PROJECT 4: PARALLEL CYCLIC AND HOUSEHOLDER REDUCTIONS

Programming Graphics Processing Units (GPUs)

INTRODUCTION

- Problématique du projet
- Méthodes
  - Householder
  - Parallel cyclic reduction (PCR)

HOUSEHOLDER

L'algorithme Householder réduit une matrice symétrique A de dimension n*n en forme tridiagonale par n-2 transformations orthogonales.

$$\mathbf{P}_1 \cdot \mathbf{A} = \left[\begin{array}{c|ccc} 1 & 0 & 0 & \cdots & 0 \\ \hline 0 & & & & \\ 0 & & & & \\ \vdots & & {}^{(n-1)}\mathbf{P}_1 & & \\ 0 & & & & \end{array}\right] \cdot \left[\begin{array}{c|ccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ \hline a_{21} & & & & \\ a_{31} & & & & \\ \vdots & & \text{irrelevant} & & \\ a_{n1} & & & & \end{array}\right] = \left[\begin{array}{c|ccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ \hline k & & & & \\ 0 & & & & \\ \vdots & & \text{irrelevant} & & \\ 0 & & & & \end{array}\right]$$

$$\mathbf{A}' = \mathbf{P} \cdot \mathbf{A} \cdot \mathbf{P} = \left[\begin{array}{c|ccc} a_{11} & k & 0 & \cdots & 0 \\ \hline k & & & & \\ 0 & & & & \\ \vdots & & \text{irrelevant} & & \\ 0 & & & & \end{array}\right]$$

$$\mathbf{P}_2 \equiv \left[\begin{array}{cc|ccc} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \hline 0 & 0 & & & \\ \vdots & \vdots & & {}^{(n-2)}\mathbf{P}_2 & \\ 0 & 0 & & & \end{array}\right]$$

$$\mathbf{Q} = \mathbf{P}_1 \cdot \mathbf{P}_2 \cdots \mathbf{P}_{n-2}$$

PARALLEL CYCLIC REDUCTION (PCR)

$$inv(Q) = Q'$$
$$Q' * A * Q = tridiag\_A$$
$$Q * tridiag\_A * Q' = A$$
$$A * x = b$$
$$Q * tridiag\_A * Q' * x = b$$
$$tridiag\_A * Q' * x = Q' * b$$
$$Q' * x = z$$
$$Q' * b = y$$

$$\longrightarrow \quad tridiag\_A * z = y \quad \longrightarrow \quad \begin{pmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & 0 \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & \ddots \\ & 0 & & \ddots & \ddots & c_{d-1} \\ & & & & a_d & b_d \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_d \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_d \end{pmatrix} \longrightarrow \quad x = Q * z$$

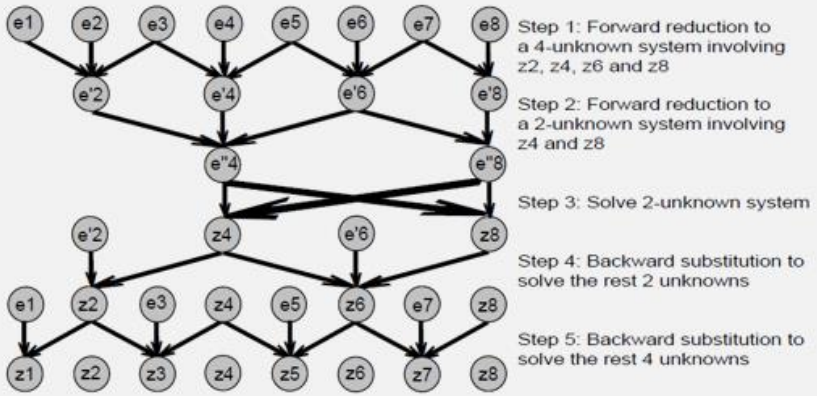


Figure 1: CR when $d = 8$.

La réduction cyclique parallèle (PCR) est une méthode alternative à l'algorithme de Thomas pour la résolution de systèmes linéaires tridiagonaux.

Il est stable pour les matrices à diagonale dominante ou les matrices symétriques et positives.

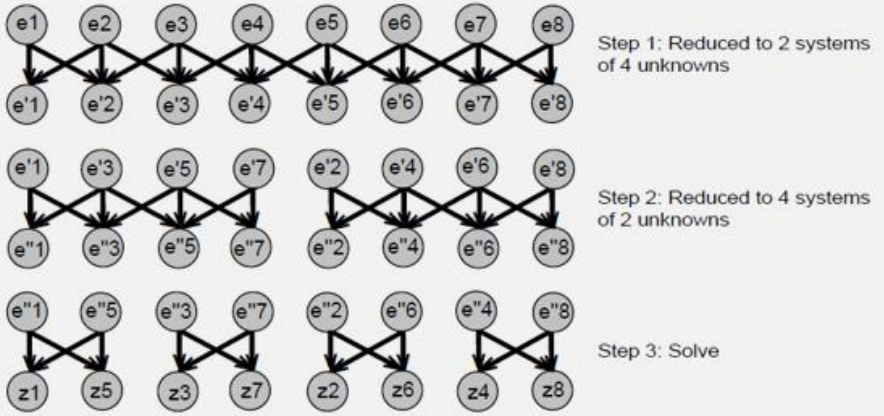


Figure 2: PCR when $d = 8$.

__global__ void PCR_Kernel(float* Q, float* alist, float* blist, float* clist, float* dlist, float* xlist, int iter_max, int DMax)

Calculer plusieurs matrices simultanément (batch)

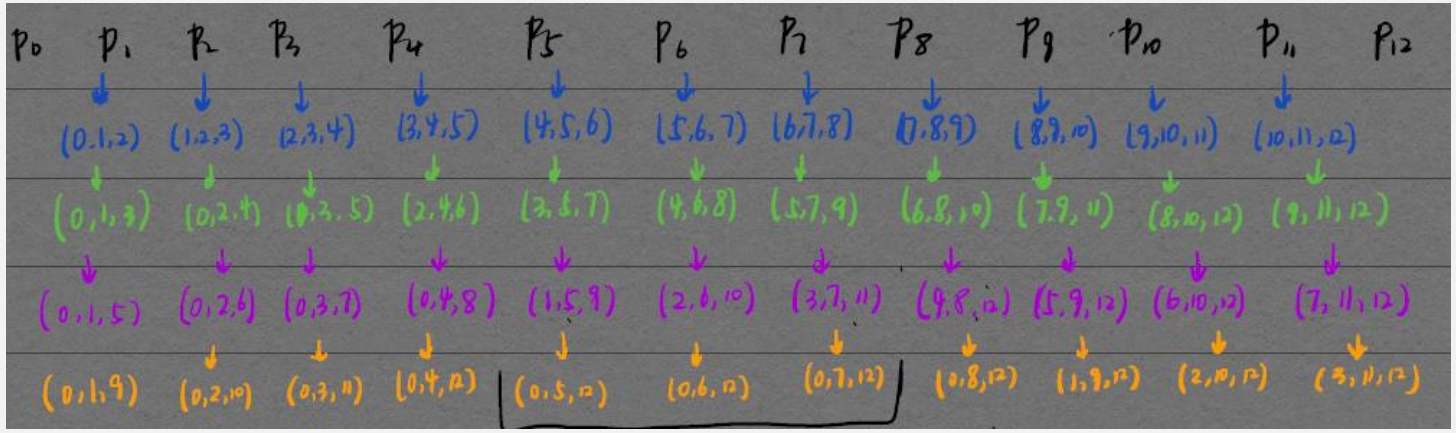
```
tidx = threadIdx.x%d;
Qt = (threadIdx.x-tidx)/d;
gbx = Qt + blockIdx.x*(blockDim.x/d);

P = (d / 2 + (d % 2))*(tidx % 2) + (int)tidx / 2;
```

```
// Shared memory
extern __shared__ float  shared_arr[];
```

Calculer une seule matrice

```
tidx = threadIdx.x;
Qt = 0;
gbx = 0;
```

dimension = 11



La condition d'arrêt

```
int pos = idx_row-2*stride;
accum = 0;
for ( int iter = 0; iter<5; iter++ ) {
        if (pos >=0 && pos < DMax)
                accum++;
        pos+=stride;
}

if (accum < 3) {
        next_or_ot = false;//Turn of for ever
}
```

RÉSULTAT

```
This is the random symmetric matrix A and the random second member d:
39.500000 36.500000 47.500000 14.500000 16.500000 27.500000 36.500000 24.500000 32.500000 42.500000 12.500000     25.150000
36.500000 41.500000  6.500000  9.500000 29.500000  8.500000 46.500000 19.500000  4.500000 48.500000 45.500000     59.150002
47.500000  6.500000 38.500000 16.500000 13.500000 23.500000  4.500000 11.500000 14.500000 36.500000 24.500000     44.150002
14.500000  9.500000 16.500000 44.500000  9.500000 23.500000 29.500000  7.500000  8.500000 21.500000 14.500000     64.150002
16.500000 29.500000 13.500000  9.500000 11.500000 16.500000 38.500000 15.500000 20.500000 22.500000  7.500000     90.150002
27.500000  8.500000 23.500000 23.500000 16.500000 26.500000 37.500000 35.500000 20.500000 19.500000  2.500000     31.150000
36.500000 46.500000  4.500000 29.500000 38.500000 37.500000 28.500000 28.500000 35.500000 22.500000  3.500000     99.150002
24.500000 19.500000 11.500000  7.500000 15.500000 35.500000 28.500000 33.500000 11.500000 15.500000 35.500000     21.150000
32.500000  4.500000 14.500000  8.500000 20.500000 20.500000 35.500000 11.500000  9.500000 27.500000 15.500000     93.150002
42.500000 48.500000 36.500000 21.500000 22.500000 19.500000 22.500000 15.500000 27.500000 34.500000 44.500000     12.150000
12.500000 45.500000 24.500000 14.500000  7.500000  2.500000  3.500000 35.500000 15.500000 44.500000 15.500000     91.150002

This is the orthogonal matrix Q effecting the transformation in householder:
 0.403393 -0.227855 -0.466289  0.105253 -0.289330 -0.122297  0.192509  0.326628 -0.539200 -0.153952  0.000000
 0.032950  0.027391  0.159629 -0.331361  0.278660 -0.403326  0.527641  0.084265  0.151731 -0.560385  0.000000
-0.439959  0.027320 -0.159180  0.771288  0.199935  0.062119  0.206896 -0.068615 -0.055396 -0.301746  0.000000
-0.037413  0.628588 -0.514316 -0.161044  0.032171 -0.264316 -0.304677 -0.329256 -0.094838 -0.178584  0.000000
-0.749610 -0.162576 -0.029751 -0.407724 -0.362505  0.073790  0.043231  0.025669 -0.310829 -0.092371  0.000000
 0.063817  0.081039  0.602840  0.235490 -0.245782 -0.485311 -0.152121 -0.263860 -0.425255 -0.030790  0.000000
 0.030509 -0.034311  0.126396 -0.173857  0.696382  0.292390 -0.167271  0.000861 -0.592800 -0.043107  0.000000
 0.048428 -0.628116 -0.118252 -0.010936  0.044159 -0.117351 -0.536540 -0.238221  0.193678 -0.437223  0.000000
 0.247761 -0.020161  0.028397 -0.068873 -0.225746  0.486193  0.353076 -0.692119 -0.071963 -0.190900  0.000000
 0.102834  0.349786  0.268639  0.038951 -0.260673  0.414767 -0.288745  0.408700  0.051182 -0.548069  0.000000
 0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  1.000000
```

```
 This is the diagonal elements of the tridiagonal matrix in householder:
 -3.106331
 29.146309
 -0.560986
 41.366047
-16.924112
 -2.777148
  5.007463
  6.589115
 86.690353
162.569305
 15.500000

 This is the off-diagonal elements of the tridiagonal matrix in householder:
  0.000000
 -4.644855
-17.246964
-15.756526
-24.935919
-14.538093
 21.388615
 31.862530
 33.575016
116.066551
-81.194214

 The maximum number of PCR iteration is: 3

 The final resolution is :
 -7.361708
-10.326294
 -9.163260
 -0.251056
 -0.432338
 -1.302601
 -1.338186
 -1.061947
 21.210640
 11.389147
  6.094026
```

MERCI!