Randomness and Computation
or, “Randomized Algorithms”

Mary Cryan

School of Informatics
University of Edinburgh

Graphs and Colourings
A common concept in graph theory is the concept of a colouring of a graph. If we have \(k\) different colours, we usually identify them with the set \(\{1, \ldots, k\}\).

- We can consider the different ways of colouring the vertices of a graph \(G = (V, E)\) with those \(k\) colours.
  - A \(k\)-colouring is any assignment \(c : V \rightarrow \{1, \ldots, k\}\) of colours to vertices (every \(v \in V\) gets some colour \(c(v)\)).
  - A proper \(k\)-colouring is any \(c : V \rightarrow \{1, \ldots, k\}\) such that for every \(e = (u, v), e \in E\), we have \(c(u) \neq c(v)\).
  - For a given graph \(G = (V, E)\), it is often of interest to ask what is the minimum \(k\) needed to properly colour \(G\). For sure, we know \(k \leq \text{max degree of } G + 1\).
  - Lots of research effort have gone into polynomial-time algorithms to approximate (exact is NP-hard) the minimum \(k\) for a given \(G\). **Not our concern today**
  - Alternatively we can consider the different ways of colouring the edges of a graph \(G = (V, E)\).

Our example - Ramsay numbers

Our focus today is 2-colouring the edges of the complete graph \(K_n\).

- \(K_n\) is the complete graph on \(n\) vertices (for every \(i, j \in [n], i \neq j\), we have the edge \((i, j)\)).
- We are **not** interested in vertex 2-colourings of \(K_n\), every vertex “blue” or “red”. (cannot give a proper colouring if \(n \geq 3\)).
- Our concern is whether we can colour \(K_n\)’s edge with our two colours and make sure that we do not have any “all-blue” or “all-red” subgraph which is “too large”.
- The **“Ramsay number”** \(R(k, k)\) is the smallest value for \(n\) such that in any two-colouring of the edges of \(K_n\), there must be either a red \(K_k\) (“all-red” of size \(k\)) or a blue \(K_k\) (“all-blue” of size \(k\)).

The value of \(R(k, k)\) increases with \(k\).

**class:** What is \(R(2, 2)\)? And \(R(3, 3)\) (board)?
Lower Bound on $R(k,k)$

We prove a lower bound on $R(k,k)$ for general $k$. This was first shown by Erdős in 1947.

**Theorem (Theorem 6.1)**
Consider $R(k,k)$ for some $k \geq 2$. For any $n$ such that \( \binom{n}{k} 2^{1-\frac{(k)}{2}} < 1 \), we have $R(k,k) > n$.

**Proof.**
Write down the expected number of “all red” or “all blue” $K_k$ subgraphs, when the edges of $K_n$ are coloured uniformly at random by red/blue.

For a particular $K_k$ subgraph, probability of being monochromatic is $2 \cdot 2^{1-\frac{(k)}{2}} = 2^{1-\frac{(k)}{2}}$.

There are \( \binom{n}{k} \) different $K_k$ subgraphs to consider in $K_n$.

The expected number of monochromatic subgraphs of $K_n$ is therefore

\[
\left( \binom{n}{k} \right) \frac{2}{2^{1-\frac{(k)}{2}}}. 
\]

As required.

---

**Corollary**
If $k \geq 3$, then for $R(k,k) > \lfloor \frac{2k}{2} \rfloor$.

**Proof.**
Just algebraic manipulation.

Consider $\binom{n}{k} \cdot 2^{1-\frac{(k)}{2}}$ for the given value of $n = \lfloor \frac{2k}{2} \rfloor$. This is

\[
\begin{align*}
\frac{n(n-k+1)}{k!} \cdot 2^{1-\frac{(k)}{2}} &< \frac{2^{k^2} \ldots (2^{k^2-k+1})}{k!} \cdot 2^{1-\frac{k(k-1)}{2}} \\
&= \frac{n^k}{k^2} \cdot 2^{1+\frac{k}{2}} \\
&= \left( \frac{n}{2^2} \right)^k \cdot 2^{1+\frac{k}{2}} \\
&< 1 \cdot 1,
\end{align*}
\]

as required.

---

**Making this method constructive (“derandomization”)**

In the proof of Theorem 6.1 about random colourings of $K_n$ and the presence of any monochromatic $K_k$s, we focused on the situation when we have $\binom{n}{k} 2^{1-\frac{(k)}{2}} < 1$. However, the argument shows ...
Making this method **constructive** (“derandomization”)

Using the second Corollary on slide 8, if the expectation is at most \(\binom{n}{4}2^{-5}\) (over all random 2-colourings), then there is some specific 2-colouring of \(K_n\) that has \(\leq \binom{n}{4}2^{-5}\) monochromatic \(K_4\) copies.

We can construct a specific 2-colouring to satisfy this using the **method of conditional expectation** (and “deferred decisions”).

The idea:
- Let \(f\) be a specific edge of \(K_n\).
- A random 2-colouring has probability \(1/2\) of setting \(f\) blue, and probability \(1/2\) of setting \(f\) red.
- The colours of all the other edges are set uniformly and independently with probability \(1/2\).
- Hence, for at least one of the events \(c(f) = \text{red}\), \(c(f) = \text{blue}\), the (conditional) number of expected monochromatic \(K_4\) is \(\leq \binom{4}{4}2^{-5}\).
- Find a way of determining this colour for \(f\), and iterate.

---

**Theorem**

For every integer \(n\), we can construct a specific 2-colouring of \(K_n\) such that the expected number of monochromatic \(K_4\) subgraphs is at most \(\binom{n}{4}2^{-5}\).

**Proof.**

To help with the construction, we define a **weight function** \(w\) on copies of \(K_4\) which will allow us to measure the expected “value” of colouring particular edges blue or red.

Suppose we are part-way through the construction, and some (but not all) edges have their colour fixed.
- We have some partial colouring \(c : F \rightarrow \{\text{blue, red}\}\), where \(F \subseteq E(K_n)\).
- We maintain the invariant that the expected number of monochromatic \(K_4\) copies, taken over the remaining random 2-colourings for the edges in \(E(K_n) \setminus F\), is \(\leq \binom{n}{4}2^{-5}\).

---

**Theorem**

For every integer \(n\), we can construct a specific 2-colouring of \(K_n\) such that the number of monochromatic \(K_4\) subgraphs is at most \(\binom{n}{4}2^{-5}\).

**Proof cont’d.**

The weight function \(w\) assigns a non-negative value to every subgraph \(K\) which is a copy of \(K_4\) in \(K_n\). Let \(c(K)\) be the set of colours already seen on edges of \(K\), at this stage of the partial colouring. Define

\[
w(K) = \begin{cases} 
0 & \text{if } c(K) = \{\text{blue, red}\}, \\
2^{-5} & \text{if } c(K) = \emptyset \text{ (all edges uncoloured),} \\
2r^{-6} & \text{if } |c(K)| = 1, \text{ and } r \text{ of } K\text{'s edges have this colour}
\end{cases}
\]

The total weight of the partially coloured \(K_n\) is

\[
W_F = \sum_{K \text{ a } K_4 \text{ copy in } K_n} w(K).
\]
Making this method constructive ("derandomization")

Proof.

THE ALGORITHM:
1. for $i \leftarrow 1$ to $n(n - 1)/2$ do
   
   ($F$ is $e_1, \ldots, e_{n-1}$, and these edges are coloured)
2. Calculate $W_{\text{red}}$, the effect on $W_F$ of colouring $e_i$ red.
3. Calculate $W_{\text{blue}}$, the effect on $W_F$ of colouring $e_i$ blue.
4. if $W_{\text{red}} < W_{\text{blue}}$ then Set $c(e_i) = \text{red}; W_F \leftarrow W_{\text{red}}$
5. else Set $c(e_i) = \text{blue}; W_F \leftarrow W_{\text{blue}}$
6. $F \leftarrow F \cup \{e_i\}$

- Note that the value of $W_F$ never increases through the iteration of this process. Hence we end up with a colouring $c$ which has fewer than $W_F = \binom{n}{2}2^{-k}$ monochromatic $K_4$s.
- $e_i$ can belong to at most $n^2K_4$s in $K$, so the $W_{\text{red}}, W_{\text{blue}}$ values can be calculated in $\Theta(n^2)$ time.

The Probabilistic Method in Derandomization

- The theorem on slides 10-13 can be considered to be a "derandomization" of the result on expected number of monochromatic $K_4$s in $K_n$.
  - We were able to use conditional expectation to construct a specific colouring with less than or equal to the expected number of monochromatics.
  - Our algorithm was in fact polynomial-time (about $n^2$ iterations, each doing $\Theta(n^2)$ work, so roughly $\Theta(n^3)$).
- We can use the method of conditional probabilities to derandomize algorithms like the MAX-CUT algorithm from lectures 3-4.

Changes to this slide set

The original slides handed out on 16th February had a number of slides on the derandomization of the process for generating a "good" 2 edge-colouring. However, I have moved these to lecture 11 as we didn't cover the material till then (9th March).

There were a number of typos in the second half of the slides, which have now been fixed.
- Terrible confusion with notation for the edge of consideration on slide 9 (all mentions should have been $f$, and now are).
- Use of $K_k$ instead of $K_4$ on slides 9 and 10 (slide 10 now moved to lecture 11).
- Missing detail about updates to $W_F$ and $F$ now added to the derandomization algorithm (now moved into lecture 11).
- Under the algorithm, have corrected the claim about $W_F$ (it never increases).

Reading and Doing

Reading
- You will want to read Sections 6.1, 6.2, 6.3 from the book.
- The Theorem on derandomizing monochromatic $K_4$ is not in the book.

Doing
- Write down the details of how to derandomize the MAX-CUT ("random subset") algorithm so it always (deterministically) returns a cut of value $\leq \frac{|E|}{2}$. Some similarity to what's in Section 6.2 (but we now try to minimize wrt the $\frac{|E|}{2}$).
Updates regarding course structure

Balls in Bins  Decided to cut short this section (due to upcoming strike) and move straight to “The probabilistic method”. I will probably put the material on average-case of Bucket Sort into a tutorial.

2nd Coursework  I am late with this but will ship it by early next week at the latest. You’ll still be getting it over three weeks in advance of the deadline.