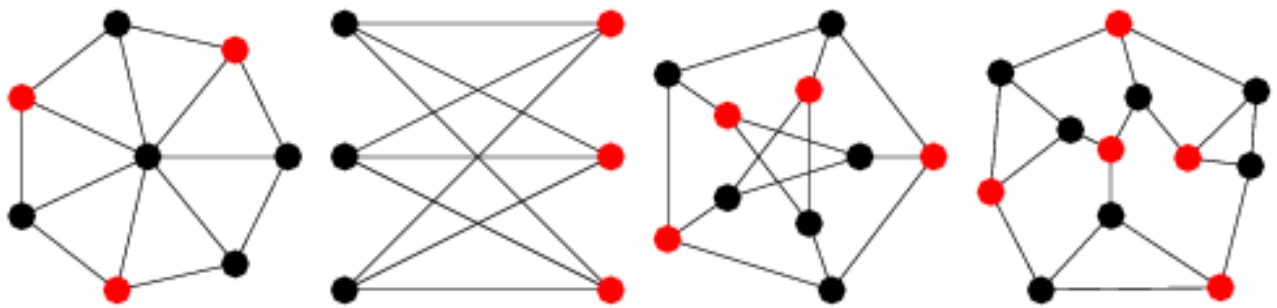
Algorithmic Strategies 2023/24

Week 7 – Branch-and-Bound

UNIVERSIDADE DE COIMBRA

Maximum independent set

Given an undirected graph $G = (V, E)$ a subset of nodes $S \subseteq V$ is an independent set iff there is no edge in E between any two nodes in S. The problem consists of finding an independent set in G of maximum cardinality.



Maximum independent set

Function $mis(x, v)$
  if $|x| > |x^*|$ **then**
    $x^* = x$
  for $i = v + 1$ **to** n **do**
    if $(i, j) \notin E$ for every node $j \in x$ **then**
      $mis(x \cup \{i\}, i)$ {recursive step}

- Backtracking that checks whether node i is neighbor of any other node already selected.

Maximum independent set

Function $mis(x, v)$
  if $|x| > |x^*|$ **then**
    $x^* = x$
  for $i = v + 1$ **to** n **do** {mark neighbors}
    if $(i, v) \in E$ **then**
      $neighbor[i] = neighbor[i] + 1$
  for $i = v + 1$ **to** n **do**
    if $neighbor[i] = 0$ **then** {rejection test}
      $mis(x \cup \{i\}, i)$ {recursive step}
  for $i = v + 1$ **to** n **do** {undo the counting}
    if $(i, v) \in E$ **then**
      $neighbor[i] = neighbor[i] - 1$

- Backtracking with *look-ahead* to avoid unnecessary recursions.

Maximum independent set

Function $mis(x, v)$
  if $|x| > |x^*|$ **then**
    $x^* = x$
  if $g(x) \leq |x^*|$ **then** {bounding test}
    return
  for $i = v + 1$ **to** n **do** {mark neighbors}
    if $(i, v) \in G$ **then**
      $neighbor[i] = neighbor[i] + 1$
  for $i = v + 1$ **to** n **do**
    if $neighbor[i] = 0$ **then** {rejection test}
      $mis(x \cup \{i\}, i)$ {recursive step}
  for $i = v + 1$ **to** n **do** {undo the counting}
    if $(i, v) \in G$ **then**
      $neighbor[i] = neighbor[i] - 1$

- $g(x)$ returns $|x|$ plus the number of nodes where $neighbor[j] = 0$, $j \geq v$.
- It is a bounding function – the value is larger than or equal to the maximum that can be achieved with the current subset.