

**СООБЩЕНИЯ  
ОБЪЕДИНЕННОГО  
ИНСТИТУТА  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ  
ДУБНА**

P11-87-421

**В.В.Гусев\*, М.С.Касчиев, В.И.Пузынин\***

**АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ СЕТКИ  
В ПАКЕТЕ ПРОГРАММ MULTIMODE**

---

\* Институт физики высоких энергий,  
Протвино, СССР

## ВВЕДЕНИЕ

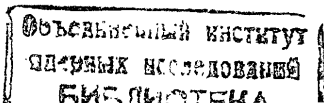
Автоматизация построения сеток в методе конечных элементов (МКЭ) является одним из важнейших этапов при его реализации. Этому вопросу уделяется большое внимание со стороны разработчиков пакетов прикладных программ (ППП) МКЭ для решения различных задач математической физики. Однако описания соответствующих программ практически не публикуются. Примером могут служить известные ППП, как PAFIC<sup>/1/</sup> и MODULEF<sup>/2/</sup>, в которых этот вопрос практически не рассматривается за исключением отдельных модулей, реализующих построение сетки в простейших случаях. В последнее время для решения двумерных задач математической физики активно используются четырехугольные конечные элементы — билинейные, биквадратичные и другие. Эффективное использование таких элементов для областей сложной формы требует разработки специальных алгоритмов и программ автоматического построения соответствующих конечно-элементных сеток.

Цель данной работы состоит в разработке и описании алгоритмов нерегулярного разбиения областей сложной формы на восьмиузловые изопараметрические элементы<sup>/3/</sup>. Приводится описание программы GRID8, реализующей эти алгоритмы. Особое внимание уделено вопросам простоты задания входных данных и возможности избежать дублирования исходной информации. Программа GRID8 легко может быть подключена к ППП МКЭ, использующим восьмиузловые изопараметрические элементы. В настоящее время она реализована в ППП MULTIMODE для расчета спектра частот осесимметричных резонаторов<sup>/4/</sup> и является его составной частью, поэтому данная работа может служить и руководством для пользователей указанного пакета при построении сетки, необходимой для его работы.

## ДИСКРЕТИЗАЦИЯ ОБЛАСТИ НА ОСНОВЕ СУПЕРЭЛЕМЕНТОВ

Пусть в пространстве  $R^2$  задана ограниченная область  $\Omega$ , в которой необходимо построить конечно-элементную сетку из четырехугольных элементов  $e^1, e^2, \dots, e^n$ , удовлетворяющих условиям согласованности<sup>/3/</sup>:

$$a) \quad \Omega = \sum_{i=1}^n e^i ;$$



б) два элемента  $e^i$  и  $e^j$  могут иметь либо только общую вершину, либо общую сторону.

Для решения поставленной задачи разобьем предварительно исходную область на крупные восьмиузловые изопараметрические элементы (суперэлементы <sup>6/7</sup>), которые удовлетворяют условиям согласованности. Каждый суперэлемент определяется координатами восьми узлов  $P_i = (x_i, y_i)$ ,  $i = 1, 8$  (рис. 1) Теперь вместо исходной задачи решим

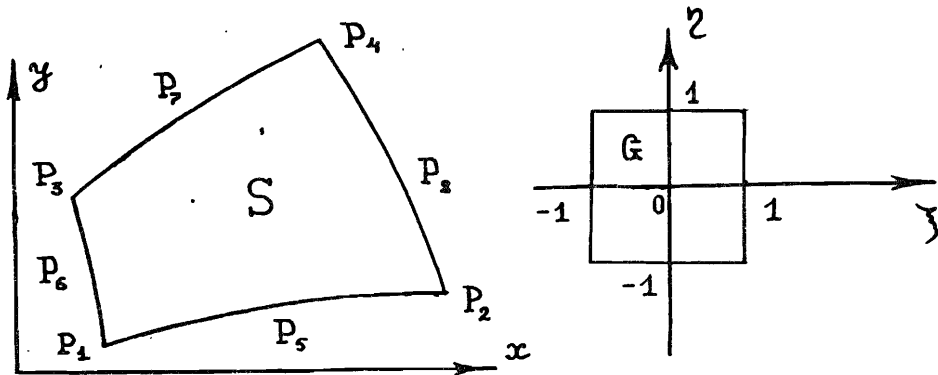


Рис. 1.

более простую — построим сетку в каждом суперэлементе при заданных разбиениях на сторонах. Пусть стороны  $P_1 P_2$ ,  $P_2 P_3$ ,  $P_3 P_4$ ,  $P_4 P_1$  суперэлемента  $S$  разбиты на  $L_1, L_2, L_3, L_4$  интервалов соответственно. Отметим, что построение треугольных элементов возможно при любых  $L_i, i = 1, \dots, 4$ , построение же сетки из четырехугольных элементов осуществимо только, если их сумма — четное число. Для того, чтобы провести это разбиение, суперэлемент  $S$ , заданный в координатах  $(x, y)$ , отображается в квадрат  $G = (-1 \leq \xi, \eta \leq 1)$ , заданный в локальных координатах  $(\xi, \eta)$ . Последний разбивается с использованием некоторого стандартного алгоритма, а обратное отображение  $G \rightarrow S$  осуществляется посредством преобразования

$$\begin{aligned} x &= \sum_{i=1}^8 x_i \Psi_i(\xi, \eta); \\ y &= \sum_{i=1}^8 y_i \Psi_i(\xi, \eta). \end{aligned} \quad (1)$$

Здесь  $(x_i, y_i)$  — координаты восьми узлов суперэлемента. При этом образы всех узлов сетки, построенной в  $G$ , образуют искомую сетку. В качестве  $\Psi_i$  используются функции формы 8-узловых изопараметрических элементов, которые имеют следующий вид:

$$\Psi_1 = -0,25(1 - \xi)(1 - \eta)(\xi + \eta + 1),$$

$$\Psi_5 = 0,5(1 - \xi^2)(1 - \eta),$$

$$\Psi_2 = 0,25(1 + \xi)(1 - \eta)(\xi - \eta - 1),$$

$$\Psi_7 = 0,5(1 - \eta^2)(1 + \xi),$$

$$\Psi_4 = 0,25(1 + \xi)(1 + \eta)(\xi + \eta - 1),$$

$$\Psi_8 = 0,5(1 - \xi^2)(1 + \eta),$$

$$\Psi_3 = 0,25(1 - \xi)(1 + \eta)(\eta - \xi - 1),$$

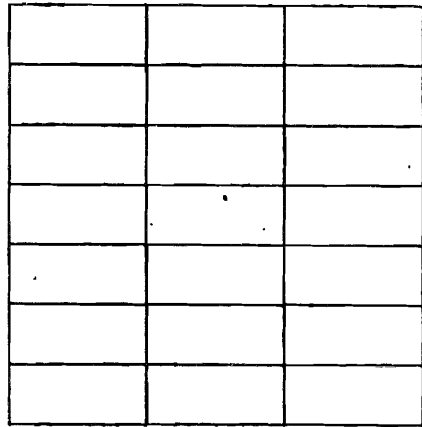
$$\Psi_6 = 0,5(1 - \eta^2)(1 - \xi).$$

Функции  $\Psi_i(\xi, \eta)$  равны единице в узле  $i$  квадрата  $G$  и нулю в остальных узлах. При этом образы всех внутренних узлов сетки, построенной в  $G$ , образуют искомую сетку в  $S$ . Особо отметим, что преобразованию (1) подвергаются лишь внутренние узлы сетки квадрата  $G$ . Это связано с тем, что преобразование (1) может исказить криволинейные стороны суперэлемента, имеющие заданную форму (например, дуга окружности). Объединяя сетки, построенные в каждом суперэлементе, получаем конечно-элементную сетку во всей области  $\Omega$ .

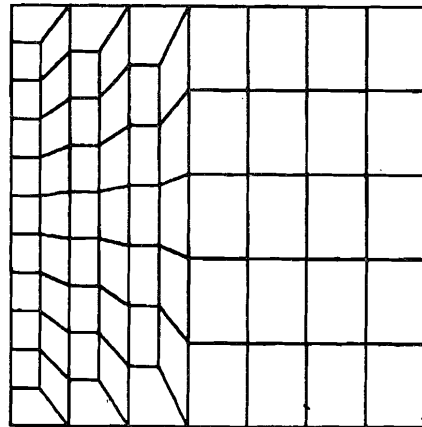
Таким образом задача сводится к тому, чтобы провести разбиение внутри квадрата  $G$  при заданных  $L_1, L_2, L_3, L_4$ . Когда  $L_1 = L_3$  и  $L_2 = L_4$ , построение сетки не вызывает затруднений, так как в этом случае в  $G$  строится равномерная сетка с шагом  $h_1 = 2/L_1$  по оси  $\xi$  и  $h_2 = 2/L_2$  по оси  $\eta$ .

Такой подход осуществлен в работах <sup>6,7/</sup>. Однако наличие только этого способа построения сетки вызывает определенные трудности в случае необходимости сгущения сетки в определенных местах, например, в области быстрого изменения решения задачи. Для построения нерегулярного разбиения квадрата  $G$ ,  $L_1 \neq L_3$  и  $L_2 \neq L_4$  предлагается алгоритм, основная идея которого заключается в построении плавного перехода от каждой стороны  $G$  к некоторому внутреннему квадрату, противоположные стороны которого разбиваются на одинаковое число интервалов. Не уменьшая практической ценности алгоритма, будем полагать, что  $L_1 - L_3 = 2k$ ,  $L_2 - L_4 = 2j$ ,  $k, j = 0, 1, 2, \dots$ . На рис. 2 показаны способы а)-г) разбиения  $G$ , реализующие такой подход при различных соотношениях  $L_i$ .

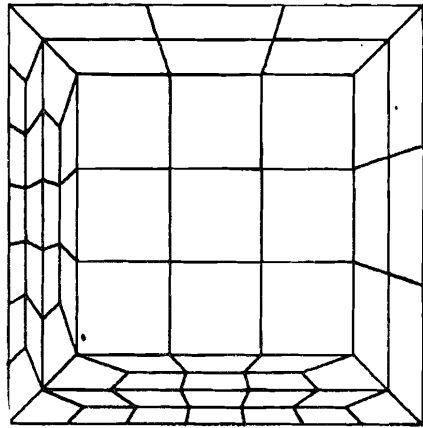
Практический опыт показывает, что этих схем достаточно для эффективного построения нерегулярного разбиения области. Их использование позволяет достаточно рационально распределять узлы внутри области, сгущая сетку в необходимых местах. Схема г) используется в случае, когда суперэлемент является окружностью. Для полу-



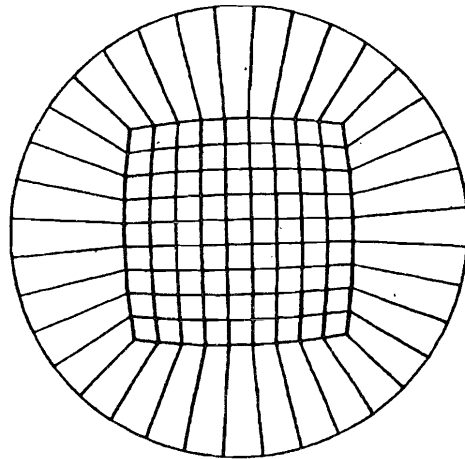
а)



б)



в)



г)

Рис. 2.

чения плавного изменения размеров полученных элементов пользователь может использовать процедуру релаксации (сглаживания) внутренних координат узлов. Пусть  $(x_k, y_k)$  — некоторый внутренний узел сетки,  $m, (x_j, y_j)$  — число соседних узлов и их координаты соответственно. Тогда процедура сглаживания имеет следующий вид:

$$x_k^{\ell+1} = \frac{1}{m} \sum_{j=1}^m x_j^{\ell};$$

$$y_k^{\ell+1} = \frac{1}{m} \sum_{j=1}^m y_j^{\ell}, \quad \ell = 1, 2, \dots$$

Для получения плавной сетки обычно достаточно применить эту процедуру 3-5 раз. Следует заметить, что для четырехугольных элементов эта процедура может приводить к вырожденным элементам, когда область не является выпуклой и имеет внутренние углы. Поэтому пользоваться ею следует с известной степенью осторожности. На рис. 3 показан пример использования сглаживания для схемы г).

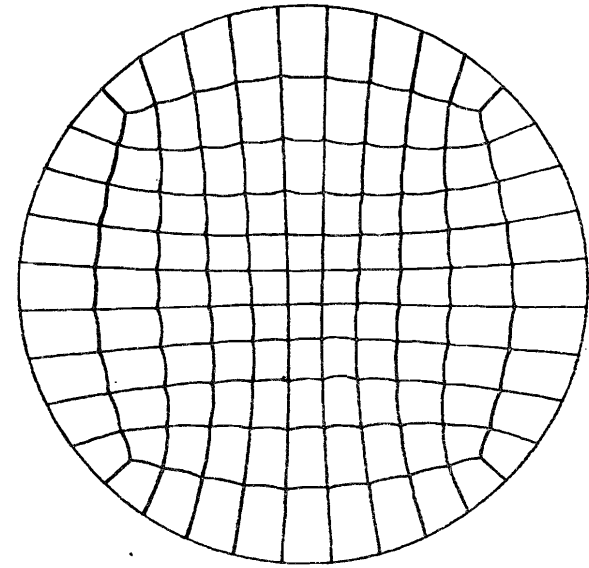


Рис. 3.

#### ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

Программа GRID8, состоящая из двух частей — SERV и GRID, реализована на языке фортран. Для повышения ее эффективности предусмотрено динамическое распределение памяти. В качестве языка входных данных используется стандартный фортрановский оператор MAMELIST. Для отображения исходных данных и конечных результатов применяется графический модуль, выполненный с использованием пакета АТОМ<sup>8/</sup>, осуществляющего интерфейс с различными графическими устройствами. Программа SERV служит для подготовки и отладки входных данных, описывающих конфигурацию исследуемой области и учитывающих характер задачи, которая будет решаться на построенной сетке. Входные данные читаются из файла в карточном формате. Выходные данные сбрасываются в файл последовательного доступа, который потом используется в GRID, осуществляющей непосредственное построение сетки. Результаты работы программы поступают в тот же файл последовательного доступа. Информация о ходе выполнения задания выдается на АЦПУ.

Прежде чем перейти к изложению структуры входных данных, определим следующие понятия:

- 1) подобласть — часть области (может быть и вся область);
- 2) суперэлемент — обобщенный четырехугольник, определенный восемью точками;
- 3) сегмент — сторона суперэлемента, определенная тремя точками;
- 4) точка определяется двумя координатами в декартовой или полярной системах координат.

Для того, чтобы построить сетку в области  $\Omega$ , пользователь предварительно должен разбить ее на невырожденные суперэлементы. Во входных данных задаются только сегменты в произвольном порядке и число разбиений каждого из них. Построение суперэлементов и генерация сетки внутри каждого суперэлемента производится на программном уровне. Если  $\Omega$  обладает некоторой симметрией, то для уменьшения объема входных данных можно использовать операции сдвига и отражения.

Описание данных начинается с карты  $N_1$  — заголовок вводимых данных (формат 204A4).

В следующей карте  $N_2$  (формат NAMELIST /SET/) задаются константы, определяющие режим работы программ:

```
&SET IPRINT=, NOPTIM=, MUSER=, NE=, JK= &END
```

Таблица

Переменная	Значение	Назначение
IPRINT	0*	Задается уровень печати. Минимальный уровень. Печать числа узлов и числа элементов. Нормальный уровень. Выдаются входные данные, число узлов и элементов. Расширенная печать. Используется при отладке входных данных.
	1	
	2-5	
NORTIM	1	Определяет число итераций для сглаживания сетки.
	0*	Операция сглаживания не производится.
NUSER	0*	Значения этой переменной пояснены ниже.
	1	
NE	600*	Верхняя оценка предполагаемого числа элементов.
JK	35*	Предполагаемое отношение числа узлов генерируемой сетки к числу элементов (JK ~ NH/NE)

Переменные NE и JK введены с целью осуществления динамического распределения памяти.

Далее идет описание подобластей. Описание каждой подобласти начинается с карты формата NAMELIST /REGION/

```
&REGION I = i_k &END,
```

где  $i_k$  — номер подобласти.

В программе GRID8 предусмотрены два способа задания координат узлов, с помощью которых описываются узлы. Первый способ — это задание координат в исходной координатной системе  $(x, y)$ . Второй способ состоит во введении локальных полярных систем координат в плоскости  $(x, y)$ . Координаты точек в этом случае задаются полярным радиусом и полярным углом. Карта задания локальных полярных систем координат имеет формат NAMELIST /SYS/

```
&SYS NP = i_1, x_1, y_1,
```

```
i_2, x_2, y_2, ... ,
```

```
i_k, x_k, y_k
```

```
&END
```

```
&END
```

где  $i_k > 0$  — номер полярной системы координат,  $x_k, y_k$  — координаты центра в плоскости  $(x, y)$ . В программе допускается использование до 50 полярных систем в одной подобласти. Если полярные системы не используются, необходима карта

```
&SYS &END
```

После этого задаются данные, определяющие узлы крупной сетки в подобласти. Формат карты — NAMELIST /POINT/, а данные задаются:

```
&POINT N = i_1, x_1, y_1, j_1,
```

```
i_2, x_2, y_2, j_2, ... ,
```

```
i_n, x_n, y_n, j_n
```

```
&END
```

$i_k$  — номер точки,  $j_k$  — номер полярной системы координат ( $j_k = 0$ , если используется система координат  $(x, y)$ ),  $x_k, y_k$  — координаты точки (в полярной системе координат  $\phi_k = y_k$  в градусах,  $R_k = x_k$ ). Точки могут дублироваться по координатам, но должны различаться по номерам. В том же месте надо описать точки, которые могут быть использованы в некоторых операциях над подобластью (сдвиг и отражение). Номера точек ( $i_k$ ) необязательно должны образовывать непрерывную возрастающую последовательность и могут принимать значения  $500 \geq i_k \geq 1$ .

Как уже было сказано, задача пользователя состоит в построении сетки из суперэлементов в области (подобласти). Эта сетка задается узлами и сегментами, из которых программа автоматически строит суперэлементы. Данные о сегментах задаются по формату NAMELIST /SEG/ в виде списка:

&SEG N = i<sub>1</sub>, j<sub>11</sub>, j<sub>21</sub>, j<sub>31</sub>, n<sub>c1</sub>, n<sub>1</sub>, ... ,

i<sub>k</sub>, j<sub>1k</sub>, j<sub>2k</sub>, j<sub>3k</sub>, n<sub>ck</sub>, n<sub>k</sub> &END

i<sub>k</sub> — номер сегмента ( $500 \geq i_k \geq 1$ ). j<sub>1k</sub>, j<sub>2k</sub>, j<sub>3k</sub> — номера заданных в списке POINT точек, образующих сегмент. Если j<sub>2k</sub> = 0, то серединная точка сегмента вычисляется следующим образом: если j<sub>1k</sub> и j<sub>3k</sub> заданы в одной и той же полярной системе координат и имеют одинаковый радиус, то точка j<sub>2k</sub> берется на середине минимальной дуги, соединяющей j<sub>1k</sub> и j<sub>3k</sub>. Во всех остальных случаях серединная точка берется на середине отрезка, соединяющего точки j<sub>1k</sub> и j<sub>3k</sub>. n<sub>ck</sub> — граничный код сегмента (узлы сетки, лежащие на сегменте i<sub>k</sub>, будут иметь этот код). n<sub>k</sub> задает разбиение сегмента i<sub>k</sub>. Смежные границы двух прилегающих подобластей должны иметь совпадающие сегменты с одинаковым числом разбиений.

Как было замечено выше, для построения сетки в суперэлементе необходимо выполнение условия четности суммы чисел разбиения всех сторон суперэлемента. Поэтому с целью избежания ошибок при значении переменной NUSER = 0 число разбиения n<sub>j</sub> сегмента i<sub>j</sub> получает значение  $2[n_j/2] + 1$ . Если NUSER = 1, то в программе используются введенные значения n<sub>j</sub>.

В ряде случаев при наличии определенной симметрии области работа пользователя может быть значительно упрощена за счет введения сдвига и отражения заданной подобласти. Карты, указывающие на необходимость проведения таких операций, имеют формат NAMELIST /TFORM/, а переменные задаются в виде списка:

A) &TFORM IREFL = i<sub>k</sub>, NP1 =, NP2 =, N = jsk, jck, ..., jsn, jcn &END

B) &TFORM ISHIFT = i<sub>k</sub>, NP1 =, NP2 =, N = jsk, jck, ..., jsn, jcn &END

C) &TFORM IREFL = i<sub>k</sub>, NP1 =, NP2 =, N = jsk, jck, ..., jsn, jcn, JALL=1 &END

D) &TFORM ISHIFT = i<sub>k</sub>, NP1 =, NP2 =, N = jsk, jck, ..., jsn, jcn, JALL=1 &END

Здесь i<sub>k</sub> = 1, 2, ... обозначает номер операции и используется только для ее идентификации. В случае A) производится отражение данной подобласти относительно прямой, определяемой точками с номерами NP1 и NP2, а в случае B) — сдвиг подобласти относительно вектора P1P2. В случаях C) и D) производится отражение (сдвиг) всей сетки, построенной предыдущими операциями. Значения над данной подобластью jsk, jck, ..., jsn, jcn указывают на то, что в построенных сегментах после отражения (сдвига) коды jsk, ..., jsn будут заменены на новые коды, которые имеют значения jck, ..., jcn. Если массив N не задан, то построенным после отражения (сдвига) сегментам присваиваются коды их прообразов.

Эта группа заканчивается картой

&TFORM &END,

которая должна быть описана даже в тех случаях, когда операции отражения или сдвига не нужны.

Операции типа TFORM требуют увеличения объема внутренней структуры данных. В некоторых случаях, например при генерации периодических структур это приводит к переполнению используемой оперативной памяти. Тогда может быть удобным использование операций типа NFORM, которые в отличие от операций TFORM (JALL = 1) выполняются независимо друг от друга и работают с внешней памятью. Описание операций типа NFORM следует после операций TFORM и имеет следующий формат:

a) &NFORM IREFL = i<sub>k</sub>, X1 =, Y1 =, X2 =, Y2 =, NSTEP =,

NCOD = jsk, jck, ...,

jsn, jcn

&END

b) &NFORM ISHIFT = i<sub>k</sub>, X1 =, Y1 =, X2 =, Y2 =, NSTEP =,

NCOD jsk, jck, ...,

jsn, jcn

&END

i<sub>k</sub> — номер операции (используется только для ее идентификации). В случае a) производится отражение подобласти, построенной после применения операций TFORM (если TFORM не использовались, то исходной подобласти) относительно прямой, определяемой точками P1 и P2, которые задаются координатами X1, Y1, X2, Y2. В случае b) производится сдвиг на вектор, который задается точками P1 и P2. В массиве NCOD задается информация: сегмент jsk при отражении (сдвиге) будет иметь код jck. Если NSTEP k > 1, то операция применяется k раз. Операции NFORM заканчиваются картой

&NFORM &END.

На этом заканчивается описание одной подобласти и может быть начато описание другой, согласно определяемому выше порядку. Отметим, что полярные системы координат, точки и сегменты нумеруются независимо в каждой из подобластей. Последней картой, устанавливающей конец исходных данных, должна быть карта

&REGION I = 0 &END.

В приложении приведен пример задания входных данных при решении конкретной задачи.

## ПРИЛОЖЕНИЕ

Рассмотрим пример построения конечно-элементной сетки для решения МКЭ задачи отыскания собственных частот и полей колебаний в круглом диафрагмированном волноводе. На рис. 4а показан начальный

### Диафрагмированный волновод

```

&SET      IPRINT = 1,   NOPTIM = 0,   NE = 200,   JK = 4   &END
&REGION   I = 1        &END
&SYST     NP = 1,      1.047,        1,01        &END
&POINT    N = 1,      0.0,          4.051,      0,
           2,        0.847,        4.051,      0,
           3,        0.2,         180.0,      1,
           4,        0.2,         225.0,      1,
           5,        0.2,         270.0,      1,
           6,        1.047,        0.0,        0,
           7,        0.0,          0.0,        0,
           8,        0.0,          1.01,       0      &END
&SEG      N = 1,      1, 0,        2, 3,      7,
           2,        2, 0,        3, 4,      9,
           3,        3, 0,        4, 5,      3,
           4,        4, 0,        5, 5,      3,
           5,        5, 0,        6, 1,      7,
           6,        6, 0,        7, 2,      7,
           7,        7, 0,        8, -1,     7,
           8,        8, 0,        1, -1,     9,
           9,        7, 0,        4, 0,      7,
           10,       8, 0,        3, 0,      7,      &END
&TFORM    &END
&NFORM    &END
&REGION   I = 0        &END

```

этап — задание исходных данных. Область разбивается на три суперэлемента, определяемые сегментами (1,2,10,8), (3,9,7,10) и (4,5,6,9). Для каждого сегмента задается число разбиений и код, используемый при решении исходной задачи. Ниже приведена структура входных данных, а на рис.4б — результат работы программы GRID8 — конечно-элементная сетка.

Авторы благодарят В.В.Парамонова и И.В.Гонина за полезные обсуждения.

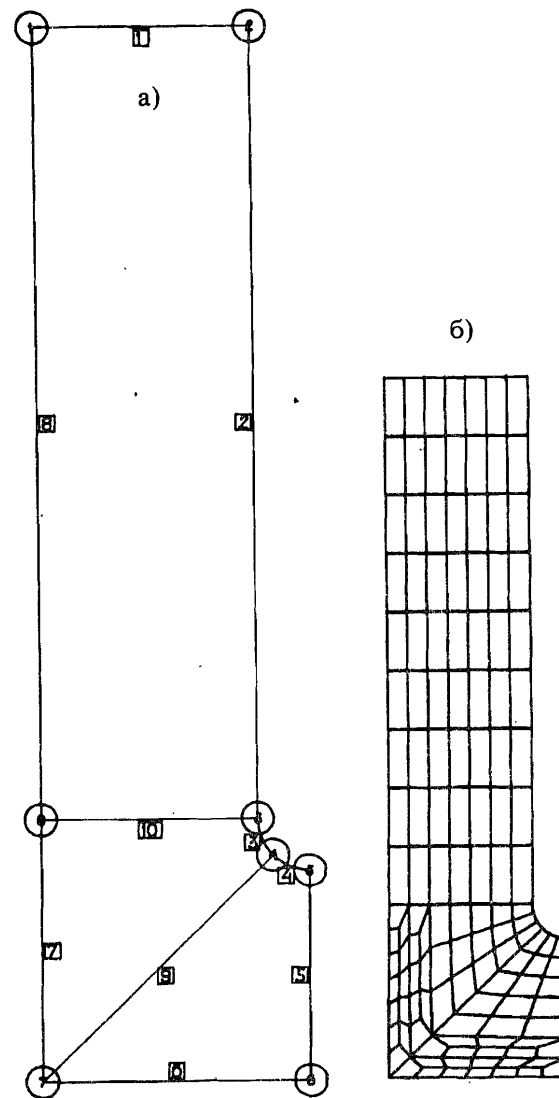


Рис. 4.

## ЛИТЕРАТУРА

1. PAFEC user guide. University of Nottingham, 1972.
2. Perronnet A. Gestion dynamique des tableaux. Structure des donnees. Modules. IPIA, 1976.
3. Стренг Г., Фикс Дж. Теория метода конечных элементов. М.: Мир, 1977.

4. Fedoseyev A.I. et al. – *NIM, A*, 1984, v.227, No.3, p.411.
5. Сегерлинд Л. *Применение метода конечных элементов*. М.: Мир, 1979.
6. Касчиева В.А. и др. *Препринт ИФВЭ-82-92, Серпухов, 1982.*
7. Касчиев М.С. и др. *Алгоритмы и алгоритмические языки. Пакеты прикладных программ*. М.: Наука, 1984, с.104.
8. Клименко С.В., Кочин В.Н. *Препринт ИФВЭ 82-9, Серпухов, 1982.*

Гусев В.В., Касчиев М.С., Пузынин В.И.

P11-87-421

**Автоматическая генерация сетки в пакете программ MULTIMODE**

Разработаны алгоритмы построения сетки в методе конечных элементов для восьмиузловых элементов. Приводится описание входных данных для программ, реализующих эти алгоритмы. Работа является одновременно и руководством для пользователей пакета программ MULTIMODE.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1987

Перевод авторов

Gusev V.V., Kaschiev M.S., Puzynin V.I.

P11-87-421

**Mesh Automatic Generation in MULTIMODE Program Packet**

Algorithms for construction of the mesh of eight-mode elements in the finite element method are developed. The description of input data for the programs is given. This paper can serve as instruction for users of MULTIMODE applied program packet.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1987

Рукопись поступила в издательский отдел  
15 июня 1987 года.