

Нижегородский государственный университет им. Н.И.Лобачевского

Факультет Вычислительной математики и кибернетики

Образовательный комплекс

Введение в методы параллельного программирования

Раздел 14.

Параллельные методы многоэкстремальной оптимизации



Гергель В.П., профессор, д.т.н.
Кафедра математического обеспечения ЭВМ

Содержание

- Введение
- Постановка задачи
- Обзор методов
- Решение одномерных задач (последовательный индексный алгоритм)
- Редукция размерности задачи
- Использование множественных отображений
- Решение многомерных задач (параллельный индексный алгоритм)
- Заключение



Введение

- Задачи минимизации некоторой функции при ограничениях типа неравенств встречаются во многих областях науки и техники
- Нахождение минимума является трудоемкой операцией, ее сложность экспоненциально растет с ростом размерности задачи

Методы оптимизации представляют собой современную динамично развивающуюся область применения параллельных вычислений



Постановка задачи…

Найти минимум функции $\varphi(y)$:

$$\varphi(y^*) = \min\{\varphi(y) : y \in D, g_j(y) \le 0, 1 \le j \le m\},$$

$$D = \{y \in R^N : a_i \le y_i \le b_i, 1 \le i \le n\}.$$

где

- $\varphi(y)$ – минимизируемая функция (критерий),
- $g_j(y)$, $1 \le j \le m$ – функциональные ограничения,
- D – область поиска,
- y – вектор варьируемых параметров

Допустимая область

$$Q = \{y : y \in D, g_j(y) \le 0, 1 \le j \le m\}.$$



Постановка задачи…

Априорная информация о задаче – предположение о выполнимости условия Липшица

$$\left|\varphi(y_1)-\varphi(y_2)\right| \le L\left\|y_1-y_2\right\|,\ y_1, y_2 \in D$$

$$\left|g_j(y_1)-g_j(y_2)\right| \le L_j\left\|y_1-y_2\right\|, 1 \le j \le m, y_1, y_2 \in D.$$



Постановка задачи…

Пример задачи глобальной условной оптимизации

Критерий:

$$\varphi(y) = -1.5y_1^2 \exp\{1 - y_1^2 - 20.25(y_1 - y_2)^2\} - \left[\frac{(y_1 - 1)(y_2 - 1)}{2}\right]^4 \exp\left\{2 - \left[\frac{(y_1 - 1)}{2}\right]^4 - (y_2 - 1)^4\right\}$$

Ограничения:

$$g_1(y)=0.01[(y_1-2.2)^2+(y_2-1.2)^2-2.25]\le 0$$

$$g_2(y)=100[1-(y_1-2)^2/1.44-(0.5y_2)^2]\le 0$$

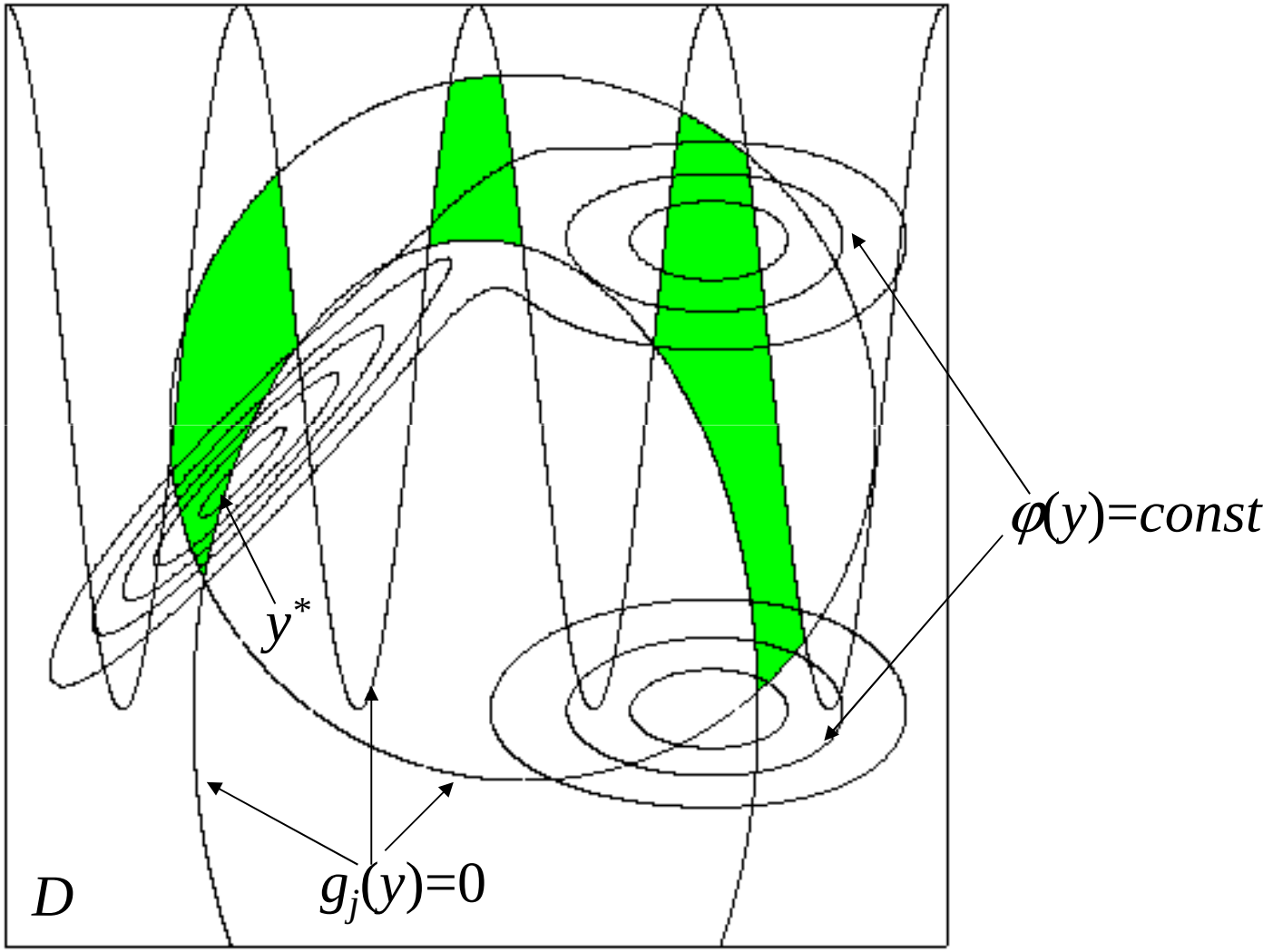
$$g_3(y)=10[y_2-1.5-1.5\sin(2\pi(y_1-1.75))]\le 0$$

Область поиска:

$$D=\{y\in R^2,\ 0\le y_1\le 4,\ -1\le y_2\le 3\}$$



Постановка задачи



Обзор методов…

Локальная оптимизация

Используется идея локального спуска

- В многоэкстремальных задачах схема локального спуска, вообще говоря, не приводит к решению.
- Мультистарт – запуск локального метода из нескольких точек.
- Низкая эффективность мульти-стартовых схем в существенно многоэкстремальных задачах, т.к. одно и то же локальное решение может быть найдено несколько раз.



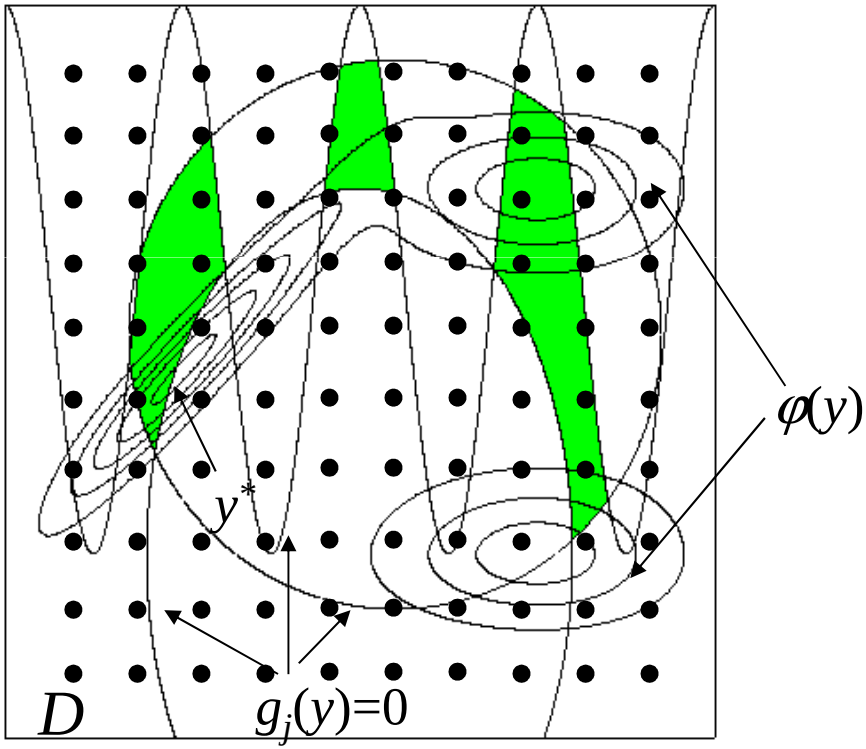
Обзор методов…

Глобальная оптимизация

- Построение случайных или детерминированных покрытий области поиска.

Количество узлов равномерной сетки увеличивается экспоненциально с увеличением размерности задачи.

$$T = T(\Delta) \approx \prod_{i=1}^{N} [(b_i - a_i)/\Delta]$$

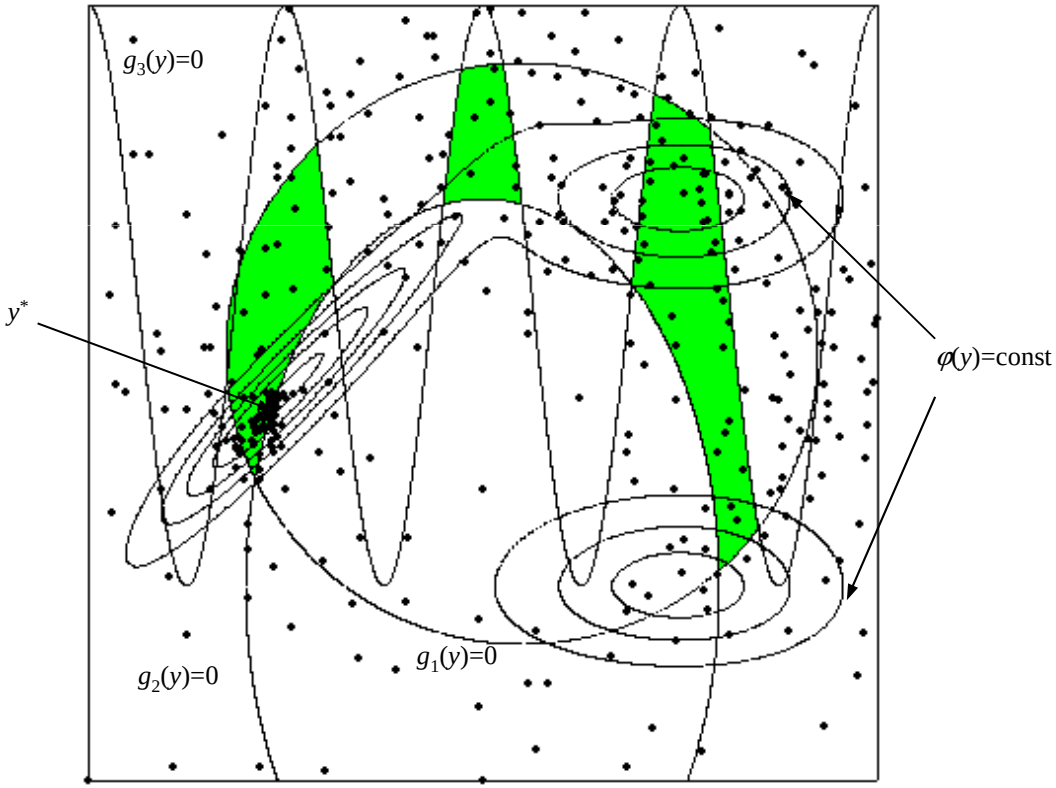


Обзор методов…

Глобальная оптимизация

- Построение неравномерных адаптивных покрытий области поиска (*алгоритм глобального поиска*)

Методы ориентированы на построение существенно более плотной сетки только в окрестности глобально-оптимального решения задачи, чем вне этой окрестности.

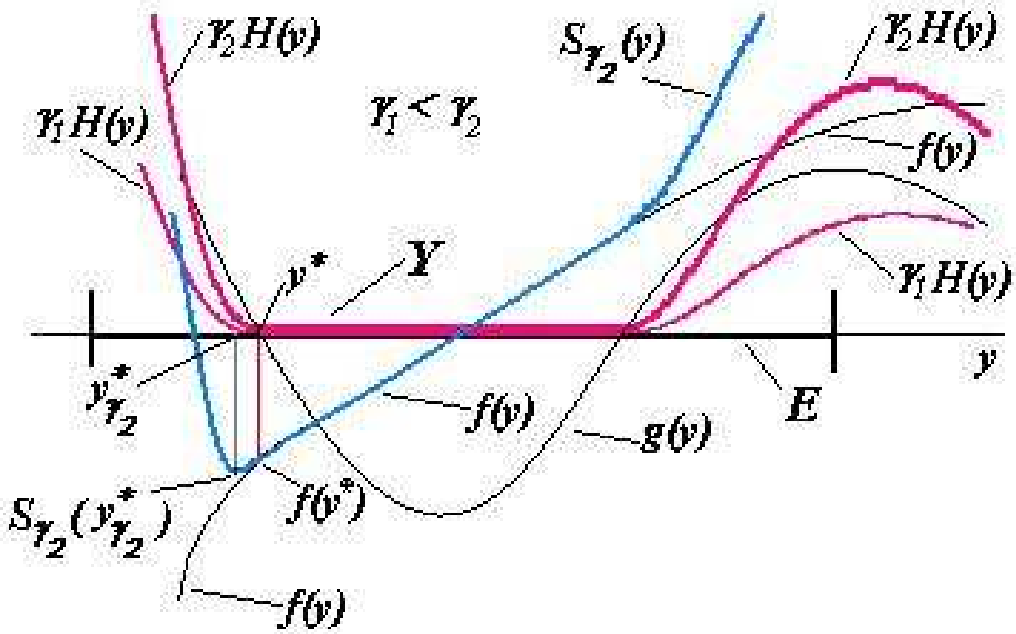


Обзор методов

Учет нелинейных ограничений

Классический подход: методы штрафных и барьерных функций

- Вычисление всех значений ограничений в точке, включая нарушенные.
- Проблема подбора коэффициентов штрафа.
- Решение серии безусловных подзадач.



Новый подход: *индексная схема учета ограничении*



Решение задач без ограничений

Рассмотрим одномерную задачу глобальной оптимизации

$$\varphi(x^*)=\min\{\varphi(x) : x\in[a,b],\ 1\le j\le m\}$$

и предположим, что функция $\varphi(x)$ удовлетворяет или условию Липшица

$$|\varphi(x_1)-\varphi(x_2)|\le L|x_1-x_2|.$$

или условию Гельдера

$$\left|\varphi(x_1)-\varphi(x_2)\right|\le K\left(\left|x_1-x_2\right|\right)^{1/N}$$



Метод Пиявского

Характеристическая схема алгоритма:…

Первые два испытания: $x^0=a$ и $x^1=b$.

1. Упорядочить точки по координате: $a = x_0<\ldots<x_i<\ldots<x_k = b$.
2. Для каждого (x_{i-1},x_i), $1\le i\le k$, вычислить **характеристику** $R(i)$,

$$R(i) = 0.5M\Delta_i - 0.5(z_i + z_{i-1}),$$

где M – значение константы Липшица, Δ_i – длина интервала.

3. Определить номер t : $R(t)=\max\{R(i): 1\le i\le k\}$.
4. Провести очередное испытание в точке из интервала (x_{t-1},x_t)

$$x^{k+1} = \frac{x_t + x_{t-1}}{2} - \frac{z_t - z_{t-1}}{2M},$$

5. Проверить условие остановки: $x_t - x_{t-1} \le \varepsilon$,

где ε – заданная точность поиска глобального минимума



Алгоритм глобального поиска (АГП)

Характеристическая схема алгоритма:…

Первые два испытания: $x^0=a$ и $x^1=b$.

1. Упорядочить точки по координате: $a = x_0<\ldots<x_i<\ldots<x_k = b$.
2. Вычислить оценку μ для неизвестной константы Липшица L,

$$\mu = \max\left\{\frac{|z_i - z_{i-1}|}{x_i - x_{i-1}},\quad i = 1,\ldots,k\right\}$$

3. Для каждого (x_{i-1},x_i), $1\le i\le k$, вычислить **характеристику** $R(i)$,

$$R(i) = \Delta_i + \frac{(z_i - z_{i-1})^2}{r^2\mu^2\Delta_i} - 2\frac{(z_i + z_{i-1})}{r\,\mu},$$

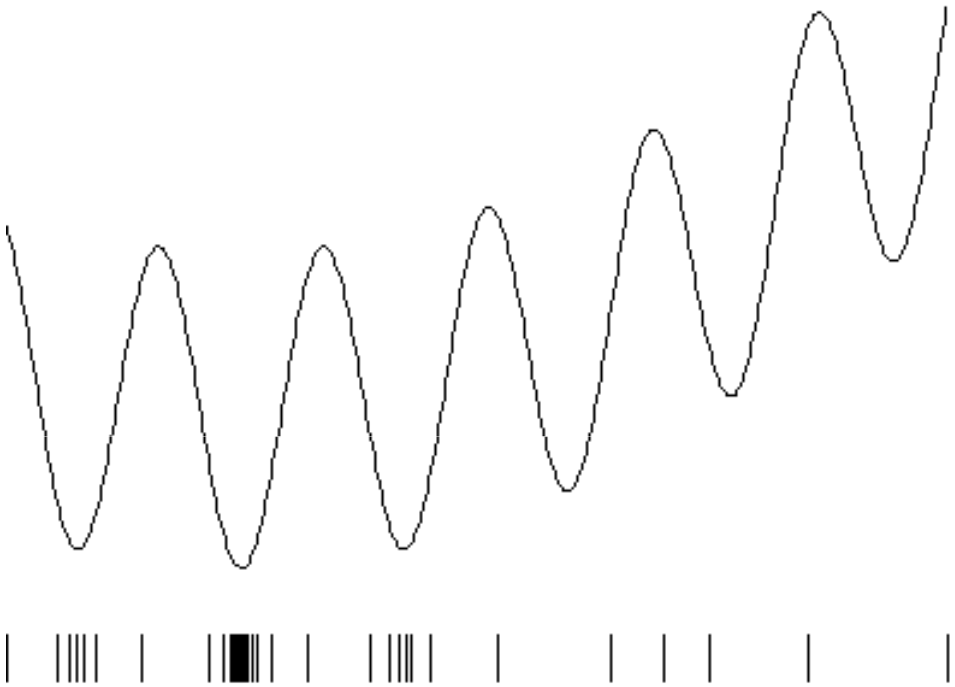
где Δ_i – длина интервала, $r>1$, – параметр метода.

4. Определить номер t : $R(t)=\max\{R(i): 1\le i\le k\}$.
5. Провести очередное испытание в точке из интервала (x_{t-1},x_t)

$$x^{k+1} = \frac{x_t + x_{t-1}}{2} - \frac{z_t - z_{t-1}}{2r\,\mu},$$



Пример использования АГП



$\varphi(x^*)=\min\{\varphi(x)=x^2+\cos(18x),\ x\in[-0.5,1.5]\}$

43 испытаний, точность поиска $\varepsilon=10^{-3}$



Условия сходимости

Если задача имеет решение y^* и выполняются условия:

- Функция $g_j(y)$, удовлетворяет в области поиска условию Липшица с константой $L > 0$, $1 \le \nu \le m+1$;
- Начиная с некоторой итерации k выполняется неравенство

$$r\mu > 2L,$$

где $r>1$ – параметр метода, μ – текущая адаптивная оценка константы Липшица;

Тогда метод будет сходиться ко всем точкам условного глобального минимума функции $\varphi(y)$, и только к ним.



Параллельный характеристический алгоритм

Основу составляют правила последовательной версии, отличия касаются инициализации и пунктов 4 – 6.

Инициализация: первые $\tau+1$ испытаний провести в произвольных внутренних точках

$$0 = x^1 < x^2 < \ldots < x^{\tau} < x^{\tau+1} = 1.$$

4. Определить τ интервалов с наибольшими характеристиками
$$R(t_1) \geq R(t_2) \geq \ldots \geq R(t_{\tau})$$
5. В этих интервалах провести τ испытаний $(n+1)$-й итерации

$$x^{k+s} = \frac{x_{t_s} + x_{t_s-1}}{2} - \frac{z_{t_s} - z_{t_s-1}}{2r\mu},$$

6. Проверить условие остановки: $\min\limits_{1 \leq s \leq \tau} \Delta_{t_s} \leq \varepsilon$



Реализация с помощью OpenMP

Описанный алгоритм реализует синхронную схему организации параллельных вычислений с единым массивом данных; легко реализуется с помощью OpenMP.

```
//Начало параллельной секции
#pragma omp parallel private(t,t1,new_x) shared(NewTrials, BestIntervals){
        i = omp_get_thread_num();//номер потока
        t = BestIntervals[i]; t1=t; t1--;//указатели на границы интервала
        //Вычисление новой точки для проведения испытания
        new_x = (t–>x+t1–>x)/2 – fabs(t–>z – t1–>z)/(2* M* rv);
        NewTrials[i] = MakeTrial(new_x);//проведение испытания в новой точке
}//Завершение параллельной секции
```



Индексная схема учета ограничений…

Рассмотрим одномерную задачу глобальной оптимизации

$$\varphi(x^*)=\min\{\varphi(x)\equiv g_{m+1}(x): x\in[a,b],\ g_j(x)\le 0,\ 1\le j\le m\}$$

и предположим, что функции задачи удовлетворяют условию Липшица

$$|g_j(x_1)-g_j(x_2)|\le L_j|x_1-x_2|,\ 1\le j\le m+1.$$

Частичная вычислимость: функция $g_j(x)$ вычислима лишь в соответствующей подобласти $Q_j\subset[a,b]$, $1\le j\le m$, где

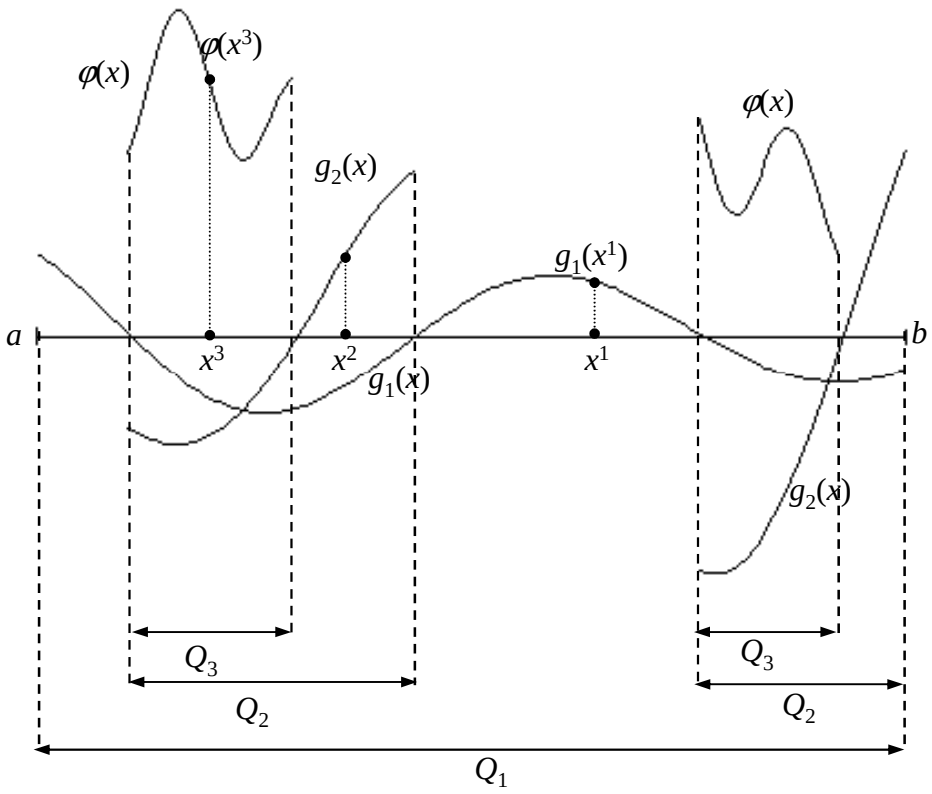
$$Q_1=[a,b],\ Q_{j+1}=\{x\in Q_j: g_j(x)\le 0\},\ 1\le j\le m.$$

↳ Исходная задача может быть приведена к виду

$$\varphi(x^*)=\min\{g_{m+1}(x): x\in Q_{m+1}\}.$$



Индексная схема учета ограничений…



Пример

Критерий:

$\varphi(x)=\cos(18x-3)\sin(10x-7)+1.5$

Ограничения:

$g_1(x)=\exp(-x/2)\sin(6x-1.5)\le 0,$

$g_2(x)=|x|\sin(2\pi x-0.5)\le 0.$

Область поиска: $x\in[0.6,2.2]$

Приведены дуги функций задачи и соответствующие области $Q_j, 1\le j\le 3$, в предположении частичной вычислимости

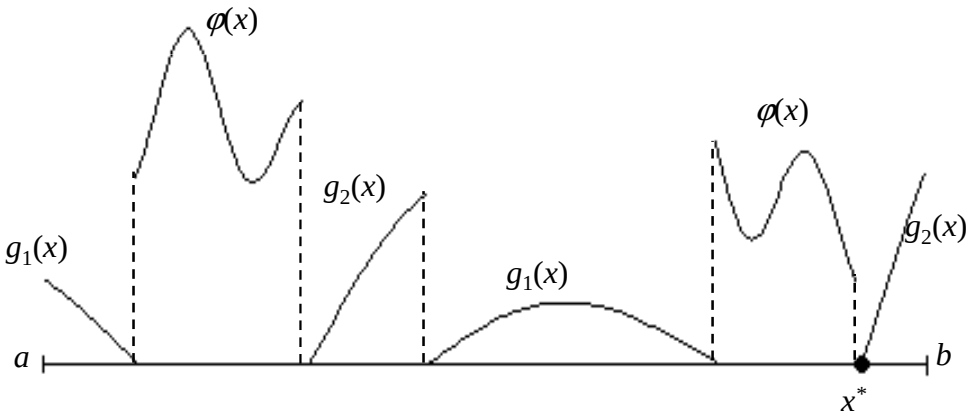


Индексная схема учета ограничений…

Введем классификацию точек x из области поиска $[a,b]$ с помощью *индекса* $\nu=\nu(x)$, где $\nu-1$ есть число ограничений, которые выполняются в этой точке. Указанный индекс ν определяется условиями

$$g_j(x)\le 0,\ 1\le j\le \nu-1,\ g_\nu(x)>0,$$

где последнее неравенство несущественно, если $\nu=m+1$.



«Индексная» функция $f(x)$

Данная классификация порождает функцию

$$f(x)=g_\nu(x),\ \nu=\nu(x),$$

определенную и вычислимую всюду в $[a,b]$

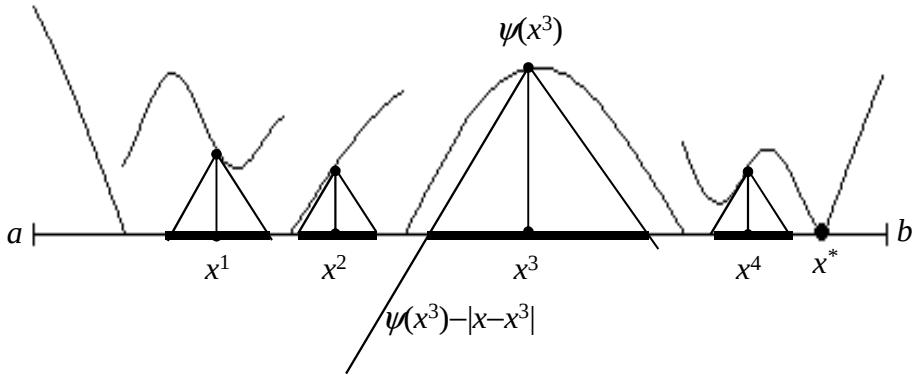


Индексная схема учета ограничений

Исходная задача преобразуется к виду

$$\psi(x^*)=\min\{\psi(x): x\in[a,b]\},\ где\ \ \psi(x)=\begin{cases} g_\nu(x)/L_\nu, & \nu<M \\ (g_M-g_M^*)/L_M, & \nu=M \end{cases},\ M=\max\{\nu(x): x\in[a,b]\}.$$

Схема исключает влияние априори неизвестных констант Липшица L_ν, $1\le\nu\le m+1$, и глобального минимума g_M^*



Оценка оптимальной точки: объединение выделенных отрезков не содержит решения задачи

Последовательный алгоритм…

Характеристическая схема алгоритма:…

Первые два испытания: $x^0=a$ и $x^1=b$.

1. Упорядочить точки по координате: $a = x_0<\ldots<x_i<\ldots<x_k = b$.
2. Классифицировать точки по индексу, т.е. построить множества $I_\nu=\{i: 1\le i\le k,\ \nu=\nu(x_i)\}$, $1\le\nu\le m+1$.
3. Вычислить текущие нижние оценки μ_ν для априори неизвестных констант Липшица $L_\nu, 1\le\nu\le m+1$,

$$\mu_\nu = \max\left\{\frac{|z_i - z_j|}{x_i - x_j},\quad i, j\in I_\nu,\ i > j\right\}$$



Последовательный алгоритм…

Характеристическая схема алгоритма:…

4. Для каждого (x_{i-1},x_i), $1\le i\le k$, вычислить **характеристику** $R(i)$,

$$R(i)=\Delta_i+\frac{(z_i-z_{i-1})^2}{r_\nu^2\mu_\nu^2\Delta_i}-2\frac{(z_i+z_{i-1}-2z_\nu^*)}{r_\nu\mu_\nu},\quad \nu=\nu(x_{i-1})=\nu(x_i),$$

$$R(i)=2\Delta_i-4\frac{(z_i-z_\nu^*)}{r_\nu\mu_\nu},\quad \nu=\nu(x_i)>\nu(x_{i-1}),$$

$$R(i)=2\Delta_i-4\frac{(z_{i-1}-z_\nu^*)}{r_\nu\mu_\nu},\quad \nu=\nu(x_{i-1})>\nu(x_i),$$

где $\Delta_i=x_i-x_{i-1}$ – длина интервала,

$r_\nu>1$, $1\le\nu\le m+1$, – параметры метода.

5. Определить интервал с максимальной характеристикой $R(t)=\max\{R(i):\ 1\le i\le k\}$.



Последовательный алгоритм…

Характеристическая схема алгоритма:

5. Провести очередное испытание во внутренней точке интервала (x_{t-1},x_t)

$$x^{k+1}=\frac{x_t+x_{t-1}}{2},\ \nu(x_{t-1})\neq\nu(x_t)$$

$$x^{k+1}=\frac{x_t+x_{t-1}}{2}-\frac{z_t-z_{t-1}}{2r_\nu\mu_\nu},\nu=\nu(x_{t-1})=\nu(x_t)$$

6. Проверить условие остановки: $x_t-x_{t-1}\leq\varepsilon$,

где ε – заданная точность поиска глобального минимума.



Последовательный алгоритм…

Условия сходимости алгоритма:

Если задача имеет решение y^* и выполняются условия:

- Функции $g_j(y)$, $1 \le j \le m+1$, удовлетворяют в области поиска условию Липшица с константами $L_\nu > 0$, $1 \le \nu \le m+1$;
- Начиная с некоторой итерации k выполняются неравенства

$$r_\nu \mu_\nu > 2L_\nu, 1 \le \nu \le m+1,$$

где $r_\nu > 1$ – параметры метода, μ_ν – текущие адаптивные оценки констант Липшица;

Тогда метод будет сходиться ко всем точкам условного глобального минимума функции $\varphi(y)$, и только к ним.



Последовательный алгоритм…

- **Результаты вычислительных экспериментов**
  - Рассмотрим одномерную задачу
    - Критерий: $\varphi(x)=\cos(18x-3)\sin(10x-7)+1.5$
    - Ограничения:
      $g_1(x)=\exp(-x/2)\sin(6x-1.5)\le 0,$
      $g_2(x)=|x|\sin(2\pi x-0.5)\le 0.$
    - Область поиска: $x\in[0.6,2.2]$
  - Точность поиска минимума 10^{-5}, решение $x^*=2.0795$, $\varphi(x^*)=0.565$

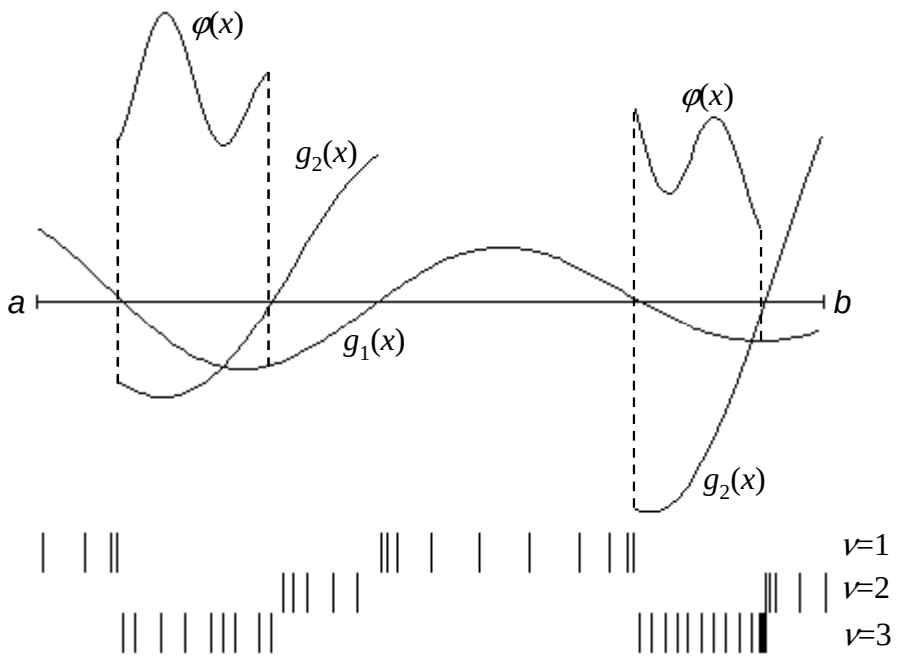
	k_1	k_2	k_3
Перебор по равномерной сетке	160000	90280	56476
Метод штрафных функций	375	375	375
Индексный метод	63	49	35

k_i – число вычислений значений i-й функции.



Последовательный алгоритм

Результаты вычислительных экспериментов



Дуги функций задачи в предположении частичной вычислимости

Координаты точек испытаний, осуществленных алгоритмом, отмечены тремя рядами вертикальных штрихов. Штрихи верхнего ряда соответствуют точкам с единичным индексом, второго – точкам, индексы которых равны 2; точки, отмеченные штрихами нижнего ряда, являются допустимыми.



Параллельный алгоритм

Основу составляют правила последовательной версии, отличия касаются инициализации и пунктов 4 – 6.

Инициализация: первые $\tau+1$ испытаний провести в произвольных внутренних точках

$$0 = x^1 < x^2 < \ldots < x^{\tau} < x^{\tau+1} = 1.$$

4. Определить τ интервалов с наибольшими характеристиками
$$R(t_1) \geq R(t_2) \geq \ldots \geq R(t_{\tau})$$
5. В этих интервалах провести τ испытаний $(n+1)$-й итерации

$$x^{k+s} = \frac{x_{t_s} + x_{t_s-1}}{2} - \frac{z_{t_s} - z_{t_s-1}}{2r_{\nu_s}\mu_{\nu_s}}, \nu = \nu(x_{t_s-1}) = \nu(x_{t_s})$$

$$x^{k+s} = \frac{x_{t_s} + x_{t_s-1}}{2}, \ \nu(x_{t_s-1}) \neq \nu(x_{t_s})$$

6. Проверить условие остановки: $\min\limits_{1 \leq s \leq \tau} \Delta_{t_s} \leq \varepsilon$



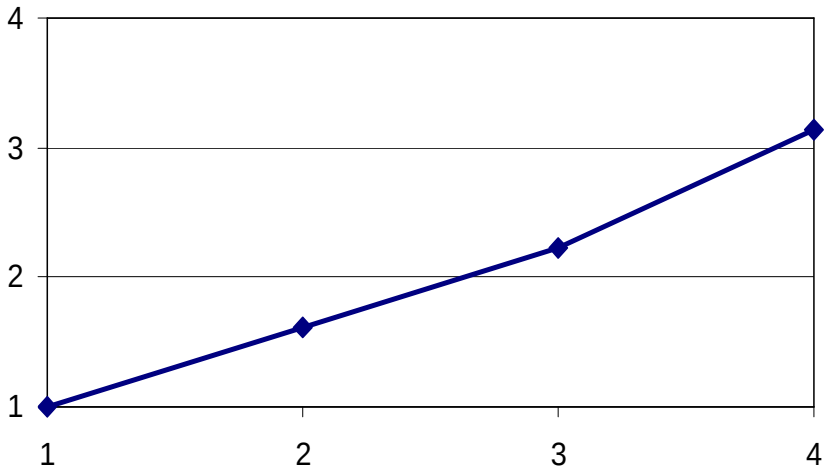
Реализация с помощью OpenMP

Описанный алгоритм реализует синхронную схему организации параллельных вычислений с единым массивом данных.

```
//Начало параллельной секции
#pragma omp parallel private(t,t1,new_x) shared(NewTrials,BestIntervals){
        i = omp_get_thread_num();//номер потока
        t = BestIntervals[i]; t1=t; t1--;//указатели на границы интервала
        //Вычисление новой точки для проведения испытания
        if(t->index!=t1–>index) new_x = 0.5*(t –>x+t1–>x);
        else new_x = (t–>x+t1–>x)/2 – sgn(t–>z – t1–>z)*pow(fabs(t–>z – t1–>z)/
                        Mv[t–>index],N)/(2*rv);
        NewTrials[i] = MakeTrial(new_x);//проведение испытания в новой точке
}//Завершение параллельной секции
```



Экспериментальная оценка ускорения



Потоки	Итерации	Ускорение
1	855	–
2	531	1.61
3	383	2.23
4	272	3.14

Под ускорением понимается отношение числа итераций последовательного и параллельного алгоритмов.



Редукция размерности…

Рассмотрим многомерную задачу глобальной оптимизации

$$\varphi(y^*) = \min\{\varphi(y): y \in D,\ g_j(y) \le 0,\ 1 \le j \le m\},$$

$$D = \{y \in R^N: a_i \le y_i \le b_i,\ 1 \le i \le N\}.$$

Пусть $y(x)$ есть отображение Пеано области поиска D на отрезок $[0,1]$

$$\{\ y(x): 0 \le x \le 1\ \} = \{\ y \in R^N: y \in D\ \}$$

Используя отображение $y(x)$, многомерная задача может быть сведена к одномерной

$$\varphi(y^*) = \min\{\ \varphi(y): y \in D,\ g_j(y) \le 0,\ 1 \le j \le m\ \} =$$

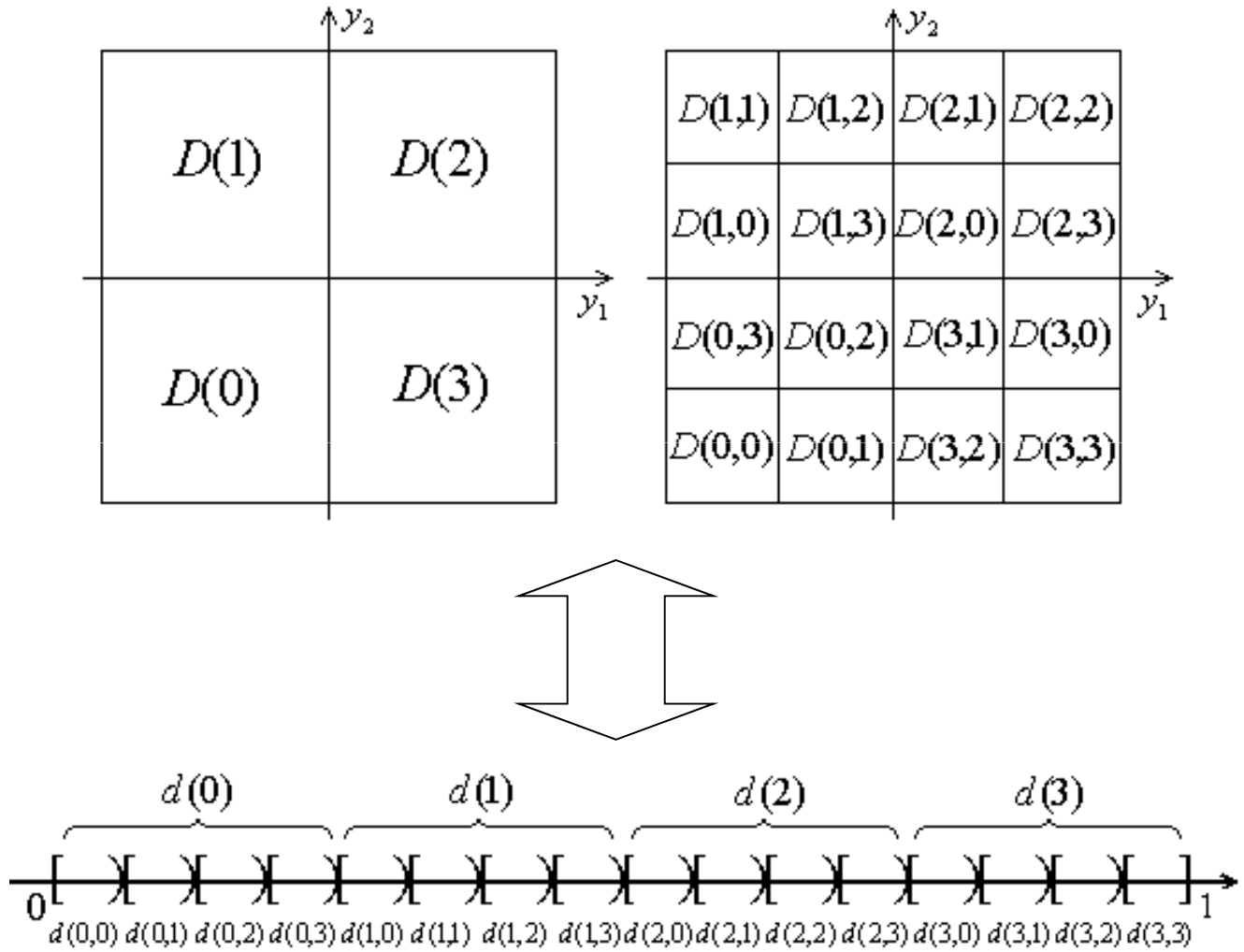
$$= \min\{\ \varphi(y(x)): x \in [0,1],\ g_j(y(x)) \le 0,\ 1 \le j \le m\ \}.$$

Если $\varphi(y)$ удовлетворяет условию Липшица, то $\varphi(y(x))$ удовлетворяет равномерному условию Гельдера

$$\left|\varphi(y(x_1)) - \varphi(y(x_2))\right| \le 4L\sqrt{N}\left(\left|x_1 - x_2\right|\right)^{1/N}$$

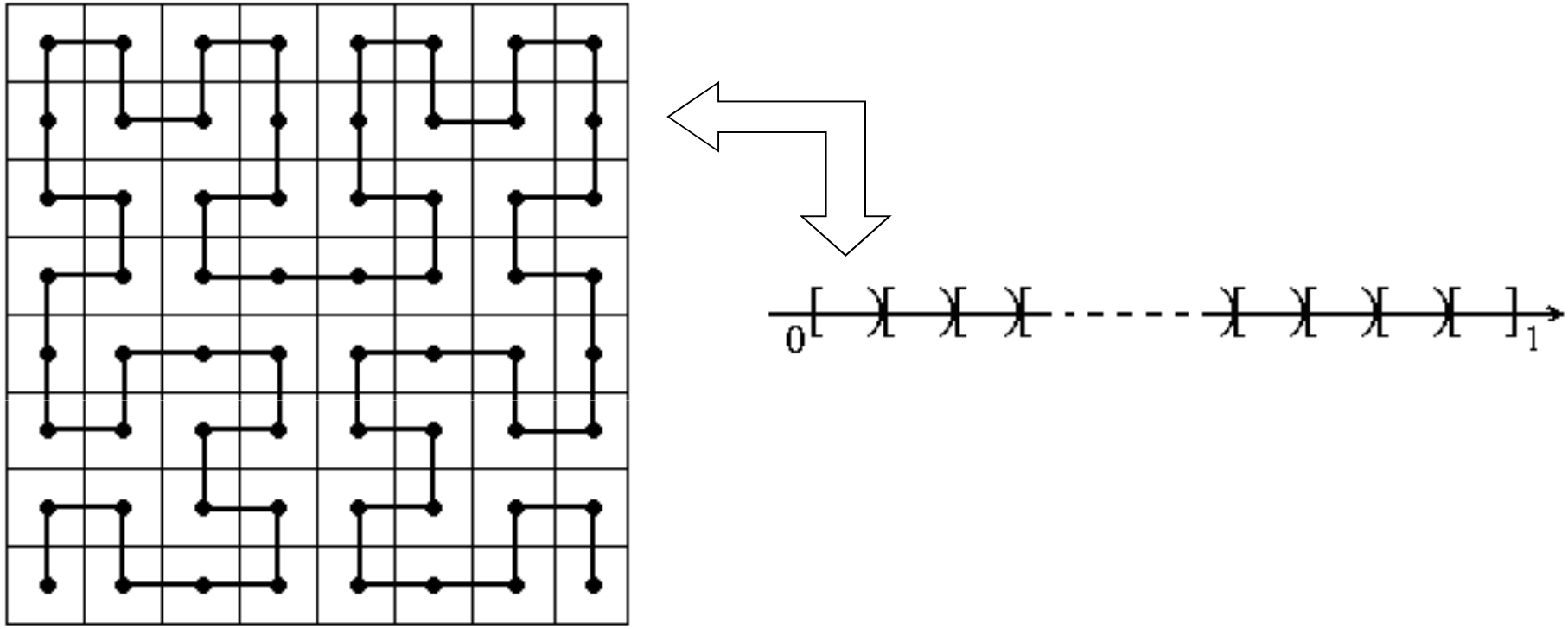


Редукция размерности…





Редукция размерности…



Численные методы для построения приближений кривых Пеано рассмотрены в работах Стронгина Р.Г. (1978, 1992) и монографии Стронгина Р.Г., Сергеева Я.Д. (2000).



Редукция размерности

- Программная реализация:
  - **Функция *mapd*** реализует прямое отображение: для заданной точки $x \in [0,1]$, плотности развертки m и размерности пространства n вычисляется образ $y(x) \in D$.
  - **Функция *xyd*** реализует обратное отображение: для заданной точки $y \in D$, плотности развертки m и размерности пространства n вычисляется ее прообраз $y^{-1}(x) \in [0,1]$.

Программа



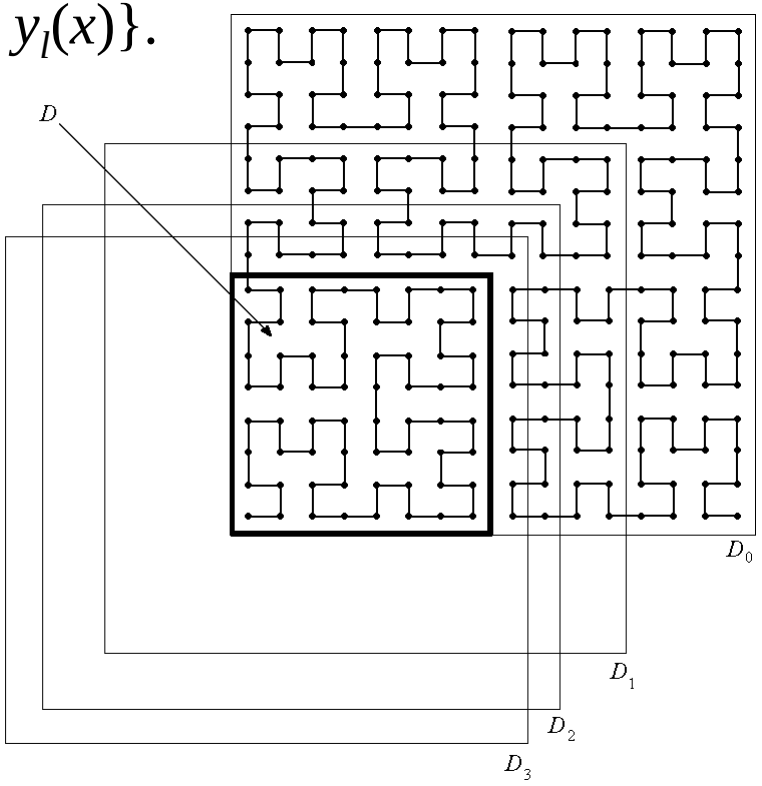
Множественные отображения…

Проблема. Близким образам $y'=y(x')$, $y''=y(x'')$ могут соответствовать существенно далекие прообразы x', $x''\in[0,1]$.

Подход. Использование множества отображений

$$Y(x)=\{y_1(x), y_2(x), \ldots, y_l(x)\}.$$

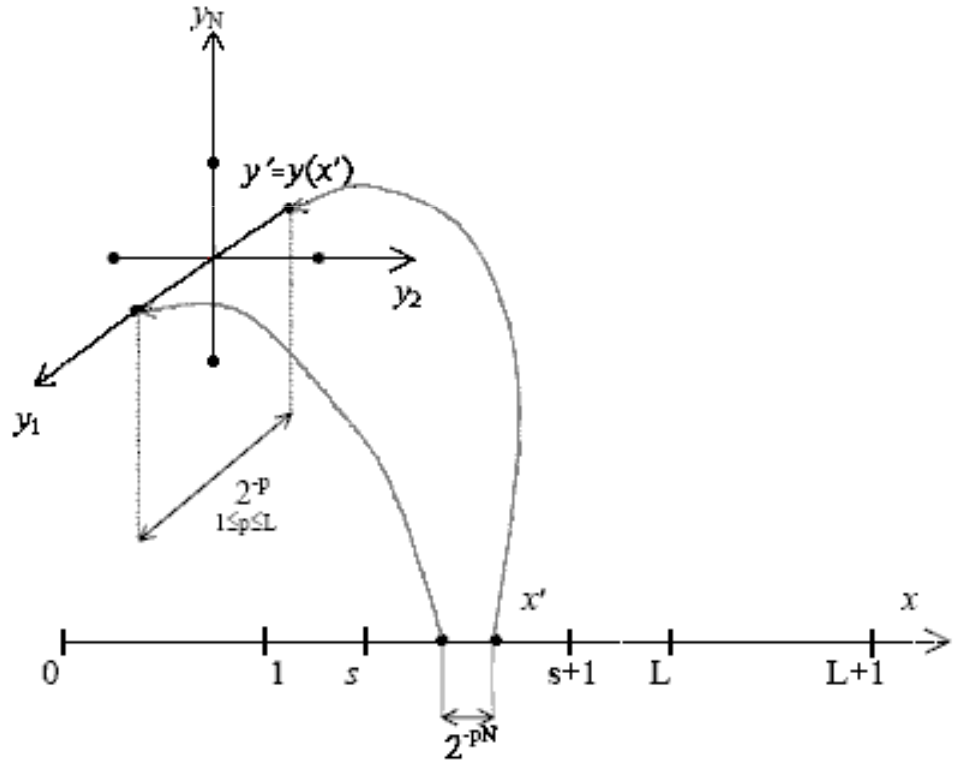
Каждая кривая Пеано $y_i(x)$ из $Y(x)$ может быть получена в результате некоторого сдвига вдоль главной диагонали гиперинтервала D.





Множественные отображения…

Таким образом сконструированное множество кривых Пеано позволяет получить для любых близких образов y', y'', отличающихся только по одной координате, близкие прообразы x', x'' для некоторого отображения $y_i(x)$.



Чтобы выделить область поиска D требуется введение в задачу дополнительного ограничения

$$g_0(y) = \max \{ |y_i| - 2^{-1} : 1 \le i \le N \}.$$



Множественные отображения

- Программная реализация:
  - **Функция *GetImage*** реализует прямое отображение: для заданной точки $x \in [0,1]$, плотности развертки m, размерности пространства n и номера развертки l вычисляется ее образ $y^l(x)$;
  - **Функция *GetPreimages*** реализует обратное отображение: для заданной точки y из многомерного пространства, плотности развертки m, размерности пространства n и числа разверток L вычисляются все ее прообразы $x^0, x^1, \ldots, x^L$.

Программа



Принцип распараллеливания

Использование множества отображений приводит к формированию соответствующего множества одномерных многоэкстремальных задач

$$\min\{\varphi(y^l(x)) : x \in [0,1],\ g_j(y^l(x)) \le 0,\ 0 \le j \le m\},\ 0 \le l \le L.$$

Каждая задача может решаться независимо; при этом любое вычисленное значение

$z_i = g_\nu(y^l(x_i))$ функции $g_\nu(y)$

может быть преобразовано к значению

$z_j = g_\nu(y^k(x_j))$ для любых задач l и k

без повторных трудоемких вычислений функции $g_\nu(y)$.

Подобное информационное единство дает возможность оптимизации всего множества функций параллельно



Параллельный алгоритм…

Решающие правила алгоритма в целом совпадают с правилами последовательного алгоритма, кроме способа проведения испытания:

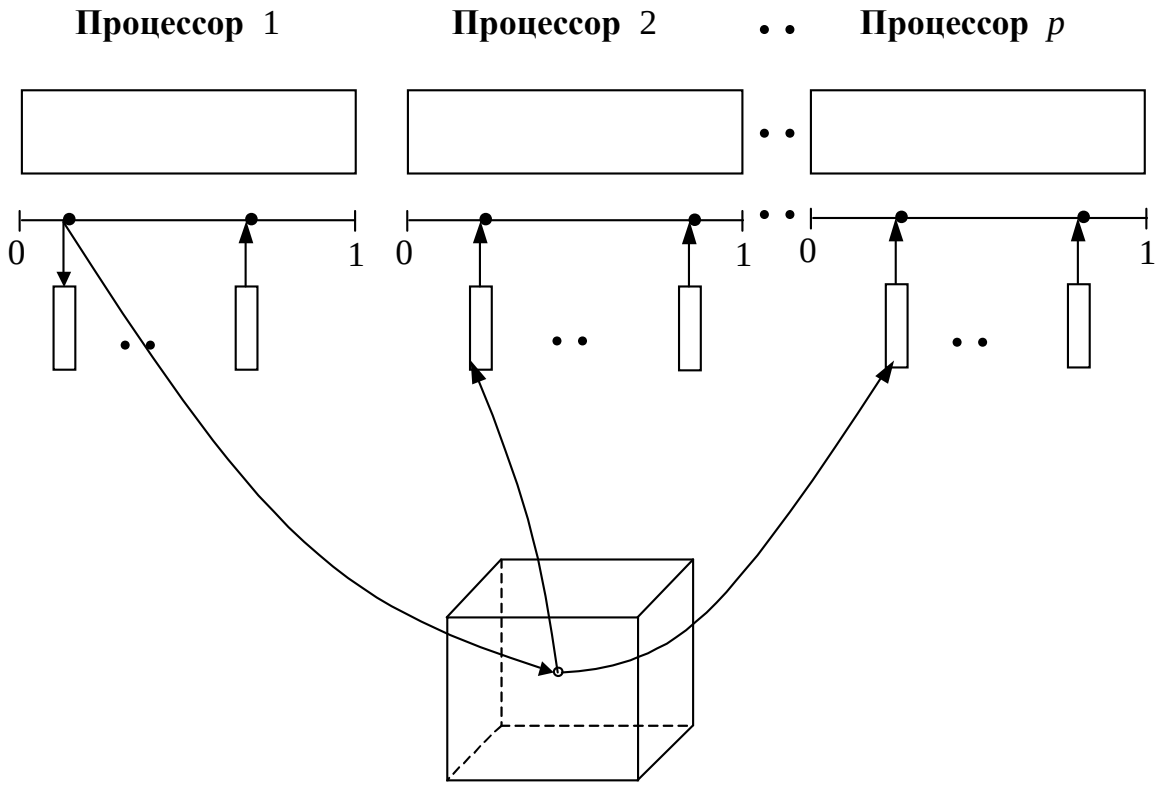
- После выбора точки очередной итерации поиска процессор информирует о произведенном выборе все остальные процессоры;
- Каждый процессор после проведения испытания в точке итерации передает полученный индекс и значение функции всем процессорам вычислительной системы;
- Перед началом очередной итерации каждый процессор использует все полученные данные для расширения имеющегося набора с поисковой информацией.

Предлагаемая схема не содержит какого-либо единого управляющего процессора, что увеличивает надежность выполняемых вычислений.



Параллельный алгоритм…

Схема информационных обменов



Параллельный алгоритм…

Результаты вычислительных экспериментов…

Рассмотрим двумерную задачу глобальной оптимизации

$$\varphi(y) = -1.5y_1^2 \exp\{1 - y_1^2 - 20.25(y_1 - y_2)^2\} - \left[\frac{(y_1 - 1)(y_2 - 1)}{2}\right]^4 \exp\left\{2 - \left[\frac{(y_1 - 1)}{2}\right]^4 - (y_2 - 1)^4\right\}$$

Ограничения:

$$g_1(y)=0.01[(y_1-2.2)^2+(y_2-1.2)^2-2.25]\le 0$$

$$g_2(y)=100[1-(y_1-2)^2/1.44-(0.5y_2)^2]\le 0$$

$$g_3(y)=10[y_2-1.5-1.5\sin(2\pi(y_1-1.75))]\le 0$$

Область поиска: $D=\{y\in R^2,\ 0\le y_1\le 4,\ -1\le y_2\le 3\}$

Решение: $y^*=(0.942,\ 0.944),\ \varphi(y^*)=-1.489$



Параллельный алгоритм…

Результаты вычислительных экспериментов…

Один процессор/развертка

l	Вычислено значений			
	g_1	g_2	g_3	$\varphi=g_4$
1	1098	623	392	152

Два процессора/развертки

l	Вычислено значений				
	g_0	g_1	g_2	g_3	$\varphi=g_4$
1	636	567	331	212	104
2	632	327	193	125	61
Всего	1268	894	524	337	165

Параметры метода

Плотность развертки m=12

Точность поиска $\varepsilon=10^{-3}$

Найдено решение y^*=(0.942, 0.945)

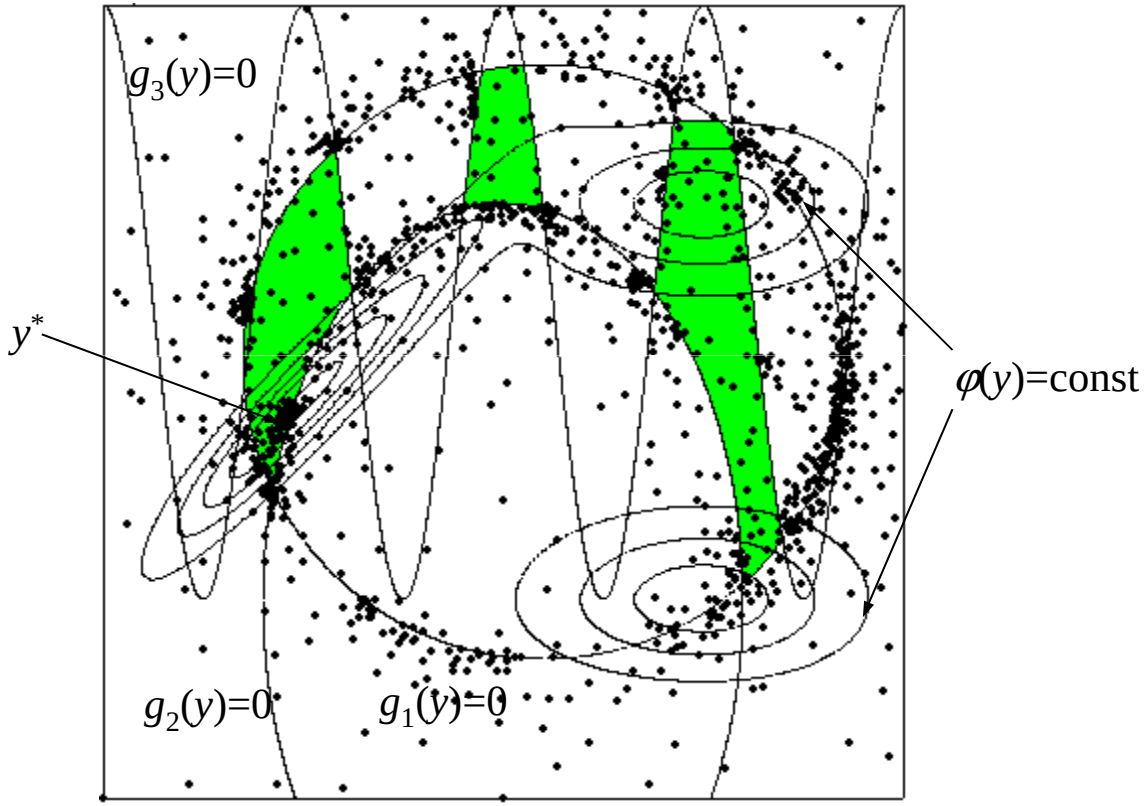
Сводная таблица

Число процессоров	Ускорение
1	—
2	1.93
4	2.16
6	3.80



Параллельный алгоритм…

Результаты вычислительных экспериментов…

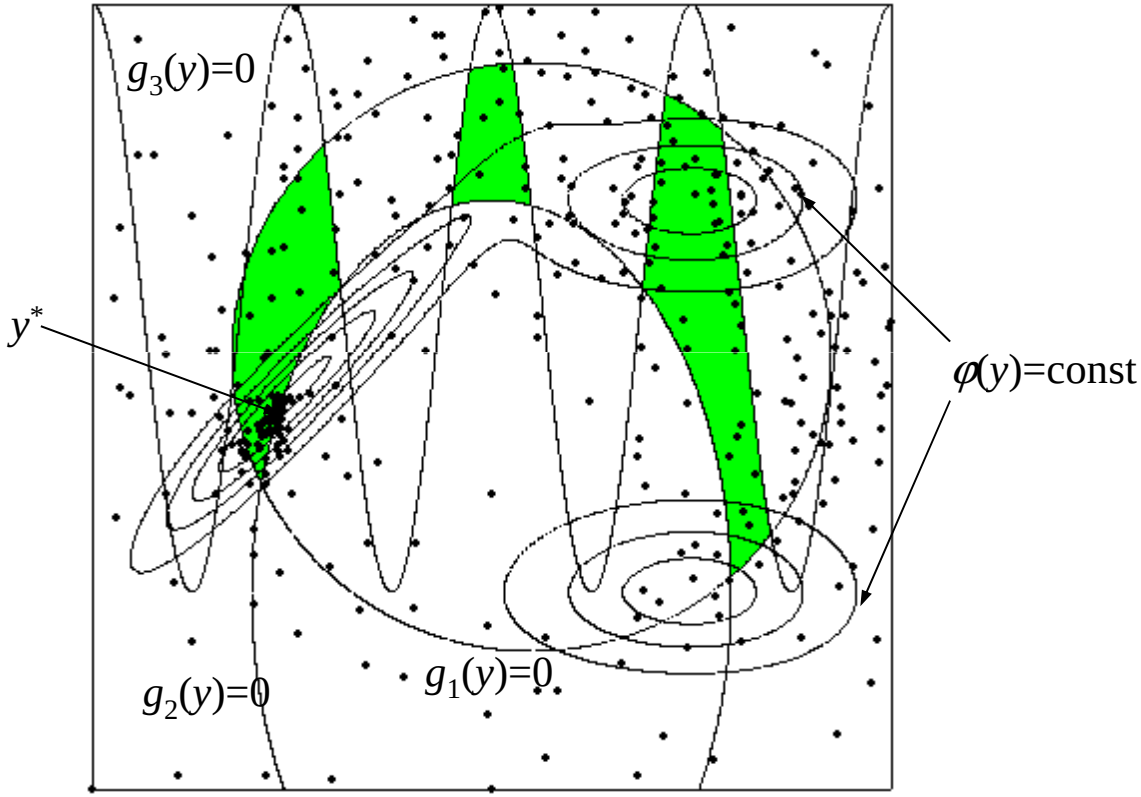


Расположение точек испытаний при использовании шести процессоров



Параллельный алгоритм

Ускорение сходимости (использование адаптивных резервов)



Расположение точек испытаний при использовании шести процессоров



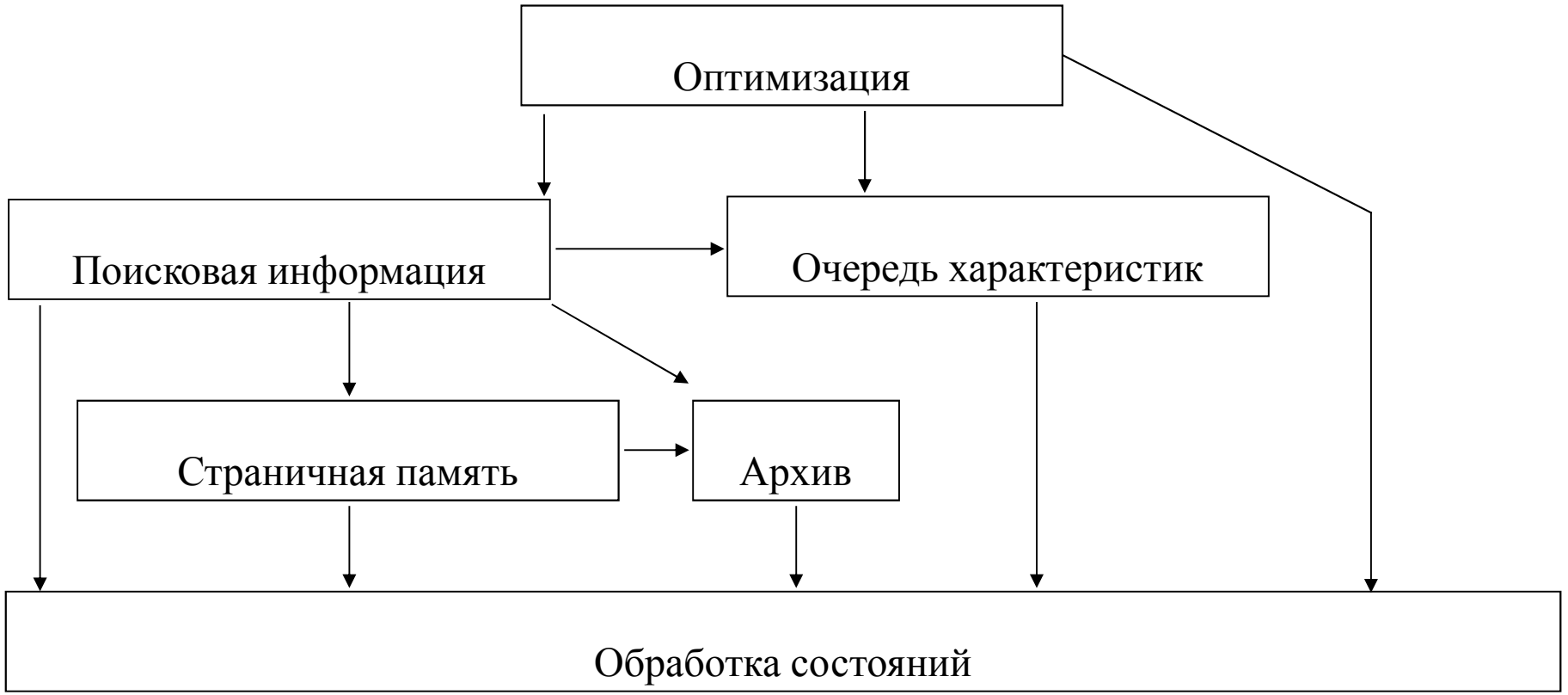
Программная реализация…

- Предварительные замечания:
  - Рассматриваются задачи, в которых вычисление значения функции в точке является достаточно трудоемким;
  - Объем вычислений, выполняемый на каждой итерации индексного метода, может существенно отличаться;
  - Вычислительные узлы при параллельном выполнении программы могут иметь разную мощность.

  Вывод: эффективная параллельная реализация индексного метода может быть только асинхронной.
- Программная система “Абсолют Эксперт” позволяет решать (исследовать) задачи многомерной глобальной оптимизации.
- Основной режим работы системы – параллельное выполнение на кластере.



Программная реализация…



Функциональная структура комплекса Абсолют Эксперт



Программная реализация

- Подсистема Оптимизация – построение объекта оптимизации, постановка задачи, определение и настройка методов, назначение заданий и выполнение процесса поиска оптимального варианта
- Подсистема Поисковая информация – структуры данных и методы, обеспечивающие получение, хранение и анализ информации, генерируемой в ходе итераций поиска
- Подсистема Страничная память – структуры данных и методы для организации страничного представления оперативной памяти, используемой для обработки поисковых данных
- Подсистема Архив – предназначена для организации хранения поисковой информации во внешней памяти
- Подсистема Очередь характеристик – ускоряет работу алгоритмов при больших объемах поисковой информации
- Подсистема Обработка состояний – унифицированная схема контроля и наблюдения за состояниями системы



Заключение

- В разделе рассмотрен индексный метод учета функциональных ограничений в задачах глобальной оптимизации.
- Представлена постановка задачи, изложена схема работы индексного метода для решения одномерных задач, приведена его программная реализация.
- Изложена схема редукции размерности задачи, основанная на развертках Пеано, приведена программная реализация разверток.
- Представлен параллельный алгоритм, основанный на множественных отображениях Пеано. Приведена программная реализация множественных отображений.
- Дана общая характеристика программной системы Абсолют Эксперт.
- Представлены результаты вычислительных экспериментов.



Вопросы для обсуждения

- В чем состоит задача поиска оптимального решения?
- В чем заключается сложность определения оптимума в задачах глобальной оптимизации?
- Чем обоснована необходимость поиска глобального оптимума на неравномерном покрытии области поиска?
- В чем состоит индексная схема учета ограничений ?
- Опишите класс характеристических алгоритмов оптимизации.
- В чем заключаются недостатки использования единственной развертки Пеано при редукции размерности?
- Каким образом множественные отображения позволяют распараллелить индексный алгоритм?
- Чем вызвана необходимость реализации асинхронной схемы расчетов в параллельной модификации индексного алгоритма?



Литература

- **Стронгин Р.Г.** (1978).Численные методы в многоэкстремальных задачах. – М.: Наука.
- **Strongin R.G.** (1992) Algorithms for multi-extremal mathematical programming problems employing the set of joint space-filling curves. Journal of Global Optimization, №2. P. 357–378.
- **Strongin R.G., Sergeyev Ya.D.** (2000). Global optimization with non-convex constraints. Sequential and parallel algorithms. – Dordrecht: Kluwer Academic Publishers.



Следующая тема

- **Программная система ПараЛаб для изучения и исследования методов параллельных вычислений**



Авторский коллектив

Гергель В.П., профессор, д.т.н., руководитель

Гришагин В.А., доцент, к.ф.-м.н.

Баркалов К.А., ст. преподаватель, к.ф.-м.н. (раздел 14)

Сысоев А.В., ассистент (разделы 1)

Лабутин Д.Ю., ассистент (система ПараЛаб)

Абросимова О.Н., ассистент (раздел 10)

Гергель А.В., аспирант (раздел 12)

Лабутина А.А., магистр (разделы 7,8,9, система ПараЛаб)

Сенин А.В. (раздел 11)



© Гергель В.П.

О проекте

Microsoft

Целью проекта является создание образовательного комплекса "Многопроцессорные вычислительные системы и параллельное программирование", обеспечивающий рассмотрение вопросов параллельных вычислений, предусматриваемых рекомендациями Computing Curricula 2001 Международных организаций IEEE-CS и ACM. Данный образовательный комплекс может быть использован для обучения на начальном этапе подготовки специалистов в области информатики, вычислительной техники и информационных технологий.

Образовательный комплекс включает **учебный курс "Введение в методы параллельного программирования" и лабораторный практикум "Методы и технологии разработки параллельных программ"**, что позволяет органично сочетать фундаментальное образование в области программирования и практическое обучение методам разработки масштабного программного обеспечения для решения сложных вычислительно-трудоемких задач на высокопроизводительных вычислительных системах.

Проект выполнялся в Нижегородском государственном университете им. Н.И. Лобачевского на кафедре математического обеспечения ЭВМ факультета вычислительной математики и кибернетики (http://www.software.unn.ac.ru). Выполнение проекта осуществлялось при поддержке компании Microsoft.