Задача №2 Сформулировать точную постановку задачи Multiway NPP с ЦФ : минимизация суммы абсолютных отклонений по подмножествам от идеальной суммы в подмножестве . Реализовать точную постановку .

Дано:

n - количество чисел

$S = \{S_1, S_2, \dots, S_n\}$ - мультимножество из n натуральных чисел

k - количество подмножеств

Надо:

Разбить мультимножество S на k подмножеств с такими суммами, чтобы минимизировать сумму модулей их отклонений от идеальной суммы

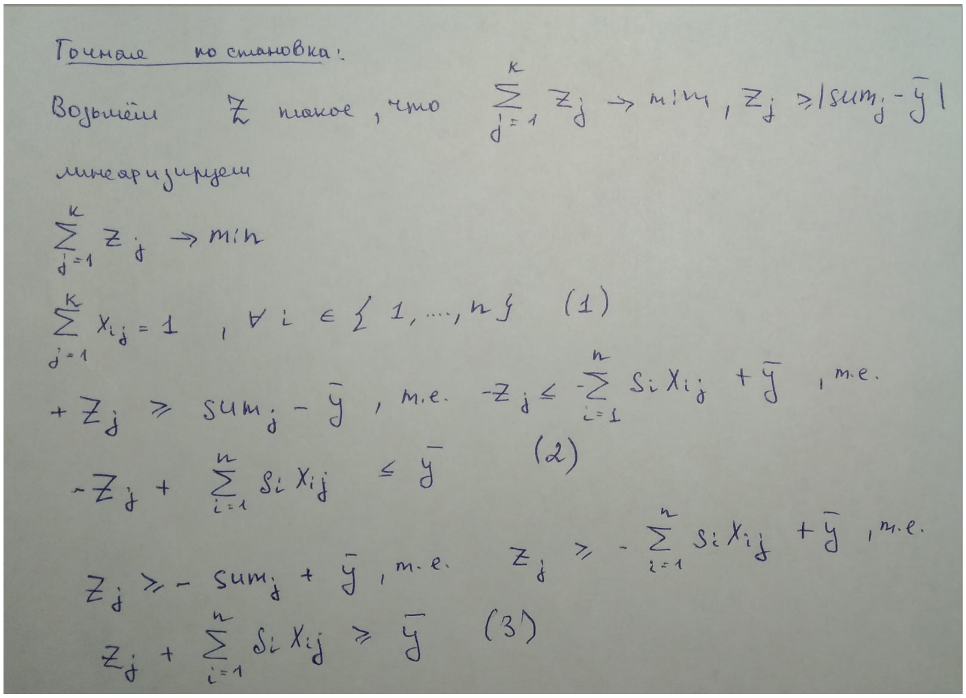
Обозначение:

sum_j, $j \in \{1, \dots, k\}$ - сумма в j группе

$\bar{y} = \dfrac{\sum_{i=1}^{n} S_i}{k}$ - идеальная сумма, к которой стремятся группы

Неизвестные: $x_{ij} = \begin{cases} 1 \\ 0 \end{cases} \quad \forall i \in \{1, \dots, n\} \quad \forall j \in \{1, \dots, k\}$

Целевая функция: $\sum_{j=1}^{k} |sum_j - \bar{y}| \to min$



Точная постановка:

Возьмём Z такое, что $\sum_{j=1}^{k} z_j \to min$, $z_j \geq |sum_j - \bar{y}|$

линеаризируем

$$\sum_{j=1}^{k} z_j \to min$$

$$\sum_{j=1}^{k} x_{ij} = 1, \quad \forall i \in \{1, \dots, n\} \quad (1)$$

$+z_j \geq sum_j - \bar{y}$, т.е. $-z_j \leq -\sum_{i=1}^{n} s_i x_{ij} + \bar{y}$, т.е.

$$-z_j + \sum_{i=1}^{n} s_i x_{ij} \leq \bar{y} \quad (2)$$

$z_j \geq -sum_j + \bar{y}$, т.е. $z_j \geq -\sum_{i=1}^{n} s_i x_{ij} + \bar{y}$, т.е.

$$z_j + \sum_{i=1}^{n} s_i x_{ij} \geq \bar{y} \quad (3)$$

Реализация

```
In[•]:= k = 3;(*количество подмножеств*)
n = 7;(*количество чисел*)
initialData = RandomInteger[{1, 10}, n];(*мультимножество*)

In[•]:= varsX = Array[x, {n, k}];
varsZ = Array[z, k];
vars = Join[Flatten@varsX, varsZ];

In[•]:= objFun = Total[varsZ]; (*целевая функция*)
c = Last@CoefficientArrays[objFun, vars];

In[•]:= con1 = Total[varsX, {2}]; (*первое условие*)
rhs1 = ConstantArray[{1, 0}, n];
(*= - '0', ≥ - '1', ≤ - '-1'*)
con2 = -varsZ + initialData.varsX;(*второе условие*)
rhs2 = ConstantArray[{Total[initialData]/k // N, -1}, k];
con3 = varsZ + initialData.varsX;(*третье условие*)
rhs3 = ConstantArray[{Total[initialData]/k // N, 1}, k];

In[•]:= lu = Join[ConstantArray[{0, 1}, n * k], ConstantArray[{0, Total@initialData}, k]];
domain = Join[ConstantArray[Integers, n * k], ConstantArray[Reals, k]];
m = Last@CoefficientArrays[Join[con1, con2, con3], vars];

In[•]:= sol = LinearProgramming[c, m, Join[rhs1, rhs2, rhs3], lu, domain];
```

In[•]:=
```
partition = Pick[initialData, #, 1] & /@
  Transpose[Partition[sol⟦ ;; -3⟧, k]] (*полученное разбиение чисел на группы*)
```
Out[•]= {{3, 8}, {6, 6}, {2, 2, 7}}

In[•]:=
```
sum = Total[partition, {2}] (*сумма чисел в каждой группе*)
```
Out[•]= {11, 12, 11}

In[•]:=
```
abs = Abs[sum - Total[initialData]/k]
 (*модуль отклонения сумм каждой группы от идеальной суммы*)
```
Out[•]= $\left\{\frac{1}{3}, \frac{2}{3}, \frac{1}{3}\right\}$

In[•]:=
```
sumOfAbs = Total[abs] (*минимальная сумма модулей отклонений*)
```
Out[•]= $\frac{4}{3}$