DDA 2010, lecture 5: Weak colouring and other tricks

- Symmetry *can* be broken very fast if nodes have odd degrees...
  - ... but we need *port numbering and orientation*

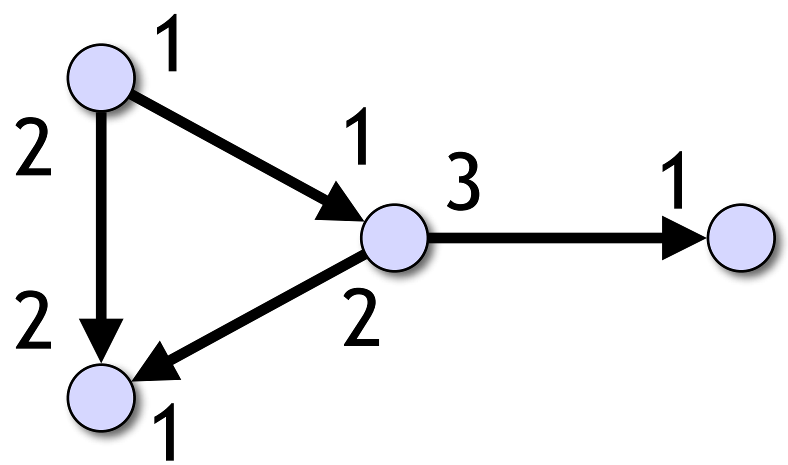
DDA 2010, lecture 5a: Port numbering and orientation

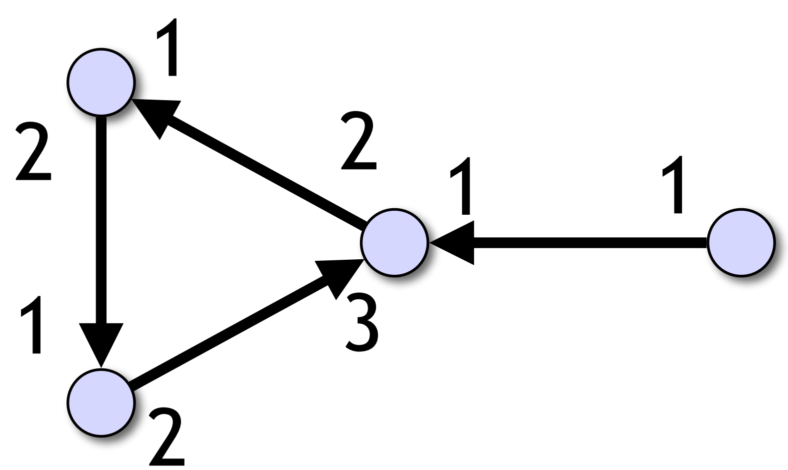
- A new model
  - stronger than the port-numbering model
  - weaker than networks with unique identifiers

Introduction

- How could we design algorithms that are faster than Cole–Vishkin? Constant-time algorithms?
  - if we try to exploit the **numerical values** of unique identifiers, we will usually get running times $\Omega(\log^* n)$ or worse
  - what if we just used the **relative order** of unique identifiers?
  - let's have a look at a model in which each pair of neighbours is ordered, and see what kinds of problems can be solved...

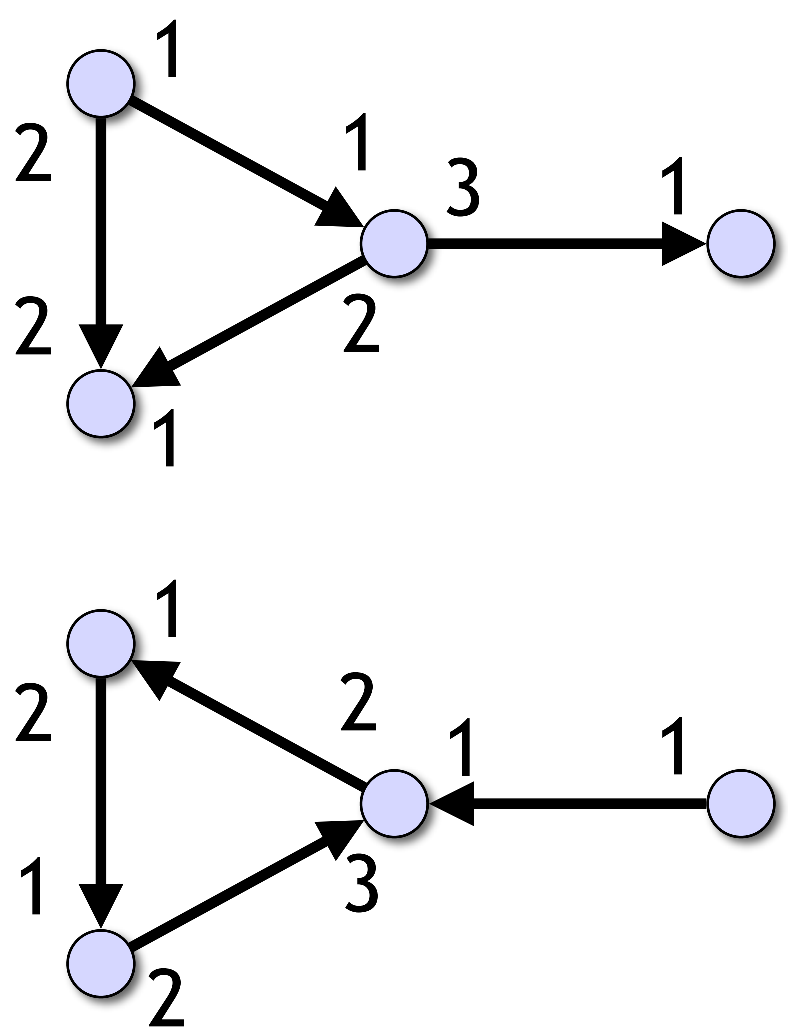
Port-numbering and orientation





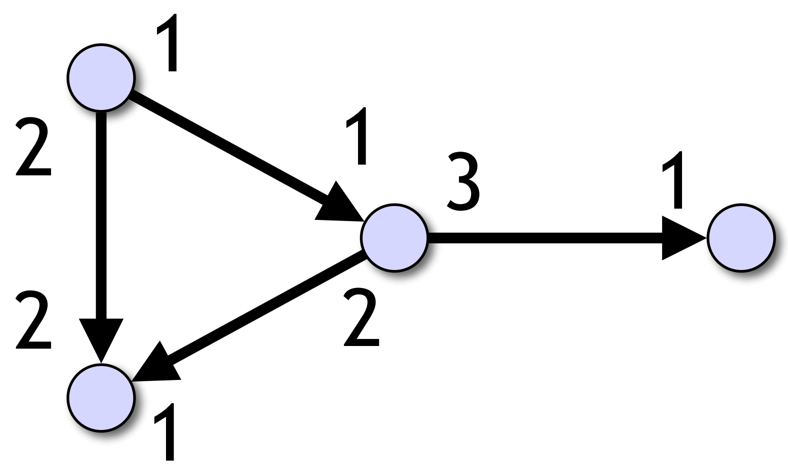
- A node of degree *d* can refer to its neighbours by integers 1, 2, ..., *d*
- Each edge has an **orientation**
  - ends labelled: head, tail
- Port-numbering and orientation chosen by adversary

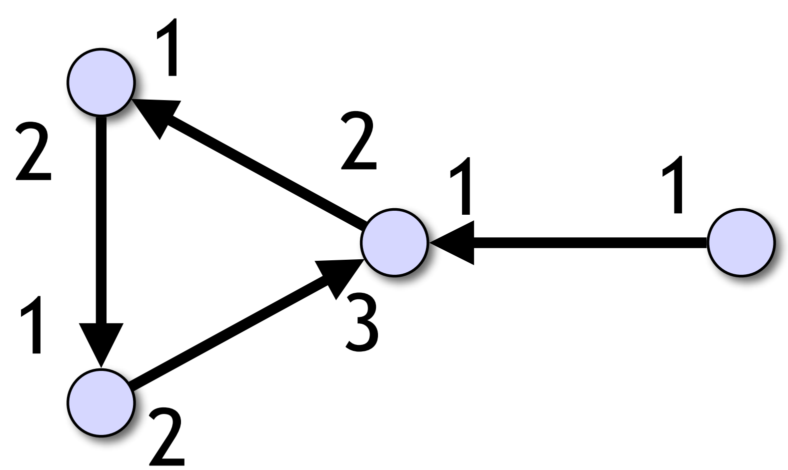
Port-numbering and orientation



- If you have unique identifiers or colouring, you can easily find an orientation
  - orient from smaller to larger ID (or colour)
  - we used this trick in lecture 2 to construct directed forests

Port-numbering and orientation



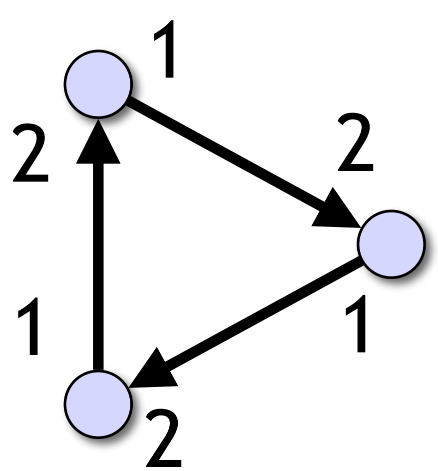


- Is this model stronger than port numbering?
- Is this model weaker than unique identifiers?

Port-numbering and orientation

- Is this model stronger than port numbering?
  - *Yes*: colouring of 2-node paths is possible
- Is this model weaker than unique identifiers?

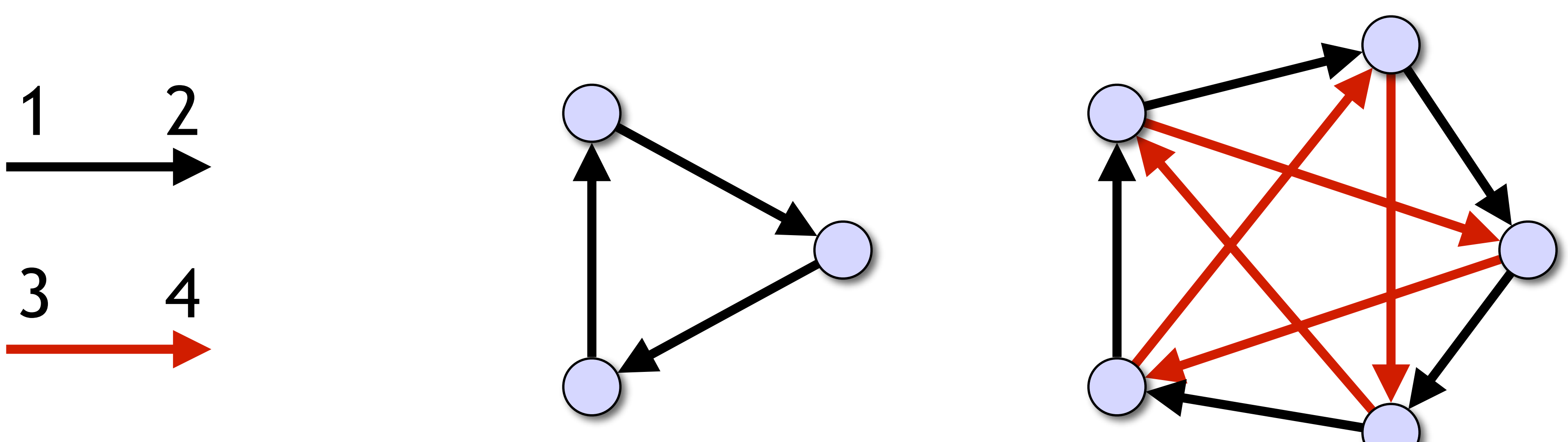
Port-numbering and orientation



- Is this model stronger than port numbering?
  - ***Yes***: colouring of 2-node paths is possible
- Is this model weaker than unique identifiers?
  - ***Yes***: colouring of 3-cycles is impossible

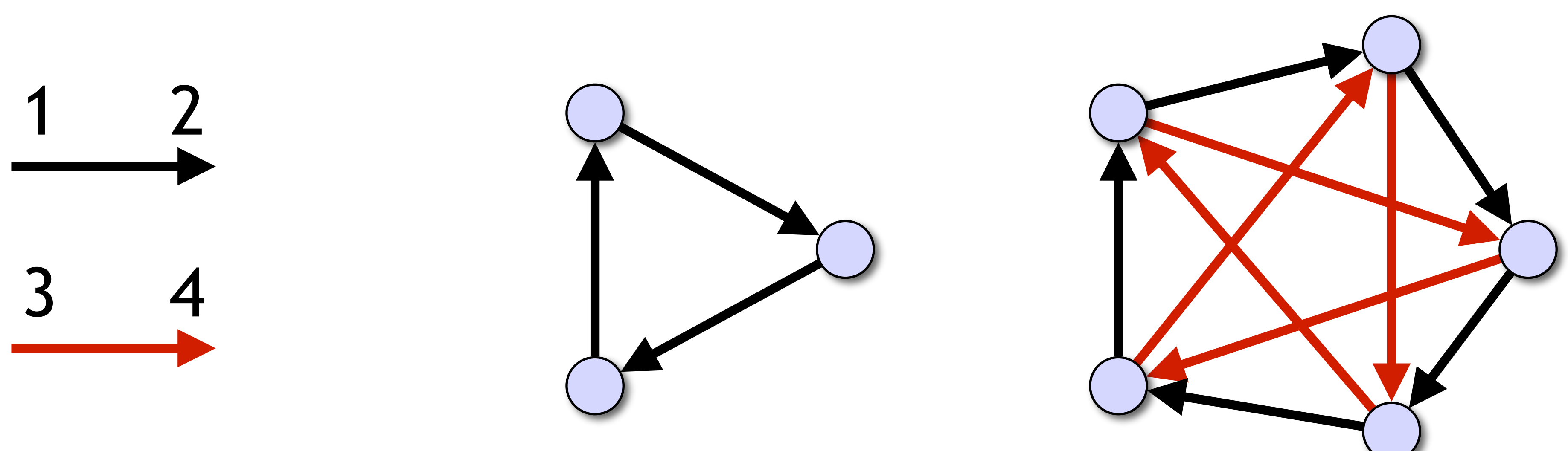
Port-numbering and orientation

- Can we solve anything non-trivial in this model?
- Looks bad: we can still have symmetric inputs



Port-numbering and orientation

- Can we solve anything non-trivial in this model?
- Looks bad: we can still have symmetric inputs
  - but in all these constructions *indegree = outdegree*, and therefore nodes must have *even degrees*!

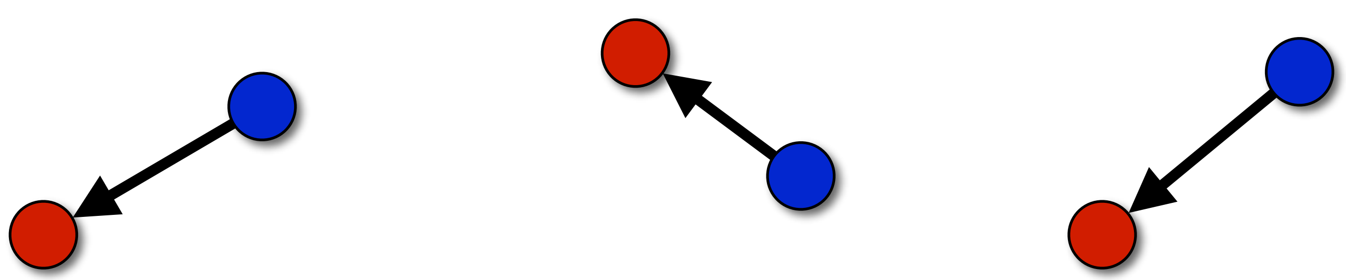


DDA 2010, lecture 5b: Weak colouring

- Naor–Stockmeyer (1995):
  - fast symmetry breaking in graphs with indegree ≠ outdegree

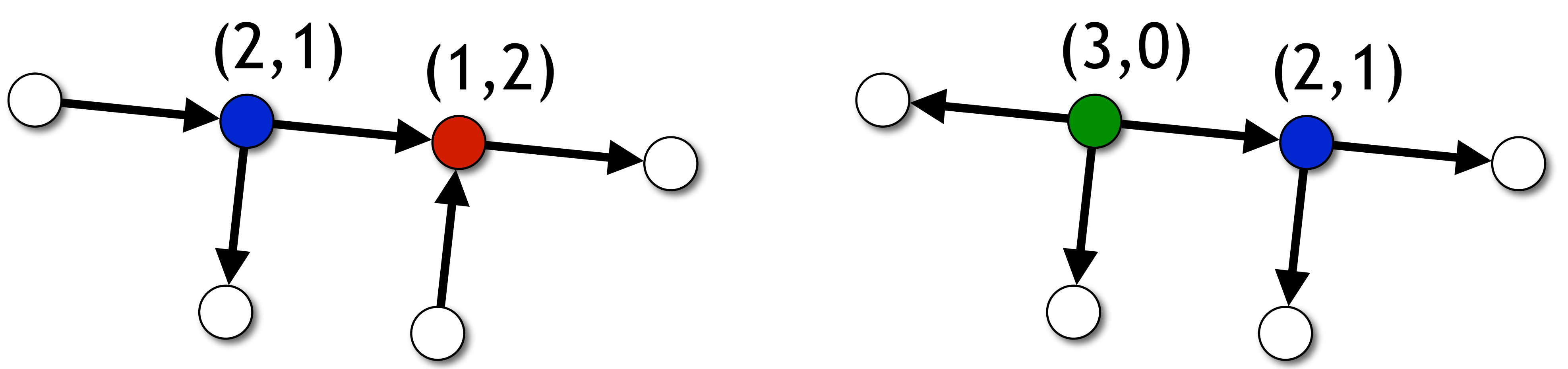
Symmetry breaking in graphs with port numbering and orientation

- The simplest case: 1-regular graphs
- Consists of isolated edges, certainly we can break symmetry for each pair of nodes
  - one is "**head**", the other one is "**tail**" (head has indegree 1, tail has outdegree 1)



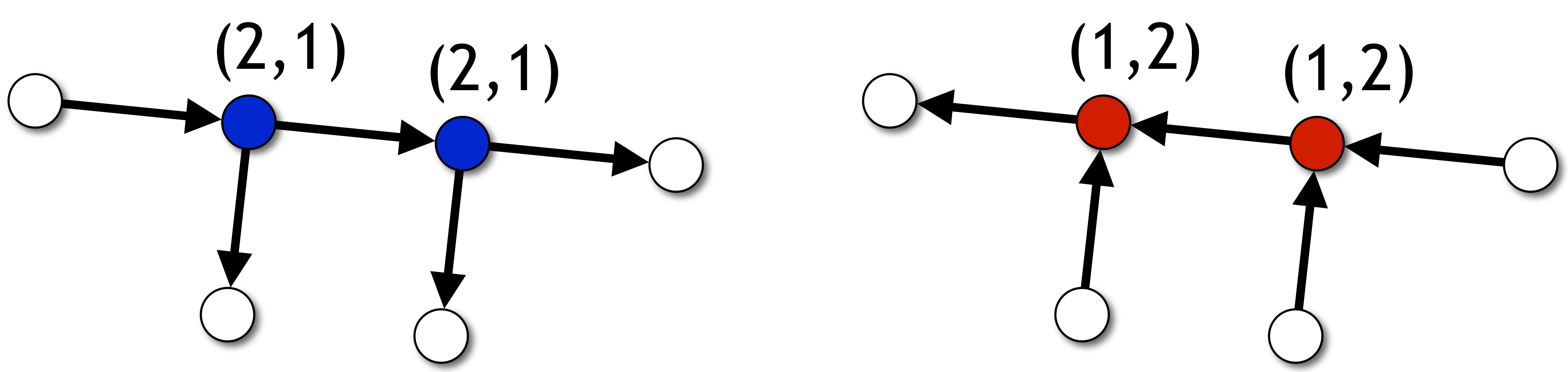
Symmetry breaking in graphs with port numbering and orientation

- In general, we can always label nodes by their (outdegree, indegree) pairs
  - different outdegrees or different indegrees: different labels, symmetry broken
  - only $O(\Delta^2)$ possible labels; easy to reduce using C–V tricks



Symmetry breaking in graphs with port numbering and orientation

- In general, we can always label nodes by their (outdegree, indegree) pairs
- But what if a *node* and *all of its neighbours* have identical (outdegree, indegree) pairs?

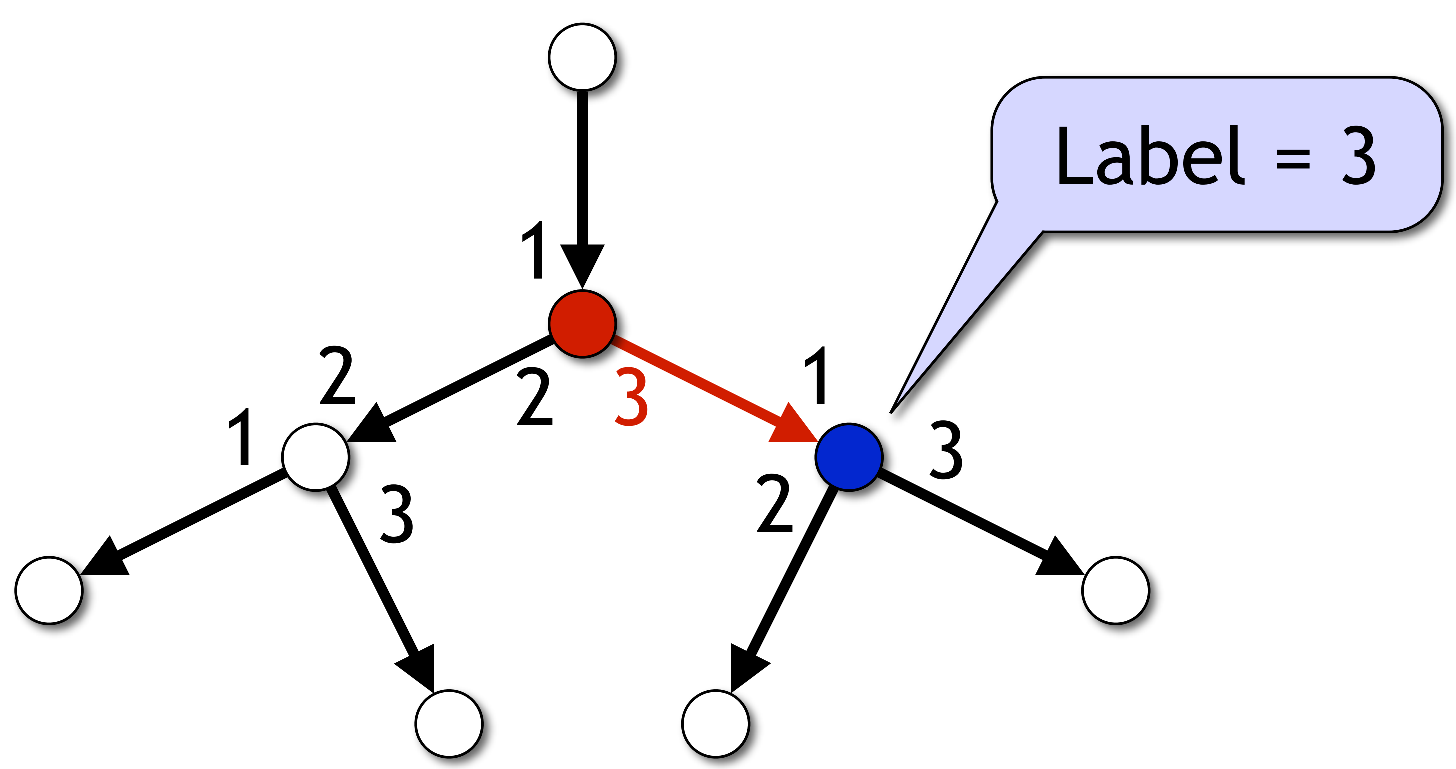


Symmetry breaking in graphs with port numbering and orientation

- In general, we can always label nodes by their (outdegree, indegree) pairs
- But what if a *node* and *all of its neighbours* have identical (outdegree, indegree) pairs?
  - we already know that if outdegree = indegree for all nodes, we are in trouble
  - but *what if we know that outdegree ≠ indegree*?
  - for example, what if all nodes have degree = 3 and therefore necessarily outdegree ≠ indegree?

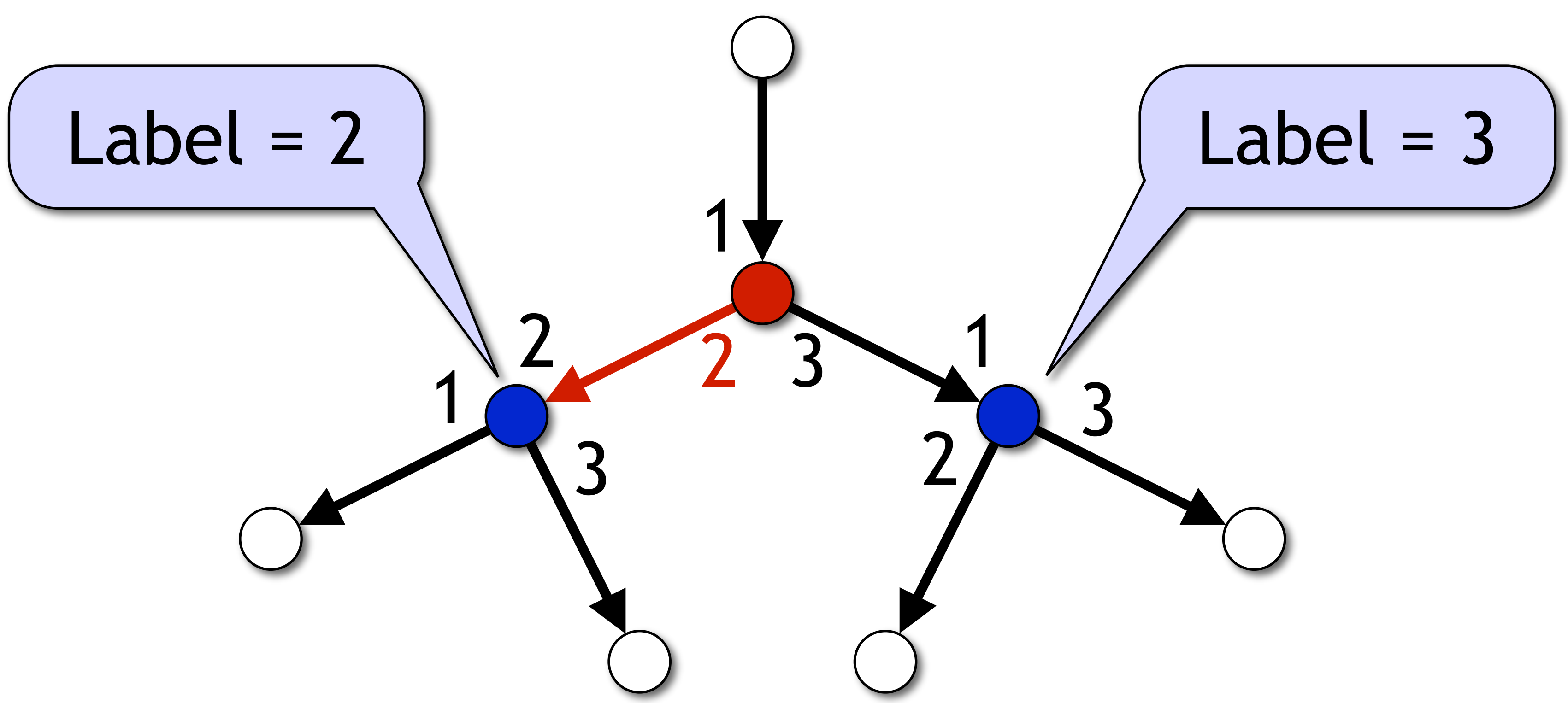
Symmetry breaking in graphs with port numbering and orientation

- Simplest case: indegree = 1, outdegree = 2
- Label = outgoing port number in **predecessor**



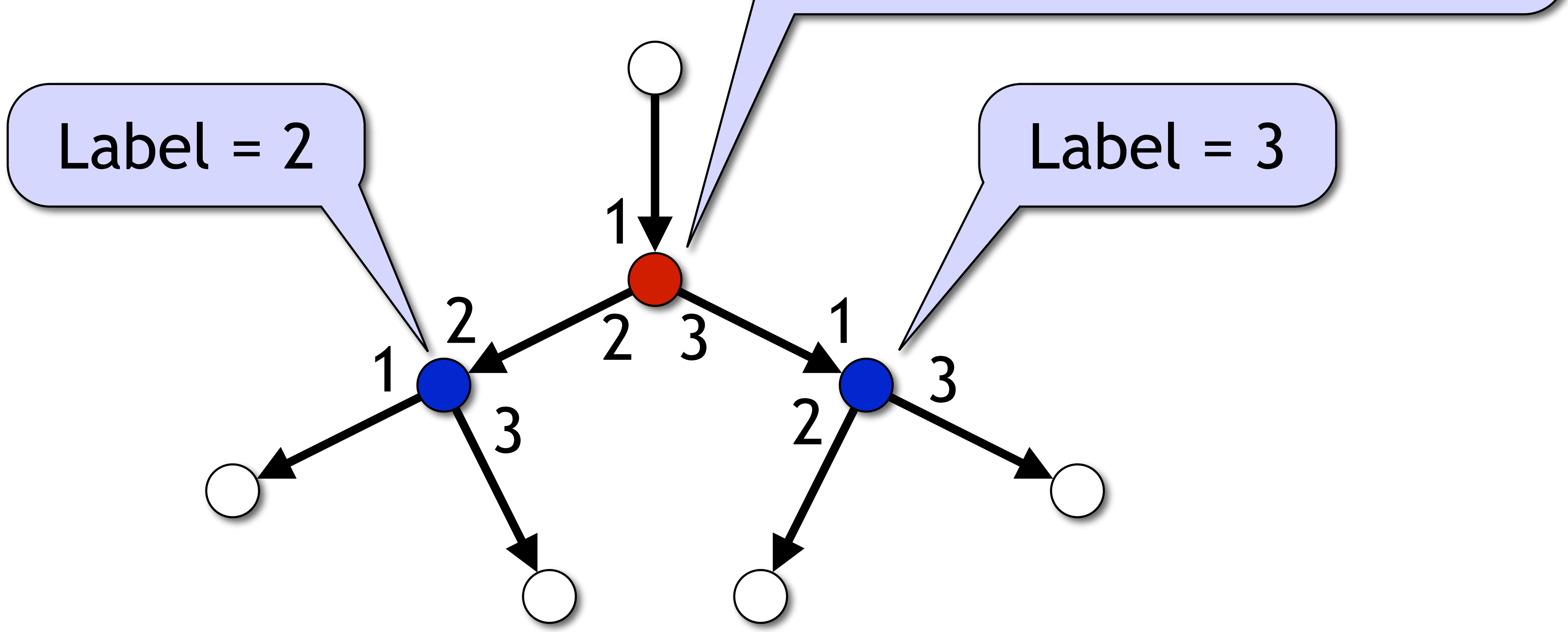
Symmetry breaking in graphs with port numbering and orientation

- Simplest case: indegree = 1, outdegree = 2
- Label = outgoing port number in **predecessor**



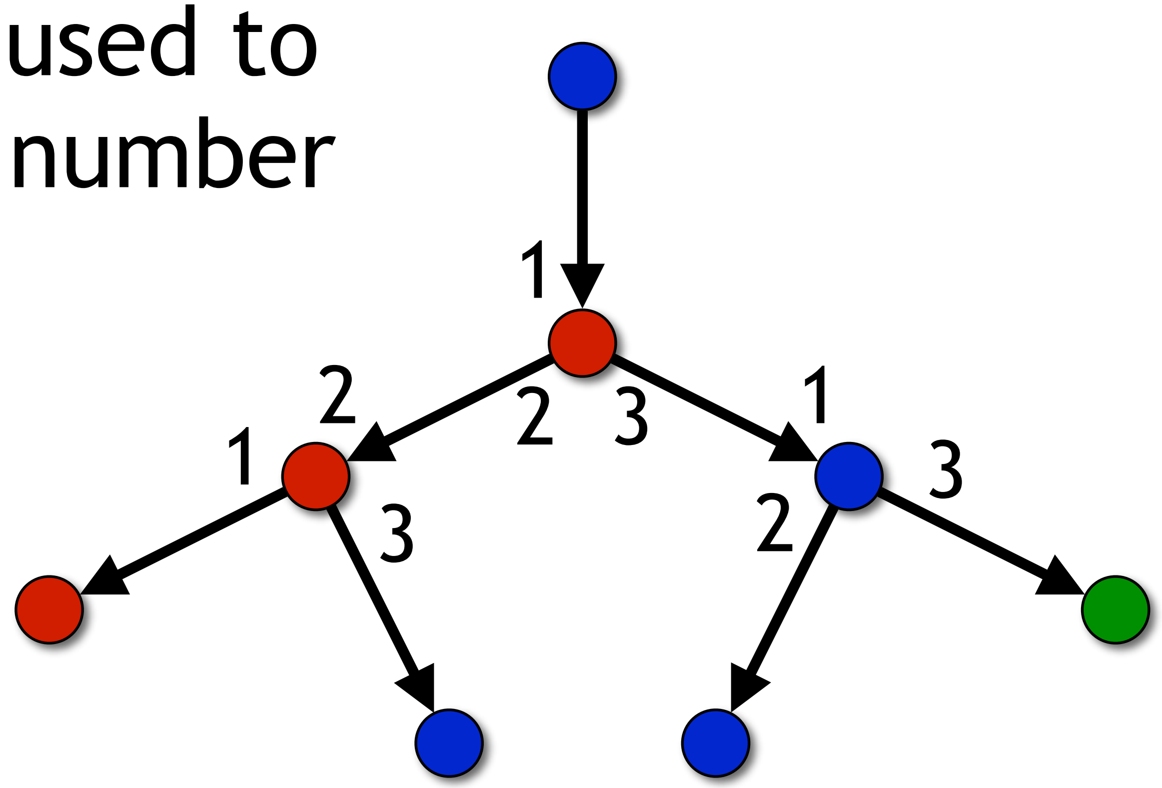
Symmetry breaking in graphs with port numbering and orientation

- Simplest case: indegree = 1
- Label = outgoing port numb



Symmetry breaking in graphs with port numbering and orientation

- We can construct a ***weak colouring***:
  - for each non-isolated node at least one neighbour has different colour
- C–V can be used to reduce the number of colours

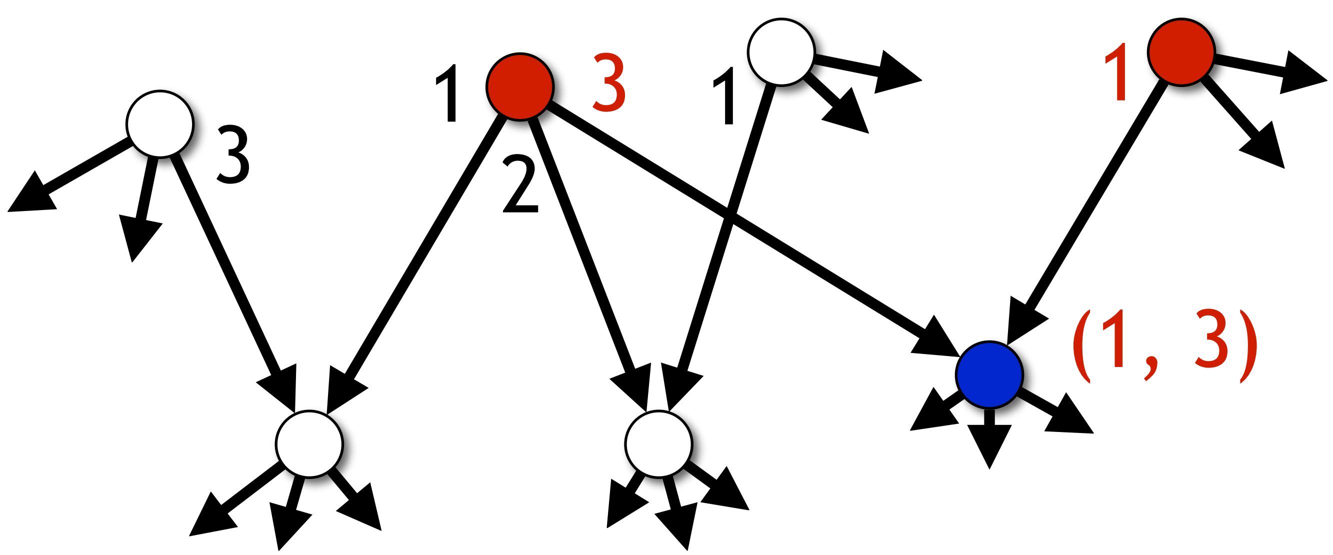


Symmetry breaking in graphs with port numbering and orientation

- Indegree = 1, outdegree = 2: ***weak colouring***
  - node takes its label from the port numbers of its parent
- Generalisation to any indegree ≠ outdegree?
  - enough to study the case indegree < outdegree
  - then we can reverse the directions and get the same result for indegree > outdegree!
  - let's present the algorithm in the general case and prove that it finds a weak colouring...

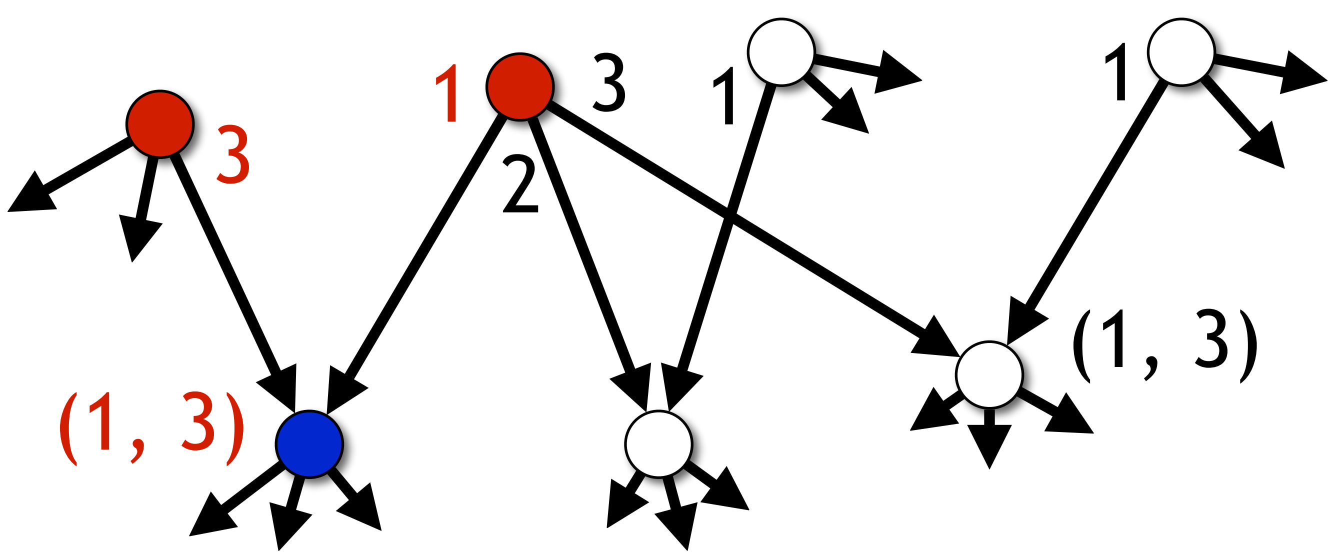
Symmetry breaking in graphs with port numbering and orientation

- General case: indegree < outdegree
- Label = list of outgoing port numbers in all **predecessors**



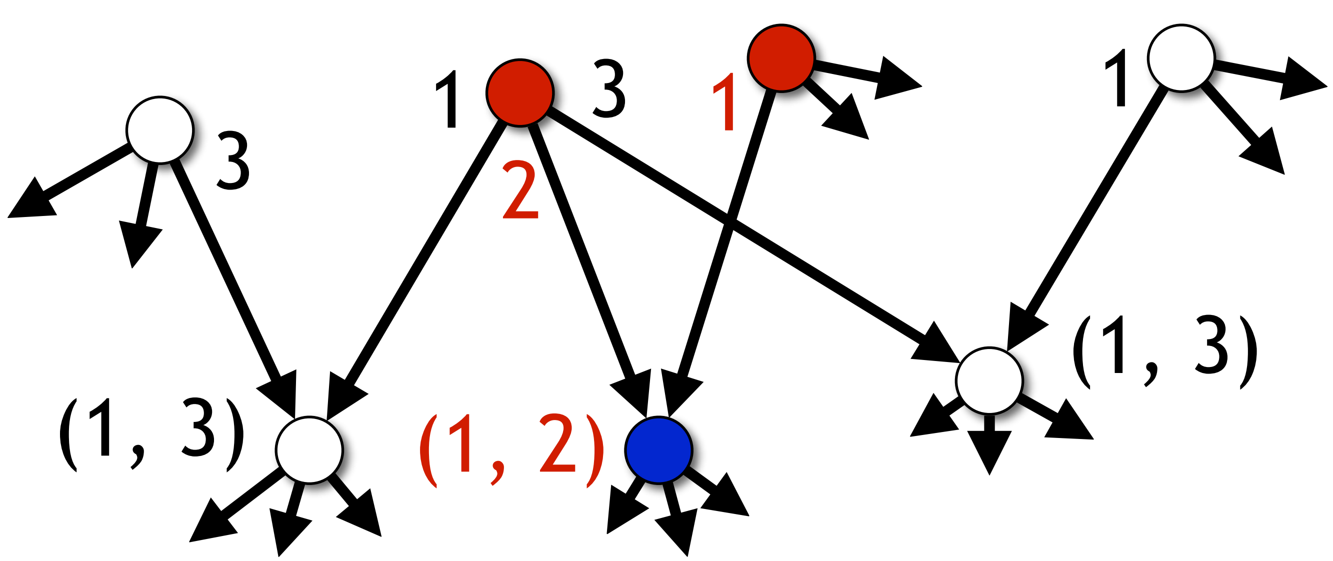
Symmetry breaking in graphs with port numbering and orientation

- General case: indegree < outdegree
- Label = list of outgoing port numbers in all **predecessors**



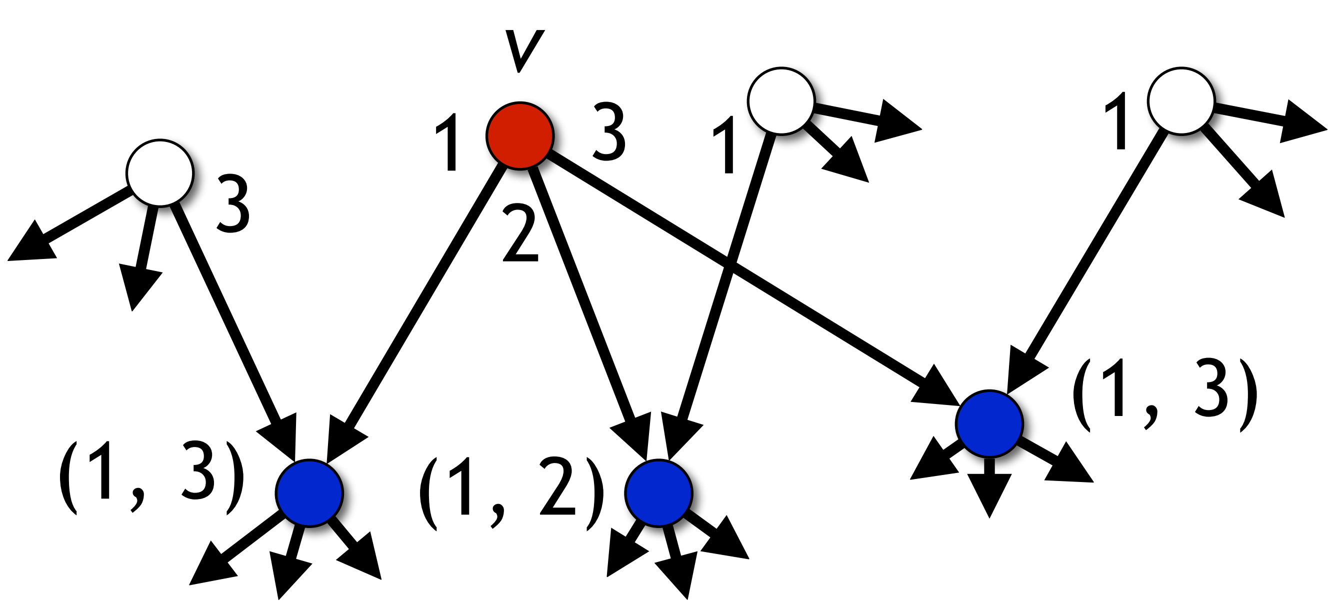
Symmetry breaking in graphs with port numbering and orientation

- General case: indegree < outdegree
- Label = list of outgoing port numbers in all **predecessors**



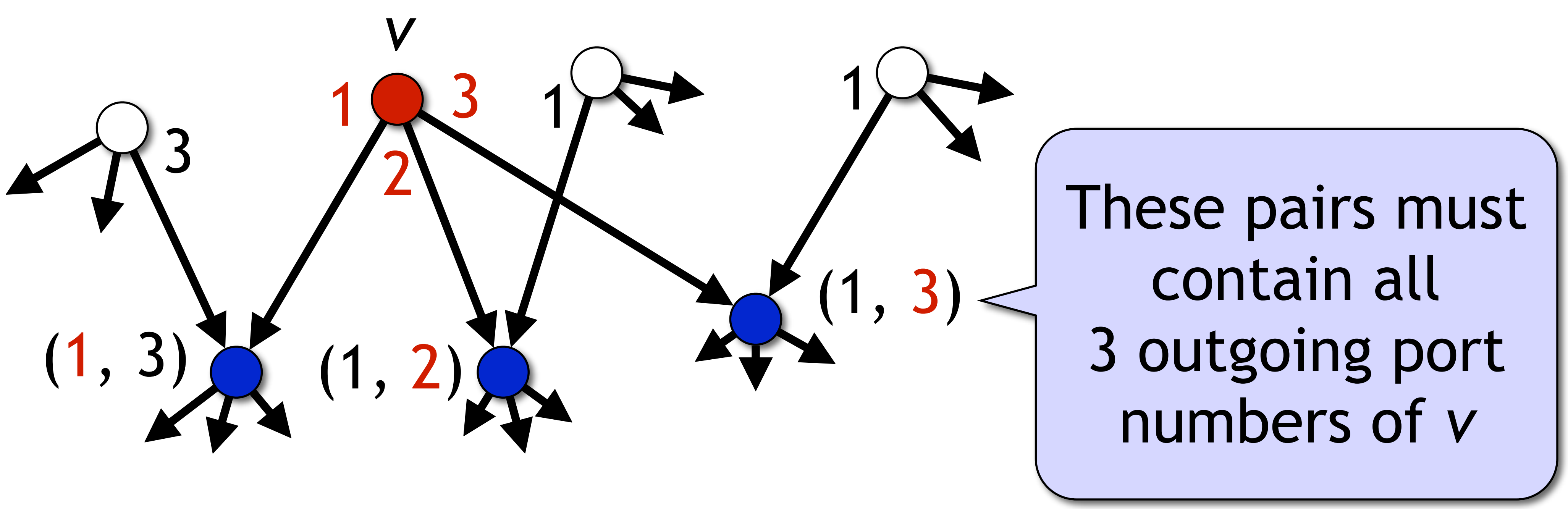
Symmetry breaking in graphs with port numbering and orientation

- **Lemma**: for each *v*, the successors of *v* have at least 2 different labels
  - Proof: pigeonhole again...



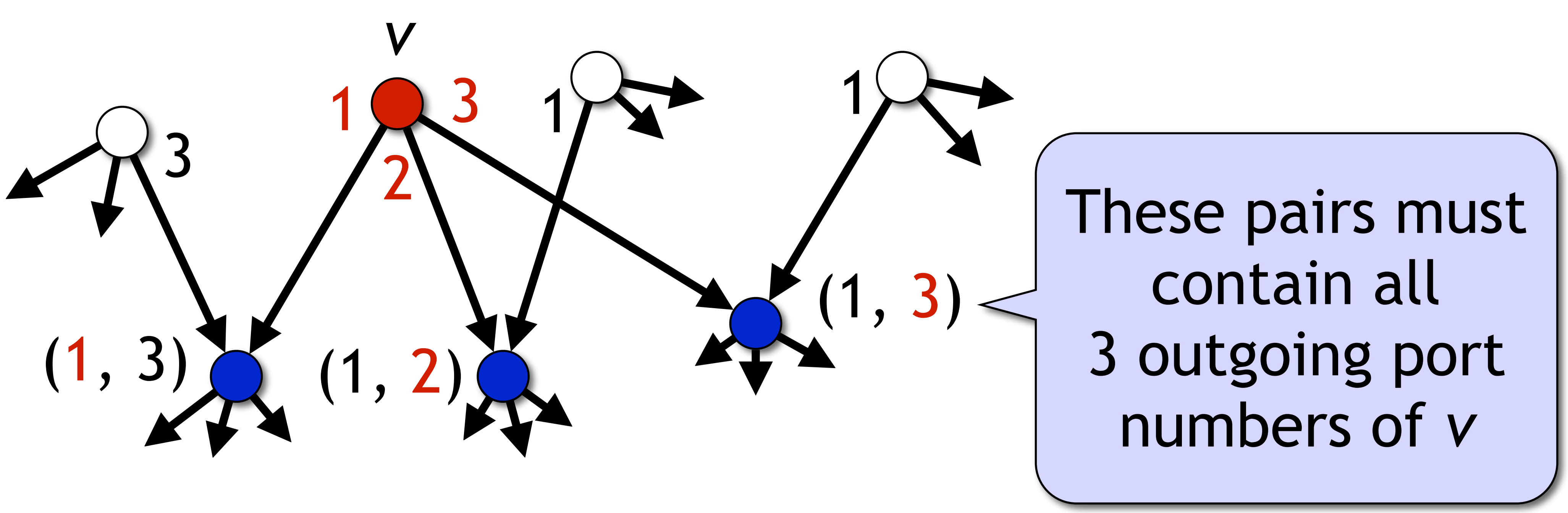
Symmetry breaking in graphs with port numbering and orientation

- E.g., outdegree = 3, indegree = 2:
  - a 2-element list can’t contain all 3 outgoing port numbers of *v*
  - must have at least 2 different 2-element lists!



Symmetry breaking in graphs with port numbering and orientation

- General case, outdegree = *s*, indegree = *t*:
  - an *s*-element list can't contain all *t* outgoing port numbers of *v* if *s* < *t*
  - must have at least 2 different *s*-element lists!



Symmetry breaking in graphs with port numbering and orientation

- **Lemma**: for each *v*, the successors of *v* have at least 2 different labels
- **Corollary**: *v* has a successor *u* such that *v* and *u* have different labels
  - i.e., we have a weak colouring
  - again, we can use C–V to reduce the number of colours
  - it is possible to construct a **weak 2-colouring**; running time is $O(\log^* \Delta)$, independent of *n*
  - assumptions: port numbering, indegree ≠ outdegree

Summary

- Model: **port numbering** and **orientation**
- If outdegree = indegree:
  - we may have a symmetric input
  - in the worst case all nodes will produce the same output
- If outdegree ≠ indegree:
  - symmetry can be broken
  - we can find a weak 2-colouring — very fast!
  - however, we can’t find a (non-weak) colouring