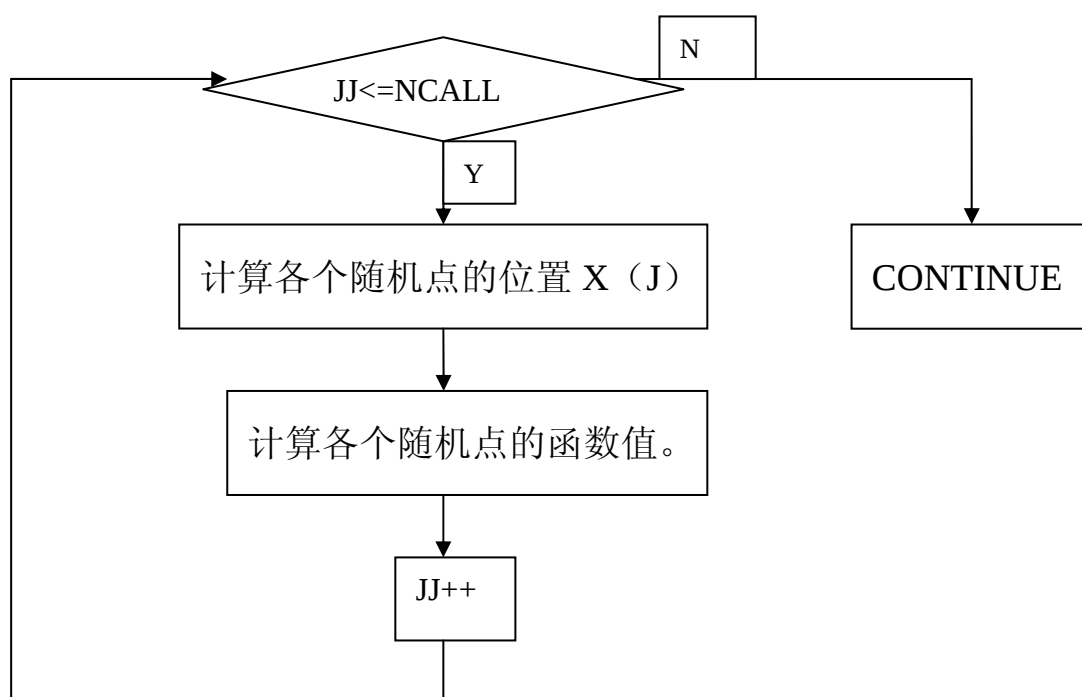
TI=0; 积分值清零。

SFUN2=0; 在 4, 5 中求误差时要用。

D (I, J) =0; 对 D 矩阵清零，在 4, 5, 7 中要用。

4) 计算各个随机点的位置 X (J)，计算各个随机点的函数值。

这里主要是算各个随机点的位置 X (J) 和各个随机点的函数值，并为下面的求误差作好准备。总的流程图如下：



下面分别看计算各个随机点的位置 X (J) 和计算各个随机点的函数值的流程图：

A) 各个随机点的位置 X (J)

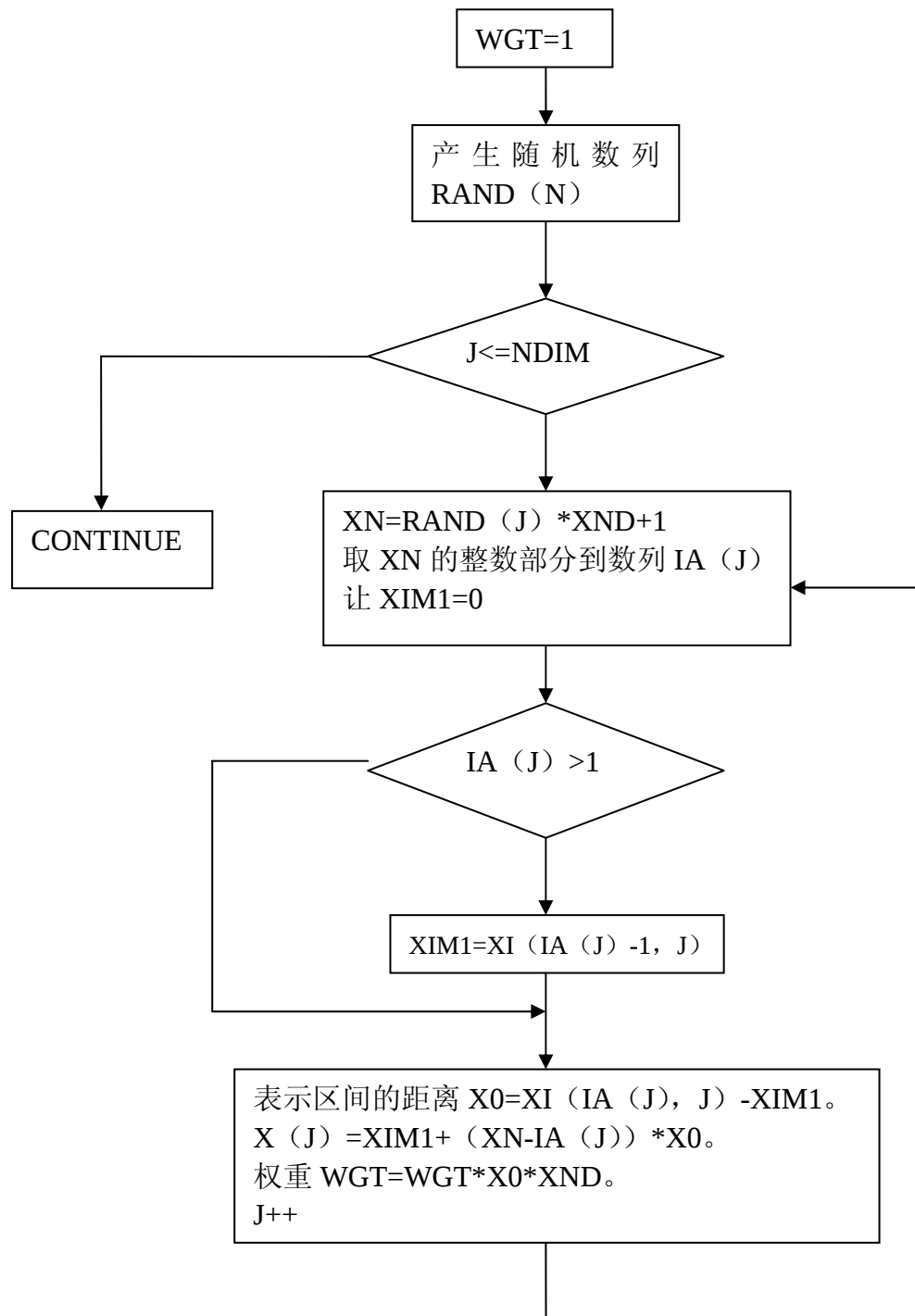
变量定义：

NDIM: 变量的个数。

XI (I, J): 变量的分布。

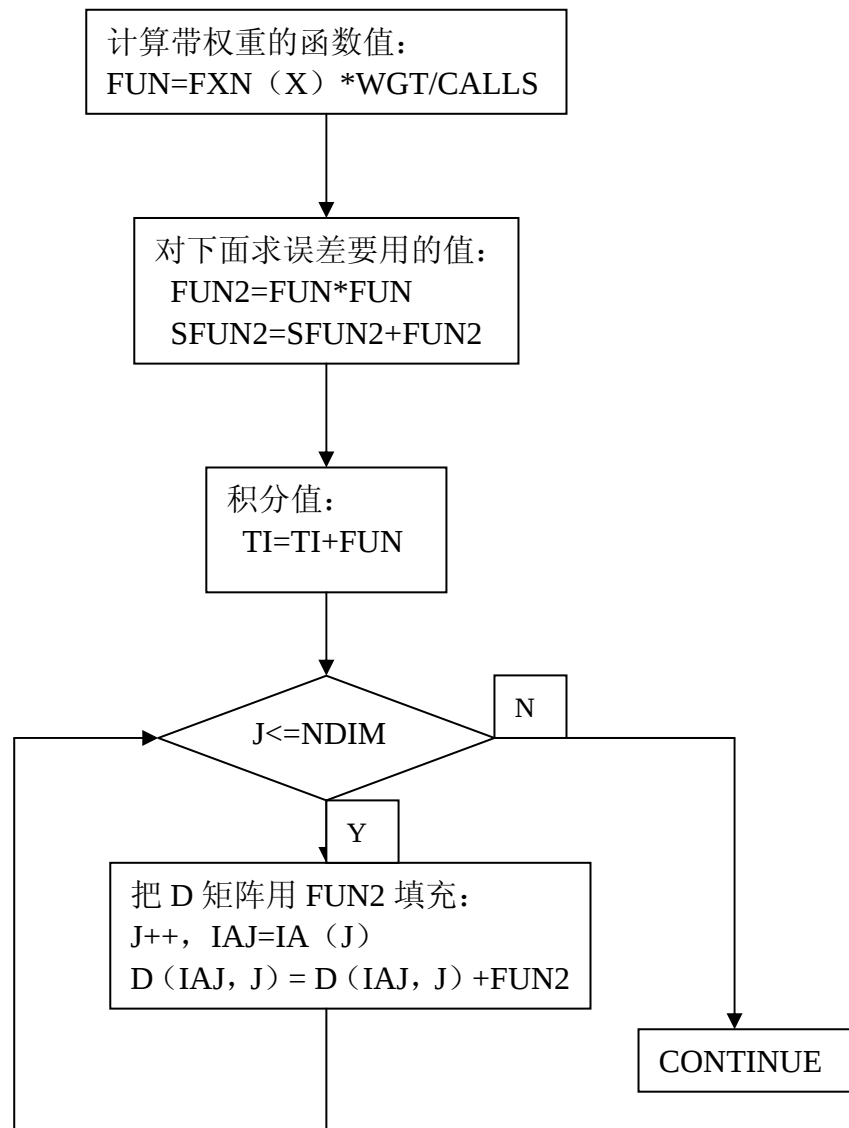
WGT: 权重。

流程图:



这里是用随机数经过 $\text{RAND}(J) * \text{XND} + 1$ 的变换值的整数部分 $\text{IA}(J)$ 选择了分层区间，用其小数部分来选择了坐标点在该分层区间的具体位置。该部分得到 $X(J)$ ，和该点的权重。

B) 计算各个随机点的函数值

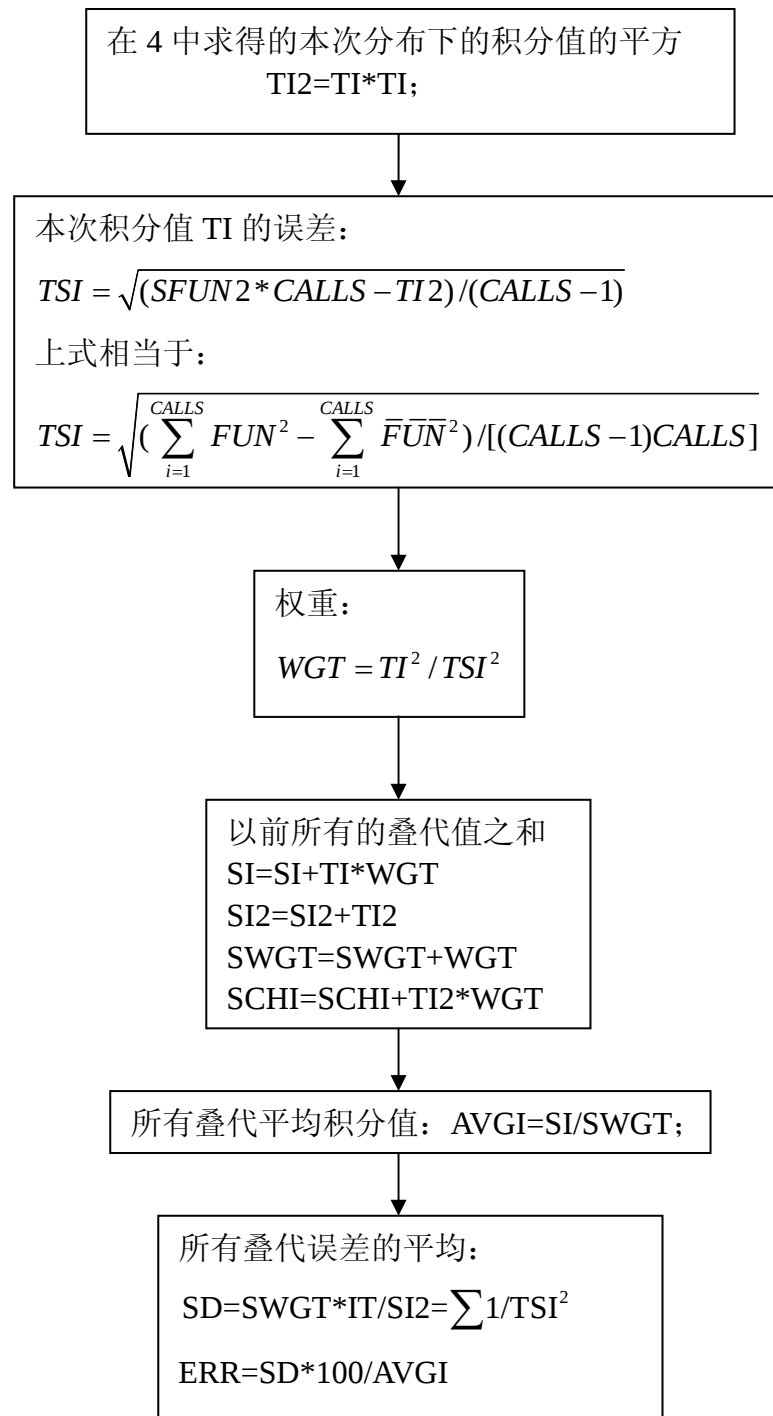


这里求出了带权重的函数值 FUN ，以及下面积分和求误差要用的几个量： SFUN2 ， FUN2 ，D 阵， TI 。其中 D 阵对第 7 个模块也很重要。D 矩阵的矩阵元是对第 J 个变量的第 I 个区间内所有投点的 FUN2 的求和。

5) 计算积分值和积分的误差

该部分是计算了在以上的投点和分布下的积分值和积分的误差值。注意这个积分值和积分误差是所有叠代的平均值。

具体的流程图如下：



该部分计算出了所有叠代的平均积分值 $AVGI$ ，所有叠代的相对误差平均 ERR ，在本次分布下的积分 TI ，本次积分的误差 TSI 。

6) 判断误差是否符合要求

该部分只是一个判断语句：

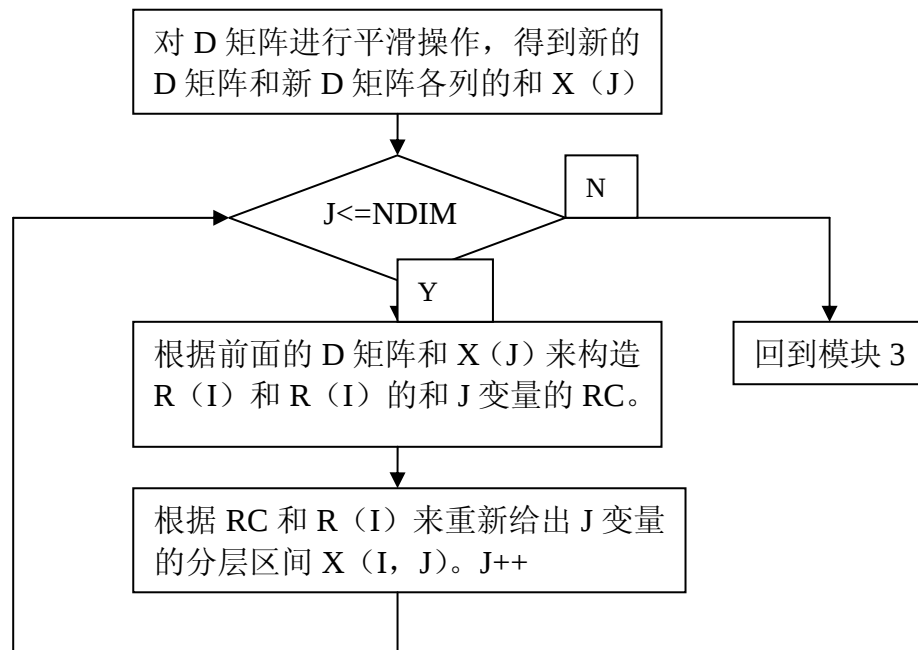
$$\text{如果 } \left(\left| \frac{SD}{AVGI}(\text{误差}) \right| > |ACC| \text{ 且 } IT < ITMX \right)$$

[其中第一个是表示精度是否符合要求，第二个是表示叠代次数是否满足要求]

如果成立就进入 7，否则就可以按照要求打印输出了，即进入 8。

7) 按误差改变各区间的大小，重新获得一个变量的分布

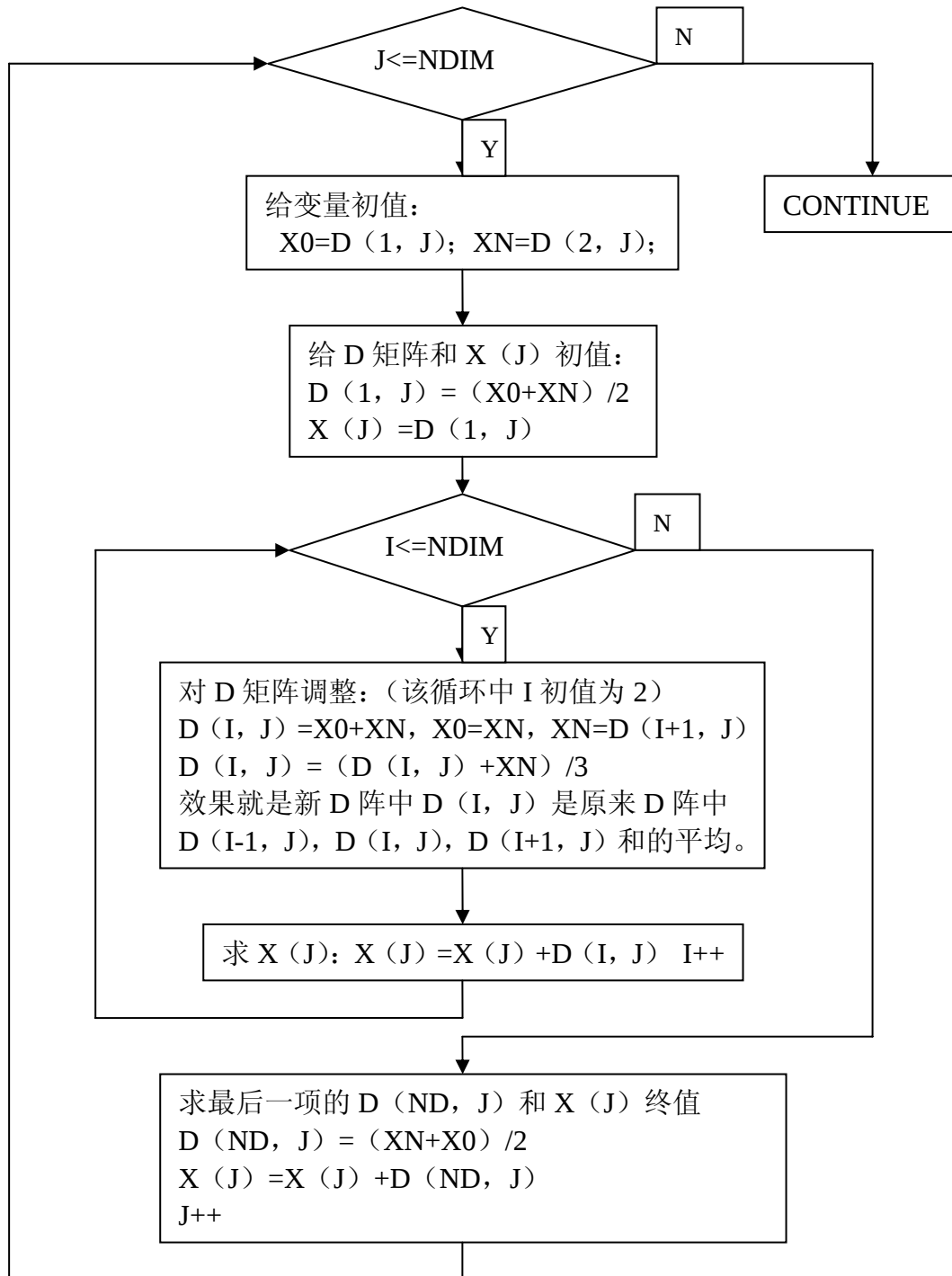
这部分是 VEGAS 的精华所在，其中的有些算式其意义也不是很明确，只能作为定义给出。该部分的主要工作就是改变 $X(I, J)$ 的分布，对每个变量的分层区间作了一个调整，其调整的具体依据是前面 4 中得到的 $D(IAJ, J)$ 矩阵。具体操作先用下面的主流程图表示：



下面来仔细看以上三部分的具体流程图：

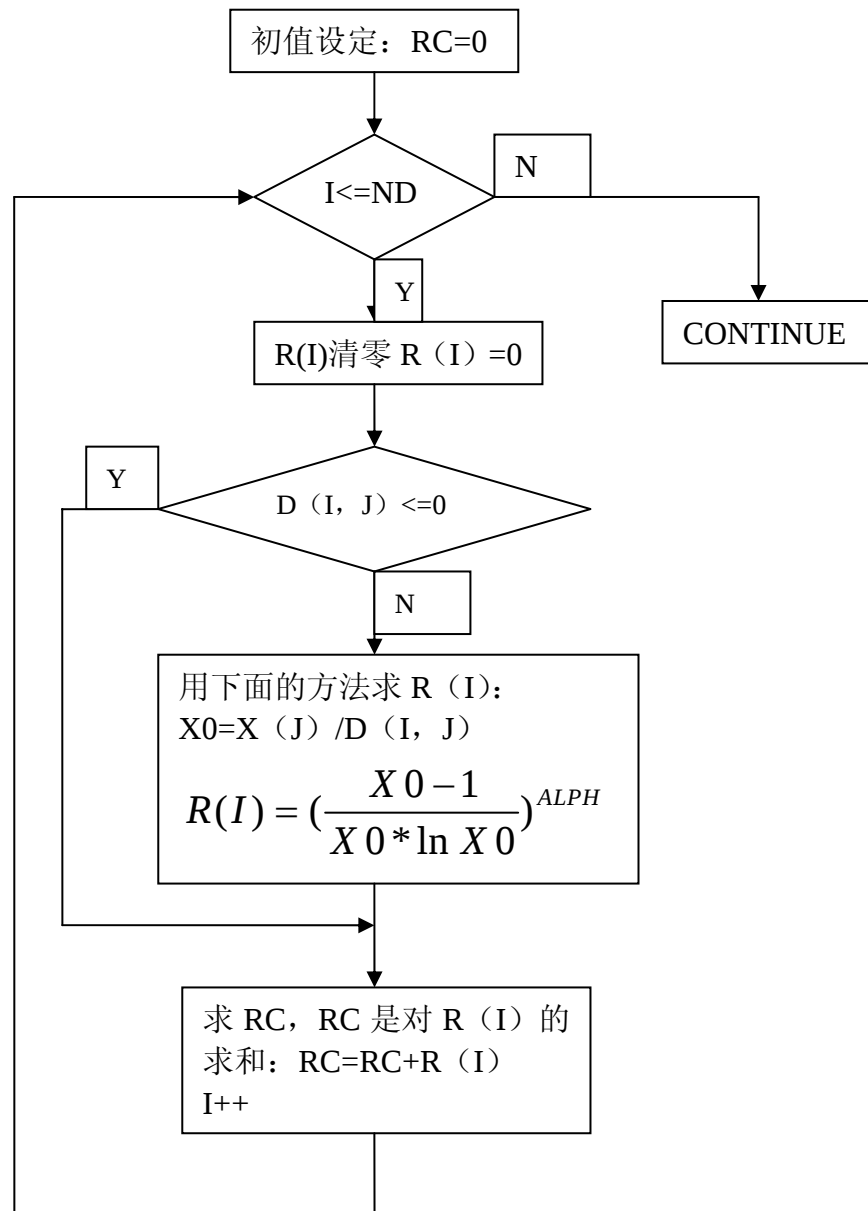
A) 对 D 矩阵进行平滑操作，得到新的 D 矩阵和新 D 矩阵各列的和 X (J)

流程图：



B) 根据前面的 **D** 矩阵和 **X (J)** 来构造 **R (I)** 和 **R (I)** 的和 **J** 变量的 **RC**。

流程图如下：



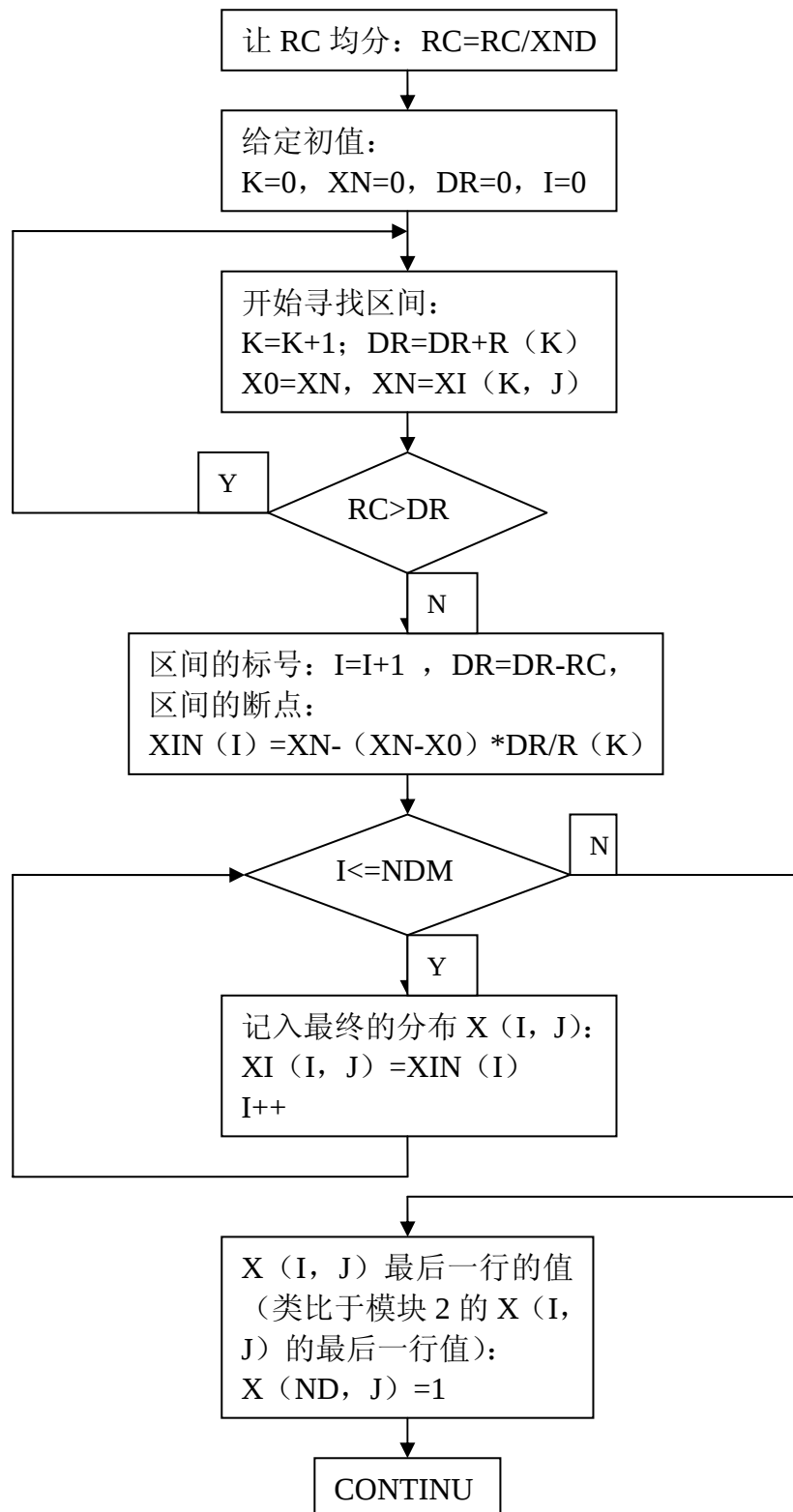
这里 ALPH 是定义的一个参数，上述的公式不清楚如何得来。

C) 根据 **RC** 和 **R (I)** 来重新给出 **J** 变量的积分区间 **X (I, J)**。

这部分重新定义了每个积分区间的长度，从而给出了新的 **X (I, J)**,

要用到前面的值 RC , $R(I)$, XND (是总的分的积分区间的个数)。

流程图如下：



控制新的分布的量是 RC 和 $R(I)$ ，刚开始时由 B 可知 RC 是所有 $R(I)$ 的和，然后把 RC 取为其原来的 XND 分之一，再由 $R(I)$ 的累加得到 DR ，当 DR 超过 RC 时，取 $R(I)$ 的标记 I 值取原来分布的 $X(I, J)$ 值来得到 XN 值，而 $X(I-1, J)$ 作为 $X0$ 值，最后得到新的 $XI(I, J) = XN - (XN - X0) * DR / R(K)$ 。这就是变换得到新分布的过程。

8) 打印输出结果

这一部分就是按要求即输入参数 $NPRN$ [打印的要求（具体要求略过）]来打印输出，这里不作详述了。

总上前面的内容，已经基本把 $VEGAS$ 程序的思想说清楚了。