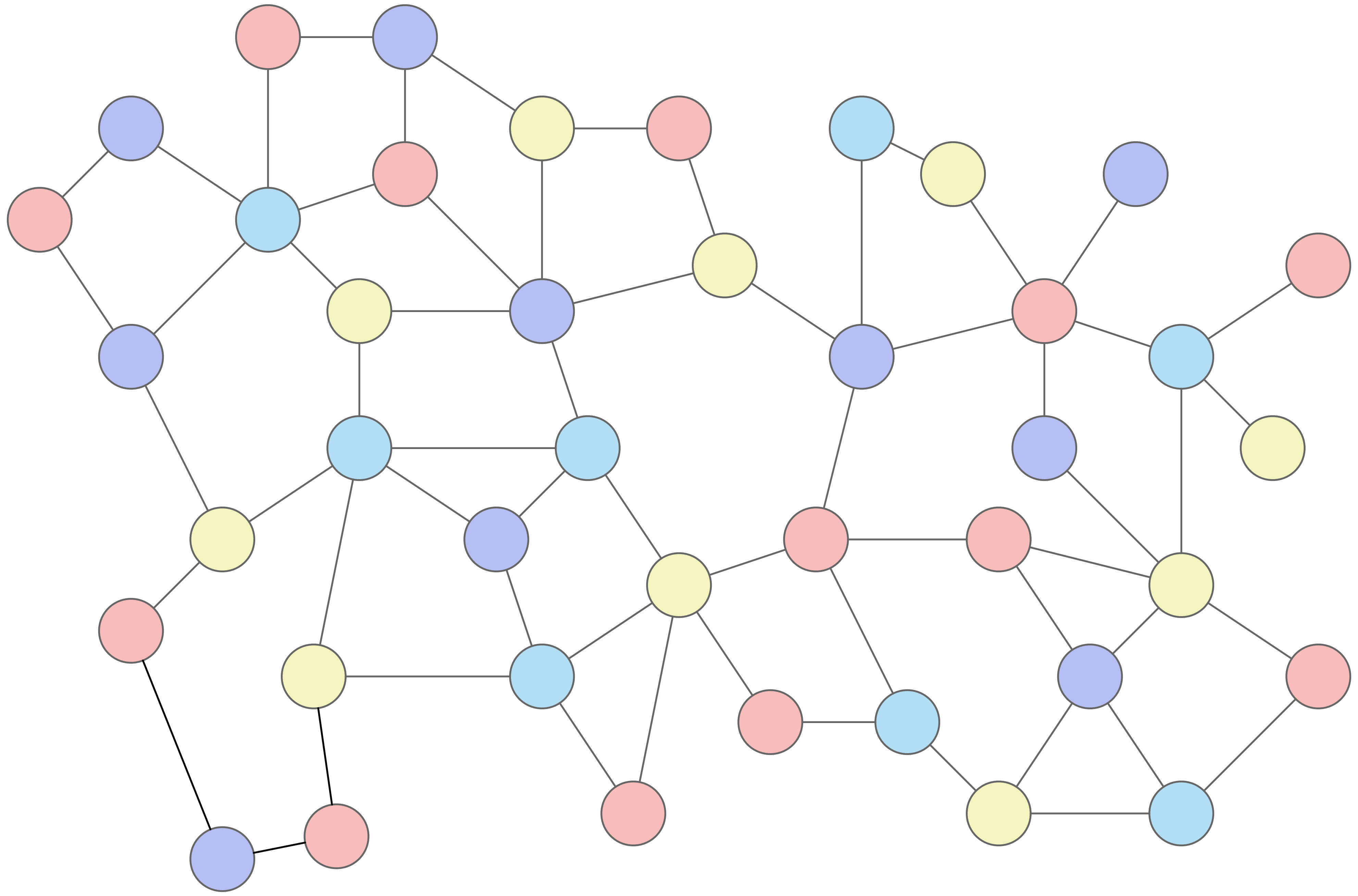
Time hierarchies of distributed complexities

Alkida Balliu
Aalto University, Finland

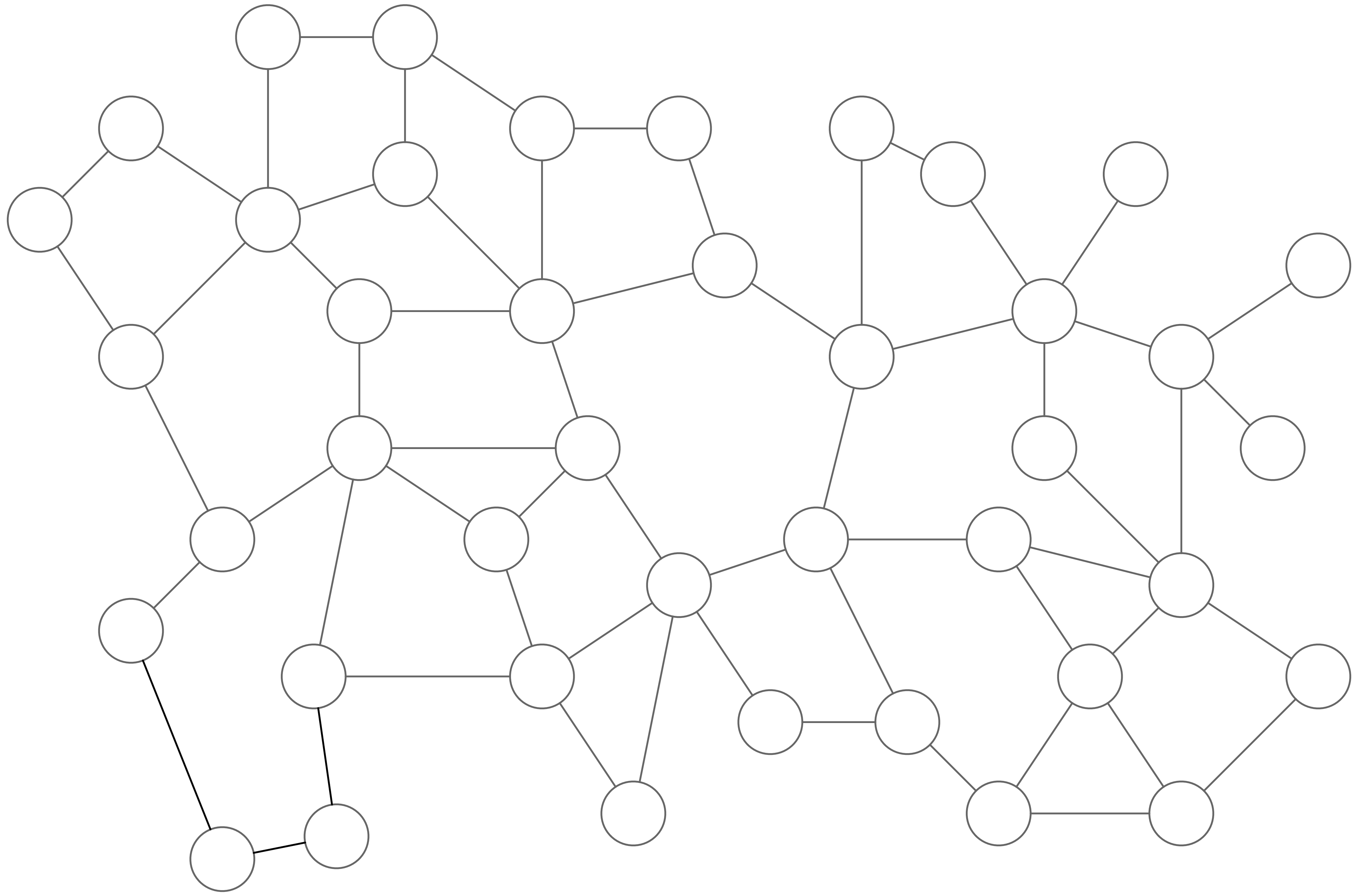
Topic: distributed graph problems

- **Family** of graph problems
- Focus on **locality**
  - How much does an entity need to know about the graph in order to solve a graph problem?
  - How local are these problems?
  - Does randomness help?



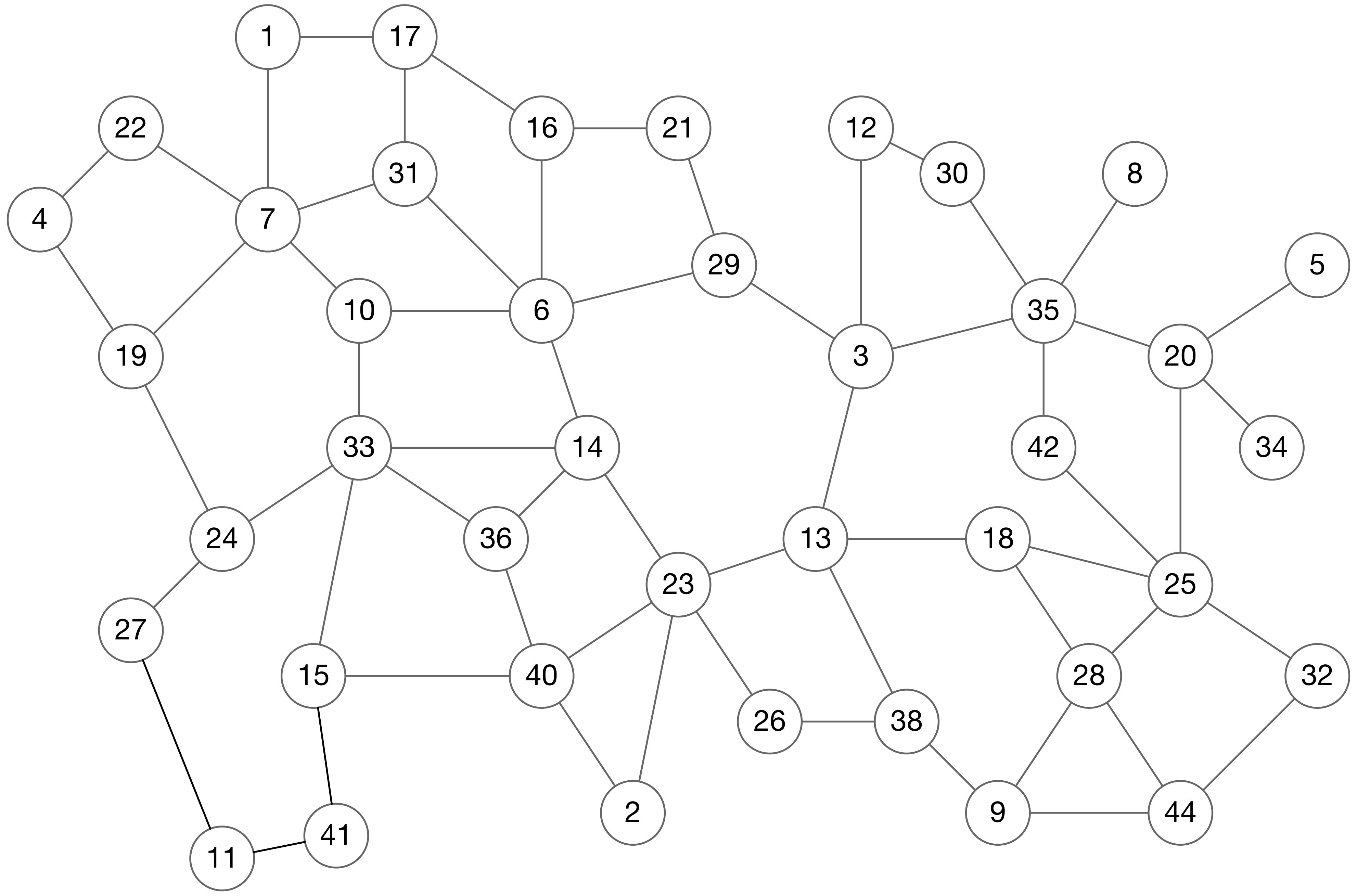
LOCAL model

- Entities = **nodes**
- Communication links = **edges**
- Input graph = communication graph



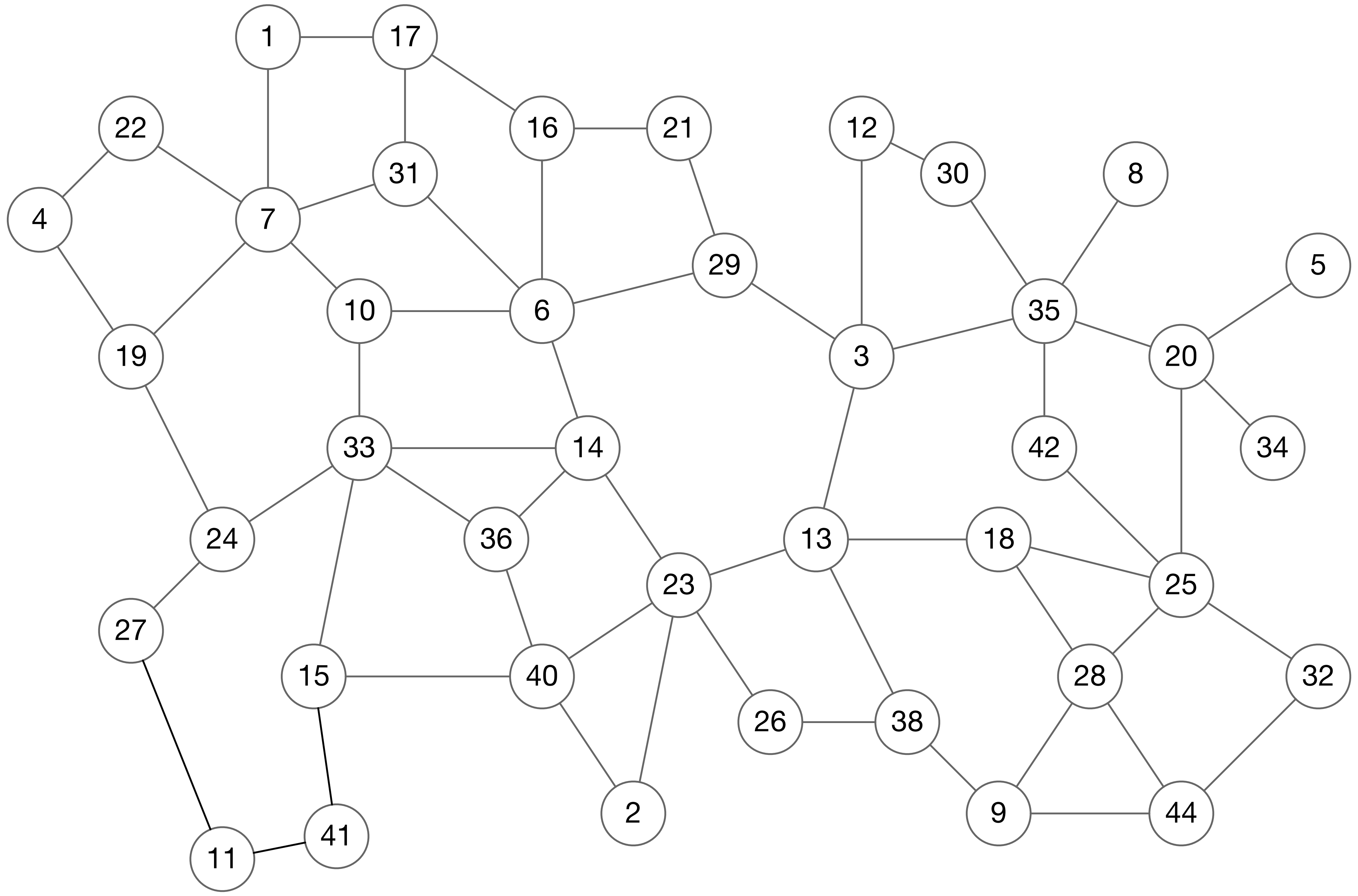
LOCAL model

- Each node has a **unique identifier** from 1 to poly(n)
- **No bounds** on the computational power of the entities
- **No bounds** on the bandwidth



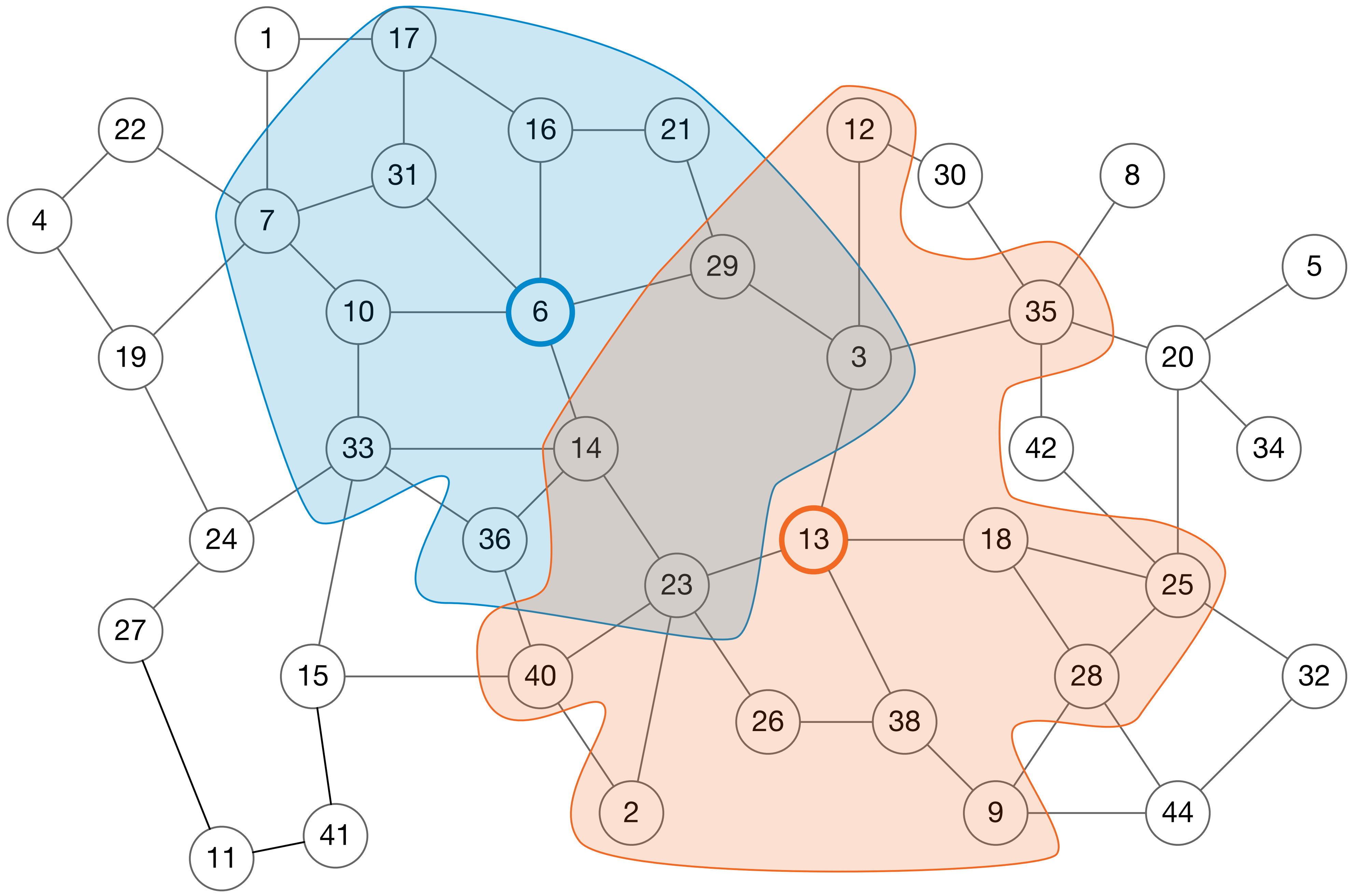
LOCAL model

- At each **synchronized** round:
  - **Send** messages to neighbours
  - **Receive** messages from neighbours
  - Perform **local computation**



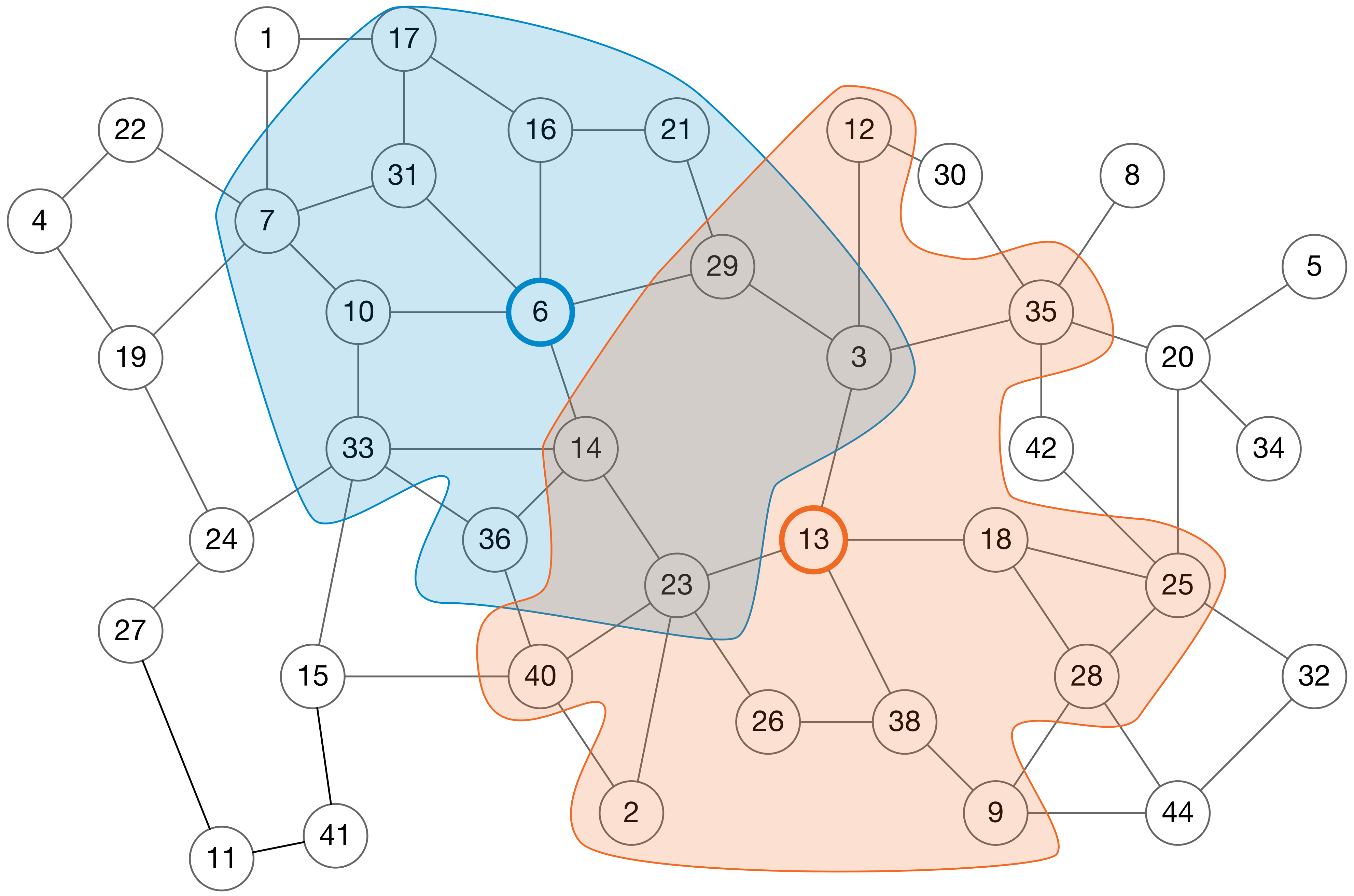
LOCAL model

- After ***t* rounds**:
  - knowledge of the graph up to **distance *t***
- Focus on **locality**:
  - time = number of rounds = distance



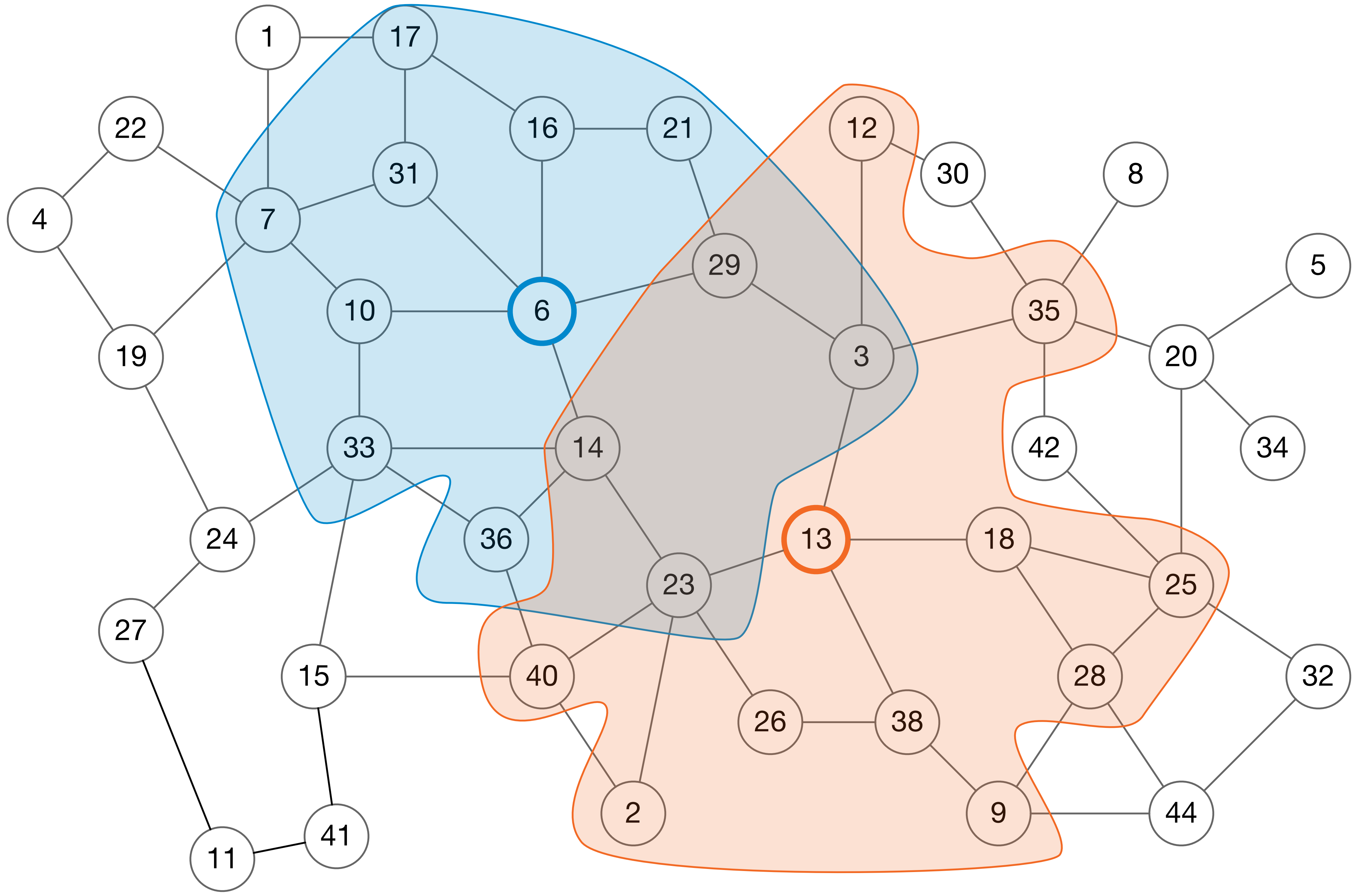
LOCAL model

- After ***t* rounds**:
  - knowledge of the graph up to **distance *t***
- Focus on **locality**:
  - time = number of rounds = distance
- **Randomized LOCAL**:
  - each node can generate an unbounded number of **random bits**



LOCAL model

- After ***t* rounds**:
  - knowledge of the graph up to **distance *t***
- Focus on **locality**:
  - time = number of rounds = distance
- **Randomized LOCAL**:
  - each node can generate an unbounded number of **random bits**



***Everything** can be solved in **Diameter time**!*

LOCAL model: initial knowledge

- **Initial knowledge** of a node:
  - n = the total number of nodes in the graph
  - Δ = the maximum degree of the graph
  - Its unique **ID**
  - A **port numbering** of its incident edges

Locally Checkable Labelings (LCLs)

A **family of graph problems** that includes many important problems

Maximal Independent Set, Maximal Matching, vertex coloring, edge coloring…

Locally Checkable Labelings (LCLs)

- **Input**
  - Graph of **constant** maximum degree Δ
  - Node labels from a **constant-size** set X
- **Output**
  - Node labels from a **constant-size** set Y, such that each node satisfies some **local constraints**
- **Correctness**
  - A solution is globally correct if it is correct in all **constant-radius** neighborhoods

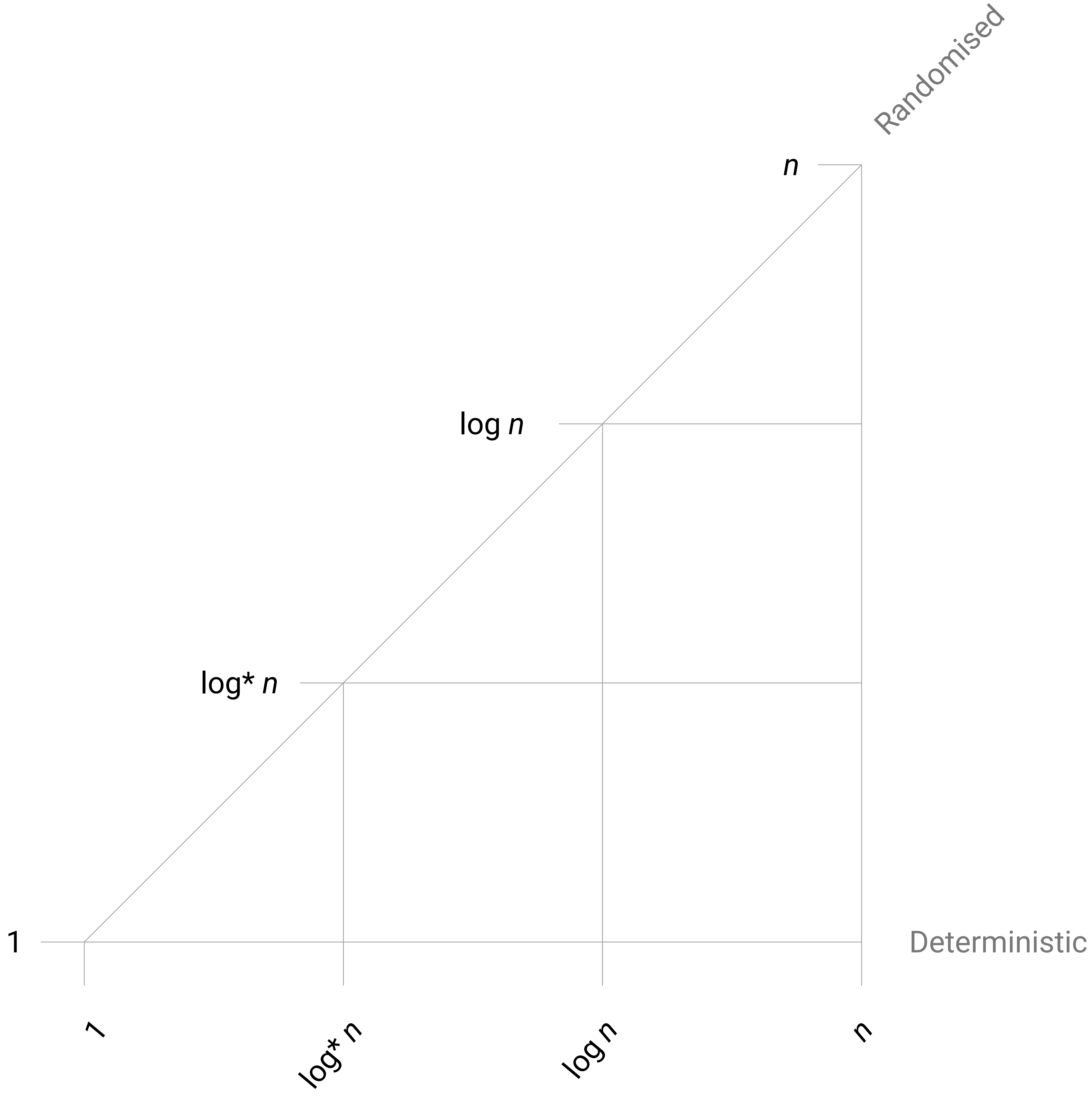
[Naor and Stockmeyer, 1995]

Example: weak 2-coloring

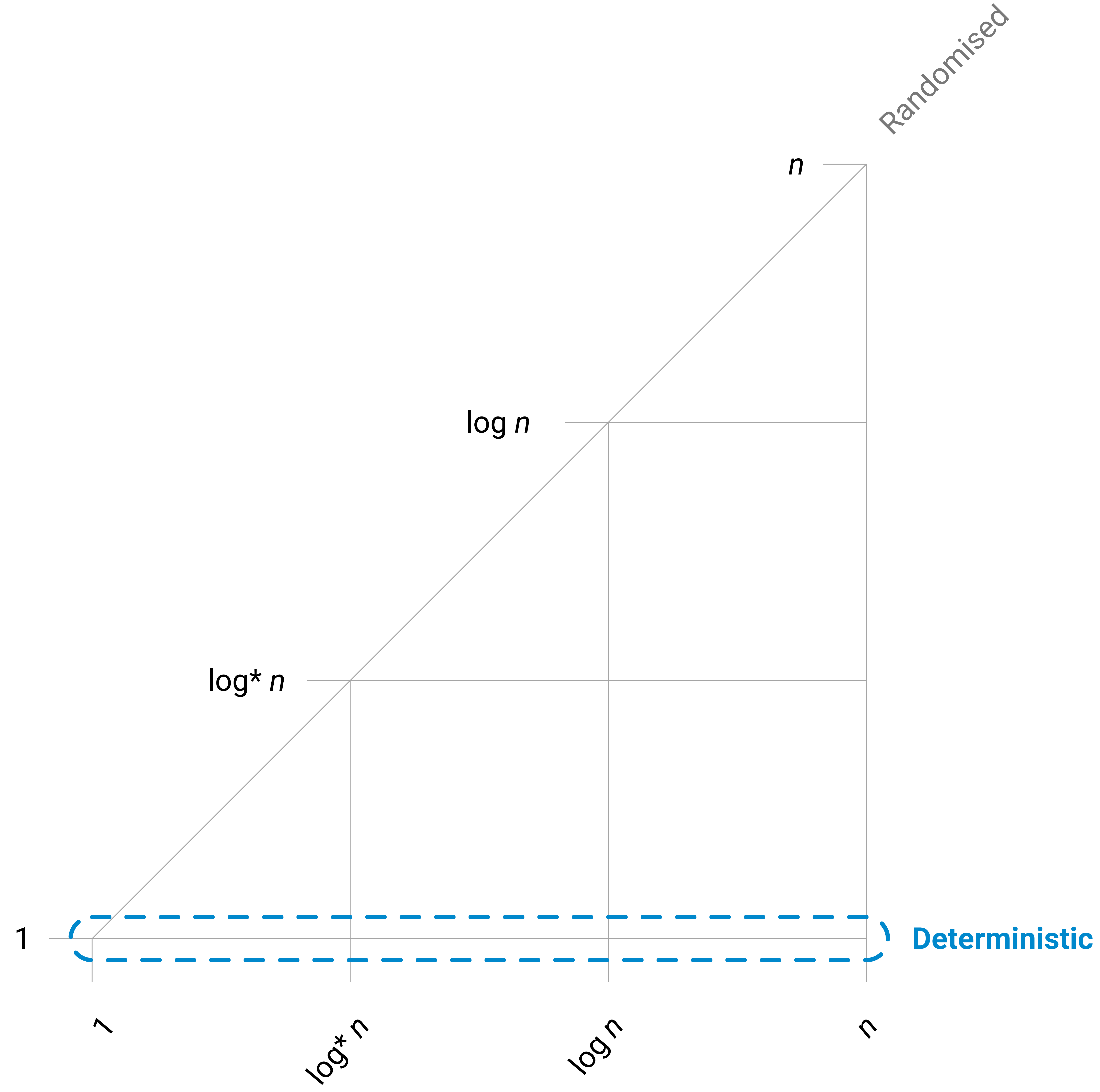
- **Output**: color nodes from a palette of **2 colors**
- **Constraint**: each node must have a **different color** from **at least 1** neighbor

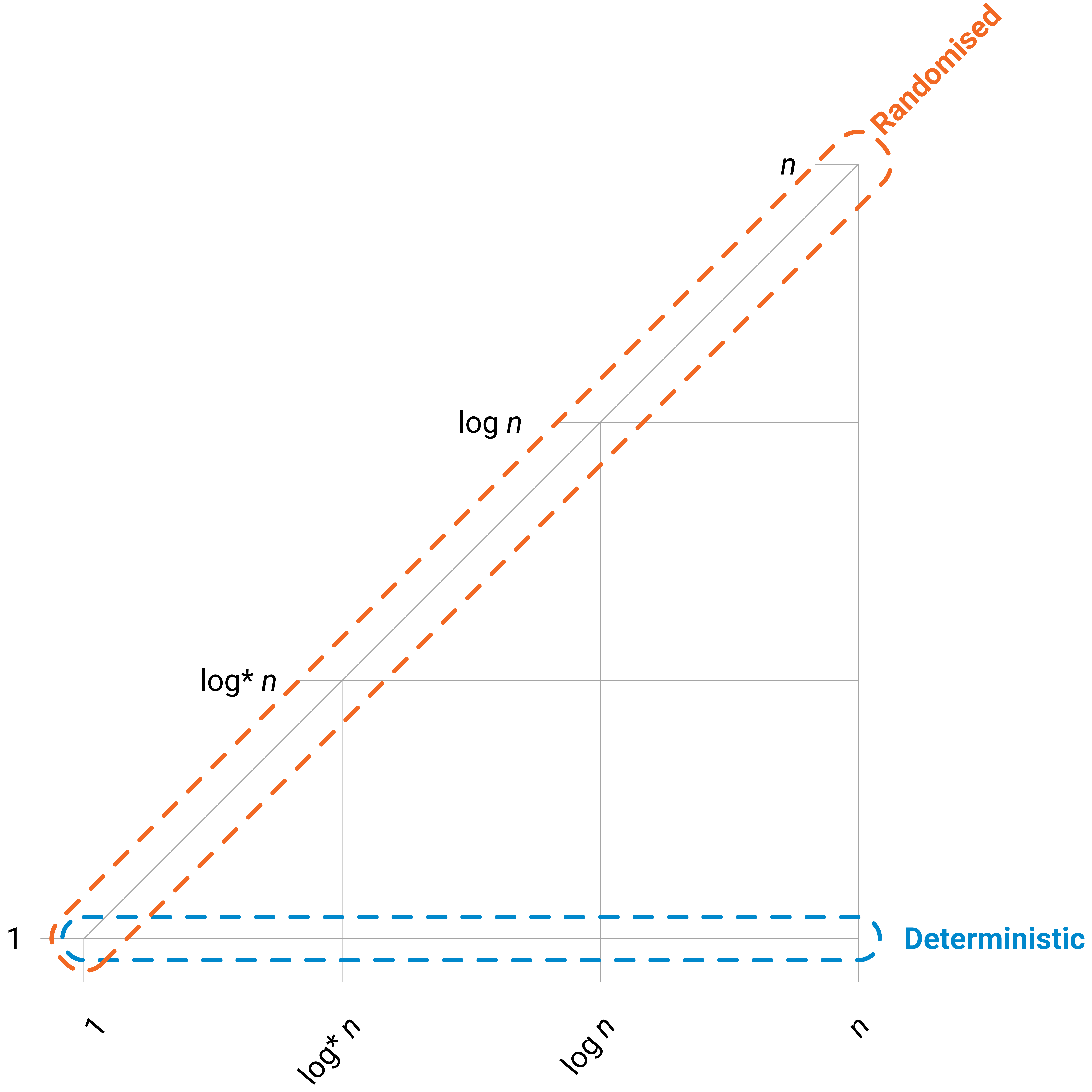
Landscape of LCLs

- **Which time complexities** are possible for **LCLs**?
- How **local** are **LCLs**?
- **Does randomness help** in solving an **LCL** faster?

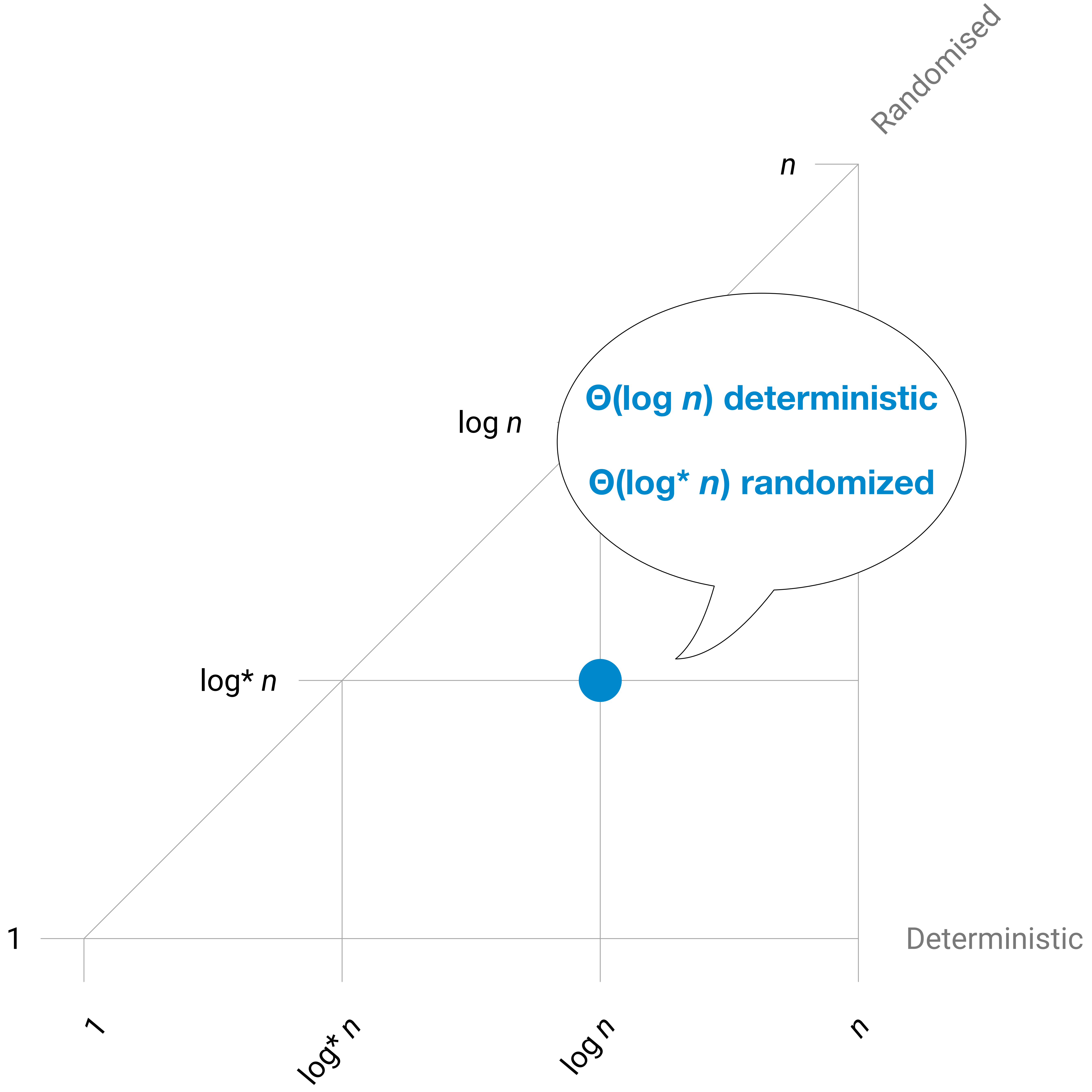


Randomised
n
log n
log* n
1
Deterministic
1
log* n
log n
n


Randomised
n
log n
log* n
1
Deterministic
1
log* n
log n
n

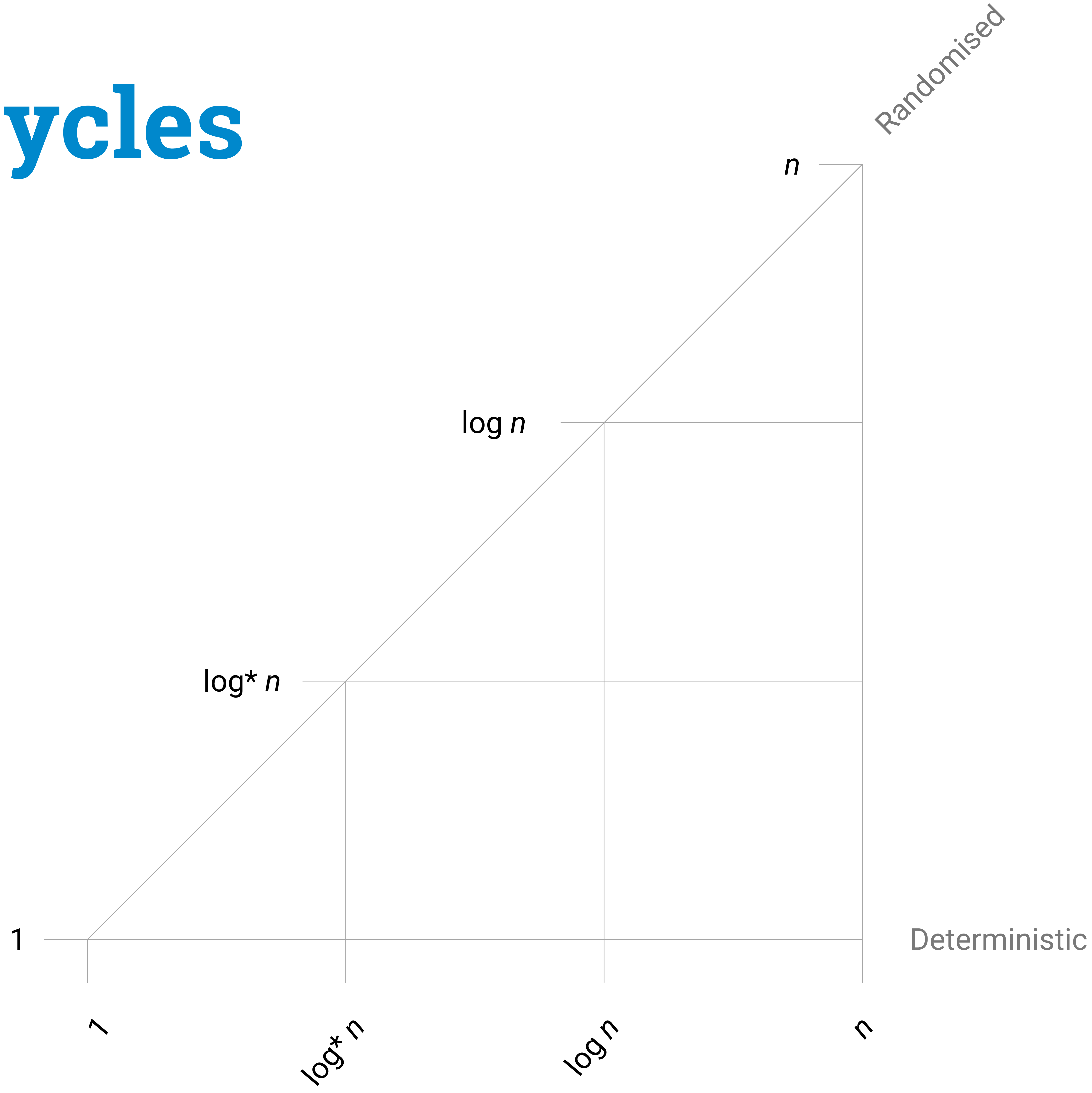


Randomised
n
log n
log* n
1
Deterministic
1
log* n
log n
n

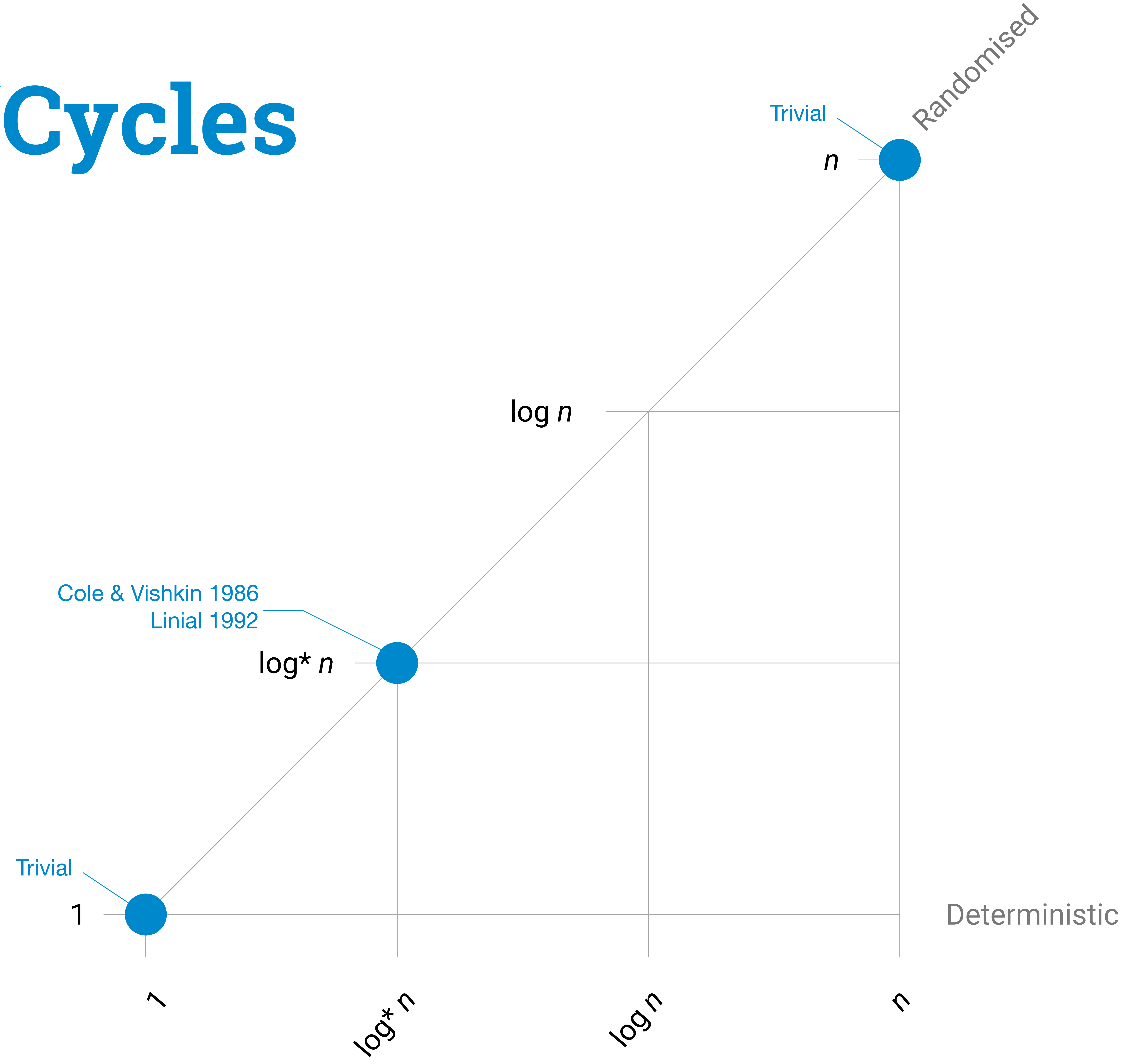


Randomised
n
log n
log* n
1
Θ(log n) deterministic
Θ(log* n) randomized
Deterministic
1
log* n
log n
n

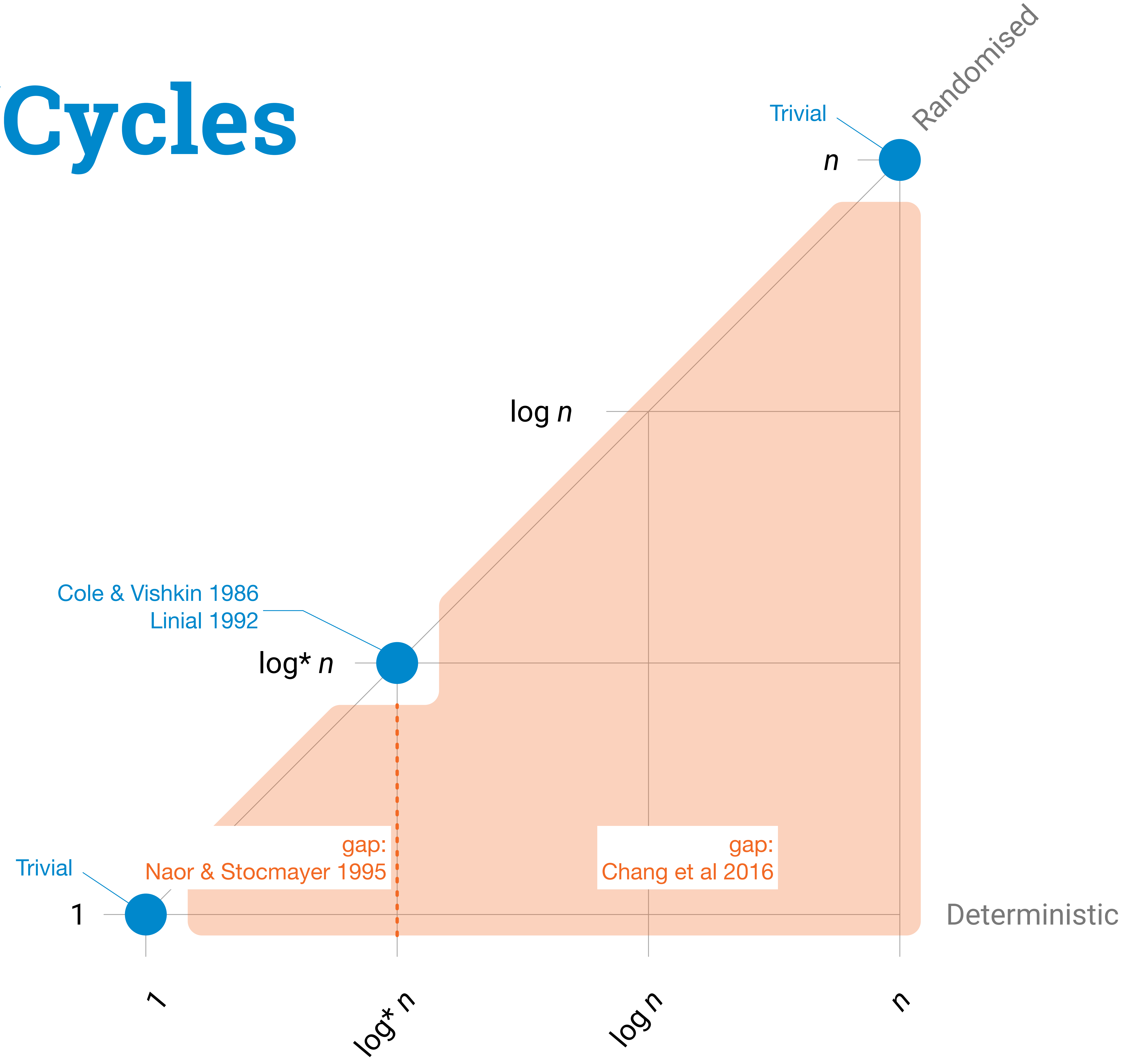
Paths/Cycles



Paths/Cycles



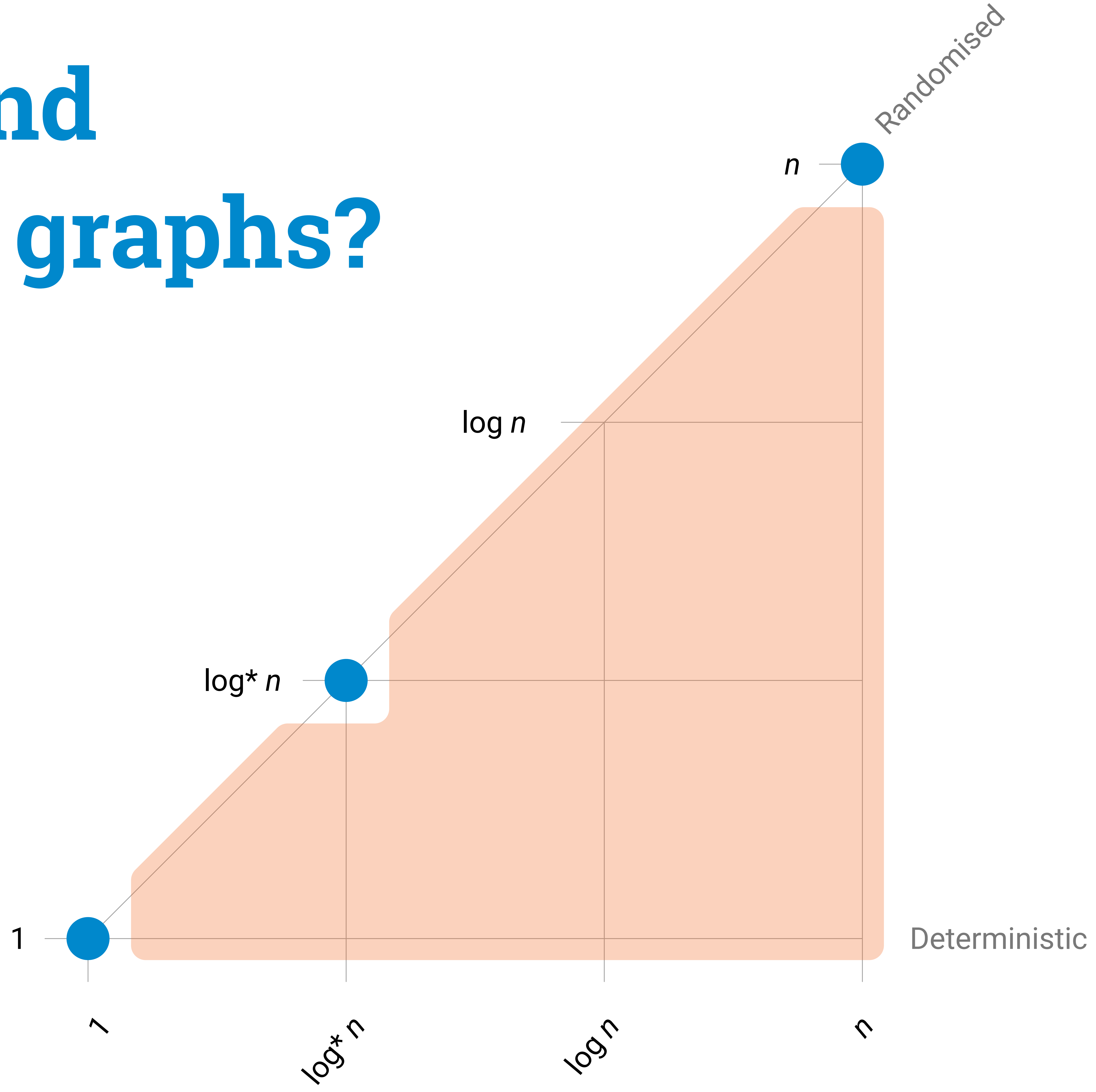
Paths/Cycles



Gaps

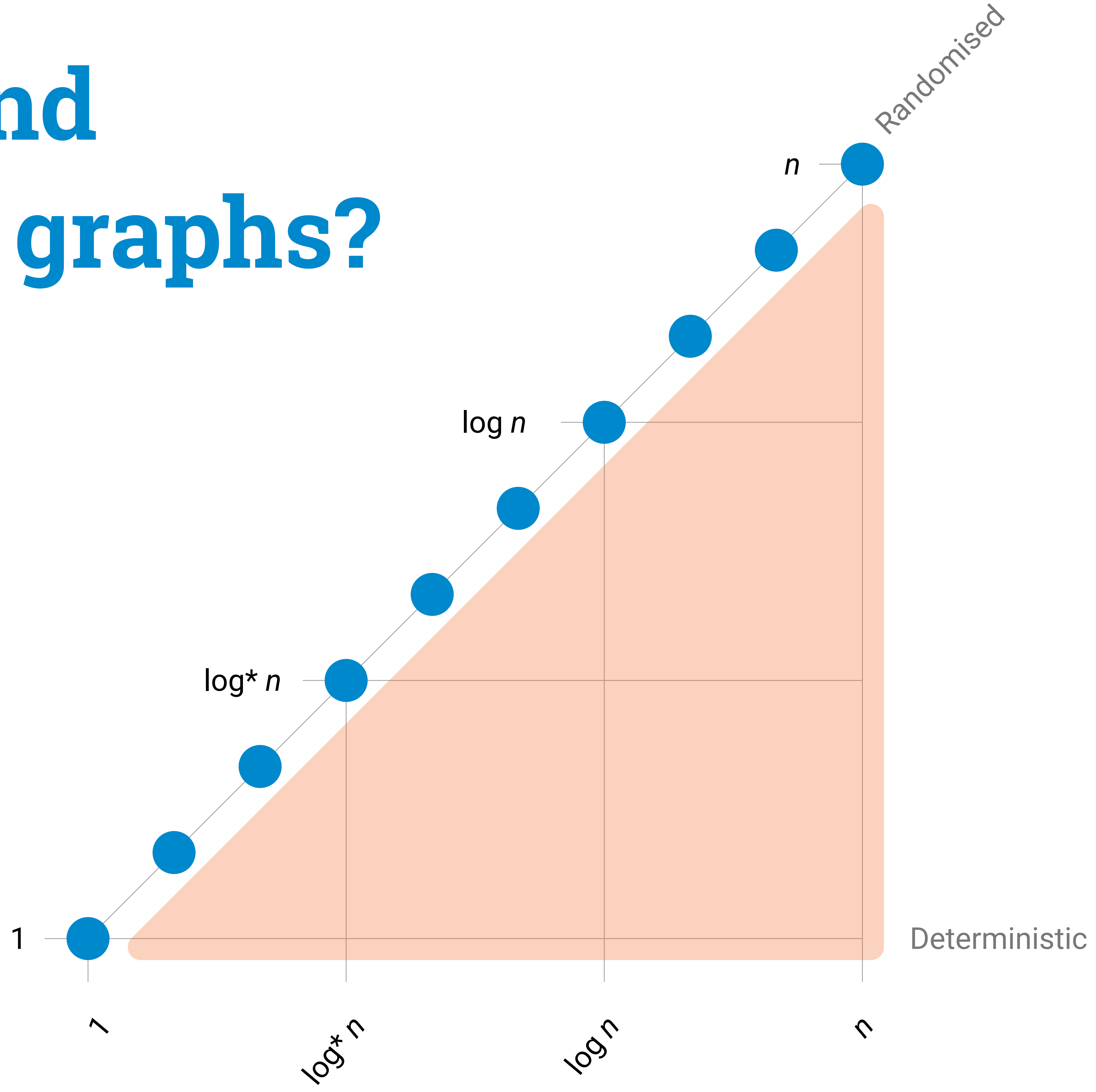
- **$\omega(1)$ — $o(\log^* n)$ gap**:
  - Every **algorithm A** that solves an **LCL P** in **$o(\log^* n)$** rounds can be **automatically sped up** into an **algorithm A'** that solves **P** in **$O(1)$** rounds
- **$\omega(\log^* n)$ — $o(n)$ gap**:
  - Every **algorithm A** that solves an **LCL P** in **$o(n)$** rounds can be **automatically sped up** into an algorithm **A'** that solves **P** in **$O(\log^* n)$** rounds

Trees and general graphs?



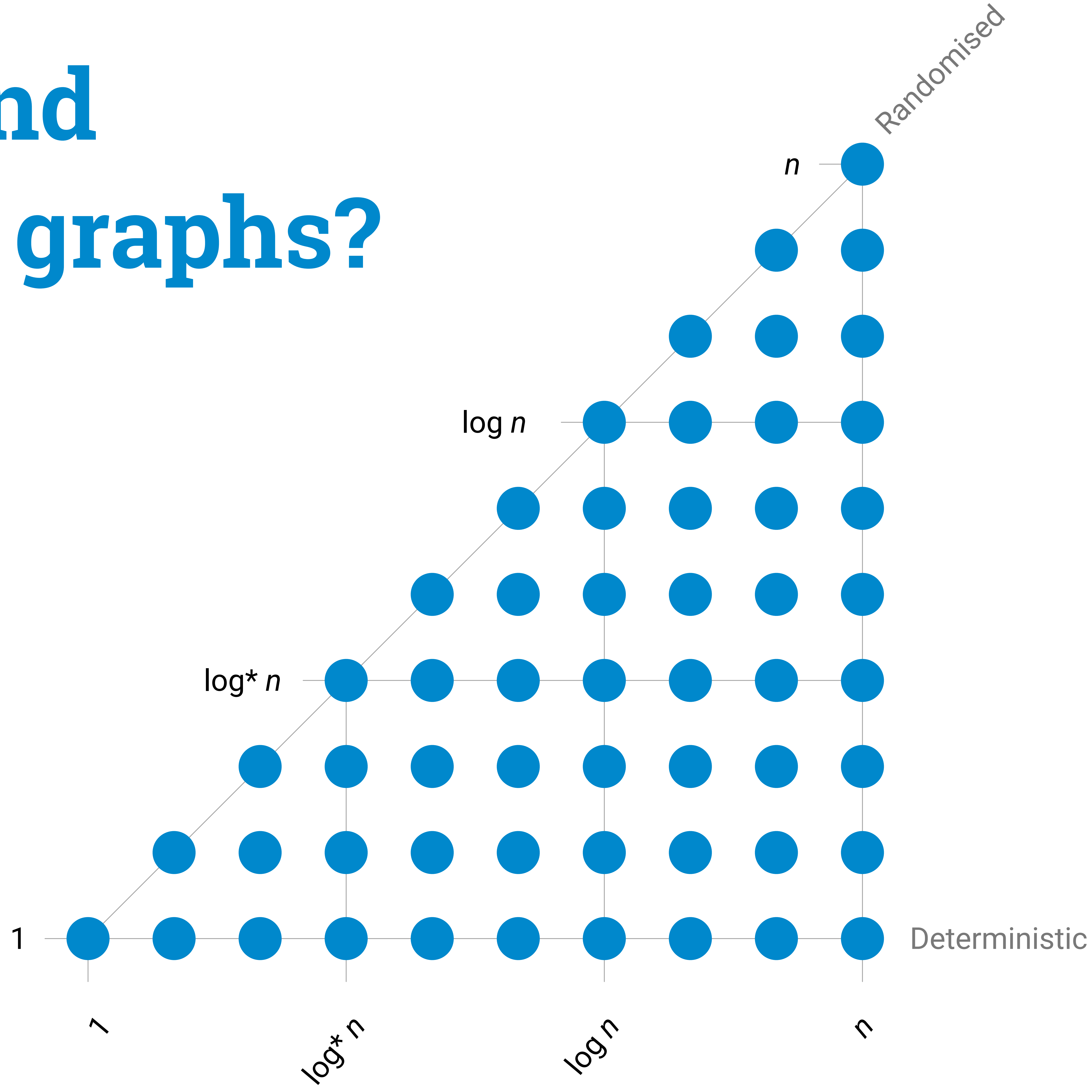
???

Trees and general graphs?



???

Trees and general graphs?

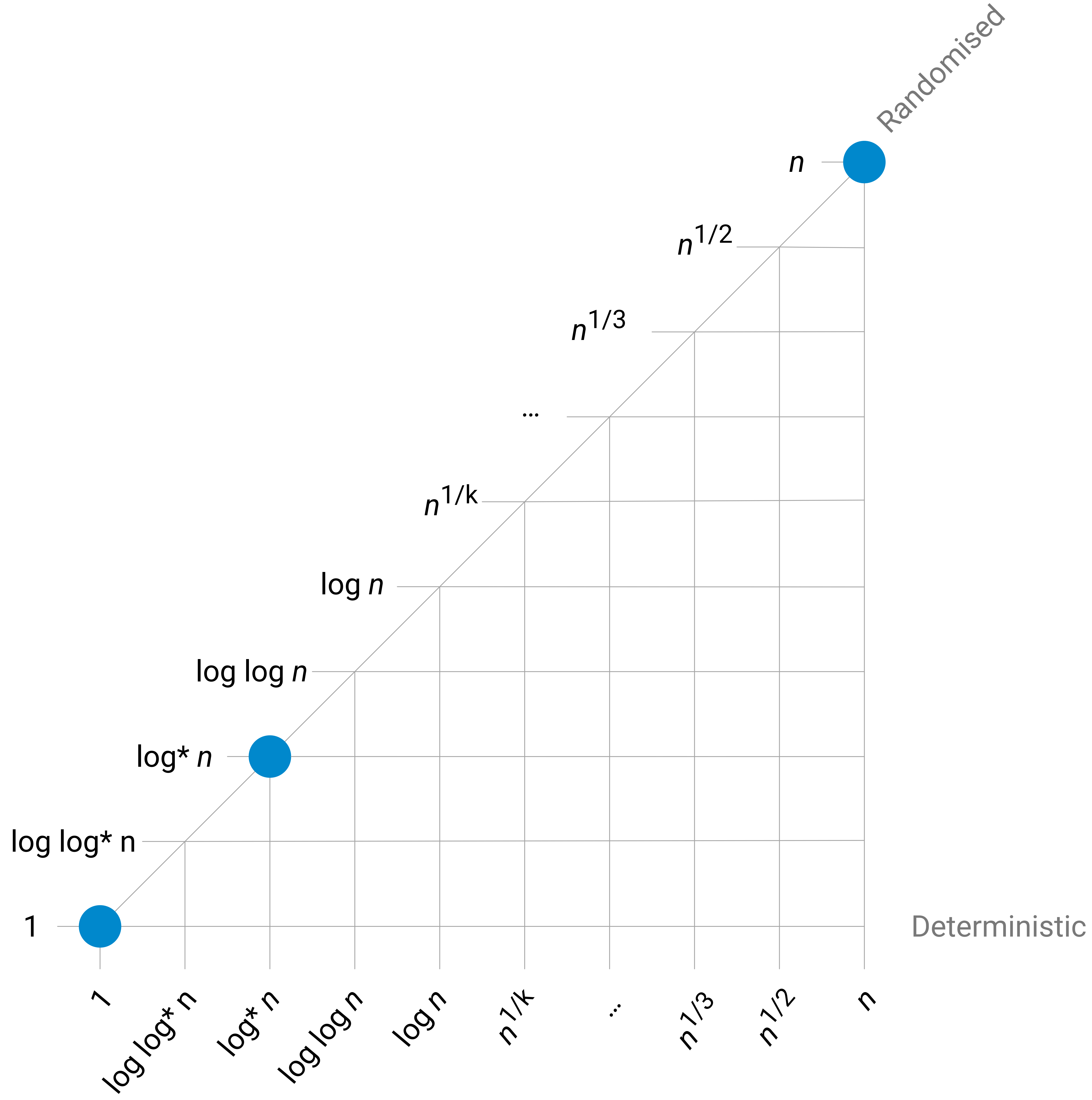


???

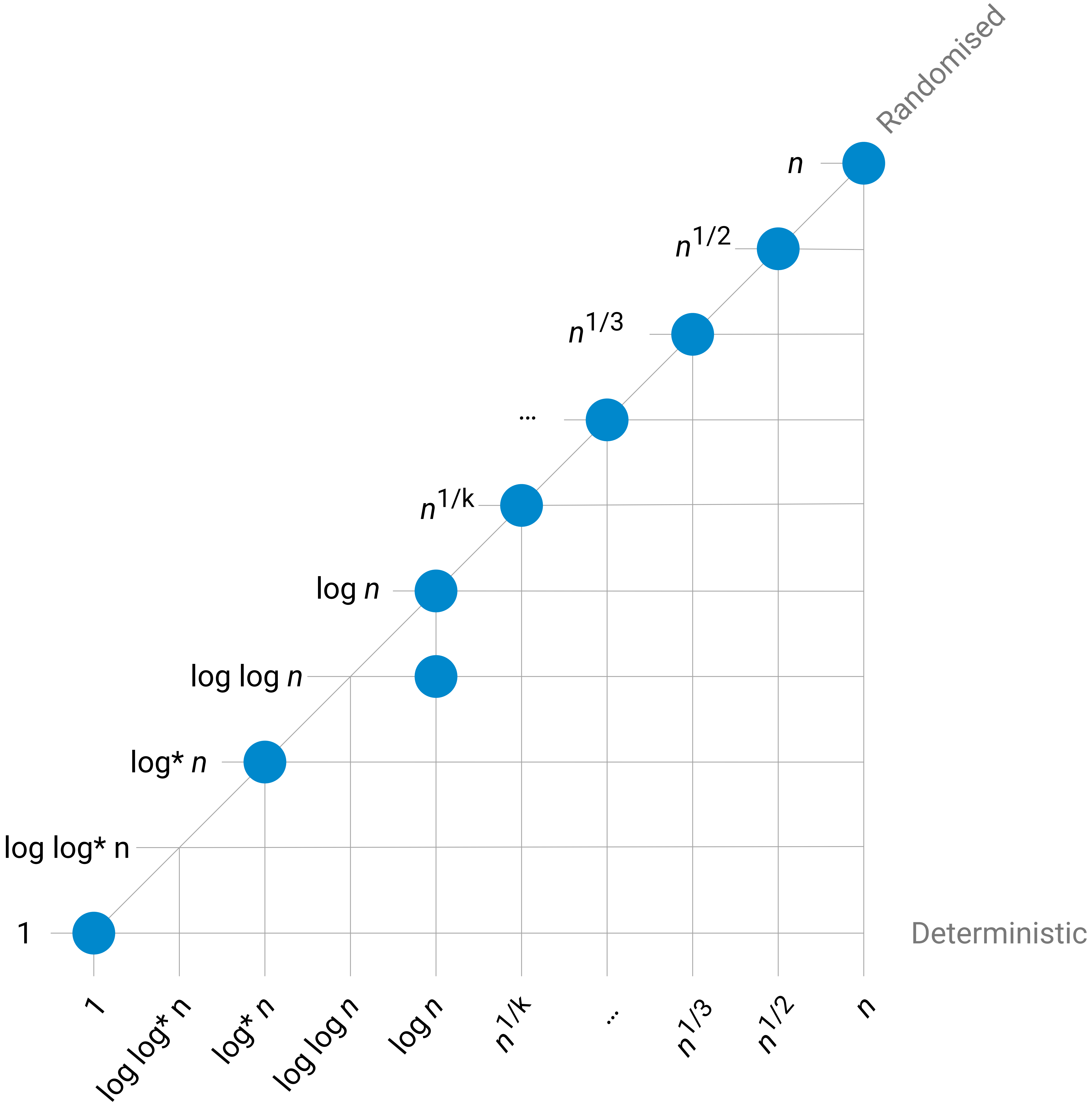
Lots of progress since 2016

- Brandt, Fischer, Hirvonen, Keller, Lempiäinen, Rybicki, Suomela, Uitto **[STOC 2016]**
- Chang, Kopelowitz, Pettie **[FOCS 2016]**
- Ghaffari, Su **[SODA 2017]**
- Brandt, Hirvonen, Korhonen, Lempiäinen, Östergård, Purcell, Rybicki, Suomela, Uznański **[PODC 2017]**
- Fischer, Ghaffari **[DISC 2017]**
- Chang, Pettie **[FOCS 2017]**
- Chang, He, Li, Pettie, Uitto **[SODA 2018]**
- B., Hirvonen, Korhonen, Lempiäinen, Olivetti, Suomela **[STOC 2018]**
- Ghaffari, Hirvonen, Kuhn, Maus **[PODC 2018]**
- B., Brandt, Olivetti, Suomela **[DISC 2018]**
- B., Brandt, Olivetti, Suomela **[Unpublished 2019]**

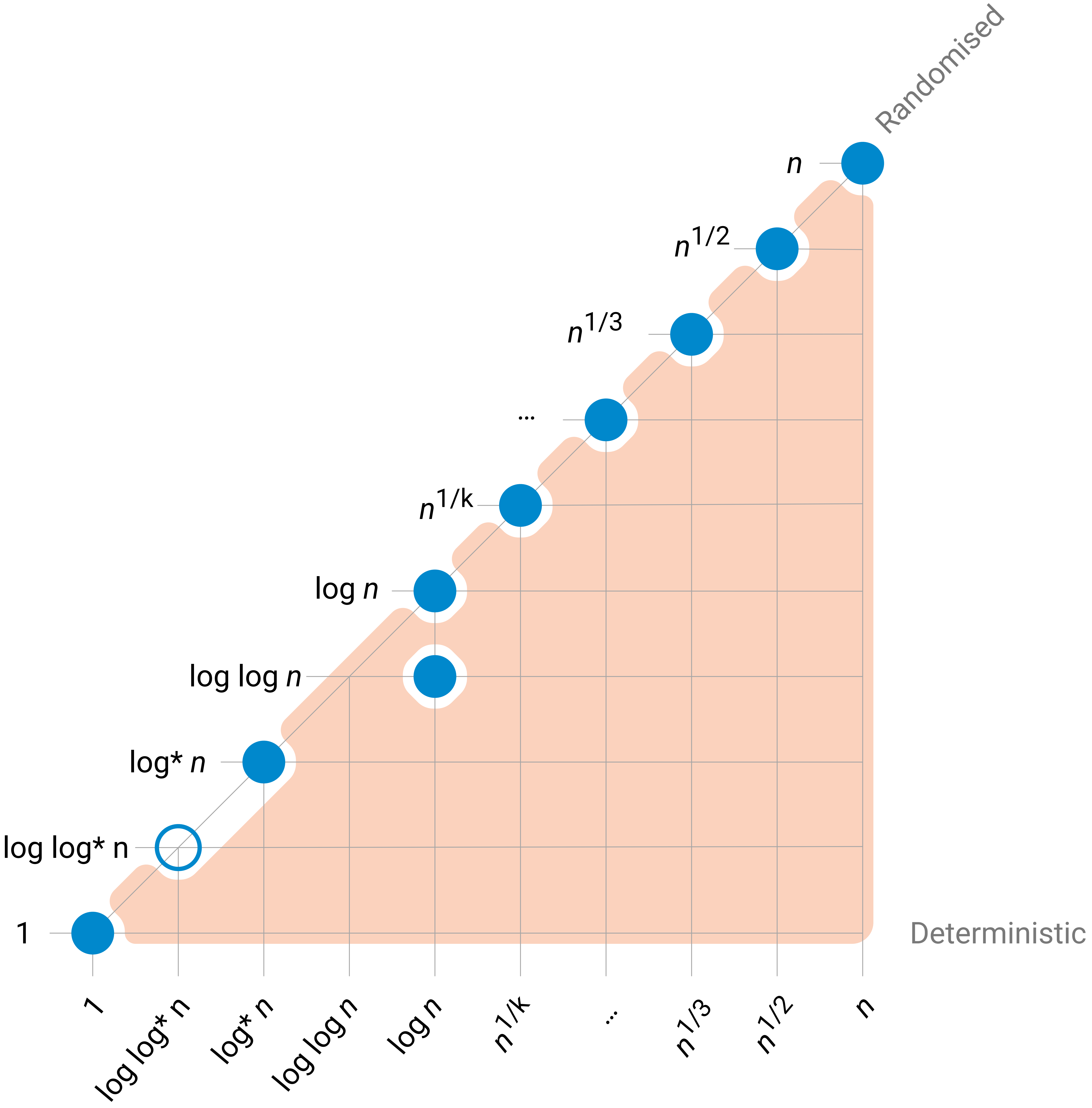
Trees


Randomised
n
n1/2
n1/3
...
n1/k
log n
log log n
log* n
log log* n
1
Deterministic
1
log log* n
log* n
log log n
log n
n 1/k
...
n 1/3
n 1/2
n

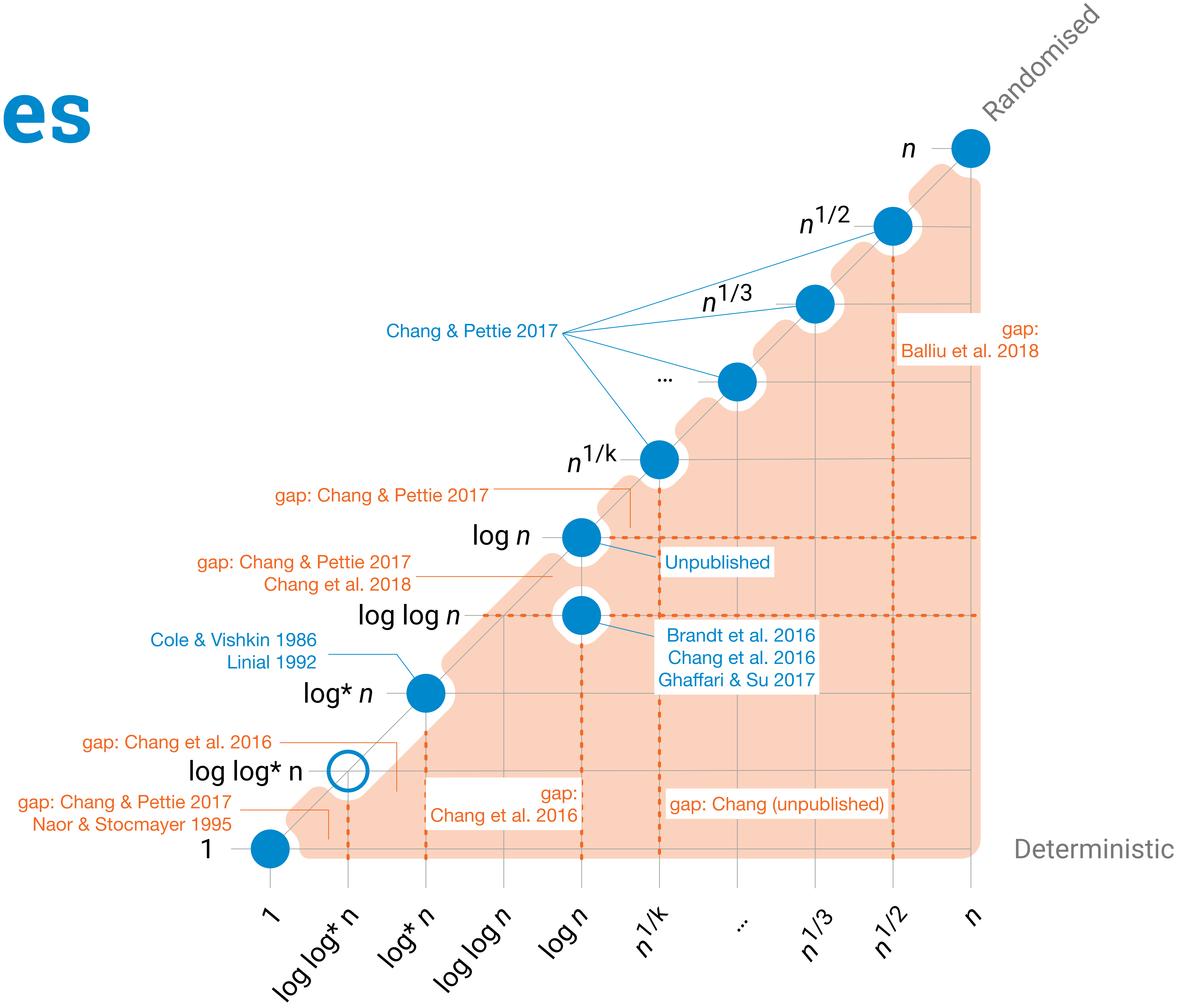
Trees


Randomised
Deterministic
n
n^1/2
n^1/3
...
n^1/k
log n
log log n
log* n
log log* n
1
1
log log* n
log* n
log log n
log n
n 1/k
...
n 1/3
n 1/2
n

Trees

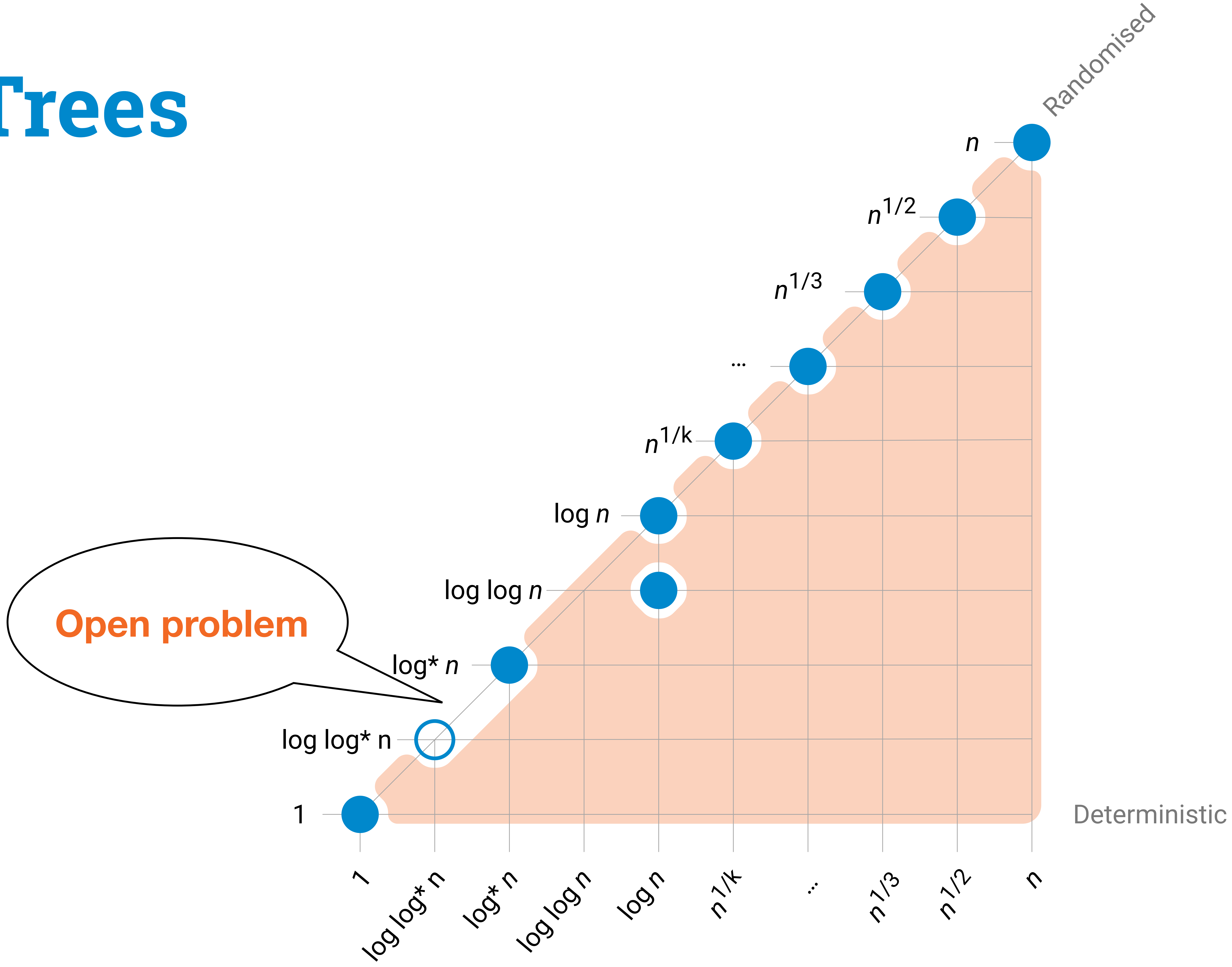

Randomised
n
$n^{1/2}$
$n^{1/3}$
...
$n^{1/k}$
log n
log log n
log* n
log log* n
1
Deterministic
1
log log* n
log* n
log log n
log n
n 1/k
...
n 1/3
n 1/2
n

Trees



Randomised
Deterministic
n
n1/2
n1/3
...
n1/k
log n
log log n
log* n
log log* n
1
Chang & Pettie 2017
gap: Balliu et al. 2018
gap: Chang & Pettie 2017
Unpublished
gap: Chang & Pettie 2017
Chang et al. 2018
Cole & Vishkin 1986
Linial 1992
Brandt et al. 2016
Chang et al. 2016
Ghaffari & Su 2017
gap: Chang et al. 2016
gap: Chang & Pettie 2017
Naor & Stocmayer 1995
gap: Chang et al. 2016
gap: Chang (unpublished)
1
log log* n
log* n
log log n
log n
n1/k
...
n1/3
n1/2
n

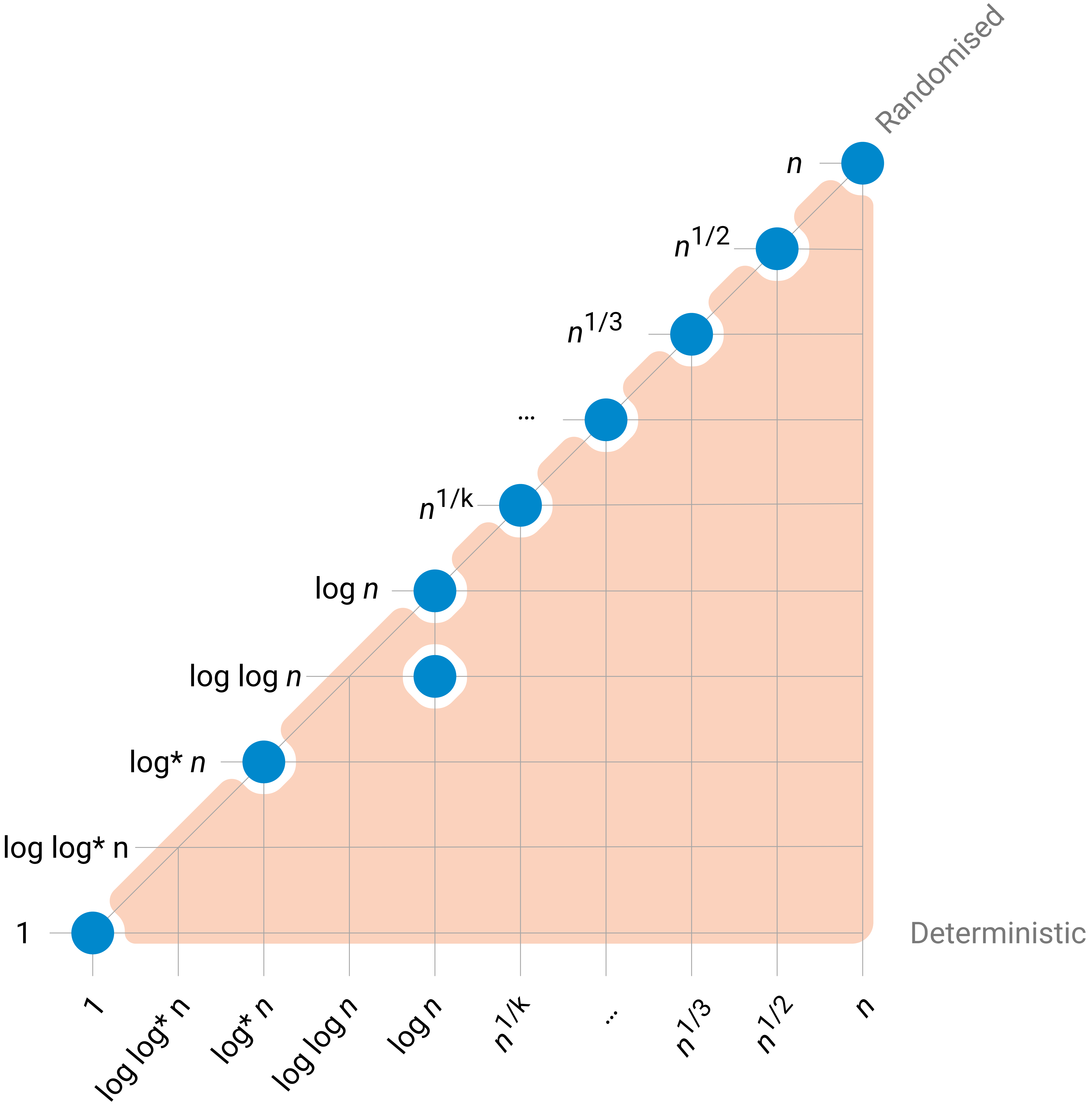
Trees



Randomised
n
$n^{1/2}$
$n^{1/3}$
...
$n^{1/k}$
log n
log log n
log* n
log log* n
1
Open problem
Deterministic
1
log log* n
log* n
log log n
log n
n 1/k
...
n 1/3
n 1/2
n

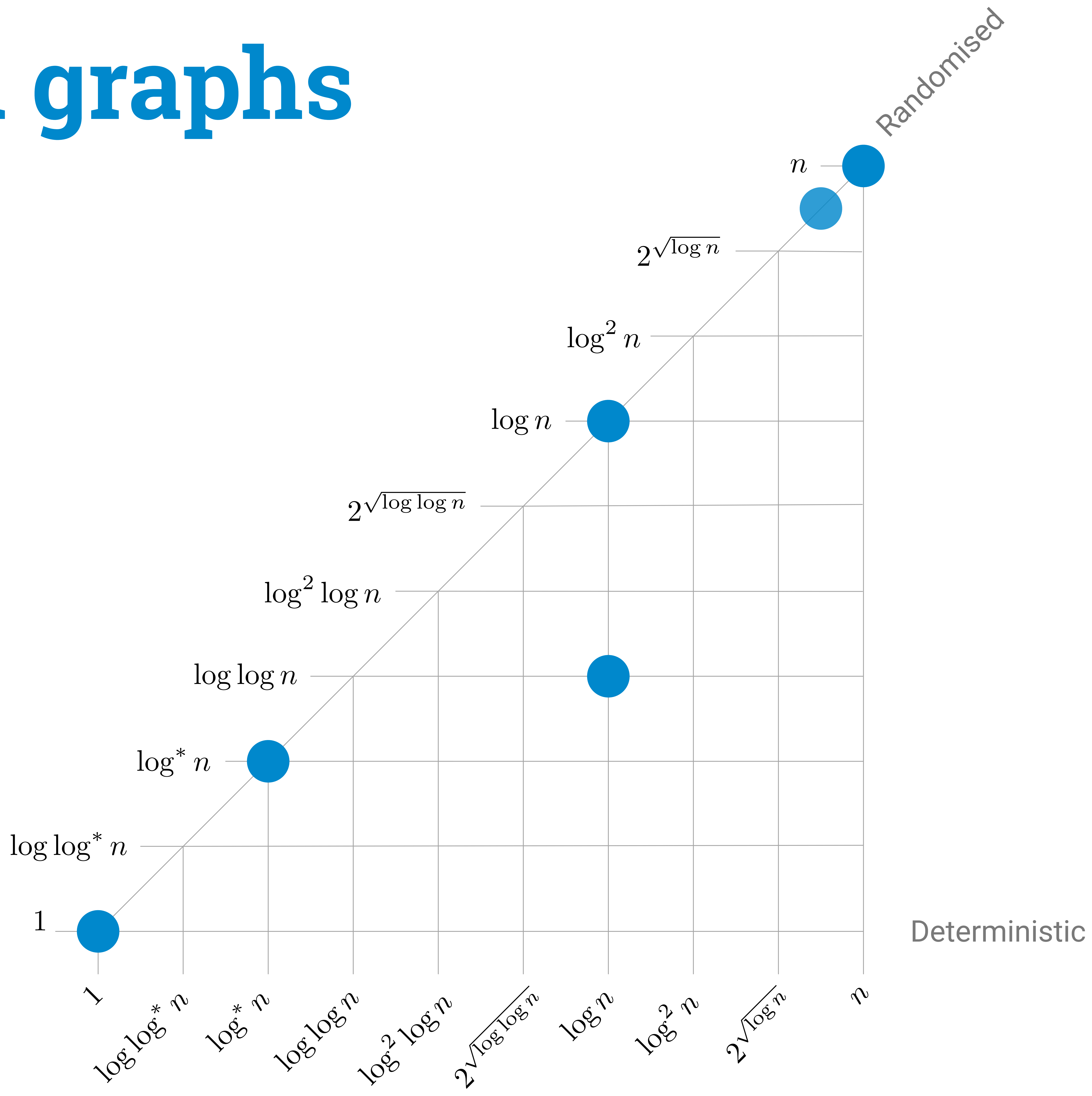
Trees

Homogeneous LCLs

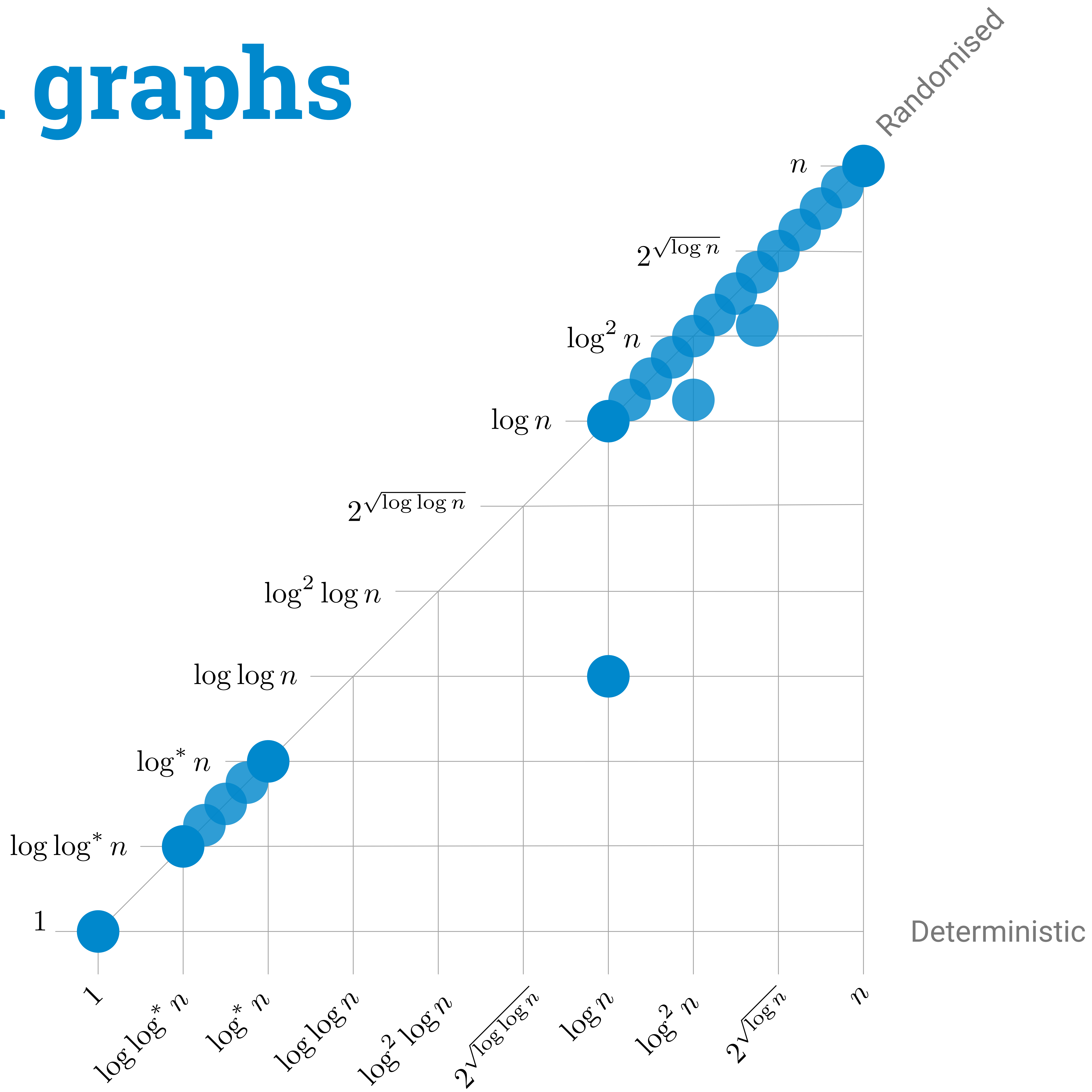


[B., Hirvonen, Olivetti, Suomela 2019]

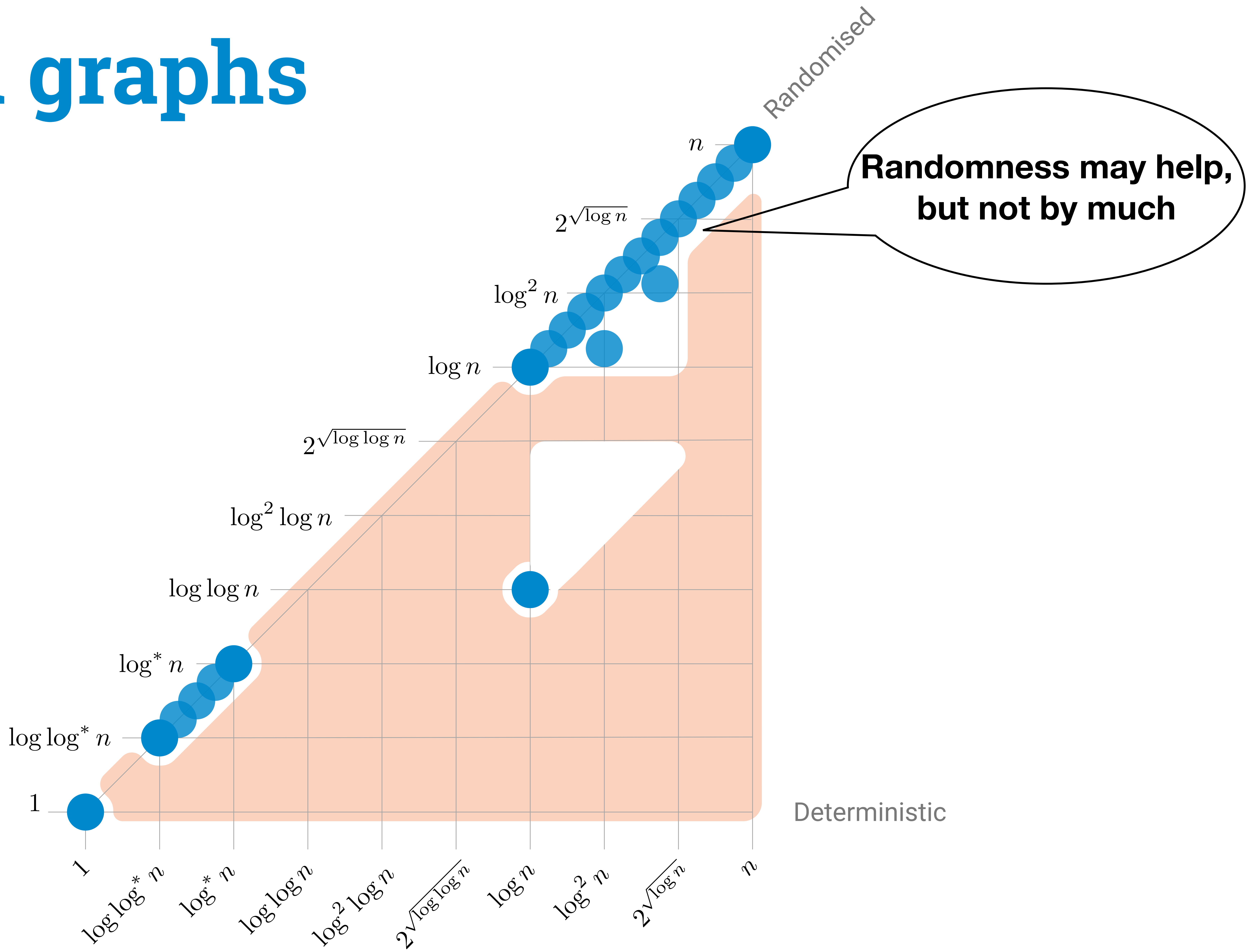
General graphs



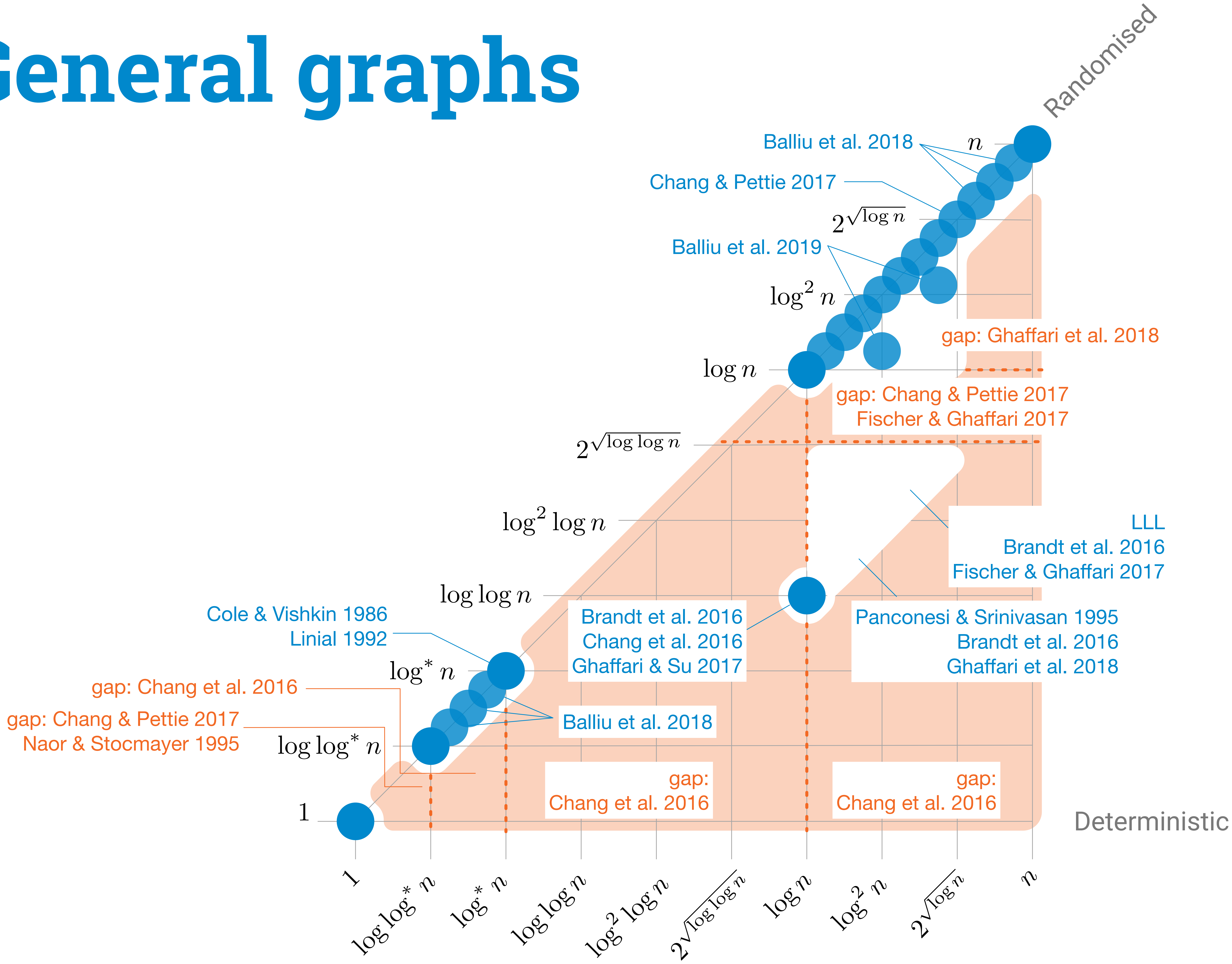
General graphs



General graphs

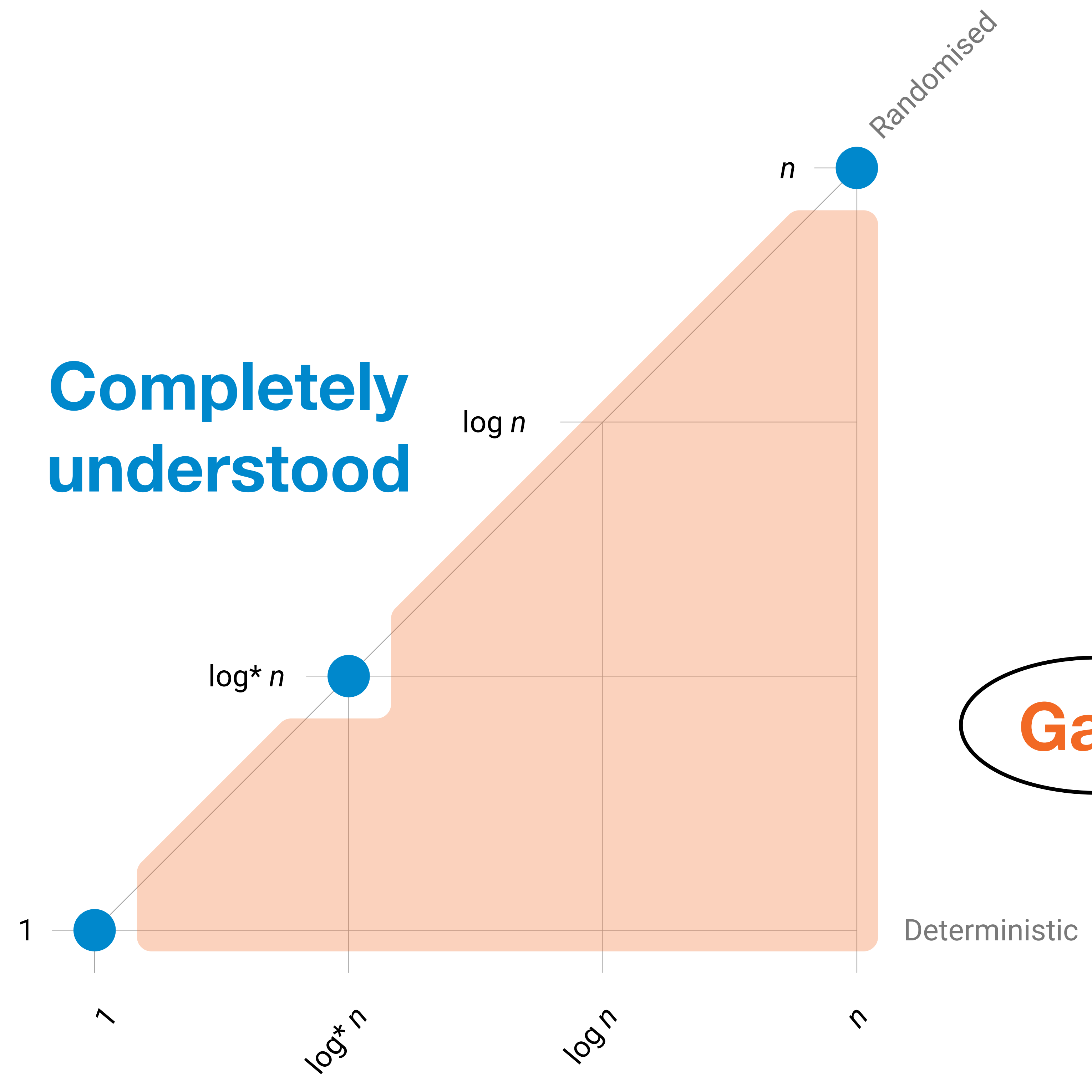


General graphs

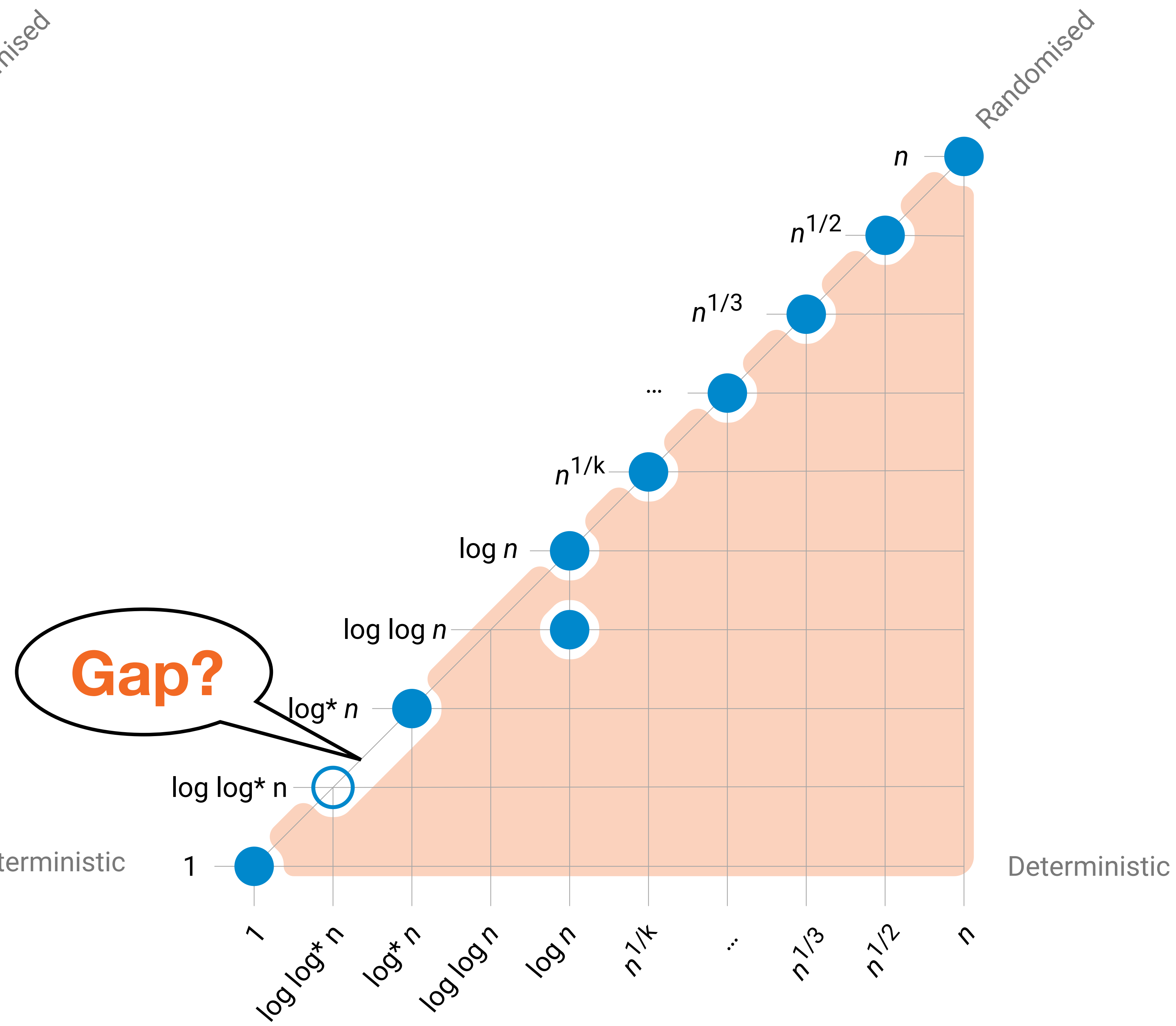


Paths/Cycles

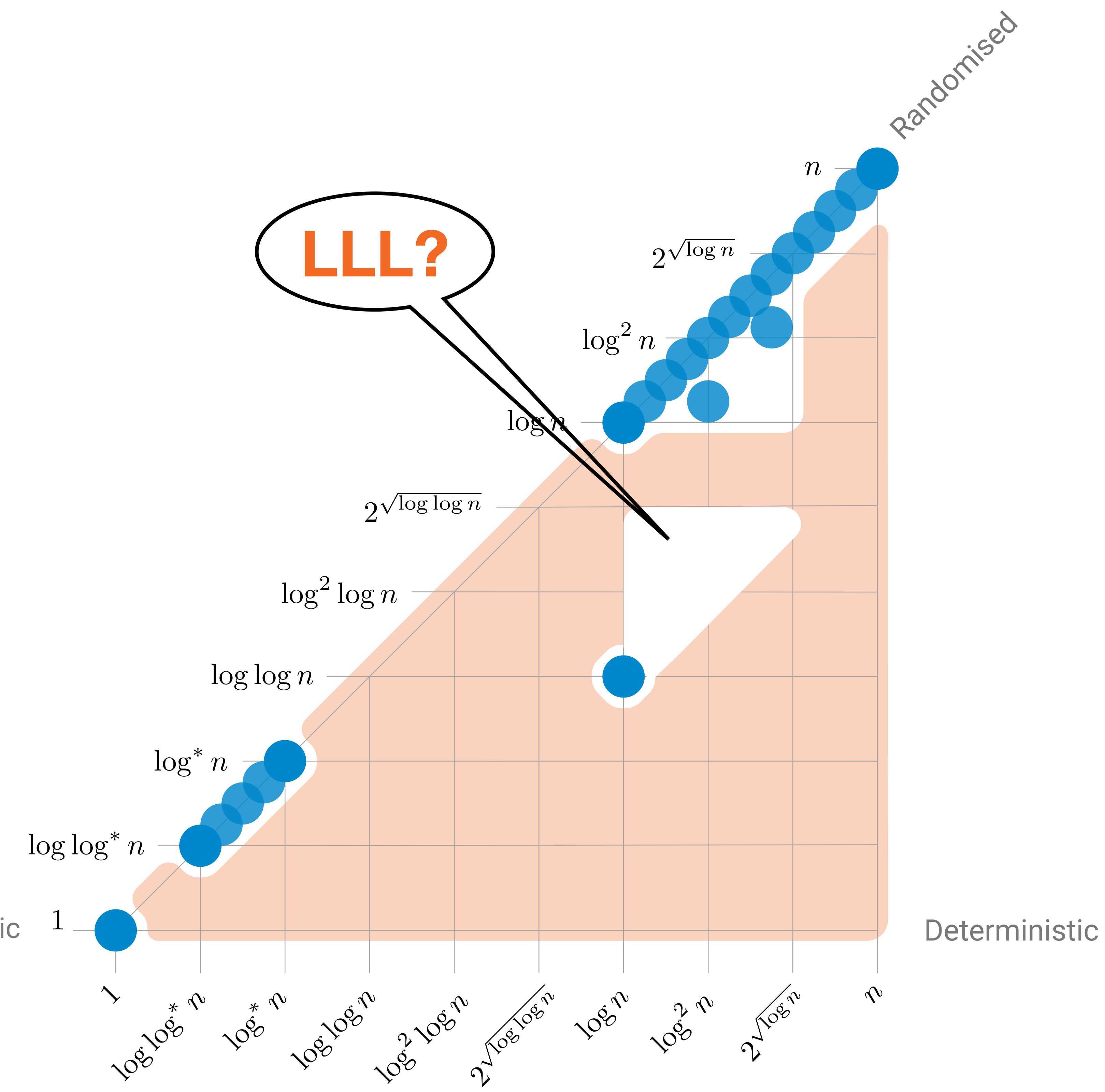
Completely understood



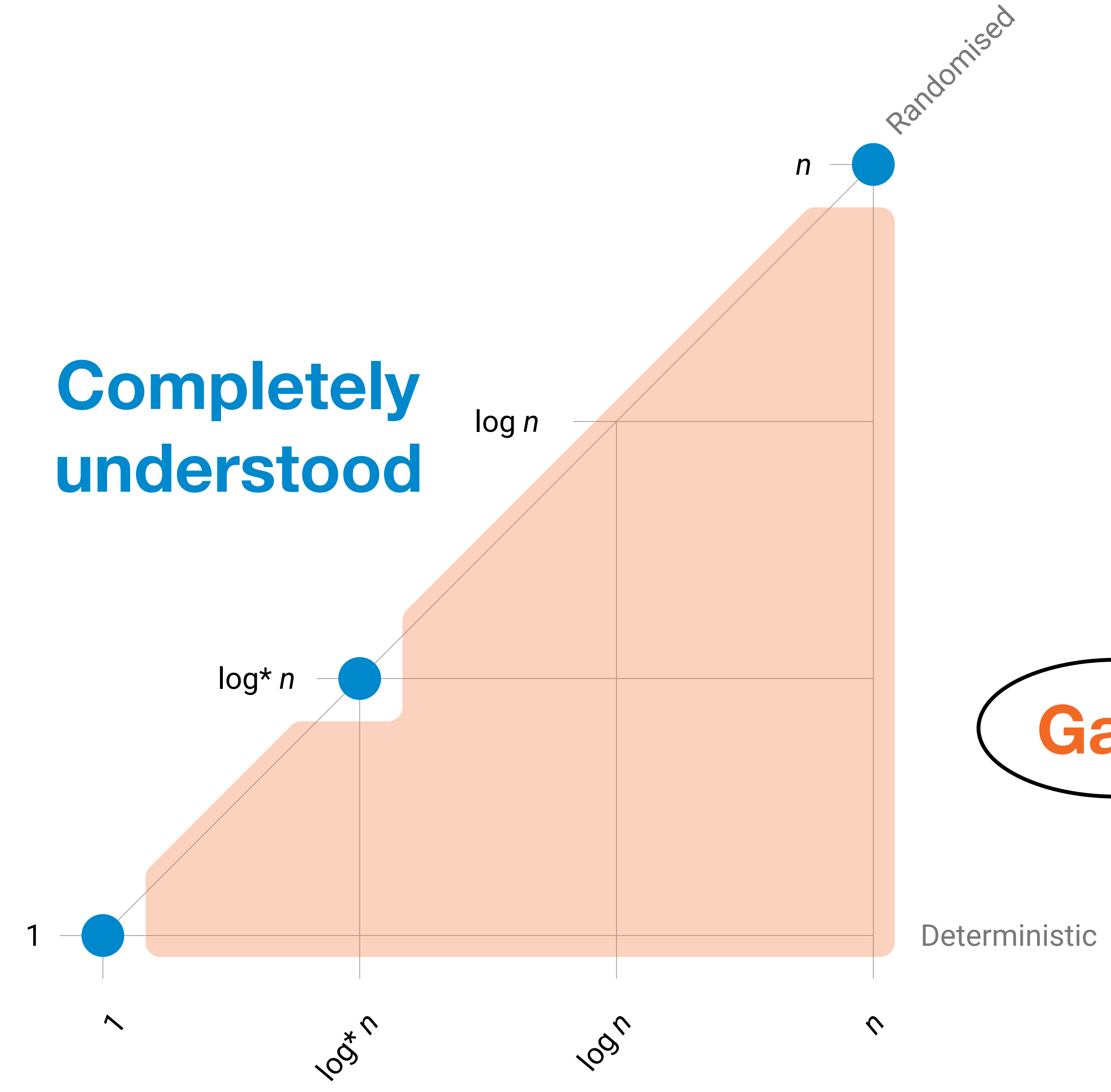
Trees



General graphs

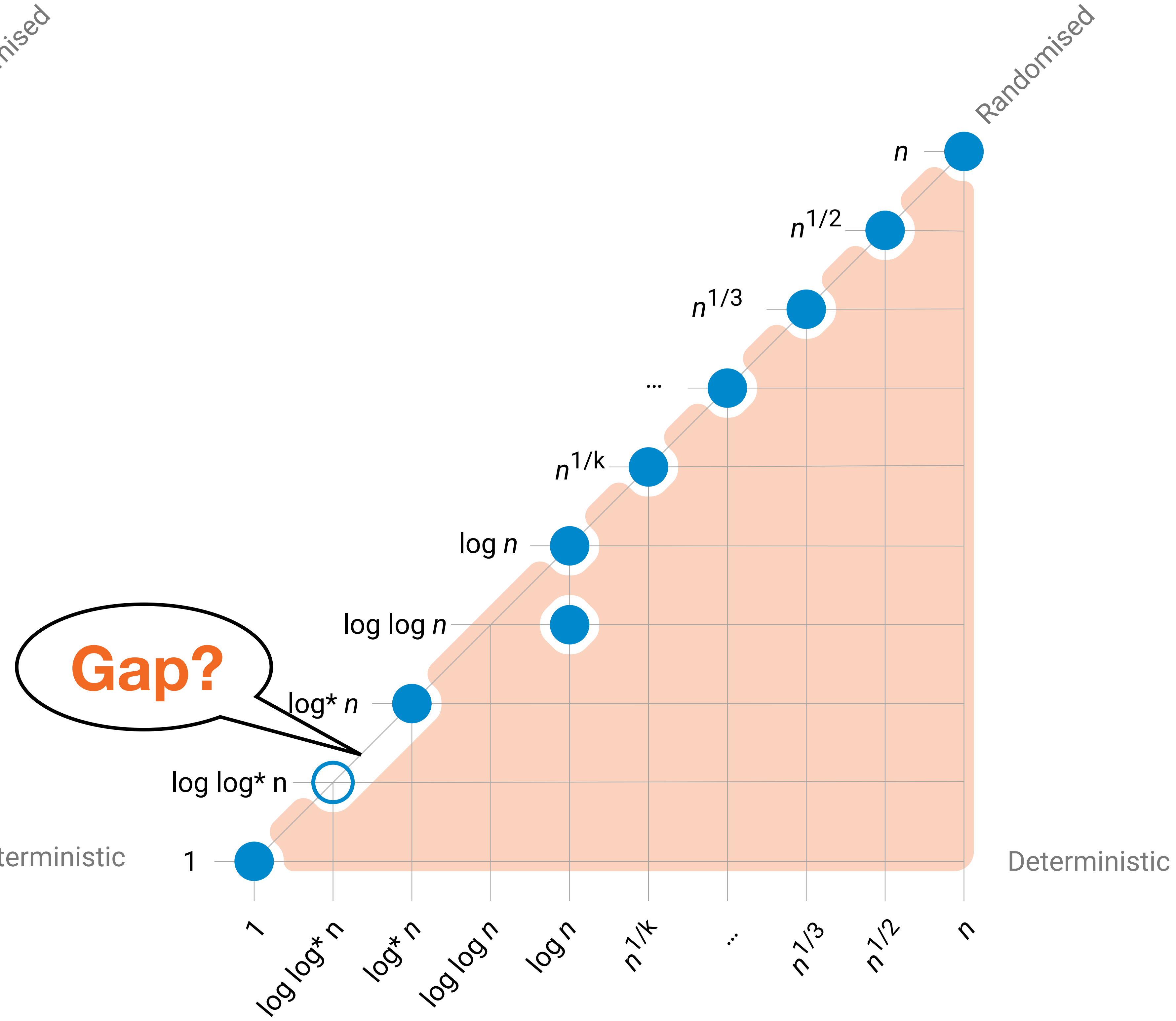


Paths/Cycles

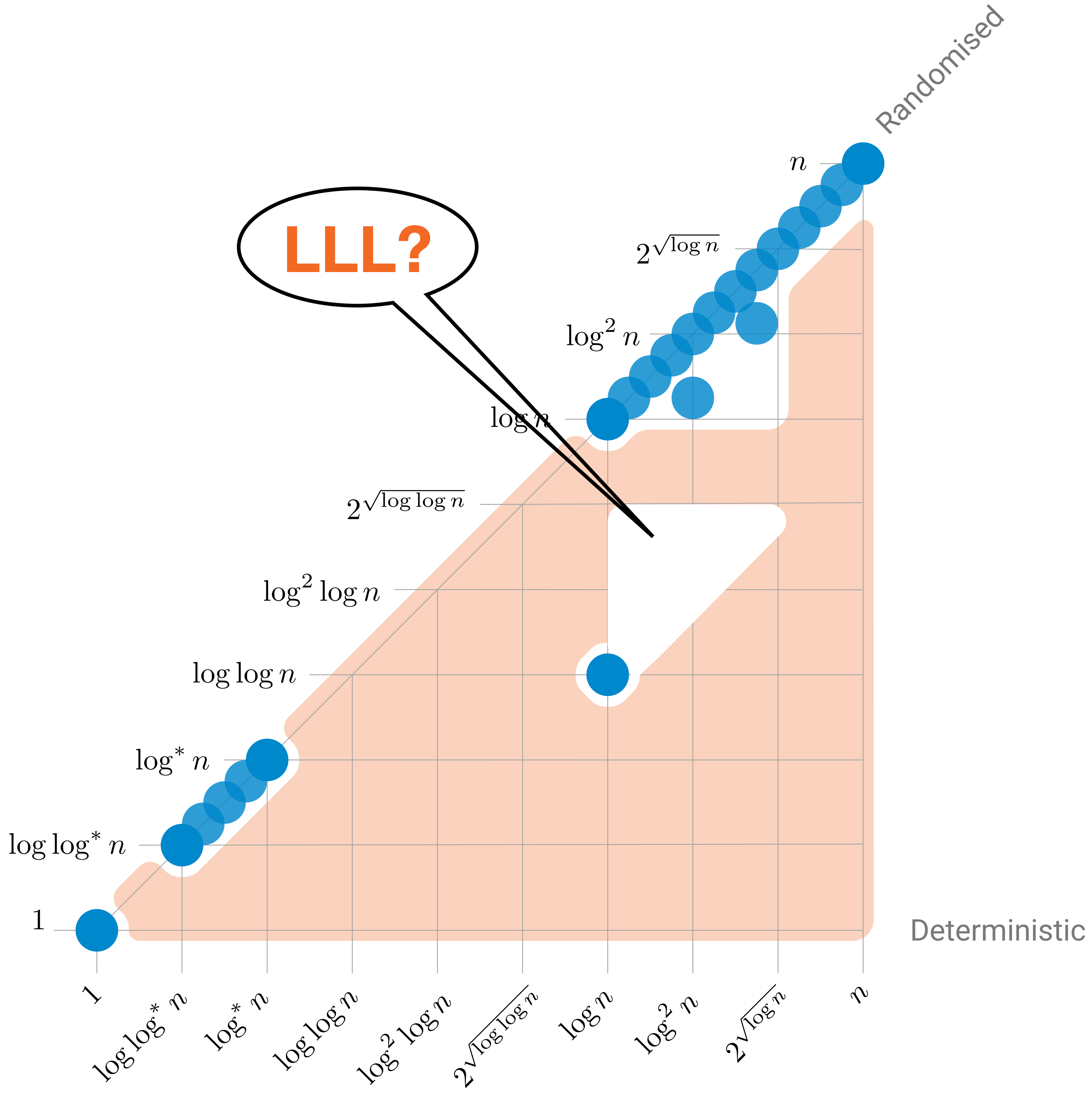


Completely understood

Trees



General graphs



Thank you!