NUMERICAL LINEAR ALGEBRA

Prof. Paola Antonietti

MOX - Dipartimento di Matematica

Politecnico di Milano

https://antonietti.faculty.polimi.it

TA: Dr. Michele Botti

MODELLISTICA E CALCOLO SCIENTIFICO
MOX
MODELING AND SCIENTIFIC COMPUTING

POLITECNICO MILANO 1863 | DEPARTMENT OF MATHEMATICS

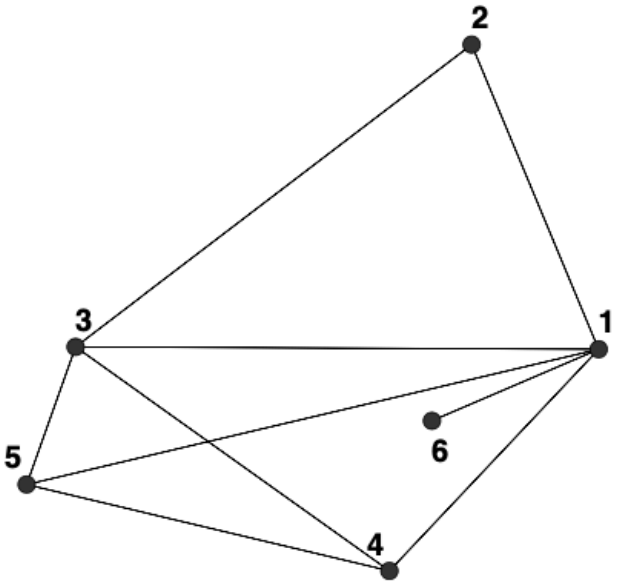
P6: Algebraic Multigrid Methods

The principle of AMG

- AMG is based on MG principles but uses matrix coefficients instead of geometric information.

- Sparse matrix (left) and the associated graph (right)

$$A = \begin{pmatrix} * & * & * & * & * & * \\ * & * & * & 0 & 0 & 0 \\ * & * & * & * & * & 0 \\ * & 0 & * & * & * & 0 \\ * & 0 & * & * & * & 0 \\ * & 0 & 0 & 0 & 0 & * \end{pmatrix}$$



“Classical” AMG
by Brandt, McCormick, Ruge

A. Brandt, S. F. McCormick, and J. W. Ruge. Algebraic multigrid (AMG) for automatic multigrid solution with application to geodetic computations. Tech. Rep., Institute for Computational Studies, Colorado State University, 1982.

A. Brandt, S. McCormick, and J. Ruge. Algebraic multigrid (AMG) for sparse matrix equations. In Sparsity and its applications (Loughborough, 1983), pages 257–284. Cambridge Univ. Press, Cambridge, 1985.

Some general remarks

- Classical AMG is based on the observation that the algebraic smooth error varies slowly in the direction of the matrix's relatively large (negative) coefficients.

- This gives us an algebraic way to track smooth errors. However, we still need to define “large”.

Definition: Strong Connection

- Given a threshold $\theta \in (0,1)$, we say that i is *strongly connected* with j if

$$-a_{i,j} \geq \theta \text{max}_{k\neq i}(-a_{i,k})$$

- Let us denote by S_i the set of vertices that i is strongly connected to by

$$S_i = \{j \in N_i : i \text{ strongly connets to } j\}$$

- where $N_i = \left\{ j \neq i : a_{i,j} \neq 0 \right\}$

- This gives us a strength matrix S, with S_i as its $i-$th row.

Standard coarsening

- Since algebraic smooth error varies slowly in the direction of strong connections, we should essentially coarsen in the direction of strong connections.
- This results in a C/F-splitting such that F-vertices strongly connect to neighbouring C-vertices, and then the idea is to represent the values of a smooth vector at F- vertices as a linear combination of the values at C-vertices.

Given a strength matrix S, a general coarsening procedure consists of two steps:

```
Algorithm: General Coarsening Algorithm

  1. Choose an independent set of C-vertices based on
     the graph of $S$;
  2. Choose additional C-vertices in order to satisfy the
     interpolation requirements.
```

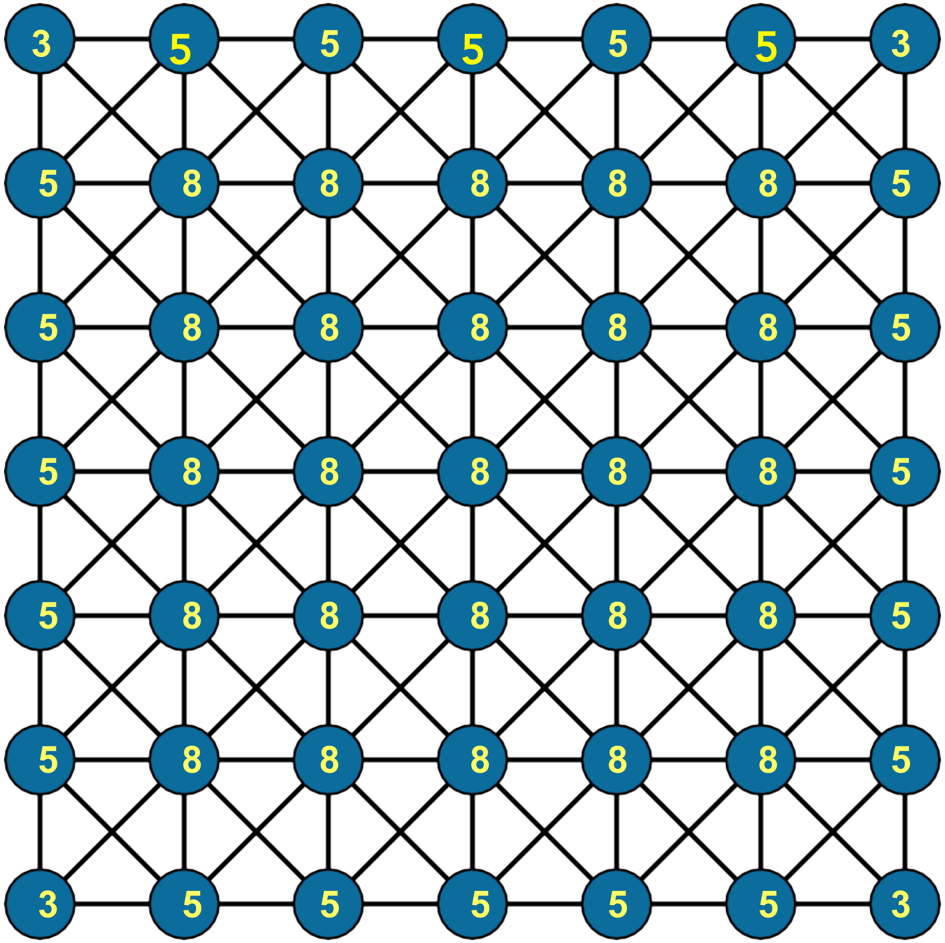
Standard coarsening

- For standard coarsening, the strength matrix S is constructed based on A directly and the general coarsening algorithm is applied.
- In step (1) of the Algorithm, the independent set can be chosen in the following way:

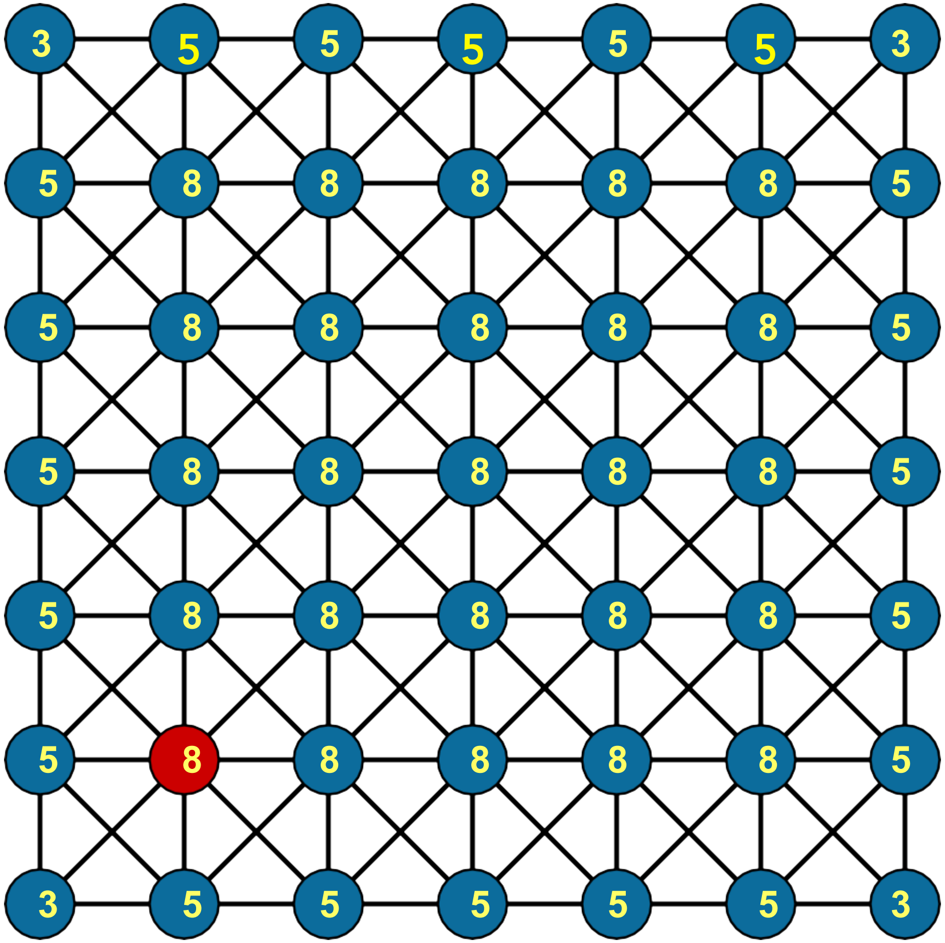
  1. Start with a first vertex i to become a C-vertex
  2. Then all vertices j which i strongly connect to become F-vertices
  3. Next, another vertex from undecided vertices becomes a C-vertex, and all vertices which are strongly connected to it become F-vertices.
  4. This procedure is repeated until all vertices become C or F-vertices.

Remark: this procedure highly depends on the order of the vertices. To obtain a relatively uniform distribution of C/F vertices a suitable measure of importance is introduced (CAMG).

Coarsening (idea)



Coarsening (idea)

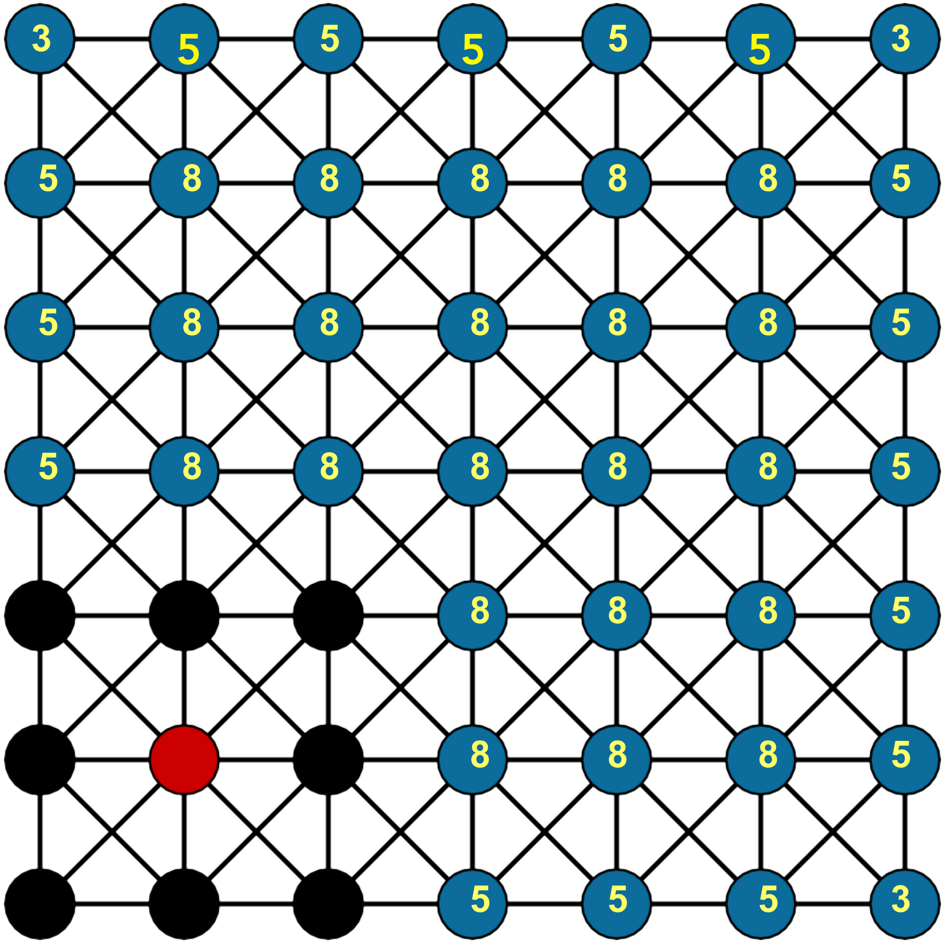


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)

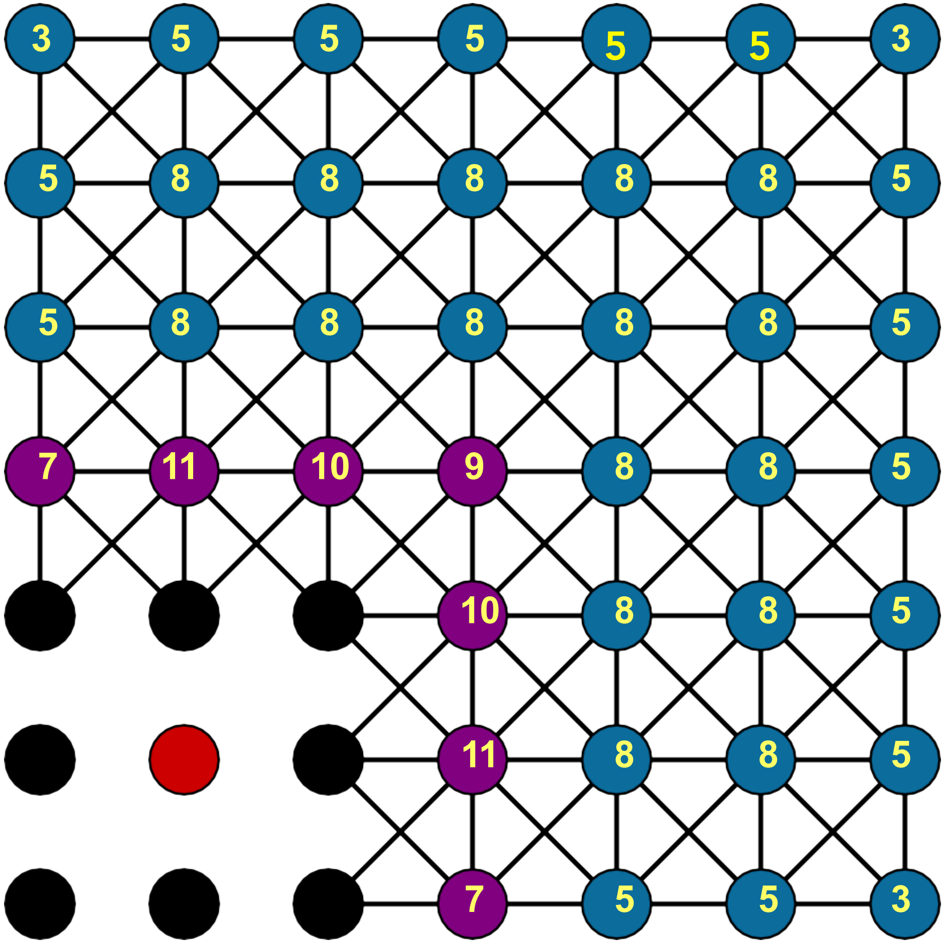


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)

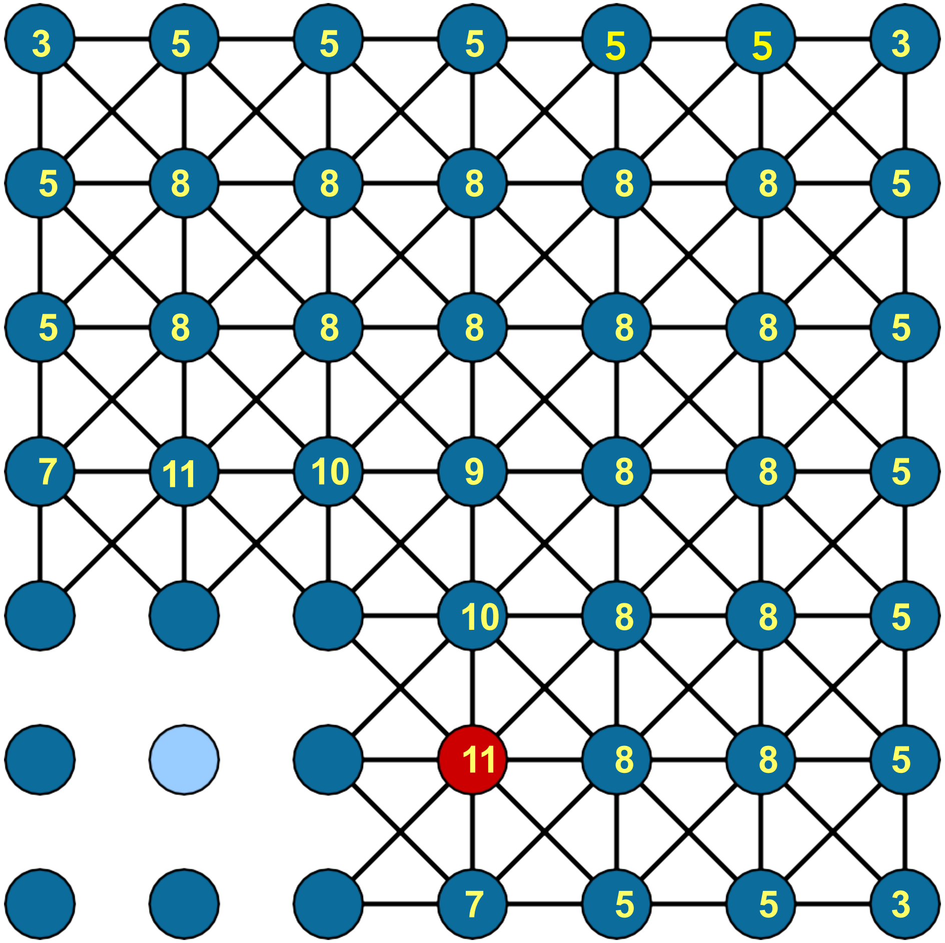


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)

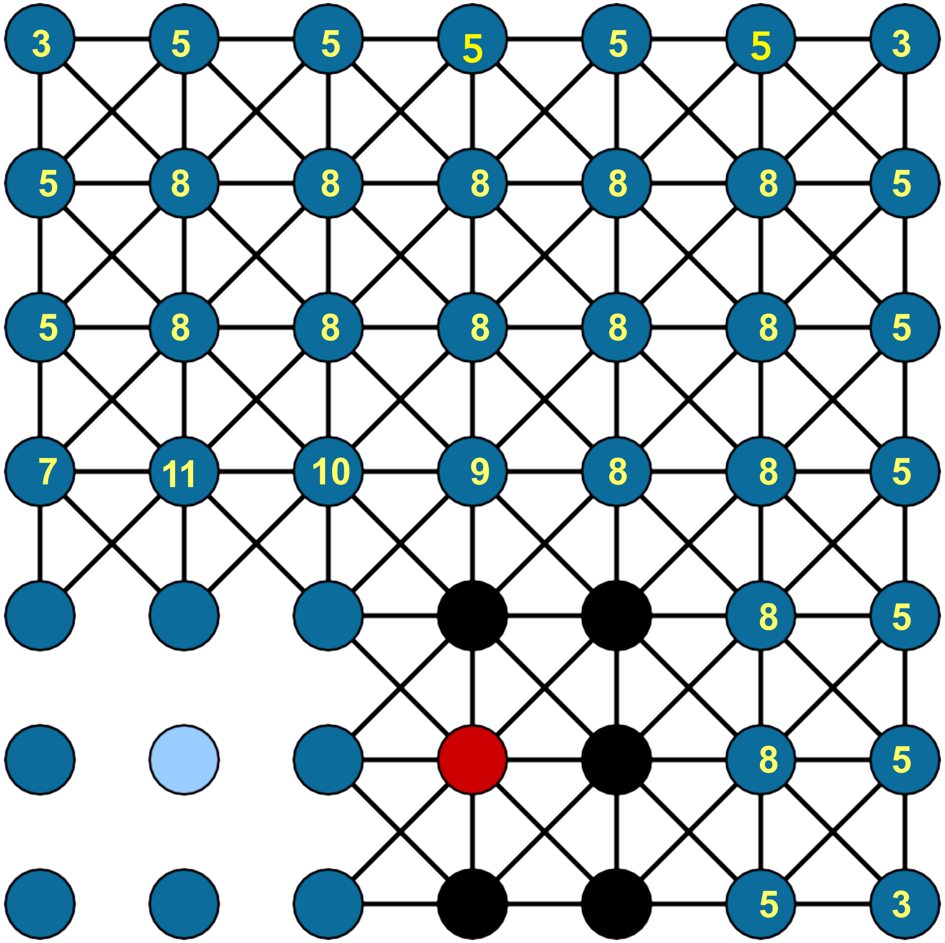


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)

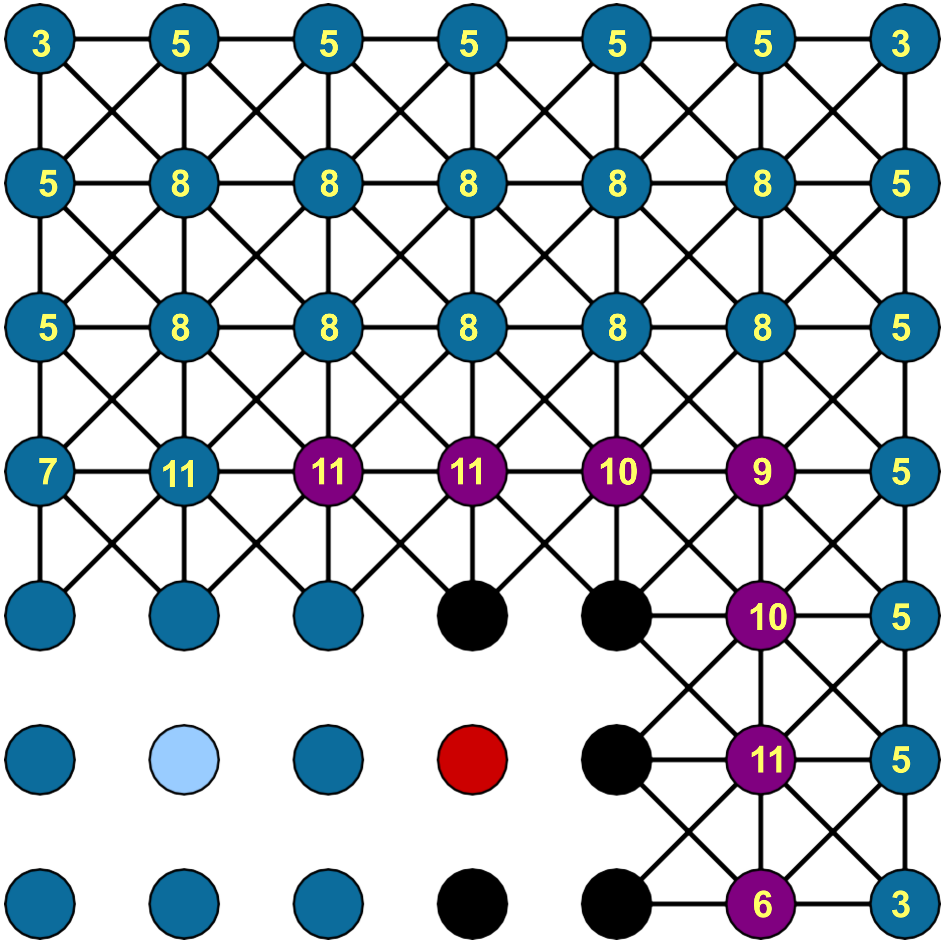


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)

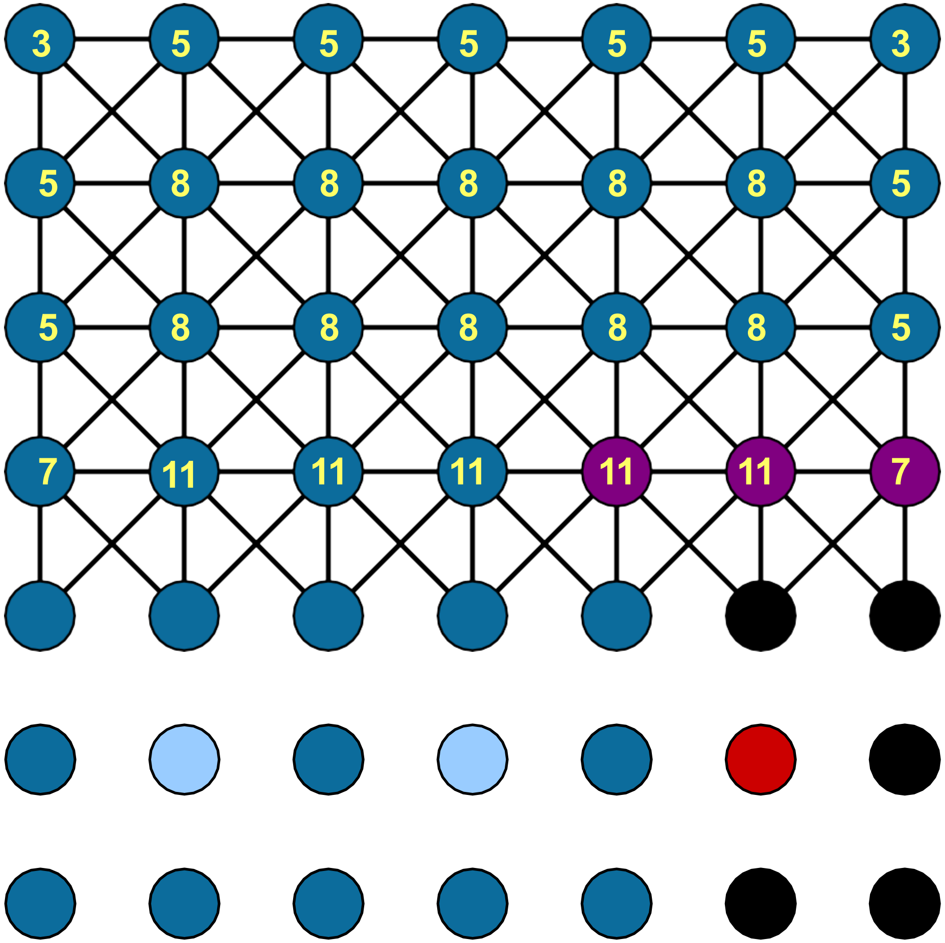


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)

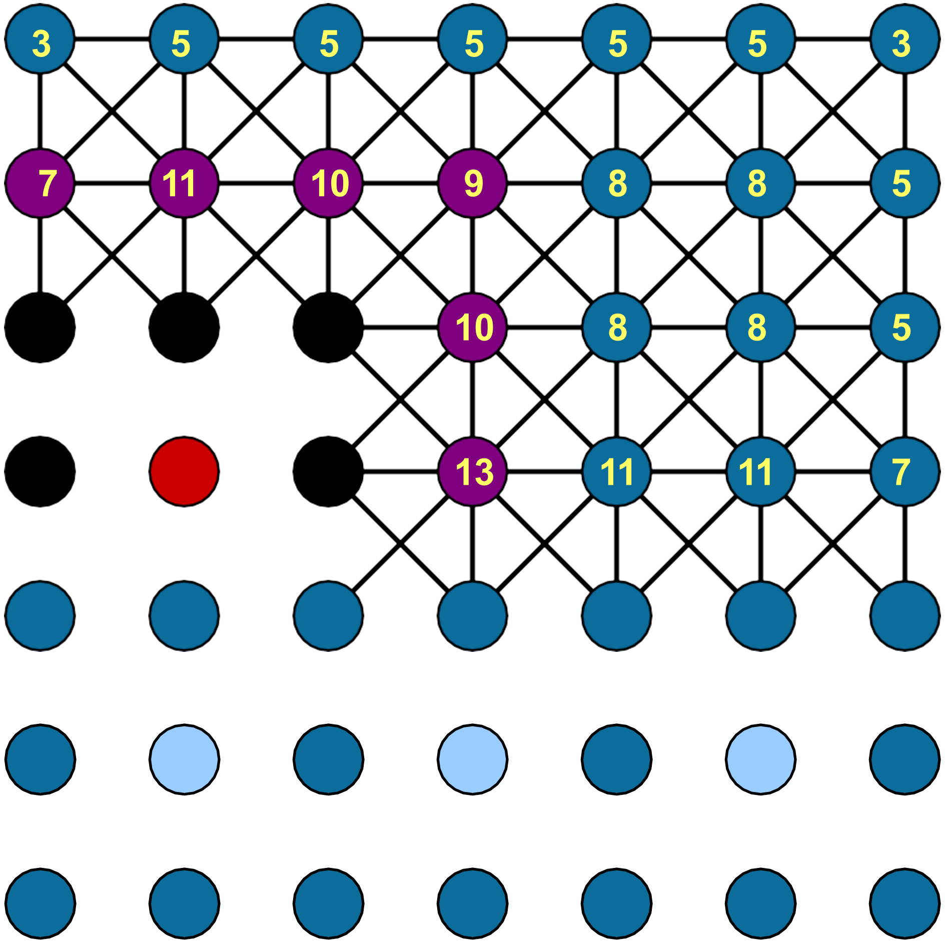


- select (suitably) C-pt

- select neighbours as F-pts

- update measures of F-pt neighbours

Coarsening (idea)



- select (suitably) C-pt
- select neighbours as F-pts
- update measures of F-pt neighbours

The interpolation step (the idea)

- Let $\mathbf{e} = (e_1, e_2, \ldots, e_i, \ldots)$ be the error.
- A simple characterization of algebraic smooth error is $A\mathbf{e} \approx 0$
- In other words

$$a_{i,i}e_i + \sum_{j \in N_i} a_{i,j}e_j \approx 0 \qquad i \in F \qquad (1)$$

- The idea is that we want to choose proper $w_{i,j}$ such that for any algebraic smooth errors

$$e_i \approx \sum_{j \in C} w_{i,j}e_j \qquad i \in F$$

Direct Interpolation (idea)

- For $i \in F$, we define

  C_i : $C-$points strongly connected to i $(C_i = C \cap N_i)$

  C_i^S :$C_i^S = C \cap S_i$

  F_i^S : $F-$points strongly connected to i $(F_i = F \cap N_i)$

  N_i^W : all points weakly connected to i $(N_i^W = N_i \backslash (C_i^S \cup F_i^S)$

- We can rewrite (1) as follows (replace $\approx$ with $=$)

$$a_{i,i}e_i + \alpha \sum_{j \in C_i^S} a_{i,j}e_j = 0 \qquad \alpha = \frac{\sum_{j \in N_i} a_{i,j}}{\sum_{j \in C_i^S} a_{i,j}}$$

Direct Interpolation (idea)

- Therefore, the formula of direct interpolation is

$$w_{i,j} = \alpha \frac{a_{i,j}}{a_{i,i}}, \quad i \in F, j \in C_i^S, \qquad \alpha = \frac{\sum_{j \in N_i} a_{i,j}}{\sum_{j \in C_i^S} a_{i,j}}$$

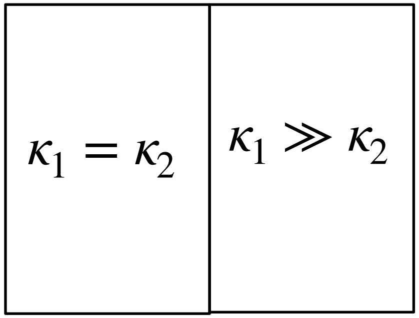
- The above direct interpolation can be applied as long as C_i^S.

Standard Interpolation (idea)

- A nature modification of direct interpolation consists of taking F_i^S into account.
- Vertices $j \in F_i^S$ are F-vertices and cannot be included in the interpolator directly.
- However, we can use its neighbourhood N_j to substitute it and include its C neighbour C_j^S.
- This leads to indirect interpolation, which usually is called standard interpolation

A numerical example

$$-\nabla(\mathbf{K}\nabla u) = f \qquad \text{in } \Omega \qquad\qquad \mathbf{K} = \begin{pmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{pmatrix}$$



	# it	Conv factor	Coarse grid	Setup time	Solution time
61×61	10	0.23	6	0.01	0.02
121×121	9	0.23	8	0.05	0.07
241×241	9	0.23	9	0.25	0.32
481×481	9	0.23	12	1.02	1.27
961×961	11	0.29	13	4.42	6.28

Parallel Coarsening Algorithms

- C-AMG coarsening algorithm is inherently sequential
- Several parallel algorithms (in *hypre*):
  - CLJP (Cleary-Luby-Jones-Plassmann): one-pass approach with random numbers to get concurrency
  - Falgout – C-AMG on processor interior, then CLJP to finish
  - CGC (Griebel, Metsch, Schweitzer): compute several coarse grids on each processor, then solve a global graph problem to select the grids with the best “fit
  - Other parallel AMG codes use similar approaches

Take home message

- AMG: the construction of the MG hierarchy is carried out using only information from the matrix and not from the geometry of the problem
- Optimal convergence (linear complexity) and good scaling potential, but exposing high parallelism is not easy at “too coarse” levels
- To achieve parallelism additional restrictions on AMG algorithmic development