Итерационные методы решения линейных уравнений

Метод Якоби

Метод простейший.

Он не является приемлемым для большинства задач, но представляет собой удобную отправную точку для обсуждения итерационных методов

Пусть А – невырожденная матрица n×n и нужно решить систему

$$A\,\mathbf{x} = \mathbf{b}$$

где $a_{ii} \neq 0$

Тогда метод Якоби имеет вид

$$x_i^{k+1} = \frac{1}{a_{ii}}\left(-\sum_{j \neq i} a_{ij} x_j^k + b_i\right)$$

$$i = 1, \ldots, n$$

$$k = 0, 1, \ldots$$

$\left\{ x_i^0 \right\}$ - начальное приближение

Метод Якоби можно переписать:

$$] \, D = diag(a_{11}, \ldots, a_{nn}); \; B = D - A$$

Тогда

$$\vec{x}^{k+1} = H\vec{x}^{k} + \vec{d}; \quad H = D^{-1}B; \quad d = D^{-1}b \qquad (\bullet)$$

Th1. Если матрица *A* имеет строгое диагональное преобладание или Является неприводимой с диагональным преобладанием, то итерации метода Якоби сходятся при любом начальном приближении $\mathbf{x}^0$

Th2. Если матрица *A=D-B* симметрична и положительно определена, то итерации метода Якоби сходятся при любом начальном приближении $\mathbf{x}^0$ тогда и только тогда, когда матрица *D+B* положительно определена.

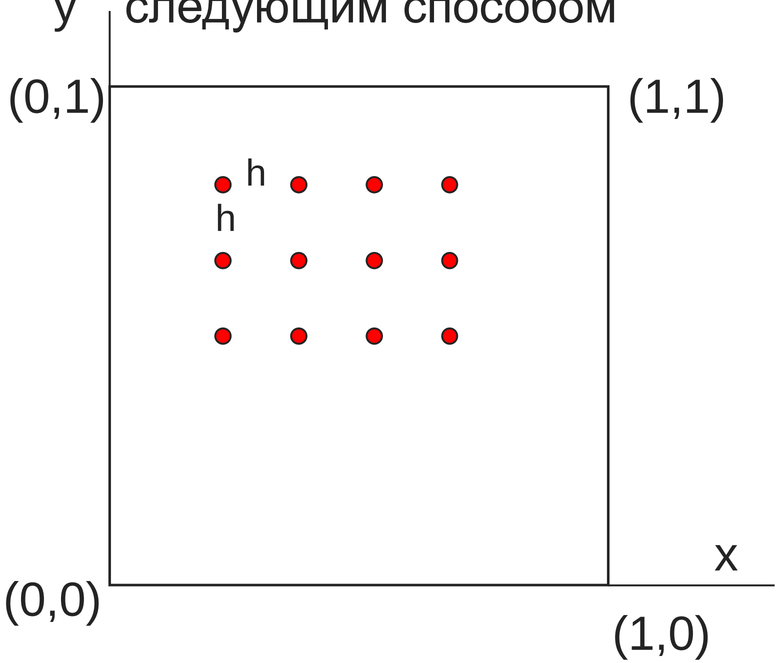
Главная операция в **(•)** – *матрично-векторное умножение (МВУ)*
Эффективная параллельная или векторная реализация *МВУ* существенно зависит от структуры *H*
Итерационные методы применяются, когда *A(H)* – большие разреженные матрицы

Уравнение Пуассона

$$u_{xx} + u_{yy} = f,$$

$$(x, y) \in \Omega = [0,1] \times [0,1]$$

Введем дискретизацию единичного квадрата Ω с шагом по пространству *h* следующим способом



(*ih*, *jh*) – сетка, u_{ij} – значения

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = h^2 f_{i,j}$$

$$i, j = 1, \ldots, N$$

ЛС из $n = N^2$ уравнений

Полученную систему можно записать

$$\mathbf{A\,x = b}$$

где $x_k = u_{1k}$, k=1,…,N – первая линия узлов
$x_k = u_{2,k-N}$, k=N+1, …, 2N – вторая линия и т.д.

$$x^T = \left(u_{1,1},\dots,u_{1,N},u_{2,1},\dots,u_{2,N},\dots,u_{N,1},\dots,u_{N,N}\right)$$

$$A = \begin{bmatrix} T & -I & & & \\ -I & T & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -I \\ & & & -I & T \end{bmatrix}; \quad T = \begin{bmatrix} 4 & -1 & & & \\ -1 & \ddots & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 4 \end{bmatrix}$$

I, T – размерностью NxN; *A* – размерность NxN

Диагонально разреженная пяти диагональная матрица

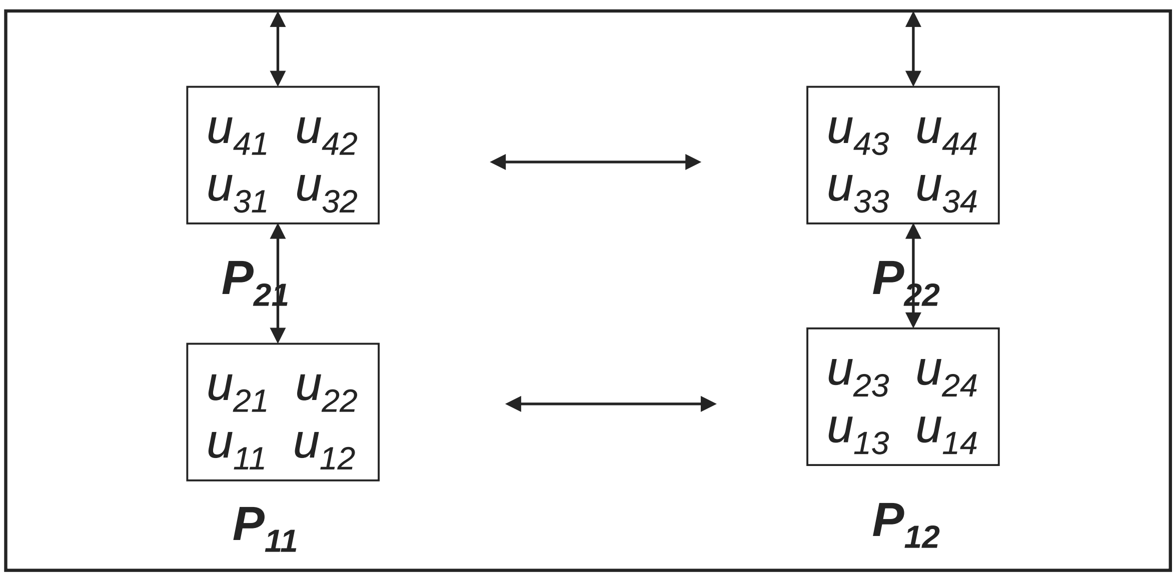
Метод Якоби применяется следующим образом

$$u_{i,j}^{k+1} = \frac{1}{4}\left(u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,j-1}^{k} - h^2 f_{i,j}\right),$$

$$i, j = 1,\dots, N; \quad k = 0,1,\dots$$

Видно, что эти значения можно вычислять ***параллельно.***
Метод Якоби иногда рассматривают, как прототип параллельного метода

Предположим, что параллельная система образована сетью из $p=q^2$ процессоров P_{ij}, упорядоченных в виде двумерной решетки.
Предположим, $N^2 = mp$.
Тогда естественна обработка каждым процессором m неизвестных.



Теоретический случай $p=N^2$ процессоров
Необходима пересылка между процессорами после каждого хода

вычисляем u_{ij}^{k+1} в процессоре $P_{i\,j}$;
посылаем значение u_{ij}^{k+1} процессорам $P_{i+1,\,j}$, $P_{i-1,\,j}$, $P_{i,\,j+1}$, $P_{i,\,j-1}$

Более обычной является ситуация, когда $N^2=mp$ и каждый процессор содержит m неизвестных.
Например, для предыдущего распределения данных вычисления, выполняемые процессором P_{22} будут

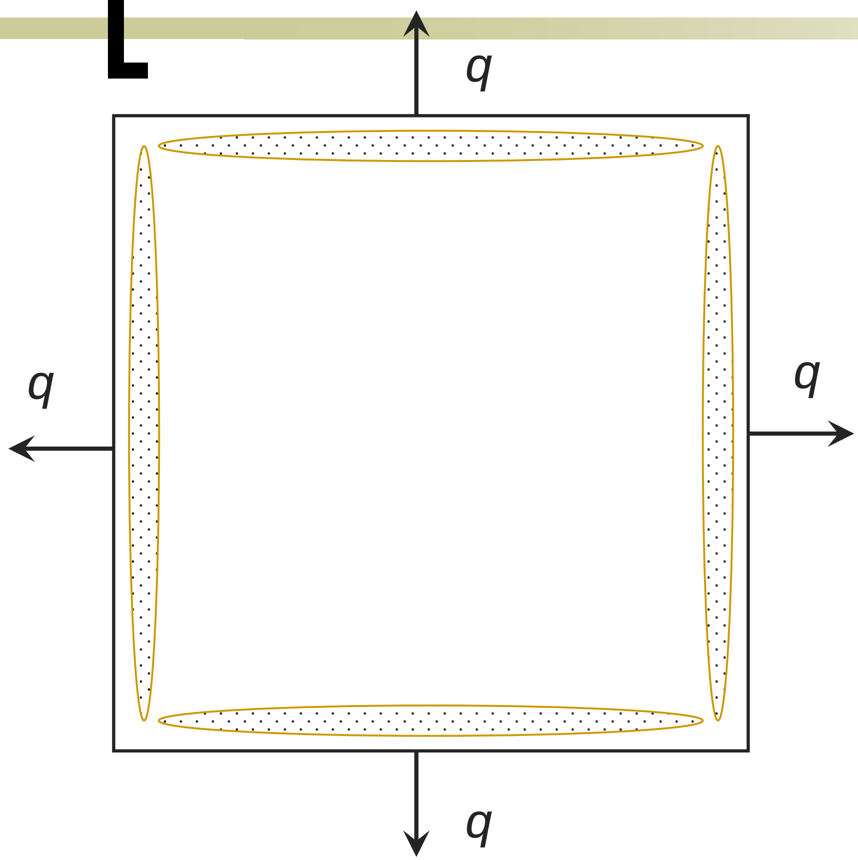
вычислить u_{33}^{k+1}, u_{34}^{k+1}; передать их в P_{12};

вычислить u_{43}^{k+1}; передать u_{33}^{k+1}, u_{43}^{k+1} в P_{21};
…………………………………………

Аналогичные вычисления параллельно выполняются в других процессорах

Будем называть *внутренними граничными значениями* те значения u_{ij}, которые необходимы другим процессорам на следующей итерации и, таким образом, подлежат пересылке.

Если процессор содержит $m=q^2$ неизвестных, то для всех m необходимо вычислить новые значения, однако переслать нужно не более $4q$ значений



Таким образом, при фиксированном числе процессоров по мере возрастания размера задачи будет увеличиваться также и отношение времени вычислений ко времени обменов

$\uparrow$ ***m* в процессоре** $\Rightarrow \downarrow \boldsymbol{t_{выч.}/t_{обм.}}$

В частности, количество внутренних точек сетки, для которых НЕ требуется обмен данными, возрастает как N^2, а число точек сетки, для которых необходим обмен данными, возрастает линейно по N. Т.о. для достаточно больших задач время обмена становится менее значительным по сравнению со временем вычислений.

Синхронные и асинхронные методы

Корректное выполнени… аций Якоби требует введени… анизма синхронизации

$$u_{22}^{k+1} \qquad u_{32}^{k}, \quad u_{23}^{k} \qquad u_{32}^{k-1}$$

Время на синхронизацию, так же как и время на ожидание того, чтобы все необходимые данные стали доступны для продолжения вычислений, вносит вклад в накладные вычислительные затраты.

Для метода Якоби может быть предложена альтернатива асинхронной работы процессоров, т.е. без какой-либо синхронизации.

Полученное новое итерационное значение не будет в точности приближением Якоби. Однако это не обязательно нанесет ущерб итерационному процессу в целом, но может оказаться более выгодным с точки зрения общих затрат.
Метод Якоби, реализованный указанным способом, будем называть *асинхронным* методом