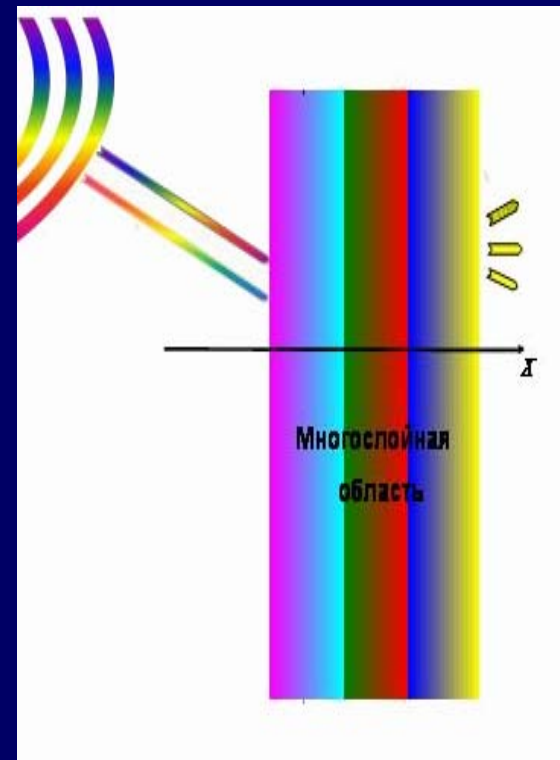



**Вопросы распараллеливания
алгоритмов решения задачи о
переносе радиационного излучения**

Е.Ф. Леликова, Л.И. Рубина, О.Н. Ульянов, М.А. Чащин

Физическая постановка задачи

Перенос радиационного излучения играет важную роль в процессах, происходящих в звездных атмосферах, плазме. Такие задачи возникают и в спектроскопии многозарядных ионов и некоторых других инженерных приложениях.



В области D имеется среда, содержащая смесь из Γ компонент (веществ), γ - номер компоненты. Пусть J - степень ионизации γ -компоненты, j - номер дискретного возбужденного состояния для данной степени ионизации. Относительная доля ионов вещества, пребывающего в состоянии (jJ) , называется населенностью этого состояния и обозначается через $c_{jJ}^{(\gamma)}$.

Математическая постановка задачи

Решается следующая система уравнений:

- Уравнения кинетики

$$A^{(\gamma)}(x)\vec{c}^{(\gamma)}(x) = \vec{b}^{(\gamma)}(x), \quad (1)$$

где $\vec{c}^{(\gamma)}(x) = (c^{(\gamma)}(x), \dots, c_n^{(\gamma)}(x))$, $\gamma = 1, \dots, \Gamma$

- Уравнение переноса в непрерывном спектре

$$\mu \frac{dI_c}{dx} = -\kappa_c(\vec{c}^{(\gamma)}(x), \varepsilon) I_c + S_c(\vec{c}^{(\gamma)}(x), \varepsilon), \quad (2)$$

$$I_c = I_c(x, \varepsilon, \mu), \quad \varepsilon \in E_c, \mu \in [-1, 1]$$

Уравнение решается на интервале $(0, L)$, при заданных условиях на границе:

$$I \Big|_{x=0, \mu > 0} = I_0(\mu, \varepsilon), I \Big|_{x=L, \mu < 0} = I_1(\mu, \varepsilon). \quad (2')$$

■ Уравнения переноса в дискретном спектре

$$\mu \frac{dI_k^{(\gamma)}}{dx} = -\kappa_k^{(\gamma)}(\vec{c}^{(\gamma)}(x), \varepsilon, \mu) I_k^{(\gamma)} + S_k^{(\gamma)}(\vec{c}^{(\gamma)}(x), \varepsilon, \mu), \quad k = 1, \dots, K, \quad (3)$$

$$I_k^{(\gamma)} = I_k^{(\gamma)}(x, \varepsilon, \mu), \quad \varepsilon \in E_k, \mu \in [-1, 1], \gamma = 1, \dots, \Gamma.$$

с аналогичными условиями на границе .

А также известны формулы, связывающие соотношения (1),(2),(3):

$$a_{ij}^{(\gamma)}(x) = a_{ij}^{(\gamma)}\left(J_c^{(\gamma)}(x), J_k^{(\gamma)}(x)\right), \quad J_c^{(\gamma)}(x) = \left(\int_{-1}^1 d\mu \int_{\varepsilon \in E_c} I_c(x, \mu, \varepsilon) \Phi_c^{(\gamma)}(\varepsilon) d\varepsilon \right) \quad (4)$$

$$J_k^{(\gamma)}(x) = \left(\int_{-1}^1 d\mu \int_{\varepsilon \in D_{ij}} I_k^{(\gamma)}(x, \mu, \varepsilon) \Phi_{ij}^{(\gamma)}(\varepsilon) d\varepsilon \right), \quad k = 1, \dots, K, \gamma = 1, \dots, \Gamma.$$



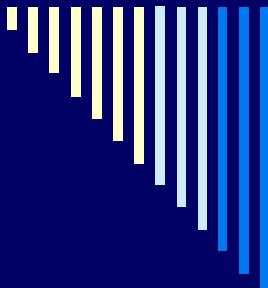
Особенности моделирования

Математические трудности - нерегулярная зависимость рассматриваемых величин от энергетической переменной, δ -образное поведение некоторых функций, наличие пограничных слоев и разрывов у ряда функций, разномасштабность изучаемых величин, сложное взаимодействие резонансных линий, нелокальность, которая связана с нелокальным характером взаимодействия излучения с веществом, нелинейность задачи.

Вычислительные проблемы - неустойчивость, плохая сходимость, необходимость использовать подробные сетки, хранить большие массивы и производить большие объемы вычислений.

История создания и этапы развития методик





Математические алгоритмы решения задачи

- [Алгоритм МАПИ](#)
- [Алгоритм МПЛЧ](#)



Методы решения

- Метод простой итерации
-

Этапы распараллеливания алгоритма МАПИ

<i>Модель распараллеливания</i>	<i>машина</i>	<i>Число процессоров</i>	<i>Физическая модель</i>
Распараллеливание по одной переменной (по переменной $N_x < 30$)	Мвс-100	8-64	Однослойная
Распараллеливание по одной переменной (по переменной $N_x < 60$ в нескольких областях)	Мвс-100 Alpha	8-128	Многослойная, подвижная.
Распараллеливание по двум переменным (по $N_x < 120$ и по $\Gamma < 5$)	мвс-1000м мвс- 1000/16	16-624	Многослойная, подвижная, многокомпонентная.
Распараллеливание по трем переменным (по N_x, Γ, K) ($\Gamma < 5, K < 40, N_x < 120$)	мвс-1000м мвс- 1000/16	16-768	Многослойная, подвижная, неоднородная, многокомпонентная.

Этапы распараллеливания алгоритма МПЛЧ

<i>Модель распараллеливания</i>	<i>машина</i>	<i>Система программ ирования</i>	<i>Физическая модель</i>
непрерывный спектр и дискретный спектр	Мвс-100	Router	Однослойная, однокомпонентная, неподвижная среда
по числу спектральных линий (до 39 линий)	Мвс-100 мвс-1000	Mpi и router	Многослойная, однокомпонентная, подвижная среда
по группе спектральных линий (до 39 линий)	мвс-1000м мвс- 1000\16	Router, DVM	Многослойная, однокомпонентная, подвижная среда
по группе спектральных линий с вторым уровнем	мвс-1000м мвс- 1000\16	DVM	Многослойная (псевдослойная), подвижная, однокомпонентная среда
по группе спектральных линий с вторым уровнем и директивой CONSISTENT	мвс-1000м мвс- 1000\16	DVM	Многослойная (псевдослойная) подвижная, однокомпонентная среда



Алгоритмы распараллеливания (схемы)

Программы МАПИ

[вариант I \(II\)](#)

[вариант III](#)

[вариант IV](#)

Программы МПЛЧ
(router)

- [вариант I](#)
- [вариант II](#)
- [вариант III](#)

Блок-схема
последовательной
программы МПЛЧ

Программы МПЛЧ
(DVM)

- [Вариант 1](#)
- [Вариант 2](#)
- [Вариант 3](#)
- [Вариант 4](#)

Основная программа

PROGRAM LAZDTASK

CDVM\$ TASK TASKDK(MAXDUK)

Описание массива задач

CDVM\$ PROCESSORS PROC (NUMBER_OF_PROCESSORS())

Описание массива виртуальных проц-ров.

NUMBER_OF_PROCESSORS()=1

Для того, чтобы программа выполнялась (как последовательная) вне среды DVM, необходимо описать операторную функцию

nump=NUMBER_OF_PROCESSORS()

.....
CDVM\$ MAP TASKDK(i) ONTO PROC(j:j+ndk)

Отображение задачи на секцию массива процессоров

.....
CDVM\$ INTERVAL 1

С ОСНОВНОЙ ИТЕРАЦИОННЫЙ ЦИКЛ

100 NIT=NIT+1

.....
CALL DCWR(C,VC, RN1,RN2,PZZN,...)

.....
CALL DKDUC(C,VC,...)

CDVM\$ TASK_REGION TASKDK, REDUCTION(SUM(pzzn),SUM(rn1),SUM(rn2)) Распределение вычислений (витков параллельного цикла) по задачам

CDVM\$ PARALLEL (jlx) ON TASKDK(jlx)

Распределение витков параллельного цикла по процессорам задачи

DO 2007 jlx = 1, LD

.....
CALL DKDUK(C,...,RR1,RR2,PZZ,...)

DO 2004 ix=1,NX*nchx

PZZN(jlx, ix)=PZZ(ix)

DO 2005 nper=1,111

RN1(nper, jlx, ix)=RR1(nper, ix)

RN2(nper, jlx, ix)=RR2(nper, ix)

2005 CONTINUE

2004 CONTINUE

2007 CONTINUE

CDVM\$ END TASK_REGION

CDVM\$ ENDINTERVAL

IF (NIT.NE.NOIT) GOTO 100

STOP

Подпрограмма DKDUK на Fortran-DVM

SUBROUTINE DKDUK(C,....,rr1,rr2,pzz,....)

.....
CDVM\$ DISTRIBUTE (BLOCK):: TT

Распределяется шаблон TT
равными блокам

CDVM\$ TEMPLATE TT(N5M)

Описание шаблона.
Выравнивание массивов.

CDVM\$ PARALLEL (IM) ON TT(IM)

Цикл распараллелить в соответствии с
распределением шаблона TT.

DO 4 IM=1,N5M

VK(*,*,IM)=...

интенсивность от угла, энергии и от X.

4 CONTINUE

CDVM\$ PARALLEL (IM) ON TT(IM)

Цикл распараллелить в соответствии с
распределением шаблона TT.

DO 444 IM=n2m,n2m

VK(*,*,IM)=.....

444 CONTINUE

CDVM\$ PARALLEL (IM) ON TT(IM),REDUCTION(SUM(pzz),SUM(rr1),SUM(rr2))

Выполнить
редукцию суммирования по массивам pzz, rn1, rn2 .

DO 350 IM=2,N5M

F(IM)=...

DO ix=1,nx

интеграл по энергии

.....
pzz(ix)=pzz(ix)+A(IM)*F(IM)

интеграл по энергии и
углу в точке X

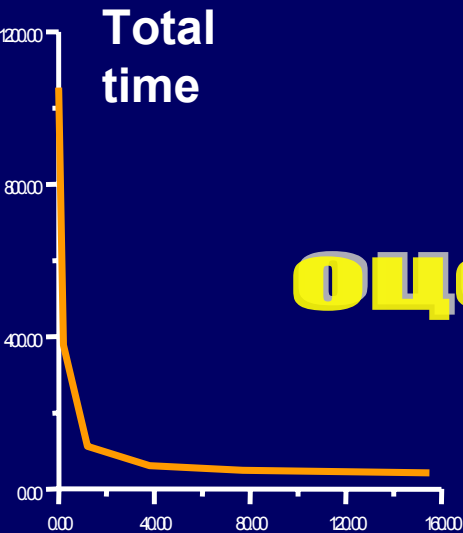
ENDDO

350 CONTINUE

RETURN

END

Определяющие факторы эффективности



можно ли быстро и эффективно оценить параллелизм?

можно ли сократить время потерь на коммуникациях?

можно ли распараллелить глубже и т.д.?

где потеряли?

НУЖНО ЛИ РАСПАРАЛЛЕЛИВАТЬ?

**ДО КАКОЙ СТЕПЕНИ
РАСПАРАЛЛЕЛИВАТЬ?**

**ДО КАКОЙ СТЕПЕНИ
УВЕЛИЧИВАТЬ
ЧИСЛО ПРОЦЕССОРОВ?**

Наши наблюдения и DVM-распараллеливание:

- Распараллеливаемость алгоритмов.
 - a) Предпочтение хорошо распараллеливаемому алгоритму?
 - b) Существует ли алгоритм?
 - c) Существуют ли средства распараллеливания?
- Приемлемость времени расчетов.
- Произвольность числа процессоров.
- Оптимальность числа процессоров.
- Выбор системы программирования.
 - a) Разработка программ на языках C - DVM и Fortran - DVM
 - b) Мобильность DVM- программ
 - c) Эффективность выполнения программ
 - d) Повторное использование программ

Программный интерфейс

(Off line)

- Основная панель
- Сетки по x
- Сетки по y
- Сетки по z
- Подсистема запуска
- Подсистема визуализации

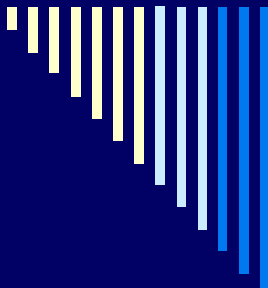
Для комплекса программ, реализующих методики МАПИ и МПЛЧ для многопроцессорных вычислительных систем и персональных компьютеров, создан единый пользовательский интерфейс управления расчетами, охватывающий

Задание физических параметров задачи, формирование расчетных сеток, запуск вычислений на различных платформах и визуализацию результатов.





спасибо за внимание



Определяющие факторы эффективности выполнения параллельных программ:

- степень распараллеливания программы - доля параллельных вычислений в полном объеме вычислений;
 - равномерность загрузки процессоров во время выполнения параллельных вычислений;
 - время, необходимое для выполнения межпроцессорных обменов;
 - степень совмещения межпроцессорных обменов с вычислениями.
-



Мобильность DVM-программ

- Среди последовательных ЭВМ мобильность DVM-программ обеспечивается “прозрачностью” DVM-указаний для стандартных компиляторов с языков Си и Фортран 77.
 - Компиляторы с языков C-DVM и Fortran DVM преобразуют DVM-указания в вызовы функций единой системы поддержки Lib-DVM, которая реализована на языке Си и базируется на стандартных коммуникационных библиотеках MPI, Router и PVM. Поэтому система поддержки переносима на любые ЭВМ, для которых существует реализация MPI или PVM.
 - Мобильность DVM-программ на параллельных ЭВМ обеспечивается вышеуказанной схемой реализации и машинно-независимой DVM-моделью. Кроме того, программа на языке Fortran-DVM может автоматически конвертироваться в программы на языках HPF1 или HPF2.
-

□ При использовании этих языков программист имеет только один вариант программы для последовательного и параллельного выполнения. Он освобождается от необходимости представлять свою программу в виде множества взаимодействующих процессов, а определяет поведение параллельной программы в глобальном пространстве имен (едином адресном пространстве).

□ Важным достоинством DVM-подхода является то, что спецификации параллелизма (DVM-указания) оформляются в виде специальных комментариев и остаются “невидимыми” для стандартных компиляторов.

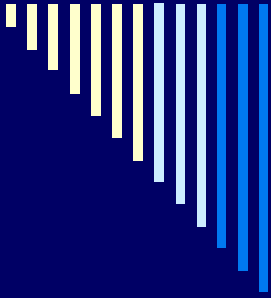
Во-первых, это упрощает внедрение новых параллельных языков, поскольку программист знает, что его программа без каких-либо изменений может выполняться в последовательном режиме на любых ЭВМ.

Во-вторых, это позволило предложить следующий метод поэтапной отладки DVM-программ.

□ На первом этапе программа отлаживается на рабочей станции как последовательная программа, используя обычные методы и средства отладки. На втором этапе программа выполняется на той же рабочей станции в специальном режиме проверки DVM-указаний. На третьем этапе программа может быть выполнена на параллельной машине (или ее модели, например MPI-машине в среде Windows95 или UNIX) в специальном режиме, когда промежуточные результаты параллельного выполнения сравниваются с эталонными результатами (например, результатами последовательного выполнения).

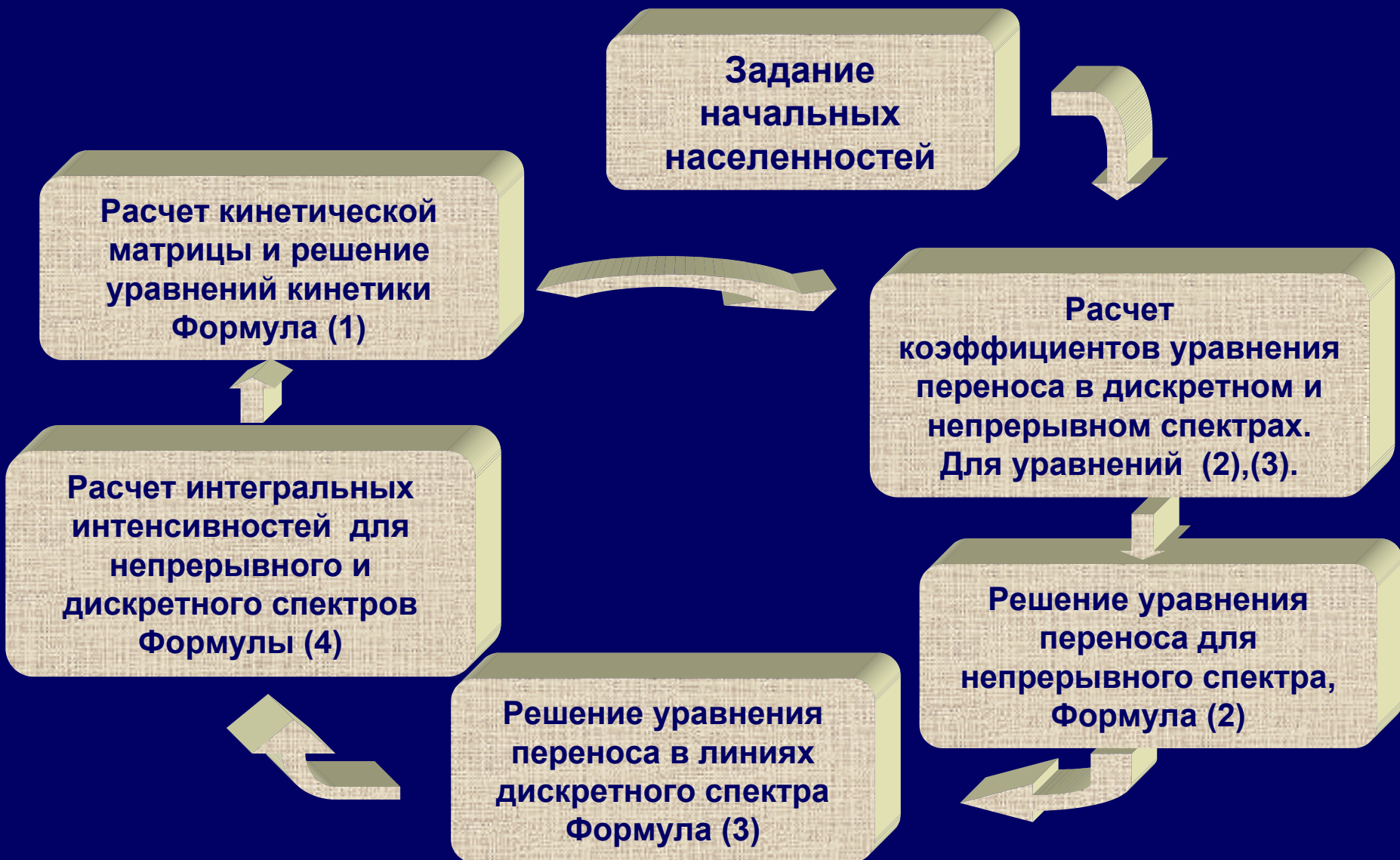
Существуют следующие возможности повышения эффективности DVM-программ:

- использование групповых асинхронных взаимодействий процессоров (одновременное выполнение нескольких редуций и обменов для нескольких массивов);
- возможность перекрытия простоев процессоров при групповых асинхронных взаимодействиях с вычислениями;
- автоматическое изменение порядка выполнения витков цикла для опережающих вычислений и рассылок данных.



- DVM-программа динамически настраивается на параметры приложения (количество и размер массивов данных).
- DVM-процедура динамически настраивается на распределение фактических параметров и подмножество процессоров параллельной ЭВМ.

● **Метод простой итерации для решения уравнений (1),(2),(3).**



Метод итераций

Задание начальных населенностей



Расчет кинетической матрицы и решение уравнений кинетики

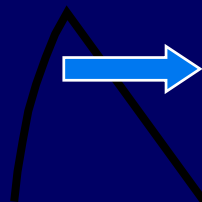
Расчет коэффициентов уравнения переноса в дискретном и непрерывном спектрах



Расчет интегральных интенсивностей для непрерывного и дискретного спектров

Решение уравнения переноса для непрерывного спектра

Решение уравнения переноса в линиях дискретного спектра



Базовая модель

два алгоритма (МАПИ и МПЛЧ)

- Однослойность
- Однородность по плотности и температуре
- Неподвижность среды
- Модель кинетики содержит две спектральные линии
- Однокомпонентность
- Последовательность программ

Вычислительные средства

- Бэсм-6, ЕС
- РС
- МВС (МВС-100,МВС-1000,МВС-000/16,
■ МВС-1000М)

Схема программы МПЛЧ (вариант dvm)

К- число резонансных линий

Одно-
временный
запуск

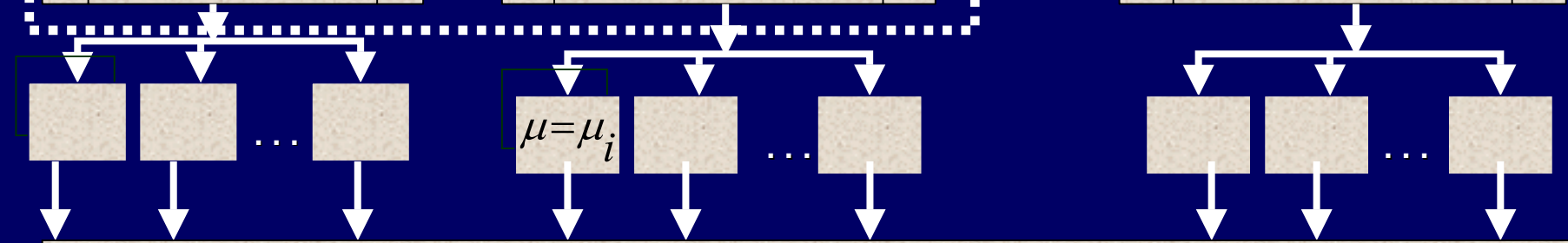
Процессор 1

Процессор К

Решение уравнений кинетики
Решение уравнений переноса для непр. спектра
Решение уравнений переноса для дискр. спектра при $k=1$
Расчеты интегралов.

Решение уравнений кинетики
Решение уравнений переноса для непр. спектра
.....
Решение уравнений переноса для дискр. спектра при $k=2$
Расчеты интегралов.

Решение уравнений кинетики
Решение уравнений переноса для непр. спектра
.....
Решение уравнений переноса для дискр. спектра при $k=K$
Расчеты интегралов.



Обмен и синхронизация по итерациям

$RR1(*,*)$, $RR2(*,*)$, $PZZ(*)$

Схема программы МПЛЧ (вариант dvm)

К- число резонансных линий

Одно-
временный
запуск

Процессор 1

Процессор К

Решение
уравнений кинетики

Решение уравнений
переноса для непр.
спектра

Решение уравнений
переноса для дискр.
спектра при $k=1$
Расчеты интегралов.

Решение
уравнений кинетики

Решение уравнений
переноса для непр.
спектра

Решение уравнений
переноса для дискр.
спектра при $k=2$
Расчеты интегралов.

Решение
уравнений кинетики

Решение уравнений
переноса для непр.
спектра

Решение уравнений
переноса для дискр.
спектра при $k=K$
Расчеты интегралов.

$$\mu = \mu_i$$

Обмен и синхронизация (опция consistent) $RR1(*,*)$, $RR2(*,*)$, $PZZ(*)$

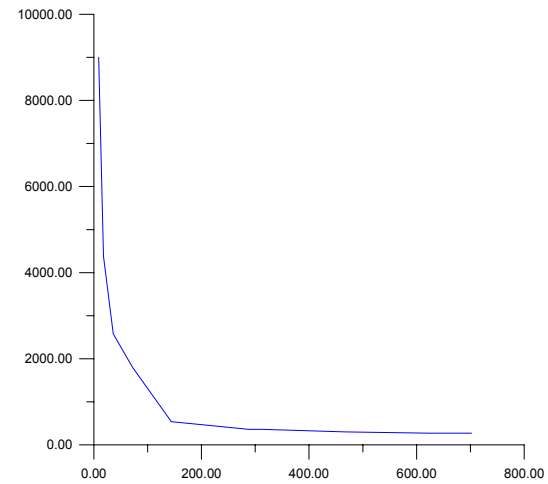
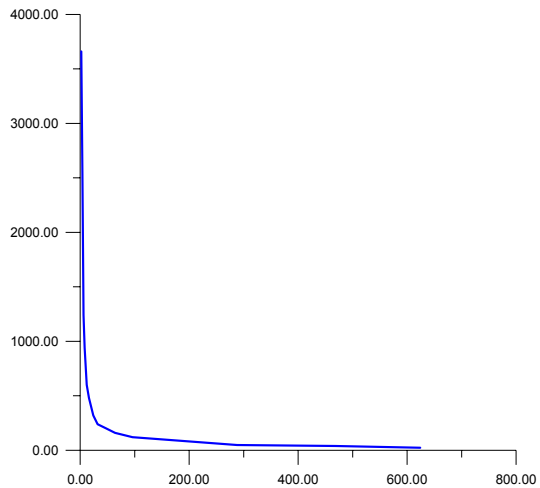
Схема программы МПЛЧ (вариант dvm)



Схема программы МПЛЧ (вариант dvm)



МВС-1000М



5 подобластей, 39 линий, число точек в сетке по x 1x=96

Одно вещество в смеси

Три вещества в смеси

Число пр.	Время	Число пр.	Время
2	3660		
4	2540		
6	1240		
8	940	9	9000
12	600		
16	480	18	4380
24	320		
32	240	36	2580
48	200		
64	160	72	1800
96	120	144	540
288	48	288	360
468	40	468	300
624	24	624	270

- Распараллеливаемость алгоритмов.
 - a. Выбор математического алгоритма
 - b. Анализ распараллеливаемости алгоритма
 - c. Выбор оптимальной схемы распараллеливания
- Произвольность числа процессоров.
- Независимость алгоритма от топологии вычислительной системы.
- Минимальность передаваемой информации при удаленном доступе.
- Выбор системы программирования

3 процессора

13 процессоров

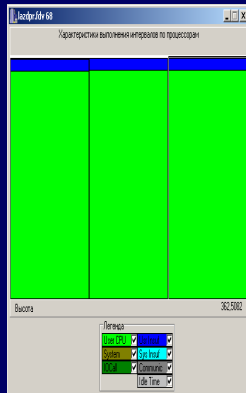
39 процессоров

78 процессоров

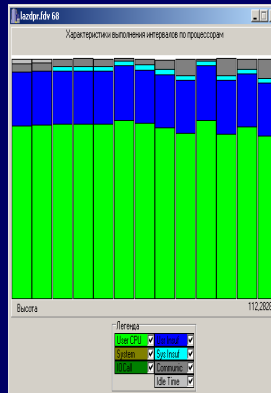
156 процессоров

версия 382

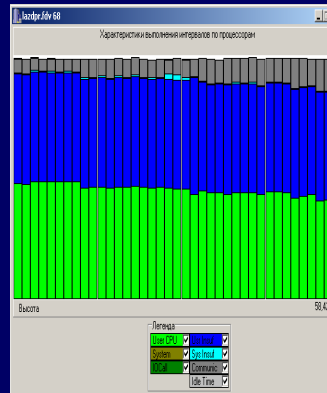
362 сек



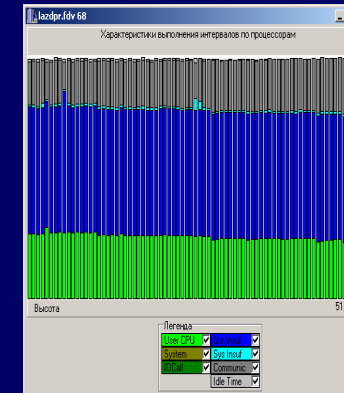
112 сек



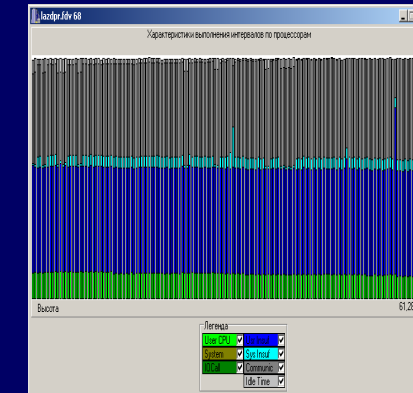
58 сек



51 сек

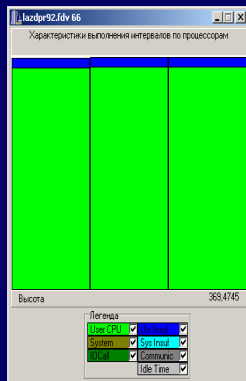


61 сек

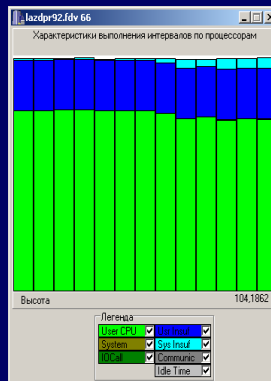


версия 392

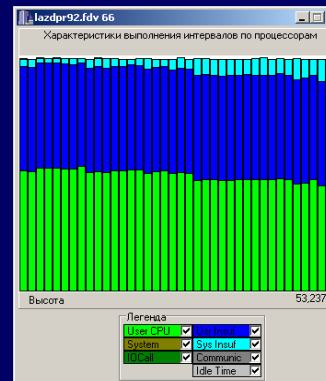
369 сек



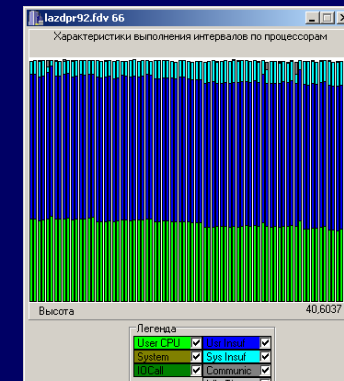
104 сек



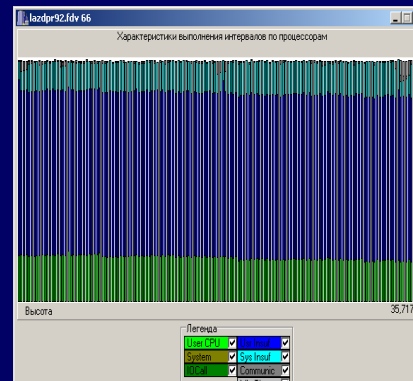
53 сек



40 сек



35 сек



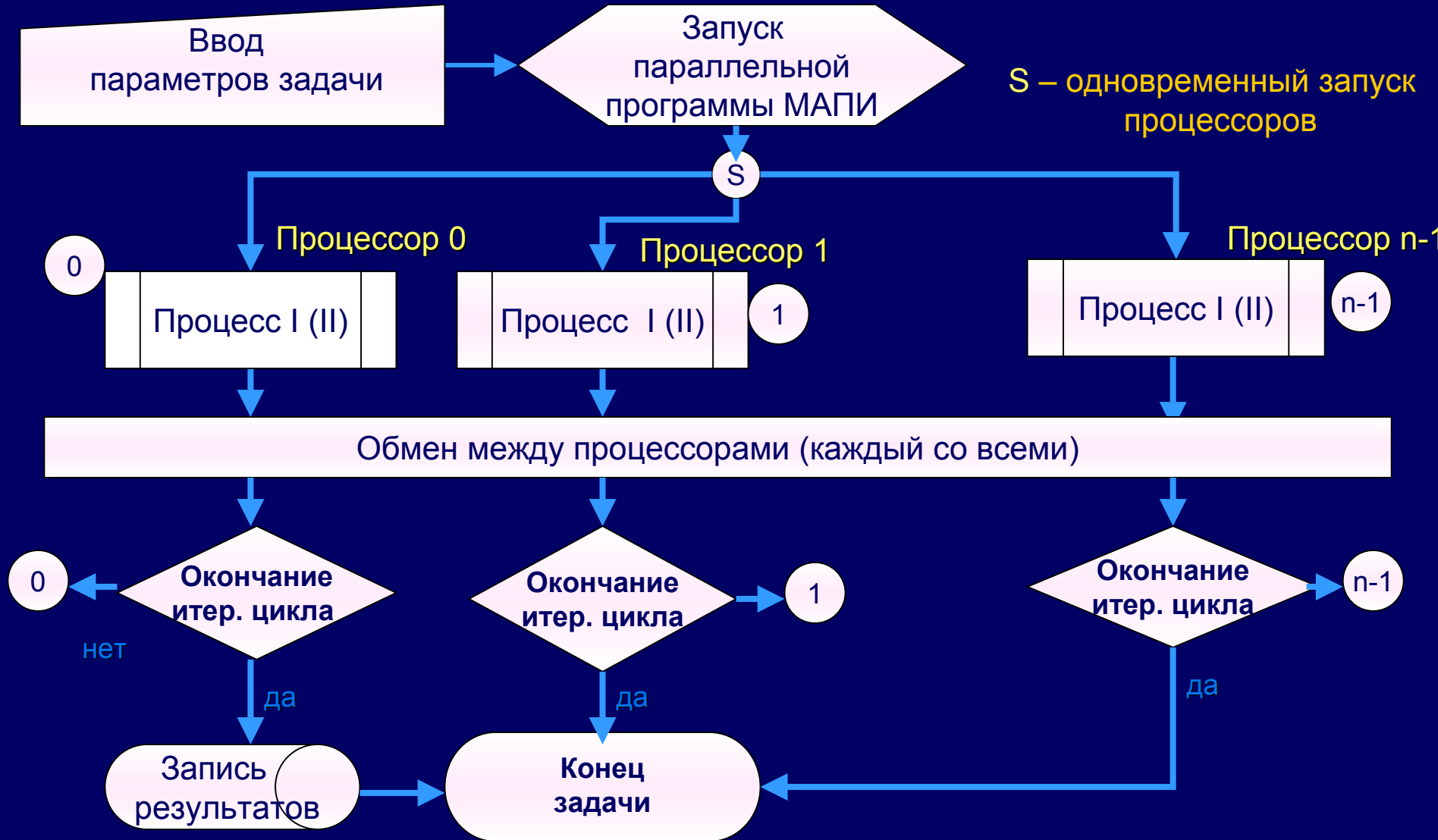
• Полезное процессорное время (User CPU_time).

• Время потерь на коммуникациях (Communication).

• Потери из-за недостаточного параллелизма (User insufficient).

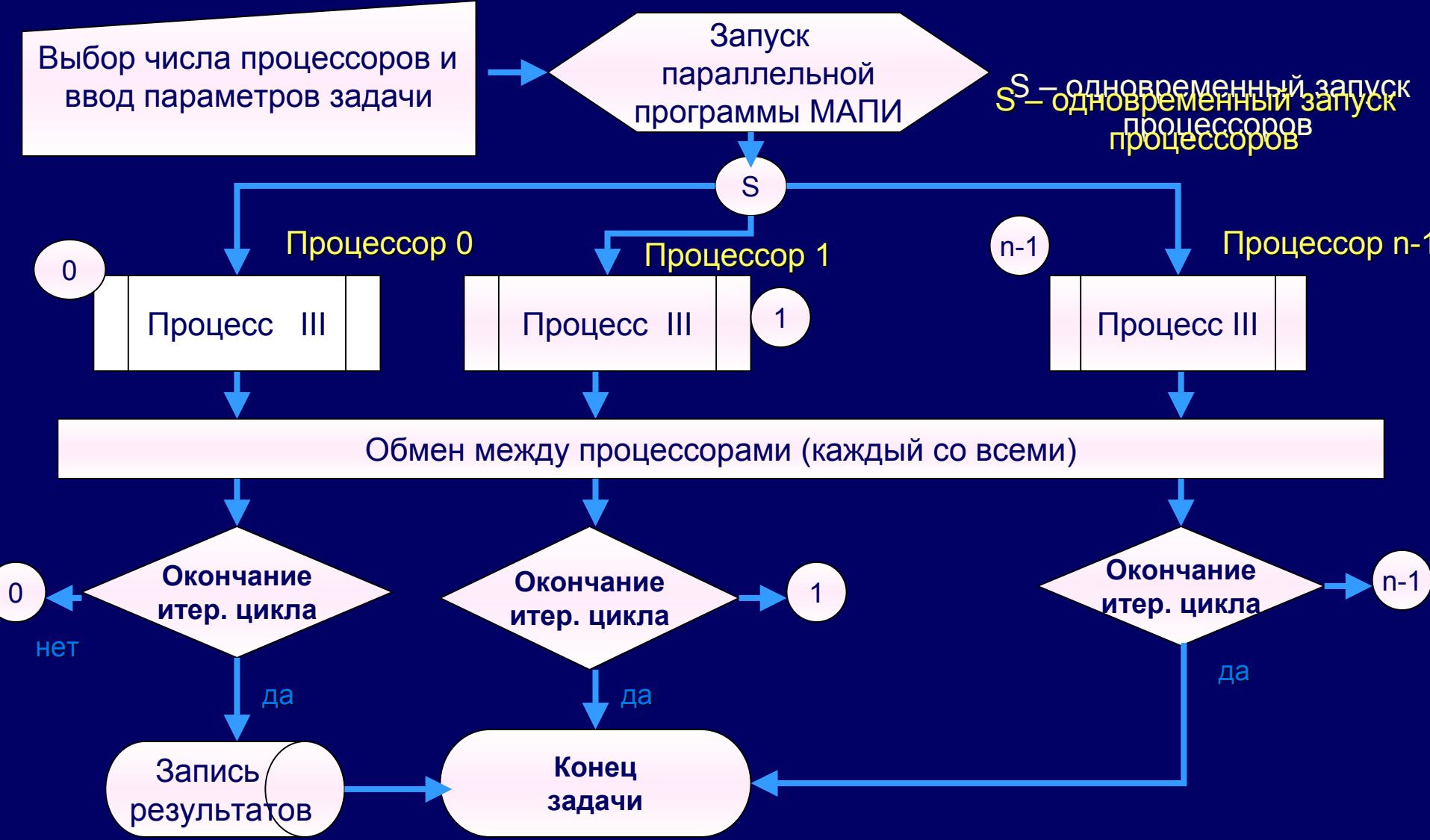
• Потери из-за недостаточного параллелизма (System insufficient).

Схема программы МАПИ, вариант I (II)



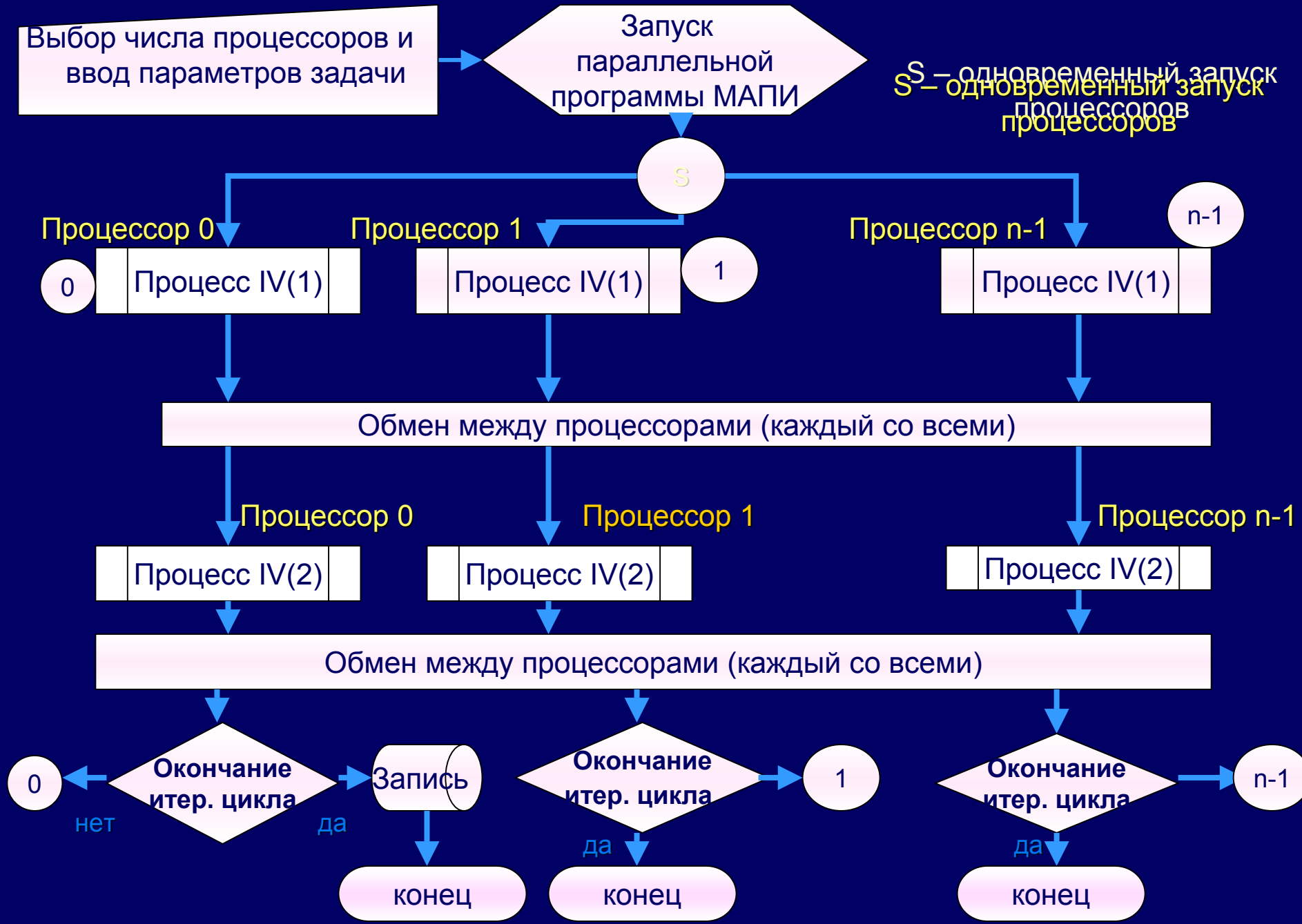
Вариант I: $n = N_x$, N_x – число точек в сетке по x
Вариант II: $n = N_x * \Gamma$, Γ – число компонент в смеси

Схема программы МАПИ, вариант II

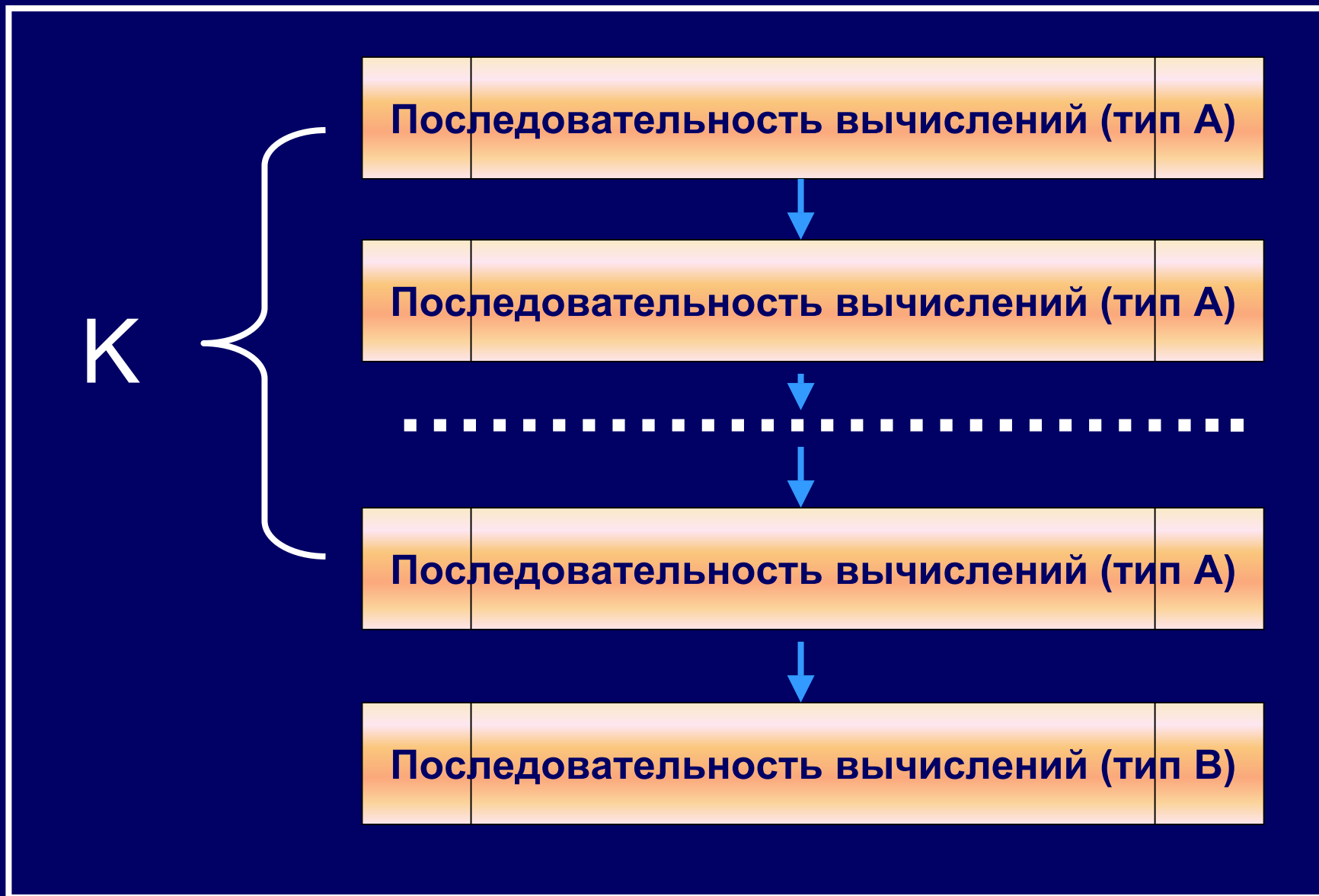


Вариант III: n – выбранное число процессоров

Схема программы МАПИ, вариант IV

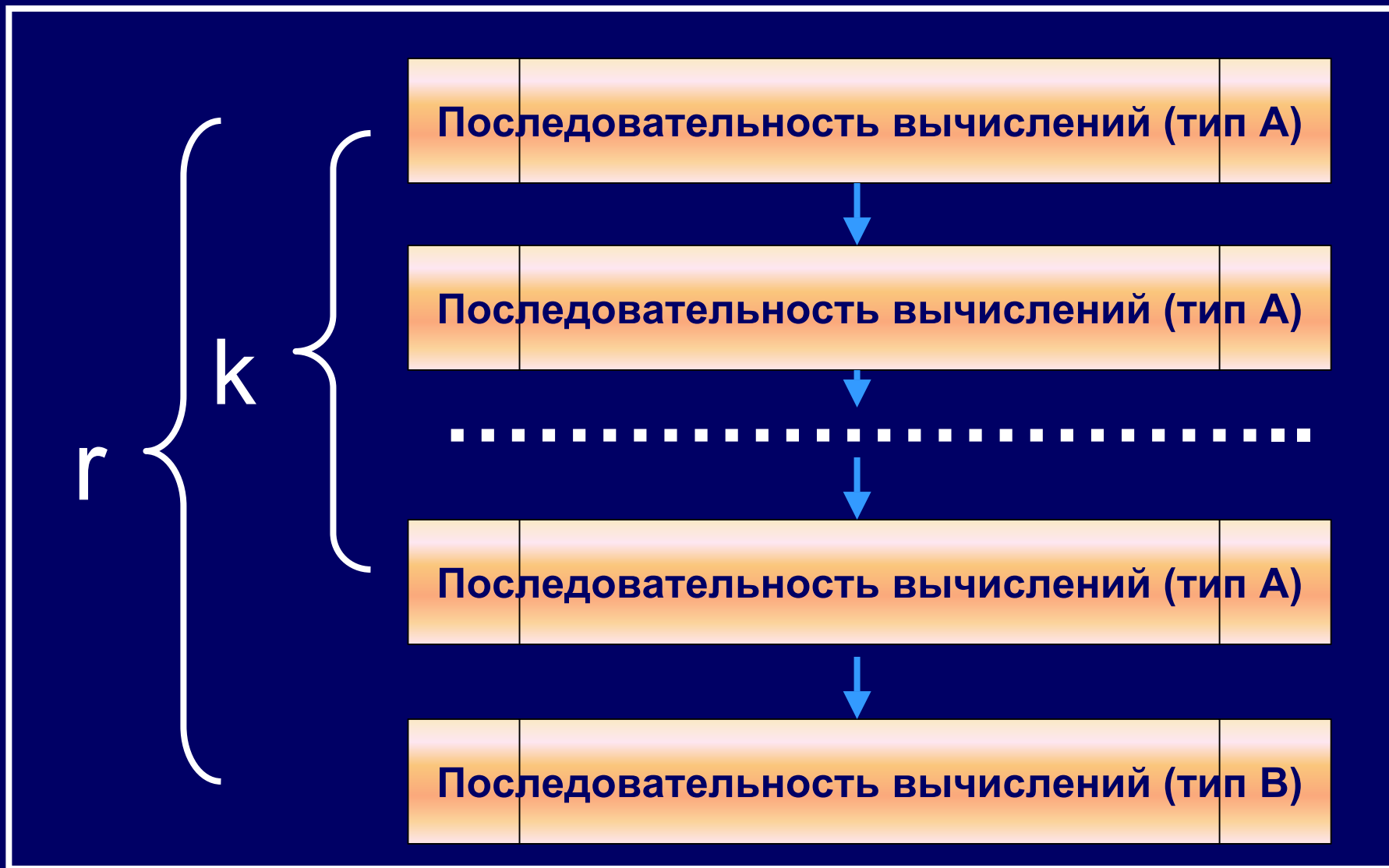


Процесс I (II)



К – число резонансных линий

Процесс III



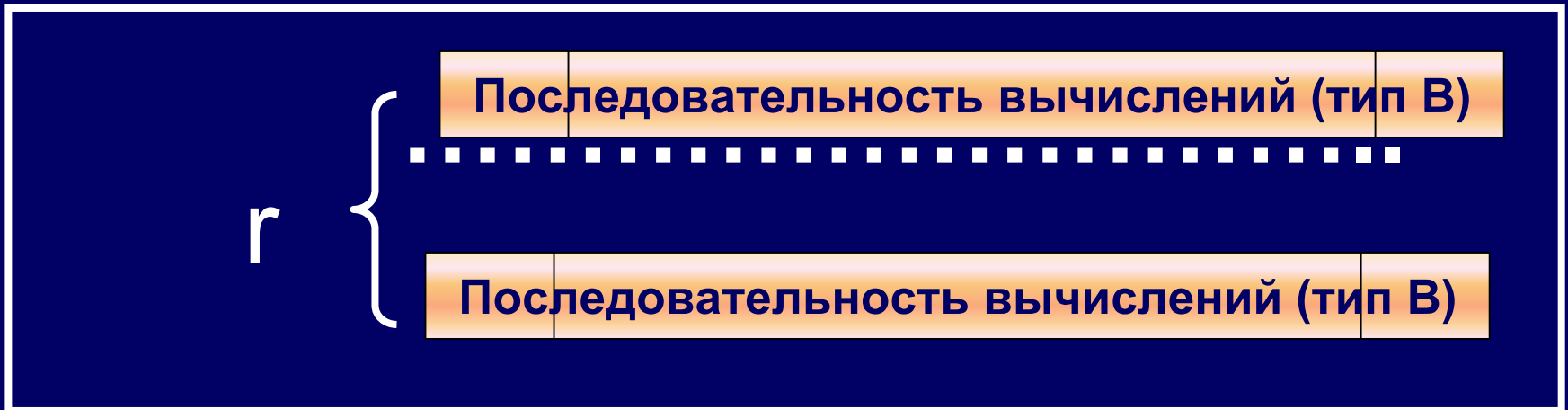
k – число резонансных линий, $r = (N_v * \Gamma) / n$

Процесс IV (1)



$$m = (N_x * \Gamma * k) / n$$

Процесс IV (2)



$$r = (N_x * \Gamma) / n$$

Схема программы МПЛЧ (вариант I)

$n=K$ - число резонансных линий



Схема программы МПЛЧ (вариант II)

$n=K$ - число резонансных линий



Схема программы МПЛЧ (вариант III)

$n=K$ - число резонансных линий



Алгоритм МПЛЧ основан на представлении искомой функции в виде интерполяционного полинома Лагранжа по узлам интерполяции - корням многочлена Чебышева. Отрезок $[0, L]$ разбивается на m интервалов $[0, x_1], [x_1, x_2], \dots, [x_{m-1}, L]$ по специальным схемам

и на каждом из интервалов $[x_{k-1}, x_k]$ решение $I = I(x, \varepsilon, \mu)$ ищется в виде интерполяционного многочлена Лагранжа степени N_k :

$$\sum_{i=1}^{N_k} L_{ki}(x) I_{ik}.$$

Здесь

$$L_{ki}(x) = \prod_{j=1, j \neq i}^{N_k} \frac{x - x_j^{(k)}}{x_i^{(k)} - x_j^{(k)}}$$

-множитель Лагранжа, построенный по узлам интервала (x_{k-1}, x_k) , $L_{ki}(x_i^{(k)}) = 1, L_{ki}(x_j^{(k)}) = 0$ при $i \neq j$,

$I_{ik} = I(x_i^{(k)}, \varepsilon, \mu)$. Для определения неизвестных коэффициентов I_{ik} получается система линейных уравнений с матрицей специального вида.

Алгоритм МАПИ основан на явном представлении решения уравнения переноса. Отрезок $[0, L]$ разбивается на m интервалов: $[0, x_1], [x_1, x_2], \dots, [x_{m-1}, L]$,

и на каждом из интервалов решается уравнение переноса.

Если обозначить $I_i = I(x_i, \varepsilon, \mu)$, то при $\mu < 0$

$$I_i = I_{i-1} \exp \left\{ \frac{-1}{\mu} \int_{x_{i-1}}^{x_i} \kappa(\xi, \varepsilon, \mu) d\xi \right\} +$$

$$\frac{1}{\mu} \int_{x_{i-1}}^{x_i} S(\xi, \varepsilon, \mu) \exp \left\{ \frac{1}{\mu} \int_{\xi}^{x_i} \kappa(\omega, \varepsilon, \mu) d\omega \right\} d\xi.$$

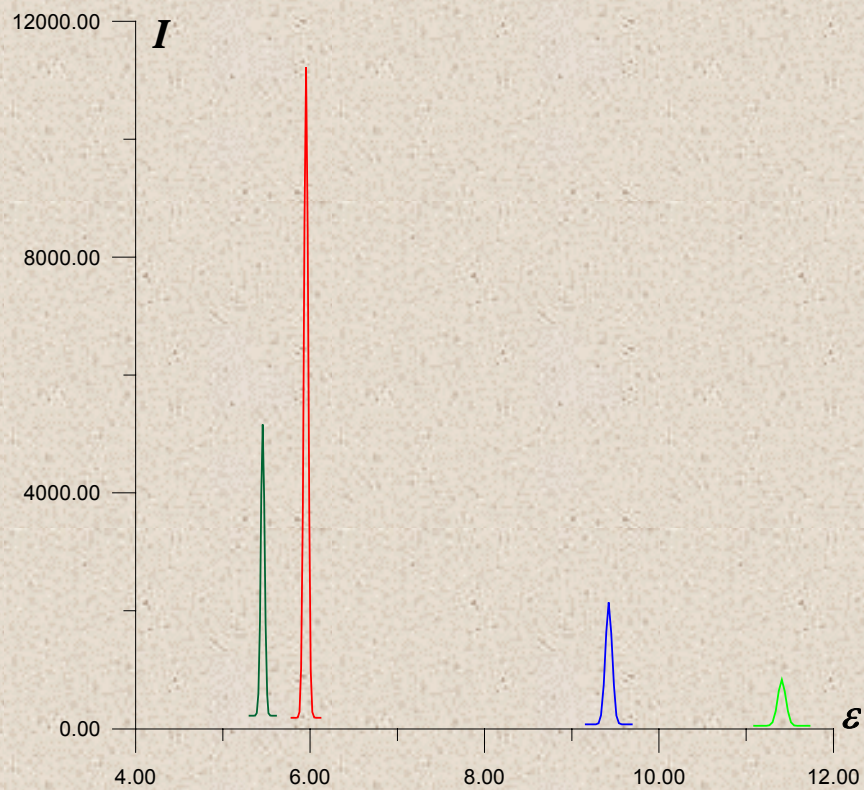
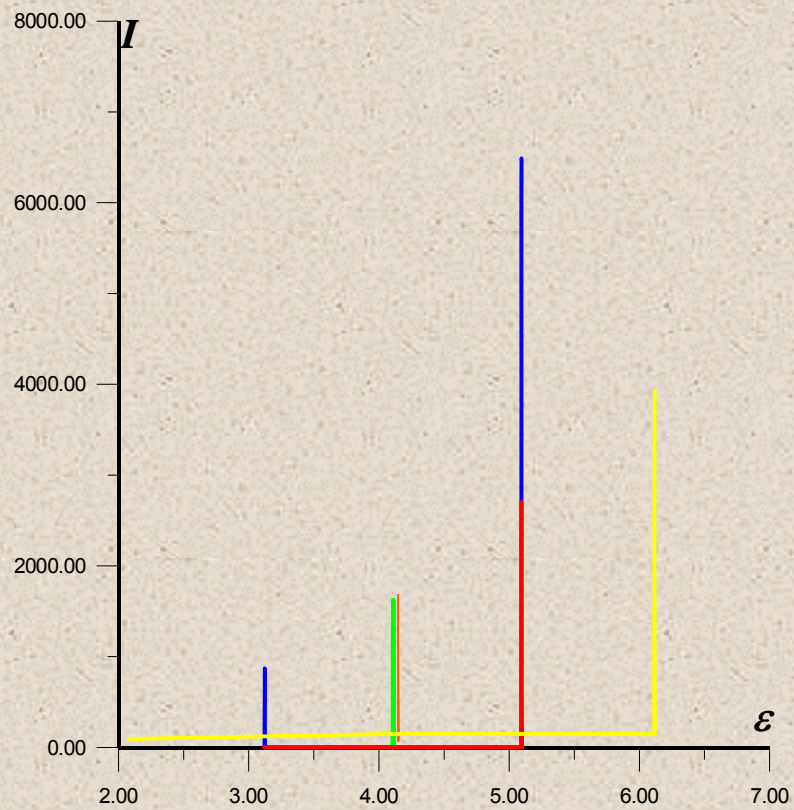
При $\mu > 0$ аналогично .

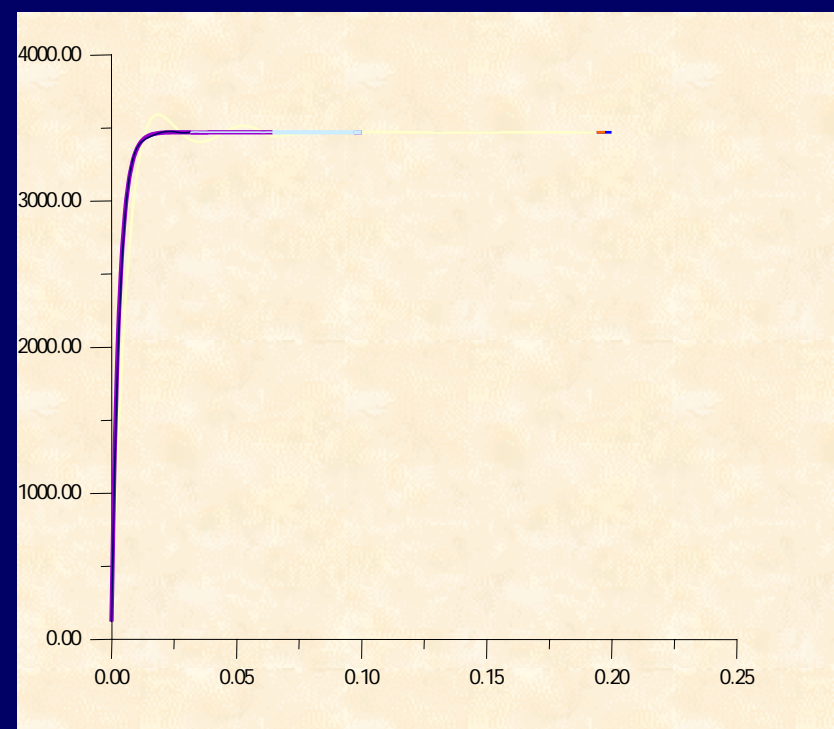
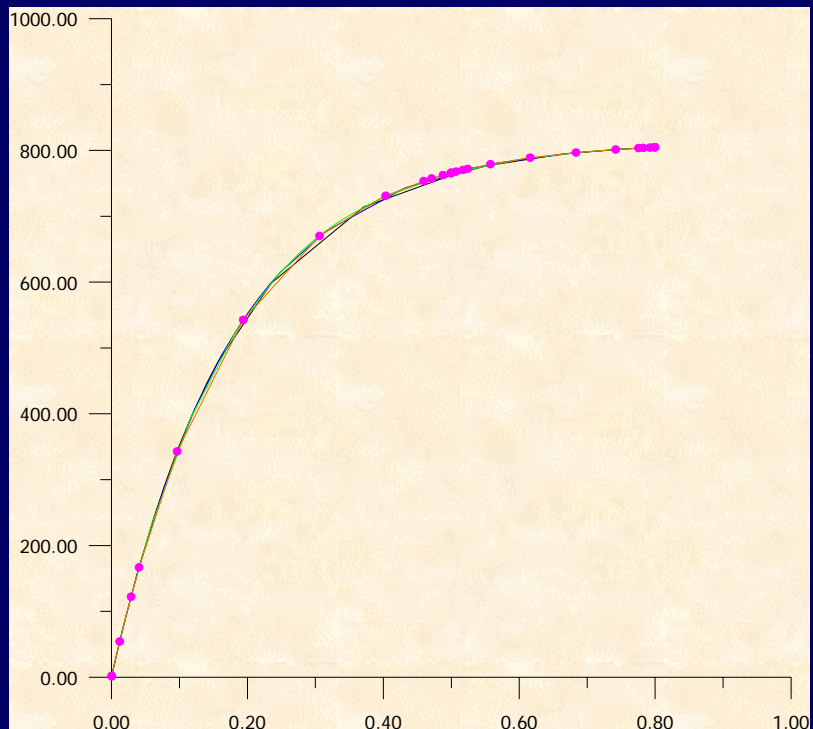
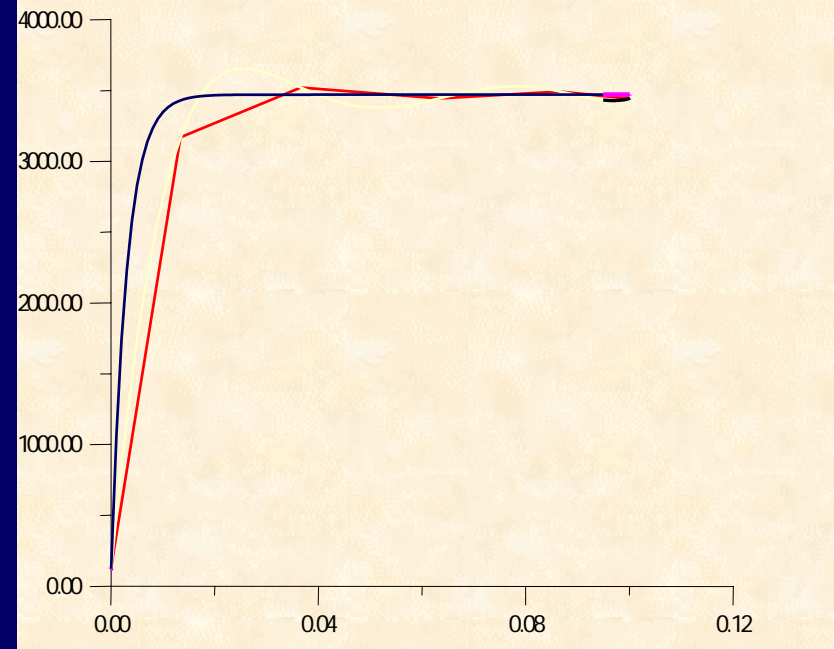
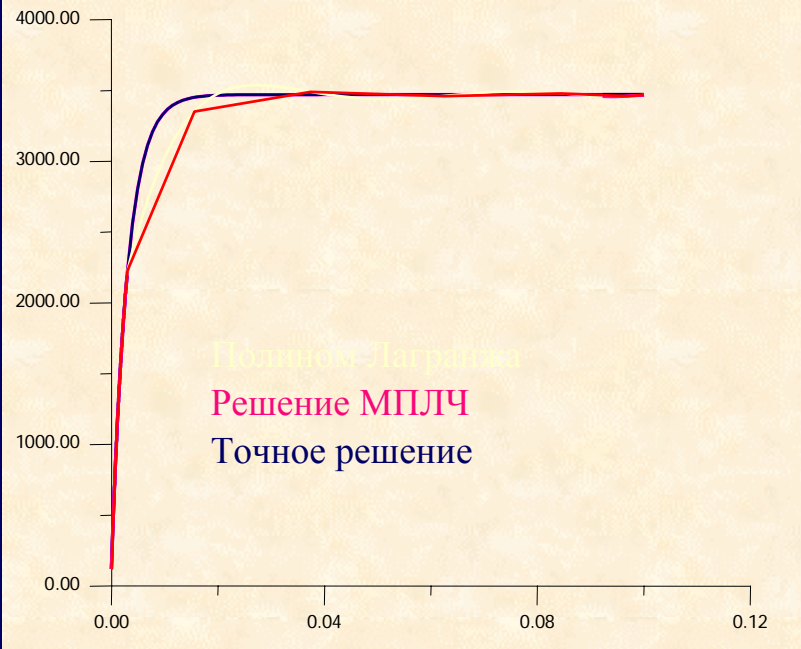
Блок схема (Последовательная программа)



Эффективность выполнения параллельных программ на многопроцессорных ЭВМ с распределенной памятью определяется следующими основными факторами:

- степенью распараллеливания программы - долей параллельных вычислений в полном объеме вычислений;**
- равномерностью загрузки процессоров во время выполнения параллельных вычислений;**
- временем, необходимым для выполнения межпроцессорных обменов;**
- степенью совмещения межпроцессорных обменов с вычислениями.**





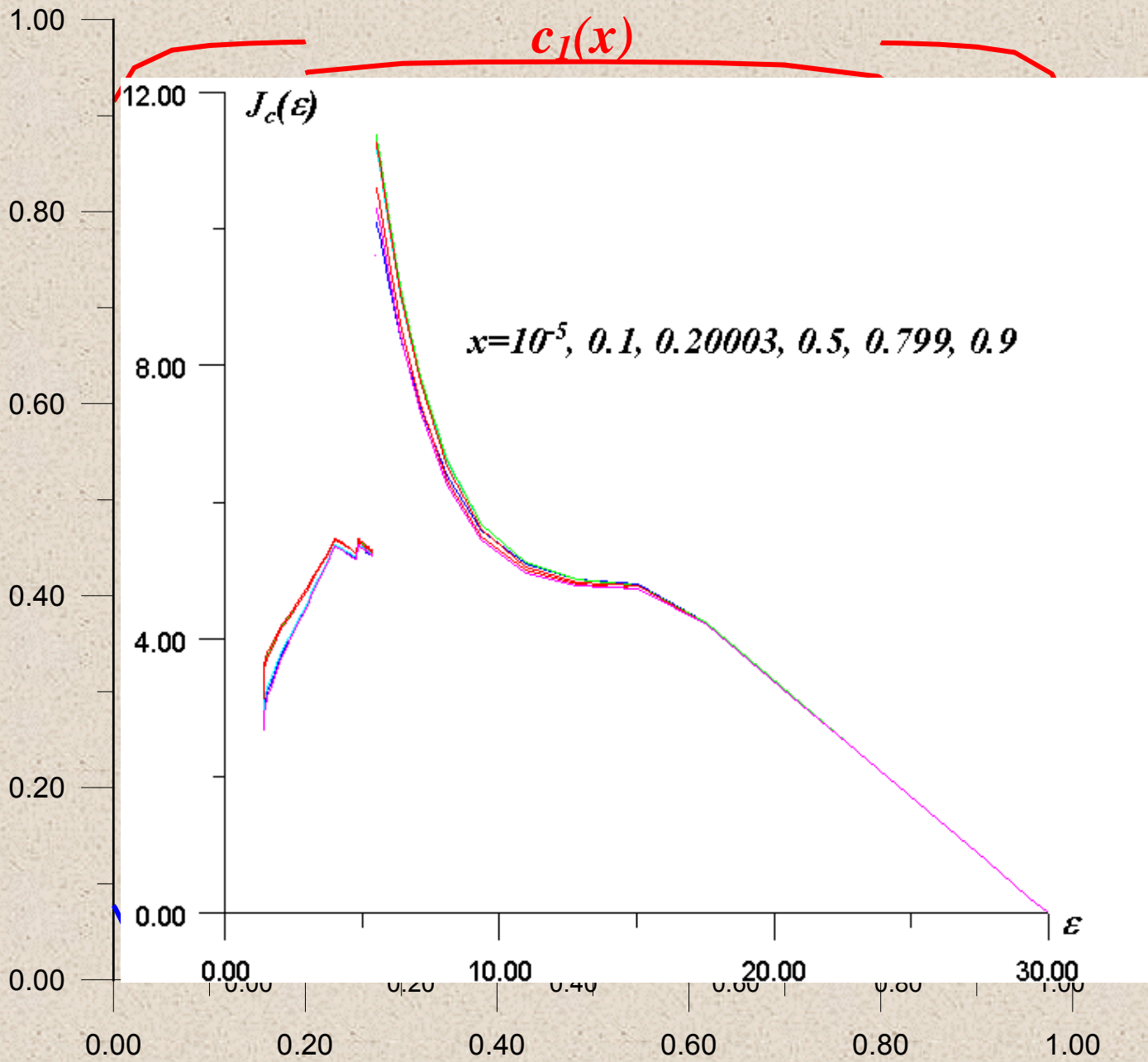
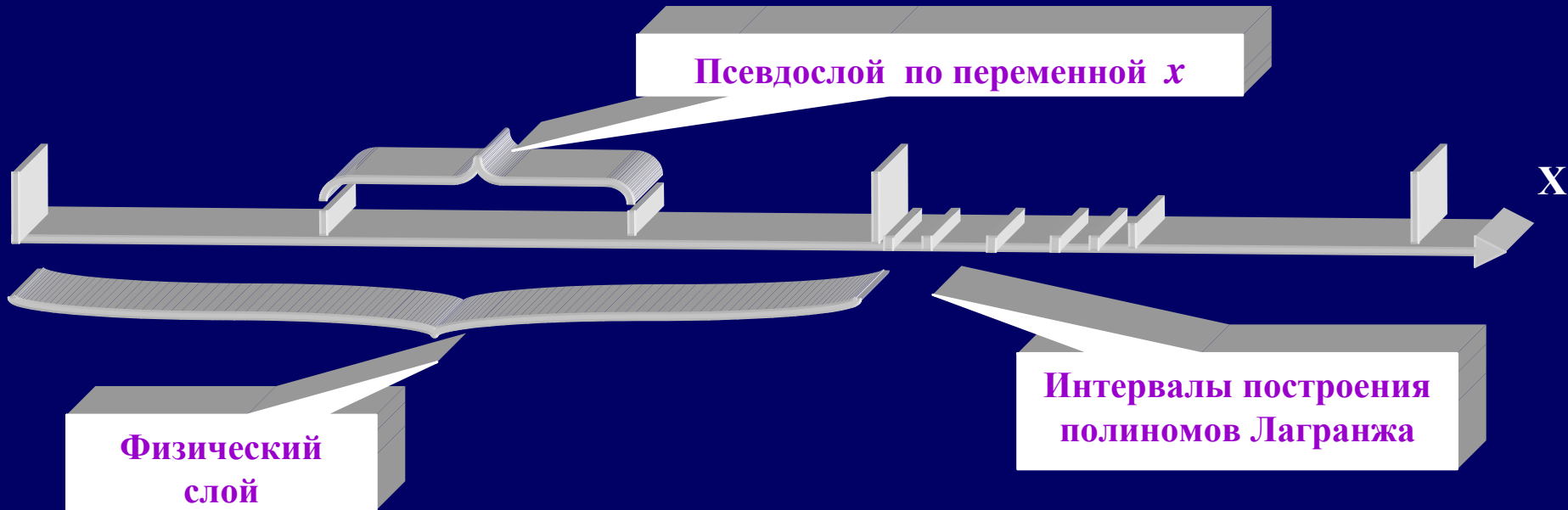


Схема разбиения области



В новой схеме псевдослои по способу расположения интервалов интерполяционных многочленов подразделяются на четыре типа:

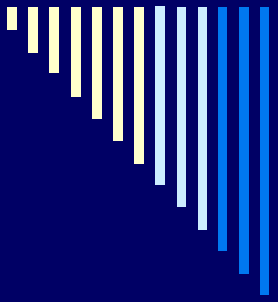
- Левый
 - Правый
 - Серединный
 - Двойной
-
- The four diagrams show different ways to place intervals on a line segment. 1. "Левый": intervals are clustered at the left end. 2. "Правый": intervals are clustered at the right end. 3. "Серединный": intervals are clustered in the middle. 4. "Двойной": intervals are placed at both ends.

Ранее использовался только тип 4.



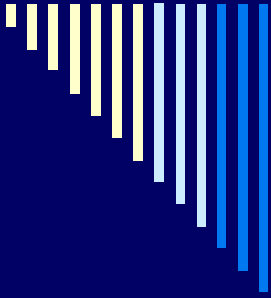
При выполнении DVM-программы обмен сообщениями между процессорами производится для осуществления следующих коллективных операций:

- глобальная редукция (редукционная операция над локальными значениями редукционных переменных, вычисленными на разных процессорах);
 - обновление теневых элементов;
 - загрузка буферов для доступа к удаленным данным;
 - перераспределение данных;
 - операции ввода-вывода (осуществляются через один специально выделенный процессор, который принимает данные от других процессоров или рассылает им данные).
-



Существуют следующие составляющие потерянного времени:

- потери из-за недостатка параллелизма, приводящего к дублированию вычислений на нескольких процессорах (**недостаточный параллелизм**). Дублирование вычислений осуществляется в двух случаях. Во-первых, последовательные участки программы выполняются всеми процессорами. Во-вторых, ветки некоторых параллельных циклов могут быть по указанию программиста полностью или частично размножены.
- потери из-за выполнения межпроцессорных обменов (**коммуникации**).
- потери из-за простоев тех процессоров, на которых выполнение программы завершилось раньше, чем на остальных (**простои**).



Модель FDVM определяет два уровня параллелизма:

- параллелизм по данным на секции массива процессоров;
- параллелизм задач – независимые вычисления на секциях массива процессоров.

Основная панель

Перенос радиационного излучения - D:\user\chma\rel9801\test

Имя Редактирование Запуск Результат Настройки Помощь

Характеристики по областям	Геометрическая область, $i = 1, 5$					Весь слой
	1	2	3	4	5	
Толщина, L_i	0,4	0,4	0,2	0,2	0,5	1,7
Плотность, ρ_i	0,03	0,03	0,03	0,03	0,03	
Температура, T_i	1,5	1,5	1,5	1,5	1,5	
Скорость, u_i	0	0,5	0,99	0	0	

Массовые концентрации компонент (%)	E1		E1		E1		E1		E1		Число состояний	линий
	100		100		100		100		100			
											3	2
											3	2
											3	2
											3	2

Вид граничных функций: $I(\epsilon, \mu) = \omega f(\epsilon) g(\mu)$

ω 0,25

$f(\epsilon)$ Трапеция

μ 0,96

Профиль контура поглощения: доплеровский

Метод решения:

- МАПИ
- МПЛЧ

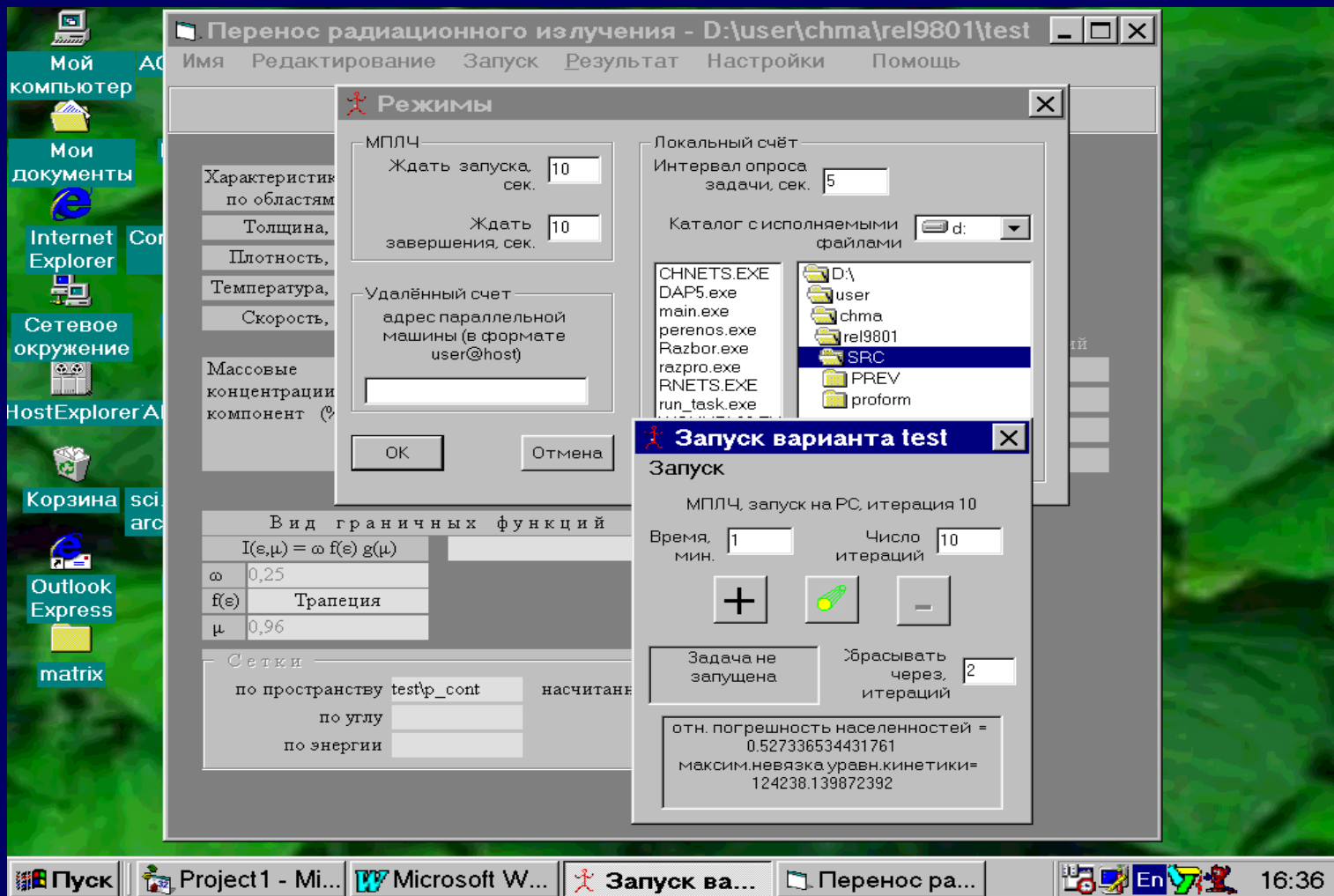
Сетки:

- по пространству test\p_cont
- по углу
- по энергии

насчитанная

Пуск Project1 - Mi... Перенос р... 16:26

Подсистема запуска задач



Подсистема визуализации

Перенос радиационного излучения - D:\user\chma\rel9801\test

Имя Редактирование Запуск Результат Настройки Помощь

Характеристики Геометрическая область, $i = 1,5$ Весь

результаты расчетов населенностей test

по объ...
Толщ...
Плотн...
Темпер...
Скор...
Массовы...
коэфф...
компл...
I(ε) в...
ω 0,24...
f(ε) 0,94...
сет...
по пр...

населенности

1 2
 3 4
 5 6
 7 8
 9 10
 11 12

сравнение вариантов (компонент) текущий вариант

вариант 1 test файл cc10t.dn cj_1ch.dn	вариант 2 файл cc10t.dn cj_1ch.dn	вариант 3 файл cc10t.dn cj_1ch.dn	вариант 4 файл cc10t.dn cj_1ch.dn
--	--	--	--

нарисовать

выход

save as delete

Пуск Microsoft W... Project1 - Mi... Перенос ра... результат... 15:04

Перенос радиационного излучения - D:\user\chma\rel9801\test

результаты расчетов интенсивностей test

МПЛЧ

интенсивность в напр. спектре для линий 1, 2, 3, 4

номер по ε	16	16	16	16
номер по мю	35	44	9	4
номер по х	60	60	60	60

интенсивность(дискр спектр)

рисуем интенсивность по x
 рисуем интенсивность по epsilon

номер линии 2

номер обл. по epsilon

1	0	0	0	0
---	---	---	---	---

номер по мю 28

номер по х 66 66 66 66

номер по ε 11 11 11 11

рисуем интегральную интенсивность

для линии

нарисовать

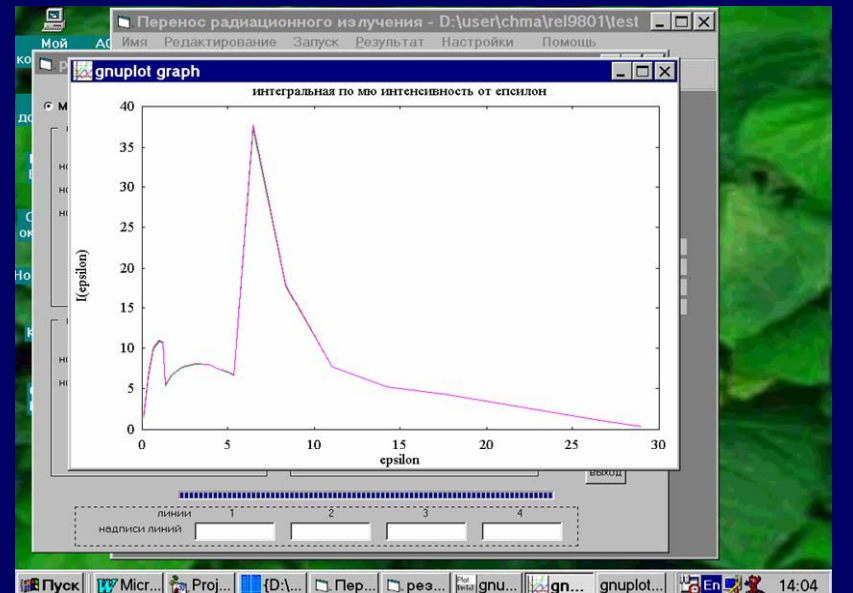
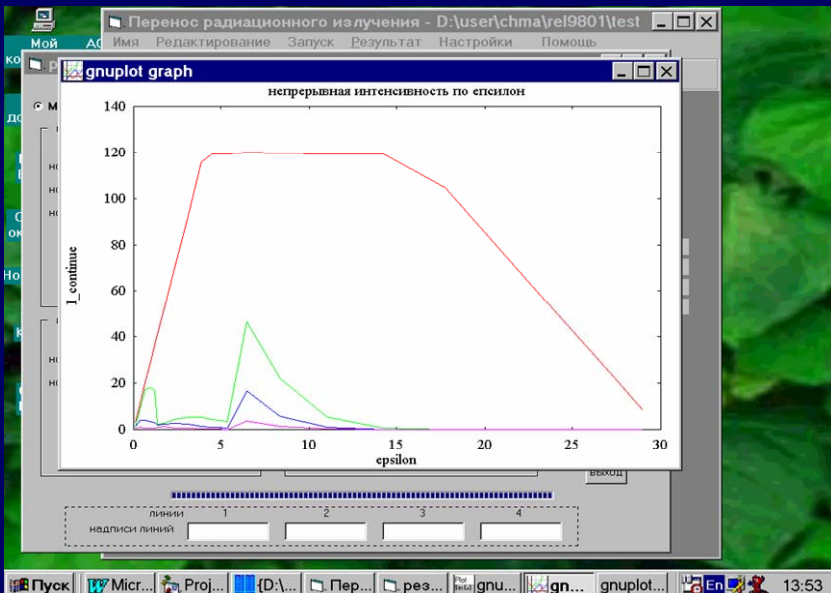
выход

вариант 1
test
файл
die_1ch.de
save as
delete

вариант 2
файл
die_1ch.de

линии 1 2 3 4
надписи линий область 1

Пуск Microsoft W... Project1 - Mi... Перенос ра... результат... 15:08



Сетка по x

Сетка по пространству - test\p_cont

Параметры сетки

δ
 N
 δN
 $L_i - 2 \delta (N+1)$
 δN
 δ
 X_1 X_{i+1}
 Всего узлов

УЗЛЫ: 1 2 3 4 5

Номер состояния	0	1	2	3	4	5
1	0,9987467	0,9987442	0,9987442	0,9987424	0,9987402	
2	1,3794e-7	1,379e-7	1,3789e-7	1,3785e-7	1,378e-7	
3	0,0011312	0,0011308	0,0011308	0,0011305	0,0011301	
4	7,9194e-5	8,0621e-5	8,0643e-5	8,1607e-5	8,2791e-5	
5	4,2731e-5	4,4229e-5	4,4254e-5	4,5406e-5	4,6757e-5	
6						
7						
8						
9						
10						
11						
12						

Н
А
С
Е
Л
Е
Н
Н
О
С
Т
И

Пуск Micro... Proje... {D:\u... Пере... Сетк... Сетк... Сет... En 13:03

Сетки по энергии

Сетки по энергии

Параметры и сетка конкретной линии

Количество узлов

$\epsilon_k (1 - \mu_i \mu_j) - \Delta_k$
 $\epsilon_k (1 - \mu_i \mu_j)$
 $\epsilon_k (1 - \mu_i \mu_j) + \Delta_k$

4,881215 4,8817171 4,882419 4,8832639 4,8841832 4,8851025 4,8859473

$\mu_i = 0,2$ $\epsilon_k = 4,1005$ $\mu_j = -0,959533$

Параметры просмотра

Область, i Переход, k Номер узла по μ, j

Сетка в непрерывном спектре

Число узлов по длине участка

Границы длин участков: 0

=0=	1	2	3	4	5	=1,3625=
=0=	0,13493	0,378065	0,68125	0,984435	1,22757	=1,3625=

Пуск Micros... Project... {D:\use... Перен... Сетка ... Сетки... En 12:58

Сетка по угловой переменной

Перенос радиационного излучения - D:\user\chma\rel9801\test

Имя Редактирование Запуск Результат Настройки Помощь

Характеристики по областям

Толщина, L_i	0,2
Плотность, ρ_i	0,03
Температура, T_i	1,5
Скорость, u_i	0

Массовые концентрации компонент (%)

E1	100
----	-----

Вид граничн

$I(\epsilon, \mu) = \omega f(\epsilon) g(\mu)$

ω	0,25
$f(\epsilon)$	Трапеция
μ	0,96

Сетки

по пространству testp_cont	насчитанная
по углу	
по энергии	

Сетка по угловой переменной

Кол-во точек по отрезкам

11	11	11	11	11
----	----	----	----	----

Сетка

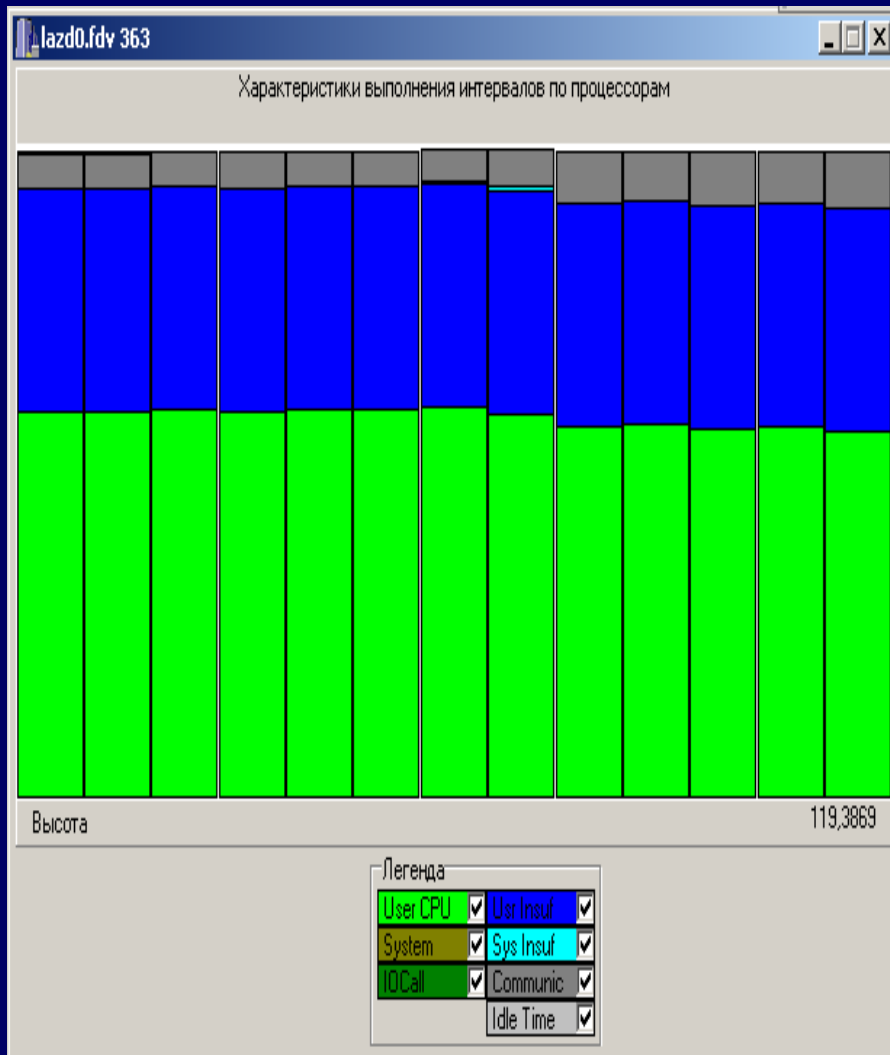
1	2	3	4	5	6	7
-1,0	-0,959533	-0,881879	-0,773327	-0,642673	-0,5005	-0,358327

Закрыть

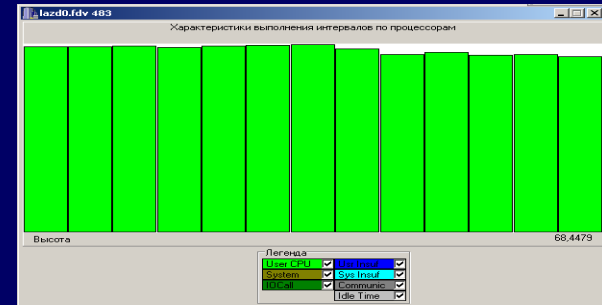
МАПН
МППЧ

Пуск Microsoft ... Project 1 - ... {D:\user\c... Перенос ... Сетка п... 13:24

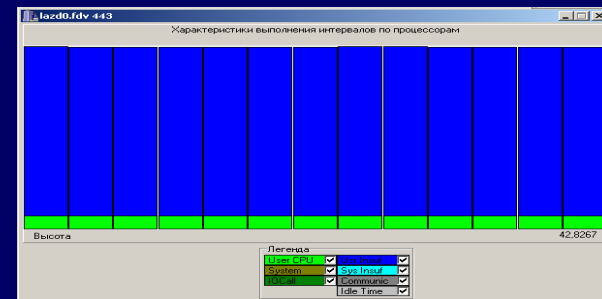
Распределение времени по интервалам



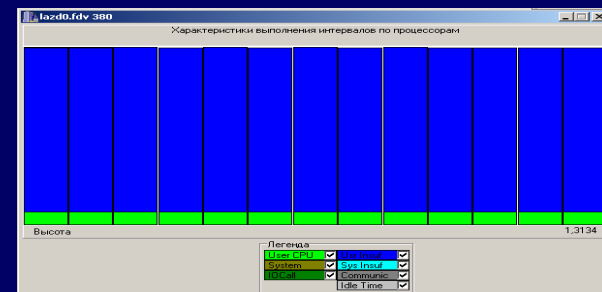
119сек.



68.4 сек



42.8сек



1.3 сек