

**МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ПЛОТНЫХ УПАКОВОК
РАВНЫХ СФЕР НА ЦИФРОВЫХ
ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНЫХ МАШИНАХ**

В. А. ВОРОБЬЕВ, В. К. КИВРАН, И. Э. НААЦ

(Представлена научным семинаром УВЛ)

Плотные упаковки равных сфер образуются путем пакования каждой последующей сферы в углубления трех соседних ранее упакованных сфер. Этот же принцип в несколько видоизмененной форме можно использовать и для получения случайных плотных упаковок.

При моделировании упаковок на цифровых вычислительных машинах (ЦВМ) удобно производить упаковку сфер в куб, расположенный в первом октанте трехмерной прямоугольной системы координат с вершиной в начале координат. Диаметр сфер можно измерять в долях ребра куба, тогда величину ребра куба можно принять за единицу. Для удобства дальнейшего изложения систему координат ориентируем в пространстве так, чтобы одна из осей координат (допустим, третья) была расположена вертикально, тогда одна грань, перпендикулярная этой оси, будет называться верхней, другая, параллельная, — нижней или дном куба.

Плотную случайную упаковку сфер в таком кубе можно получить следующим образом с учетом экономии затрат машинного времени. Предположим, что часть сфер уже упакована и расположена на дне, в нижней части куба. Для каждой очередной сферы разыгрываются случайным образом две первые координаты [1] и вблизи определяются координаты центра, чтобы сфера с этими координатами занимала по возможности одно из нижних в плоскости XV положений, касаясь нижних сфер. Если же вблизи разыгранных координат нет упакованных сфер, пакуемая сфера помещается на дно куба с теми же координатами. Можно помещать сферу в самое нижнее положение, но поиск такого положения значительно увеличивает затраты машинного времени.

При окончании процесса упаковки вышеописанным способом центр очередной пакуемой сферы часто не может принимать найденные координаты, не пересекаясь верхней гранью. В этом случае делается повторная попытка упаковки, начиная с розыгрыша координат центра. Во время опробования алгоритма на ЦВМ М-20 установлено, что при упаковке сфер диаметром 0,1 после 100 неудачных попыток вероятность возможности все же упаковать эту сферу без перемещения ранее упакованных сфер пренебрежительно мала. Поэтому за критерий окончания процесса упаковки принято количество $1/d^2$ неудавшихся подряд попыток установки очередной сферы, где d — диаметр сферы.

Алгоритм же поиска одного из нижних положений пакуемой сферы может быть различным [2]. Так, можно делать по осям координат

```

graph TD
    Start(( )) --> 1[1.  $N_1 = 1/d^2$ ]
    1 --> 2[2.  $\lambda_1, \lambda_2$ ]
    2 --> 3{3.  $2 \leq \lambda \leq 1-\epsilon$ }
    3 -- "нет" --> 18{18.  $|x - x_k| < d$ }
    3 -- "да" --> 4[4.  $x_3 = x'_3$   
9.  $\Delta = 2/10$ ]
    4 --> 5[5.  $\lambda_i = \Delta$ ]
    5 --> 6{6.  $\lambda_i \leq 1-\epsilon$ }
    6 -- "нет" --> 18
    6 -- "да" --> 7{7.  $\lambda_j > x'_j$ }
    7 -- "нет" --> 18
    7 -- "да" --> 8[8.  $g, x_3 = x'_3$ ]
    8 --> 9[9.  $x_i = \Delta$ ]
    9 --> 10{10.  $x_i \geq \epsilon$ }
    10 -- "нет" --> 18
    10 -- "да" --> 11{11.  $x_j > x'_j$ }
    11 -- "нет" --> 18
    11 -- "да" --> 12[12.  $g, x_3 = x'_3$ ]
    12 --> 13{13.  $g = 0$ }
    13 -- "нет" --> 14[14.  $g = 0$ ]
    13 -- "да" --> 15{15.  $\epsilon \leq 2$ }
    15 -- "да" --> 14
    15 -- "нет" --> 17[17.  $g, \Delta = 2/100$ ]
    17 --> 16{16.  $\Delta = 2/100$ }
    16 -- "да" --> 28{28.  $x_3 \leq 1-\epsilon$ }
    16 -- "нет" --> 18
    28 -- "да" --> 29[29.  $x_{ij}$ ]
    28 -- "нет" --> 26{26.  $N_1 \geq 0$ }
    26 -- "да" --> 27[27.  $N_1 - 1$ ]
    27 --> 28
    26 -- "нет" --> 18
    18 -- "да" --> 19[19.  $R_k^2$ ]
    19 --> 20{20.  $R_k^2 < d^2$ }
    20 -- "да" --> 21[21.  $x'_3$ ]
    21 --> 22{22.  $x''_3 > x'_3$ }
    22 -- "да" --> 23[23.  $x'_3 = x''_3$ ]
    23 --> 24{24.  $k \neq n$ }
    24 -- "да" --> 25[25. стоп]
    24 -- "нет" --> 18
    
```

Упаковка каждой очередной сферы начинается с восстановления обратного счетчика N_1 (бл. 1), по которому производится остановка процесса упаковки (бл. 25), если было произведено $1/d^2$ неудачных попыток упаковать сферу подряд. Если сфера может быть упакована, т. е. не пересекается с верхней гранью (бл. 28); сфера считается упакованной, координаты ее центра фиксируются в памяти ЦВМ (бл. 29).

и упаковка следующей сферы начинается с восстановления счетчика N_1 . Если же упакованная сфера пересекается верхней гранью, попытка упаковки повторяется, но уже минуя восстановления счетчика, при этом из счетчика предварительно вычитается единица (бл. 27) и счетчик анализируется на нуль (бл. 26). Если $1/d^2$ попыток упаковать очередную сферу не увенчалась успехом, т. е. счетчик N_1 очищен, процесс упаковки считается законченным (бл. 25).

Итак, при упаковке очередной сферы после восстановления счетчика N_1 производится розыгрыш первых двух координат сферы (бл. 2) и проверка на пересечение боковыми гранями сферы с разыгранными координатами (бл. 3). Если эта сфера пересекается хотя бы одной боковой гранью, розыгрыш координат повторяется.

Далее, при фиксированных значениях двух координат находится третья, x_3' (бл. 18 ÷ 24). Так как к этой части блок-схемы (к бл. 18 ÷ 24) производится повторное обращение из других мест программы; блоки 18 ÷ 24 объединяются в отдельную подпрограмму. Третья координата находится следующим образом. Пакуемый шар последовательно, не меняя значения первых двух координат, приводится в соприкосновение со всеми ранее упакованными сферами (с которыми можно произвести это соприкосновение) и вычисляется верхнее значение третьей координаты x_3'' . Из всех вычисленных значений выбирается максимальное. Для каждой проверяемой ранее упакованной сферы вычисляется разность соответствующих координат пакуемой и упакованной сфер, и, очевидно, что если эта разница хотя бы для одной пары координат больше величины диаметра (бл. 18), то эти сферы не могут касаться, поэтому дальше выбирается следующая упакованная сфера (бл. 24). Если эта разница меньше величины диаметра, рассчитывается расстояние (квадрат расстояния) между сферами в плоскости первых двух осей координат (бл. 19) и, если только это расстояние меньше диаметра (бл. 20), сферы приводятся в соприкосновение и рассчитывается третья координата x_3'' . А если это значение больше предыдущего значения, x_3' (бл. 22), предыдущее значение x_3' заменяется новым (бл. 23) и т. д. Максимальное значение из всех x_3'' принимается за третью координату пакуемой сферы (бл. 4).

Следует отметить, что результат работы блоков 18 ÷ 24 не изменится, если из блок-схемы удалить 18-й блок, так как результат его действия автоматически входит в результат работы блоков 19 и 20. Однако сохранение блока 18 сокращает расход машинного времени примерно в два раза.

Затем следует серия пробных шагов сначала по одной оси координат, затем по другой (бл. 15), причем вначале величина пробного шага выбирается $0,1 r$ (бл. 4), затем положение пакуемой сферы уточняется пробными шагами в $0,01 r$. Вначале делается пробный шаг от центра по оси координат (бл. 5), и если при этом боковая грань не пересечет пакуемую сферу с новой координатой (бл. 6), вычисляется третья координата x_3' (бл. 18 ÷ 24), причем, если она меньше предыдущего значения x_3 (бл. 7), старые координаты центра пакуемой сферы заменяются новыми (бл. 8) и пробный шаг повторяется. Если же новое значение x_3' не меньше старого x_3 , или сфера с новыми координатами пересекается с боковой гранью, делается пробный шаг в обратную сторону, к началу координат (бл. 9) с аналогичными вычислениями и проверками (бл. 10 ÷ 12).

Пробные шаги сразу же прекращаются, если по одной из осей координат не было ни одного исправления координат. Для выявления этого момента введена специальная переменная g , которая перед началом серии пробных шагов по любой оси координат полагается нулю,

а при исправлении координат становится отличной от нуля (бл. 8 и 12). Величина g также становится отличной от нуля перед началом пробных шагов впервые для данной пакуемой сферы (бл. 4). Итак, если к моменту завершения производства пробных шагов по одной оси координат было хотя бы одно исправление, когда $g \neq 0$ (бл. 13), величина g полагается нулю (бл. 14) и производятся пробные шаги по другой оси координат (бл. 15). Если же не было ни одного исправления, а в этом случае величина g остается равной нулю, при величине пробного шага $0,1r$ (бл. 16) она уменьшается до $0,01r$ (бл. 17) и пробные шаги продолжаются. А когда положение центра сферы найдено уже пробными шагами в $0,01r$, сферу можно считать упакованной, если она не пересекается верхней гранью.

При упаковке сфер вышеописанным способом производится постепенное заполнение пакуемого куба, причем самые нижние упакованные сферы уже не участвуют в процессе упаковки, поэтому хранение в памяти ЦВМ координат этих сфер становится нецелесообразно. Так, при упаковке сфер диаметром $0,1$ в памяти ЦВМ достаточно хранить координаты около 500 последних упакованных сфер (половина пакуемого куба). Поэтому появилась возможность «наращивать» упаковку по третьей координате без ограничения.

Программа упаковки по вышеописанной блок-схеме была опробована на ЦВМ М-20. В один куб упаковывается $860 \div 875$ сфер с плотностью упаковки $0,450 \div 0,460$. Плотность заполнения (с учетом граничного эффекта) составляет примерно $0,475$. Время упаковки одного куба — $35 \div 40$ минут, при упаковке куба при наращивании высоты упаковки затраты машинного времени несколько выше, около 50 минут.

Для получения более плотных упаковок сфер предлагается другой алгоритм поиска минимального положения сферы. Он заключается в следующем. Вначале, после розыгрыша новых двух координат, пробные шаги делаются по меридиану соприкасающейся сферы, затем в перпендикулярных направлениях до тех пор, пока пакуемая сфера может опускаться.

На рис. 2 приведена блок-схема упаковки с последним алгоритмом поиска положения пакуемой сферы. Эта блок-схема отличается от предыдущей только блоками выполнения пробных шагов, т. е. блоки $1 \div 3$, $14 \div 20$ и $21 \div 25$ на рис. 2 идентичны блокам $1 \div 3$, $18 \div 24$ и $25 \div 29$ соответственно на рис. 1.

Выполнение пробных шагов производится сериями на десять шагов, причем перед этой схемой величина пробного шага задается $0,1r$ (бл. 4); восстанавливается обратный счетчик числа шагов $q = 10$ и в отдельный массив памяти ЦВМ переписываются координаты тех сфер, с которыми может произойти соприкосновение пакуемой сферы за время этой серии (выделение сфер, координаты центров которых отстоят по обеим осям координат от центра пакуемой сферы не далее $3r$). Последние действия служат в основном для сокращения затрат машинного времени ЦВМ, затраты времени уменьшаются примерно в 2,5 раза.

Если предварительно значение третьей координаты x_3 задать заранее большим, чем оно может иметь, то после вычисления третьей координаты x_3' блоками $14 \div 20$ последующая работа начнется с блока 5, который формирует приращения по осям координат во всех вышеуказанных трех направлениях. Затем производится пробный шаг по первому направлению (бл. 6) и после проверки на пересечение сферы в этой новой точке боковыми гранями (бл. 7) вычисляется третья координата x_3' (бл. $14 \div 20$). Если это значение меньше старого x_3 (бл. 8), все старые координаты заменяются новыми, из счетчика q вычитается единица (бл. 9) и, если счетчик еще не очищен (бл. 10), вычисляются новые

```

graph TD
    Start(( )) --> 1[1.  $N_1 = 1/d^2$ ]
    1 --> 2[2.  $X_1, X_2$ ]
    2 --> 3{3.  $z \leq X \leq 1-z$ }
    3 --> 4[4.  $q = 10, \Delta = z/10, p$ ]
    4 --> 5[5.  $\Delta X_m$ ]
    5 --> 6[6.  $X + \Delta X_m$ ]
    6 --> 7{7.  $z \leq X \leq 1-z$ }
    7 --> 8{8.  $x'_3 < X_3$ }
    8 --> 9[9.  $x_3 = x'_3, q = 1$ ]
    9 --> 10{10.  $q \geq 0$ }
    10 --> 11{11.  $m \leq 7$ }
    11 --> 12{12.  $\Delta = z/100$ }
    12 --> 13[13.  $\Delta = z/100$ ]
    13 --> 14{14.  $|x - x_n| < d$ }
    14 --> 15[15.  $R_k^2$ ]
    15 --> 16{16.  $R_k^2 < d^2$ }
    16 --> 17[17.  $x''_3$ ]
    17 --> 18{18.  $x''_3 > x'_3$ }
    18 --> 19[19.  $x'_3 = x''_3$ ]
    19 --> 20{20.  $K \leq p$ }
    20 --> 21[21.  $смон$ ]
    21 --> 22{22.  $N_1 \geq 0$ }
    22 --> 23[23.  $N_1 - 1$ ]
    23 --> 24{24.  $x_3 \leq h - z$ }
    24 --> 25[25.  $x_{(j)}$ ]
    25 --> 1
    25 --> 2
    25 --> 4
    25 --> 6
    25 --> 8
    25 --> 10
    25 --> 12
    25 --> 14
    25 --> 16
    25 --> 18
    25 --> 20
    25 --> 22
    25 --> 24
    25 --> 26(( ))
    3 --> 14
    7 --> 14
    8 --> 14
    10 --> 14
    11 --> 14
    12 --> 14
    13 --> 14
    14 --> 15
    16 --> 17
    18 --> 19
    20 --> 21
    22 --> 23
    24 --> 25
    25 --> 26
    
```

The flowchart illustrates the algorithm for determining the optimal number of sections in a beam. It begins with an initial calculation of $N_1 = 1/d^2$ and proceeds through a series of steps involving calculations of X_1, X_2 , ΔX_m , and $X + \Delta X_m$. The algorithm includes several decision points (3, 7, 8, 10, 11, 12, 14, 16, 18, 20, 22, 24) that determine the flow based on various conditions. Key steps include the calculation of R_k^2 , x''_3 , and x'_3 , and the final output $x_{(j)}$. The flowchart is structured to iteratively refine the values of X and N_1 until the optimal number of sections is determined.

третьей координаты не уменьшается, производится проверка величины пробного шага (бл. 12). Когда величина пробного шага равна $0,1 r$, она заменяется на $0,01 r$ (бл. 13) и проводится подготовка к новому пробному шагу (с блока 5), а если величина пробного шага равна $0,01 r$ и при выполнении пробных шагов по всем трем направлениям величина

x_3 не уменьшается, сфера считается упакованной при условии, что она не пересекается верхней гранью.

Остальная часть блок-схемы работает аналогично блок-схеме на рис. 1. Разумеется, что блок-схема также позволяет хранить в памяти ЦВМ только координаты последних упакованных сфер (половину пакуемого куба) и производить наращивание упаковки без ограничения.

Программа, составленная по этой блок-схеме, также была опробована на ЦВМ М-20. Затраты машинного времени здесь в два раза выше, чем в предыдущей программе. Так, время упаковки одного куба составляет 1 час 20 мин, а плотность упаковки возросла на 11%. Так, в куб упаковывается примерно 975 сфер диаметром 0,1, что соответствует плотности упаковки 0,51. Плотность же заполнения (с учетом граничного эффекта) составляет $0,535 \div 0,540$. Эти результаты уже близки к результатам физического моделирования при упаковке шаров без смещения ранее упакованных.

ЛИТЕРАТУРА

1. Д. И. Голенко. Моделирование и статистический анализ псевдослучайных чисел на электронных вычислительных машинах. Изд-во «Наука». М., 1965.
 2. Д. Д. ж. Уайлд. Методика поиска экстремума. Изд-во «Наука», М., 1967.
-