• Using Ramsey’s theorem, we can show that these problems can’t be solved in $O(1)$ rounds:
  
  • finding large independent sets in cycles
  • graph colourings and maximal matchings in cycles
  • better than 2-approximation of vertex cover
  • and many more…
DDA 2010, lecture 4a: Introduction and background

- Hardness of graph colouring and other symmetry-breaking problems
Graph colouring

- Graph colouring is a central symmetry-breaking primitive in distributed algorithms
  - Colouring can be used to **schedule** the actions of the nodes: e.g., neighbours don’t transmit simultaneously
  - Given a graph colouring, we can **solve other problems**: maximal independent set, maximal matching, etc.
  - We can use colours to **simulate greedy algorithms**: finding small dominating sets, etc.
Graph colouring

• Graph colouring is a central symmetry-breaking primitive in distributed algorithms

• Many problems are as difficult as graph colouring
  • Given an algorithm that finds a maximal independent set, we can use it to find a graph colouring, and vice versa

• To understand the capabilities of distributed algorithms, it is important to know how fast we can find a graph colouring
Hardness of graph colouring

• Cole-Vishkin algorithm can be used to colour cycles in almost constant running time: $O(\log^* n)$
  • assuming we have unique identifiers

• Could we get exactly constant running time?
  • it seems very difficult to come up with an $O(1)$-time algorithm for graph colouring…
  • but how could one possibly prove that no such algorithm exists?
  • there are infinitely many algorithms!
Hardness of graph colouring

• Cole-Vishkin algorithm can be used to colour cycles in almost constant running time: $O(\log^* n)$
  • assuming we have unique identifiers

• Could we get exactly constant running time?

• This was resolved by Nathan Linial in 1992:
  • 3-colouring an $n$-cycle requires $\Omega(\log^* n)$ rounds
  • Cole-Vishkin technique is within constant factor of the best possible algorithm!
Hardness of other problems

• Linial’s result shows that it is not possible to solve these problems in cycles in $O(1)$ time:
  • vertex colouring, edge colouring, maximal independent set, maximal matching, ...

• Naor and Stockmeyer (1995): generalisations
  • using Ramsey’s theorem

• What about other problems?
Hardness of other problems

• Linial: we can’t find maximal independent sets in constant time

• However, could we perhaps find a “fairly large” independent set in constant time?
  • e.g., an independent set with at least $n/10$ nodes?

• We will see that this is not possible, either
  • strong negative result
  • proof uses Ramsey’s theorem
Finding a non-trivial independent set

- Czygrinow et al. (2008)
  - constant-time algorithms can’t find large independent sets in cycles
Lower-bound result for finding large independent sets

- Numbered directed $n$-cycle:
  - directed $n$-cycle, each node has outdegree = indegree = 1
  - node identifiers are a permutation of \{1, 2, \ldots, n\}
Lower-bound result for finding large independent sets

• We will show that the problem is difficult even if we have a numbered directed cycle
  • general case of cycles with unique IDs at least as hard
Lower-bound result for finding large independent sets

• Fix any $\varepsilon > 0$ and running time $T$ (constants)
• Algorithm $A$ finds a feasible independent set in any numbered directed cycle in time $T$
• **Theorem:** For a sufficiently large $n$ there is a numbered directed $n$-cycle $C$ in which $A$ outputs an independent set with $\leq \varepsilon n$ nodes
  • can’t find an independent set with $> 0.001n$ nodes
  • not even if the running time is 1000000 rounds
Lower-bound result for finding large independent sets

- Let $T$ be the running time of $A$, let $k = 2T + 1$
- The output of a node is a function $f'$ of a sequence of $k$ integers (unique IDs)

$T = 2, k = 5$:  
output = $f'(11, 9, 5, 2, 7)$

output = $f'(3, 11, 9, 5, 2)$
Lower-bound result for finding large independent sets

- Lets focus on **increasing** sequences of IDs
- Then the output of a node is a function $f$ of a **set** of $k$ integers

$k = 5$:

```
output = f({6, 7, 11, 13, 21})
```

```
output = f({3, 6, 7, 11, 13})
```
Lower-bound result for finding large independent sets

- Hence we have assigned a colour \( f(X) \in \{0, 1\} \) to each \( k \)-subset \( X \subset \{1, 2, \ldots, n\} \)

\[
k = 5:
\]

\[
\text{output} = f(\{3, 6, 7, 11, 13\})
\]

\[
\text{output} = f(\{6, 7, 11, 13, 21\})
\]
Lower-bound result for finding large independent sets

- Hence we have assigned a colour $f(X) \in \{0, 1\}$ to each $k$-subset $X \subset \{1, 2, \ldots, n\}$
- Fix a large $m$ (depends on $k$ and $\varepsilon$)
- Ramsey: If $n$ is sufficiently large, we can find an $m$-subset $A \subset \{1, 2, \ldots, n\}$ s.t. all $k$-subset $X \subset A$ have the same colour
Lower-bound result for finding large independent sets

• That is, if the ID space is sufficiently large…
Lower-bound result for finding large independent sets

- That is, if the ID space is sufficiently large, we can find a monochromatic subset of $m$ IDs...

$$f({2, 3, 6, 7, 11}) = f({2, 3, 6, 7, 13}) =$$
$$f({2, 3, 6, 7, 21}) = f({2, 3, 6, 11, 13}) =$$
$$... = f({6, 7, 11, 13, 21})$$
Lower-bound result for finding large independent sets

- Construct a numbered directed cycle: monochromatic subset as consecutive nodes
Lower-bound result for finding large independent sets

- Construct a numbered directed cycle: monochromatic subset as consecutive nodes

\[ f(\{2, 3, 6, 7, 11\}) = \]
\[ f(\{3, 6, 7, 11, 13\}) = \ldots \]
Lower-bound result for finding large independent sets

- Construct a numbered directed cycle: monochromatic subset as consecutive nodes

```plaintext
Same output
... and it must be 0
```
Lower-bound result for finding large independent sets

- Hence there is an $n$-cycle with a chain of $m - 2T$ nodes that output 0.
Lower-bound result for finding large independent sets

• Hence there is an $n$-cycle with a chain of $m - 2T$ nodes that output 0

• We can choose as large $m$ as we want
  • Good, more “black” nodes that output 0

• However, $n$ increases rapidly if we increase $m$
  • Bad, more “grey” nodes that might output 1

• Trick: choose “unnecessarily large” $n$ so that we can apply Ramsey’s theorem repeatedly
Lower-bound result for finding large independent sets

- Huge ID space...

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
</tr>
<tr>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
<td>40</td>
</tr>
<tr>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
<td>50</td>
</tr>
<tr>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
<td>60</td>
</tr>
</tbody>
</table>
Lower-bound result for finding large independent sets

- Find a monochromatic subset of size $m$...
Lower-bound result for finding large independent sets

• Delete these IDs...
Lower-bound result for finding large independent sets

- Still sufficiently many IDs to apply Ramsey…
Lower-bound result for finding large independent sets

• Repeat…
Lower-bound result for finding large independent sets

• Repeat until stuck
Lower-bound result for finding large independent sets

- Several monochromatic subsets + some leftovers
Lower-bound result for finding large independent sets

Large enough $n$: at most $\varepsilon n/2$ nodes in the grey area

Large enough $m$: at most $\varepsilon n/2$ nodes near the boundaries
Lower-bound result for finding large independent sets

- Thus $A$ outputs an independent set with $\leq \varepsilon n$ nodes
DDA 2010, lecture 4c: Corollaries

- Finding “anything” non-trivial in cycles is not possible in constant time
A strong negative result

• We have used Ramsey’s theorem to show that constant-time algorithms can’t find large independent sets in cycles
  • moreover, we can get a $\Omega(\log^* n)$ lower bound on the running time of any algorithm that finds a large independent set
  • trick: use a power tower upper bound for $R_2(n; k)$

• What implications do we have?
A strong negative result

- If we could find a graph colouring...
  - we could find a maximal independent set...
  - which is an independent set with at least $n/3$ nodes
  - contradiction

- Corollary: graph colouring can’t be solved in constant time in cycles
  - we got Linial’s result as a simple corollary...
A strong negative result

• If we could find a \((2 - \varepsilon)\)-approximation of vertex cover...
  • we would have a vertex cover with at most \(n - \varepsilon n/2\) nodes in an \(n\)-cycle (even \(n\))
  • its complement is an independent set with at least \(\varepsilon n/2\) nodes
  • contradiction
• This is tight: it is possible to find a 2-approximation in time independent of \(n\)
A strong negative result

• Using Ramsey’s theorem, we are able to show that these problems can’t be solved in $O(1)$ time:
  • vertex colouring, edge colouring, …
  • maximal independent set, maximal matching, …
  • $(2 - \varepsilon)$-approximation of vertex cover
  • $(\Delta + 1 - \varepsilon)$-approximation of dominating set…

• Next lecture: something positive with $O(1)$ running time…