Authors:

Arvid Heise (HPI)
Jorge Quiané (QCRI)
Ziawasch Abedjan (HPI)
Anja Jentzsch (HPI)
Felix Naumann (HPI)

معهد قطر لبحوث الحوسبة
Qatar Computing Research Institute
Member of Qatar Foundation عضو في مؤسسة قطر

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Scalable Discovery of Unique Column Combinations

VLDB 2014

Being

Unique




OK, and in DB terms?

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

Uniques: {Phone}, {Phone, City},...
Non-Unique: {First}, {First, City}..

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

~~Uniques~~: {Phone}, {Phone, City},...
~~Non-Unique~~: {First}, {First, City}..

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

~~Uniques: {Phone}, {Phone, City},...~~
~~Non-Unique: {First}, {First, City}..~~

Minimal-Uniques: {Phone}, {First, Last},...
Maximal-Non-Unique: {First, City}..

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

~~Uniques~~: {Phone}, {Phone, City},...
~~Non-Unique~~: {First}, {First, City}..

Minimal-Uniques: {Phone}, {First, Last},...
Maximal-Non-Unique: {First, City}..

(Non-)Unique Column Combinations

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

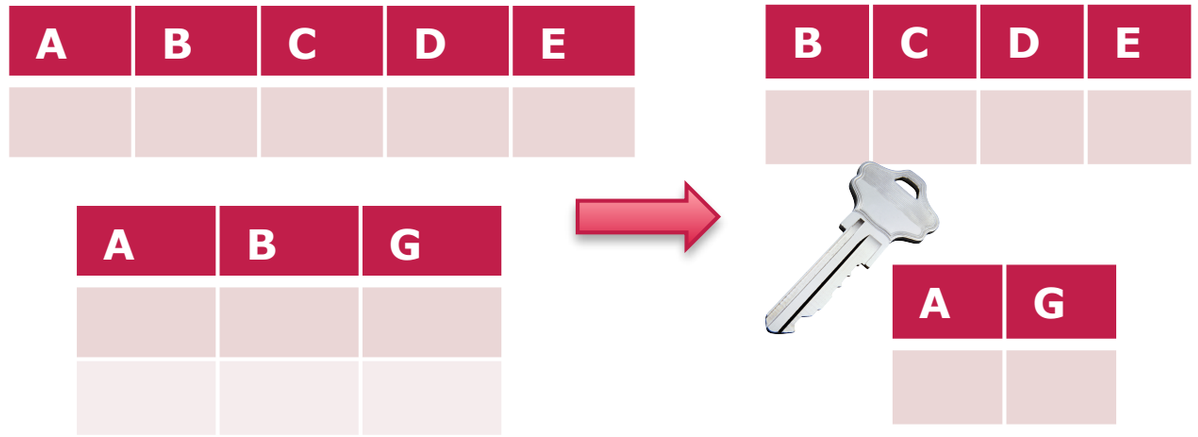
~~Uniques: {Phone}, {Phone, City},...~~
~~Non-Unique: {First}, {First, City}..~~

Minimal-Uniques: {Phone}, {First, Last},...
Maximal-Non-Unique: {First, City}..

(Non-)Uniques in Practice

(Non-)Uniques in Practice

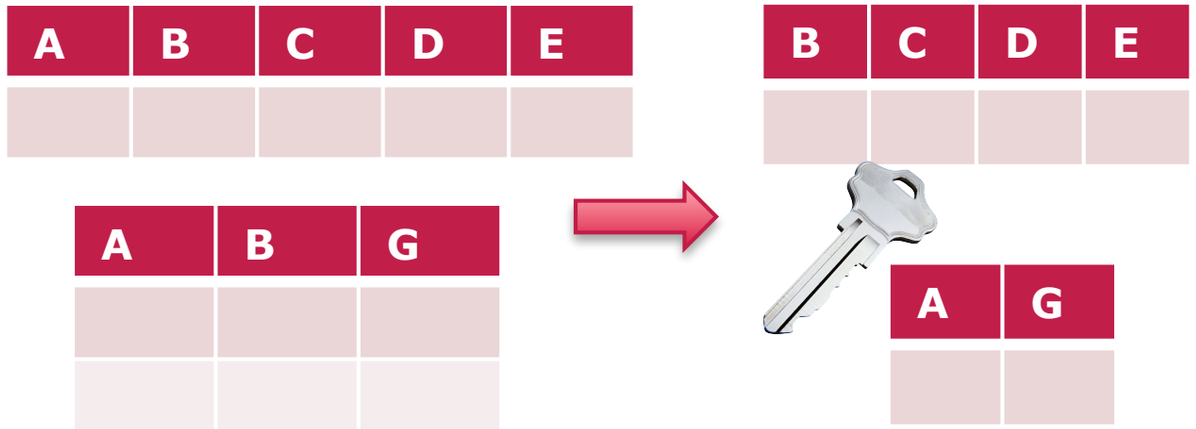
- Data reverse-engineering
  - Uniques are candidates keys



(Non-)Uniques in Practice

- Data reverse-engineering
  - Uniques are candidates keys

- Data quality monitoring
  - Anomaly detection
  - Reactive duplicate detection

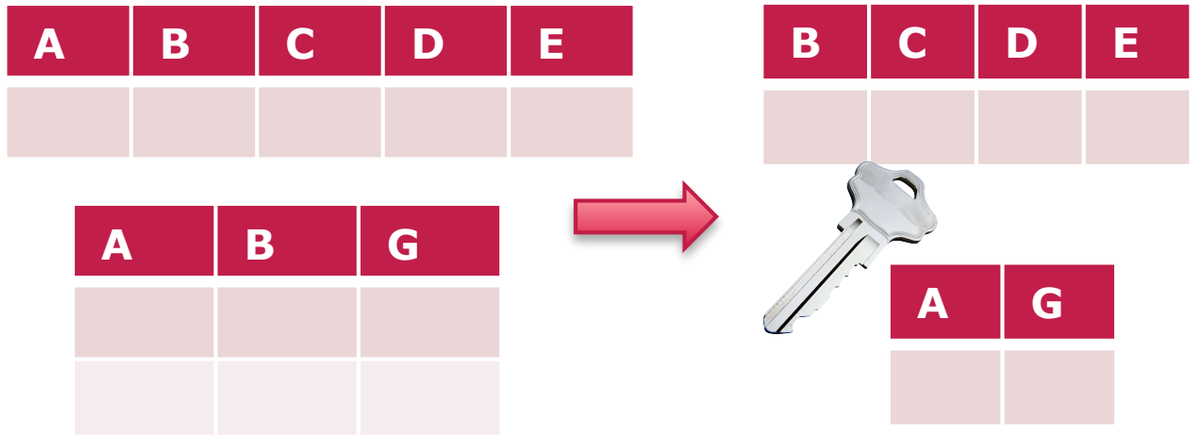


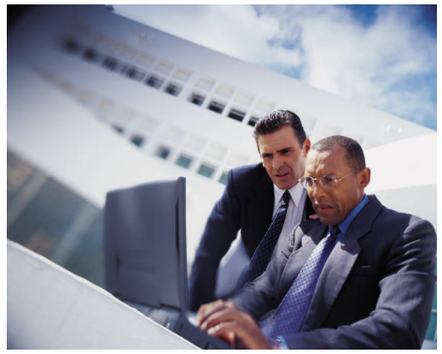
(Non-)Uniques in Practice

- Data reverse-engineering
  - Uniques are candidates keys

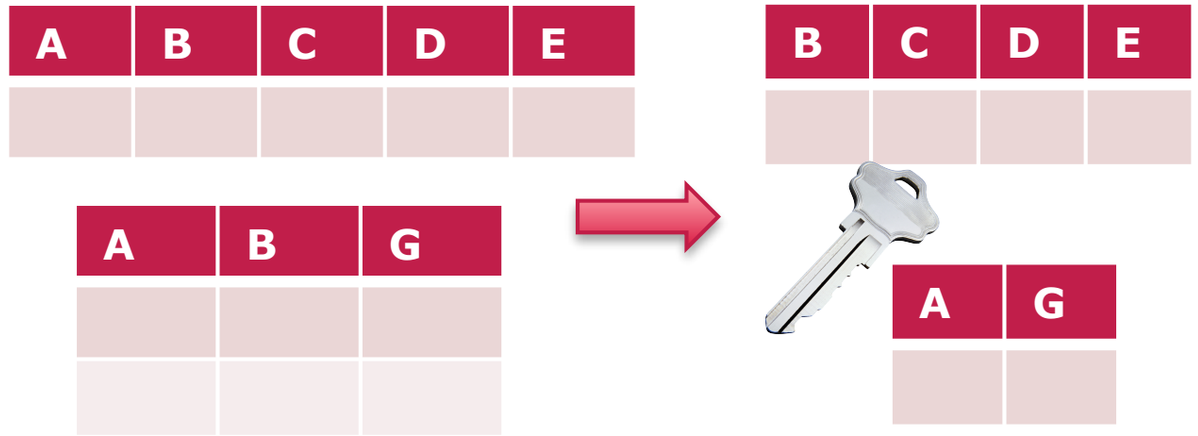
- Data quality monitoring
  - Anomaly detection
  - Reactive duplicate detection

- Database administration
  - Indexing
  - Query optimization

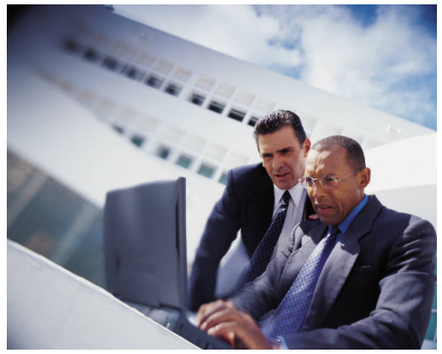




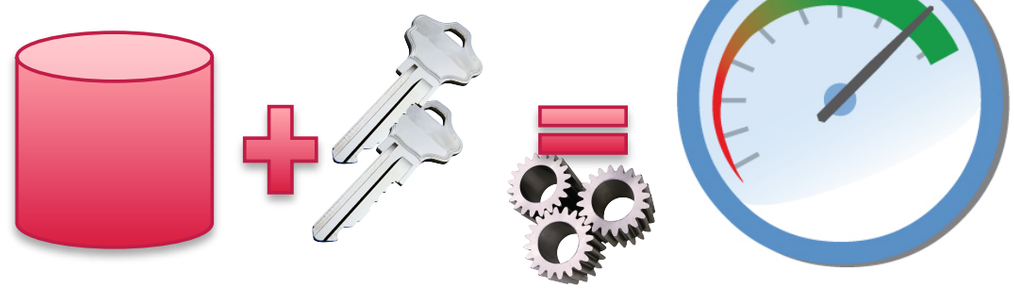
(Non-)Uniques in Practice



- Data reverse-engineering
  - Uniques are candidates keys



- Data quality monitoring
  - Anomaly detection
  - Reactive duplicate detection

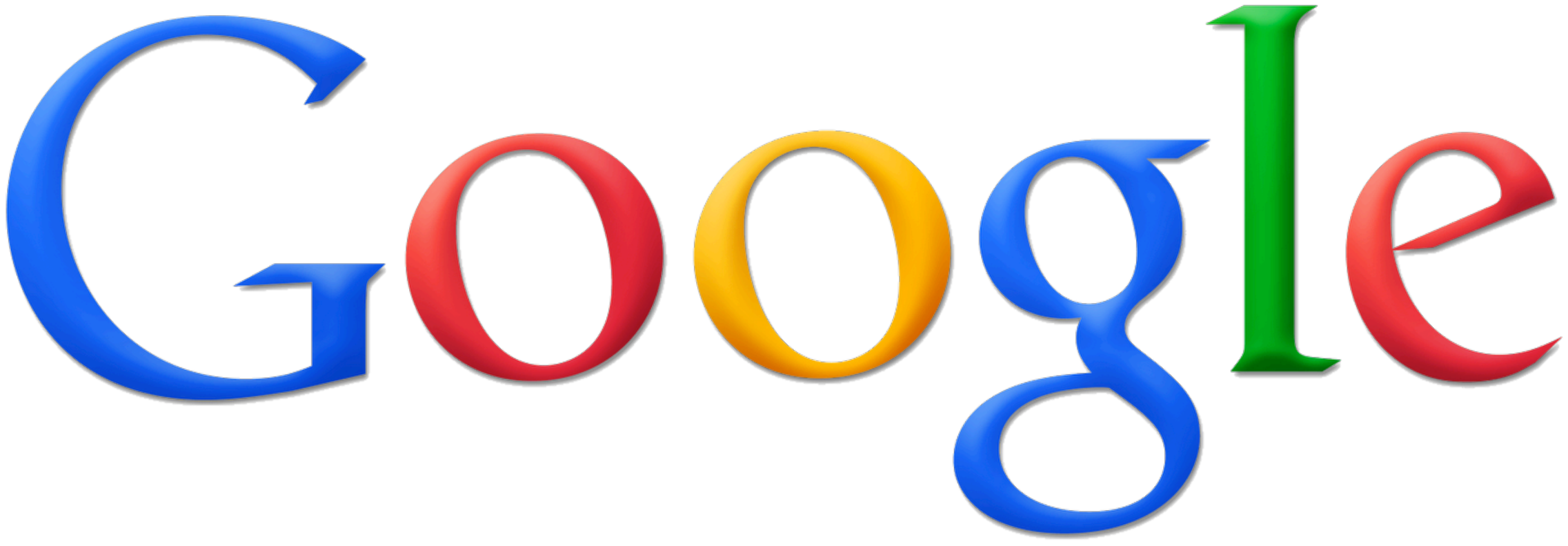


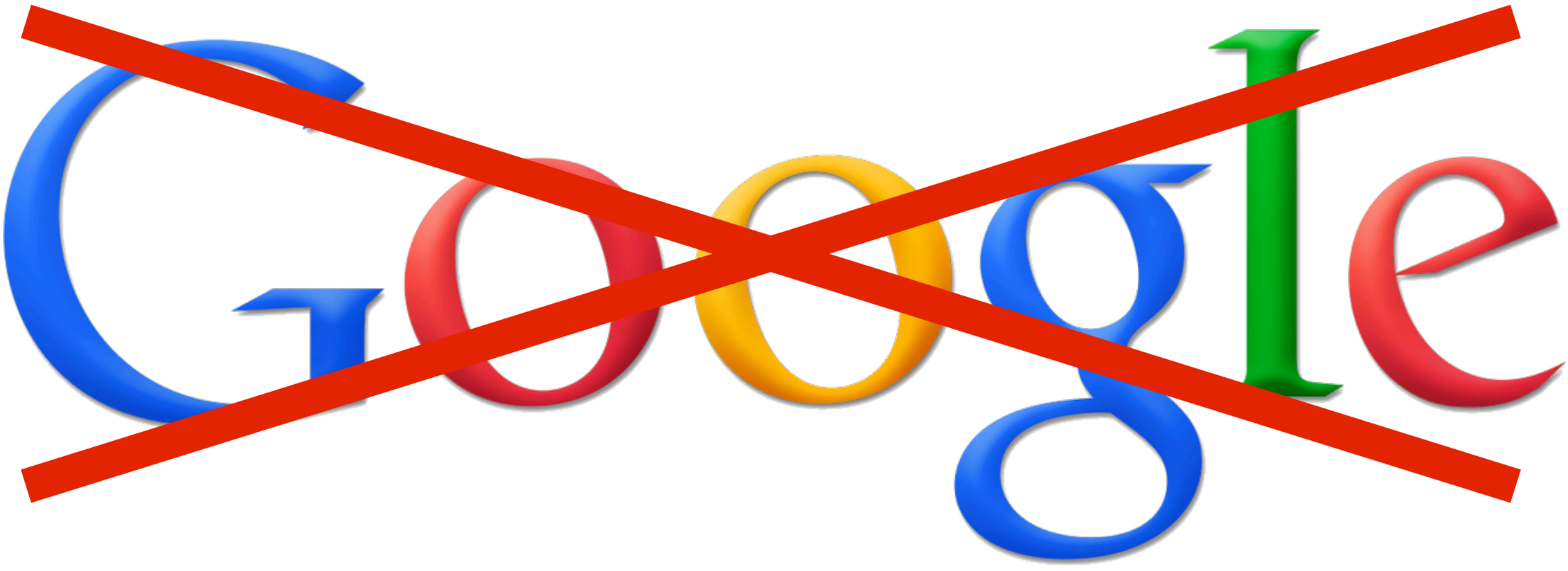
- Database administration
  - Indexing
  - Query optimization

- Identifying dependencies (in unknown data)
  - Life science data
  - Sensor data

Big Data


sugar phosphate
backbone
cytosine
deoxyribose
guanine

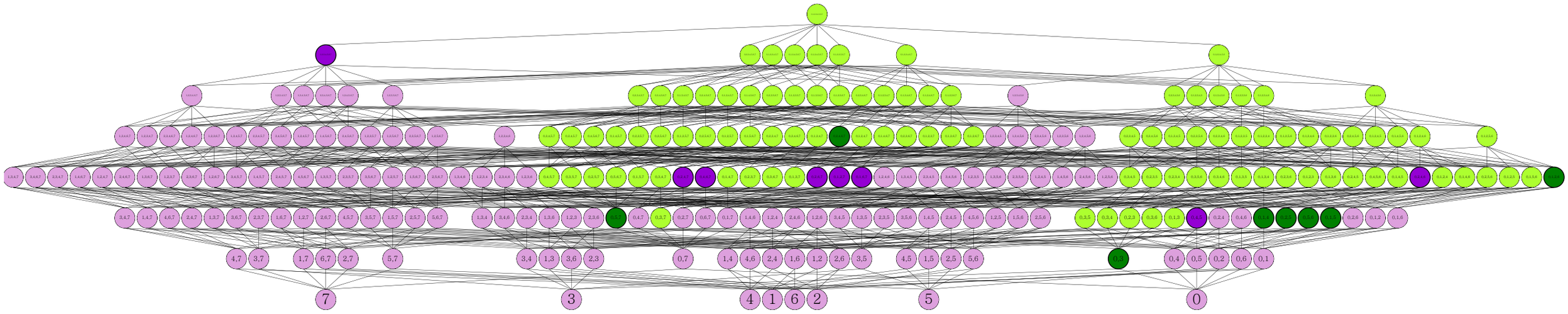

Google



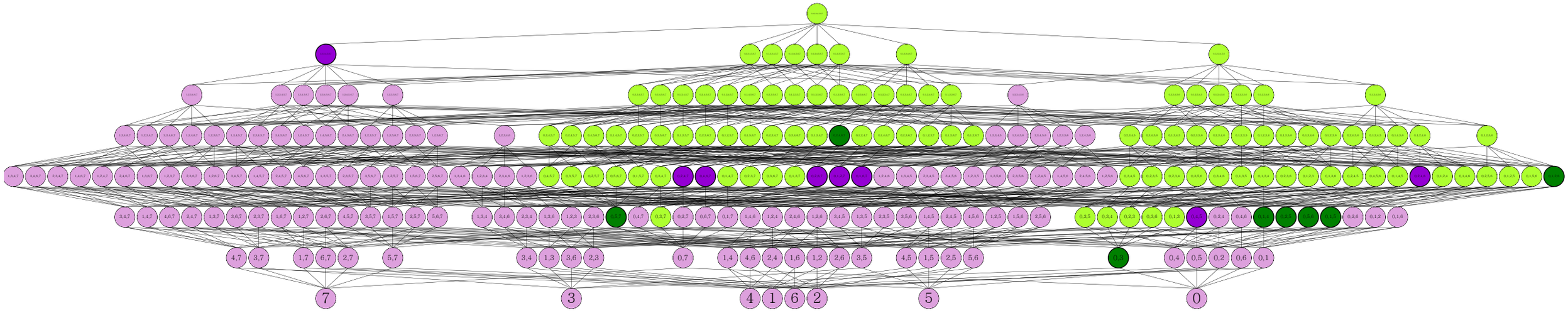
YAHOO!®

Problem?

Discovering (Non-)Uniques

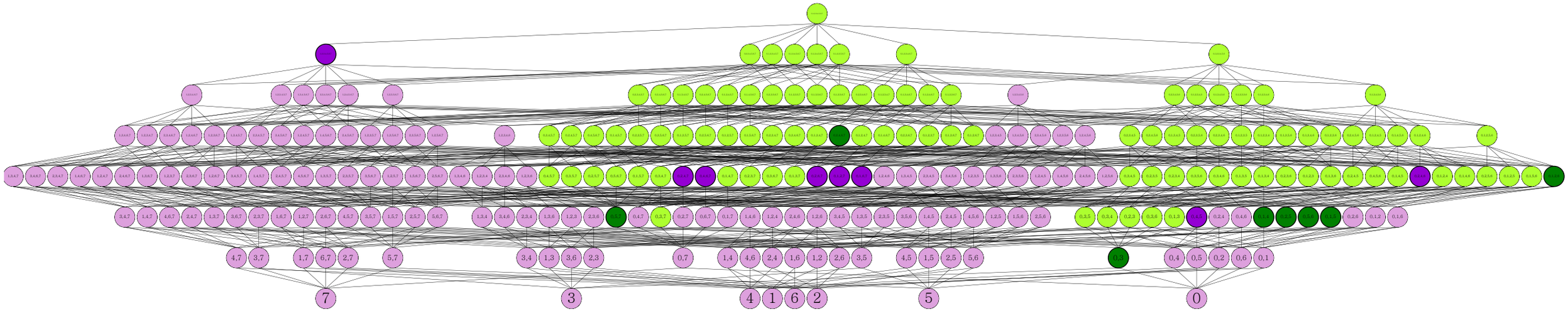


Discovering (Non-)Uniques



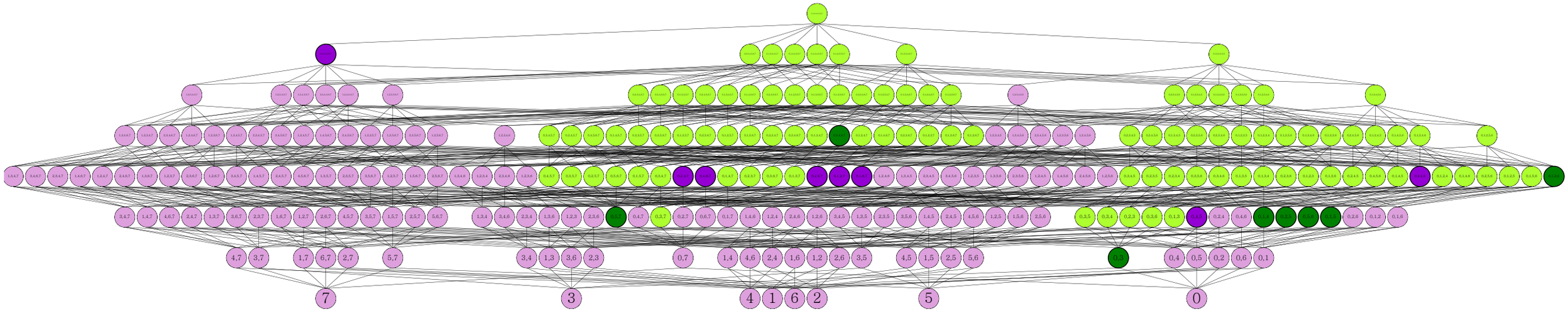
- 10 columns **1,023** combinations
- 50 columns **1,125,899,906,842,623** combinations

Discovering (Non-)Uniques



- 10 columns **1,023** combinations
- 50 columns **1,125,899,906,842,623** combinations
- Solution set might be **exponential**

Discovering (Non-)Uniques



- 10 columns **1,023** combinations
- 50 columns **1,125,899,906,842,623** combinations
- Solution set might be **exponential**

Discovery of all (non-)uniques is **NP-Hard**







NP-Hard!







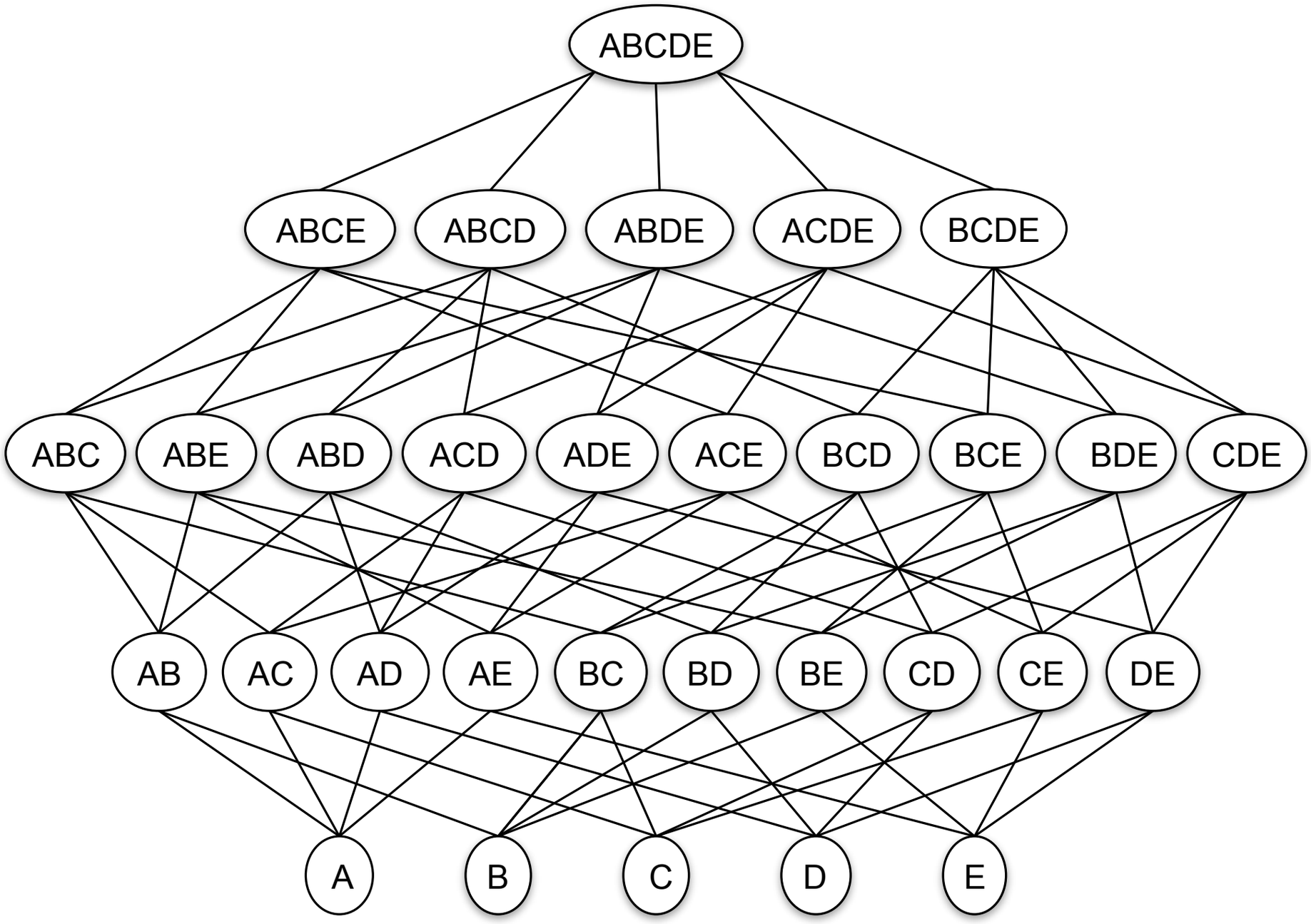




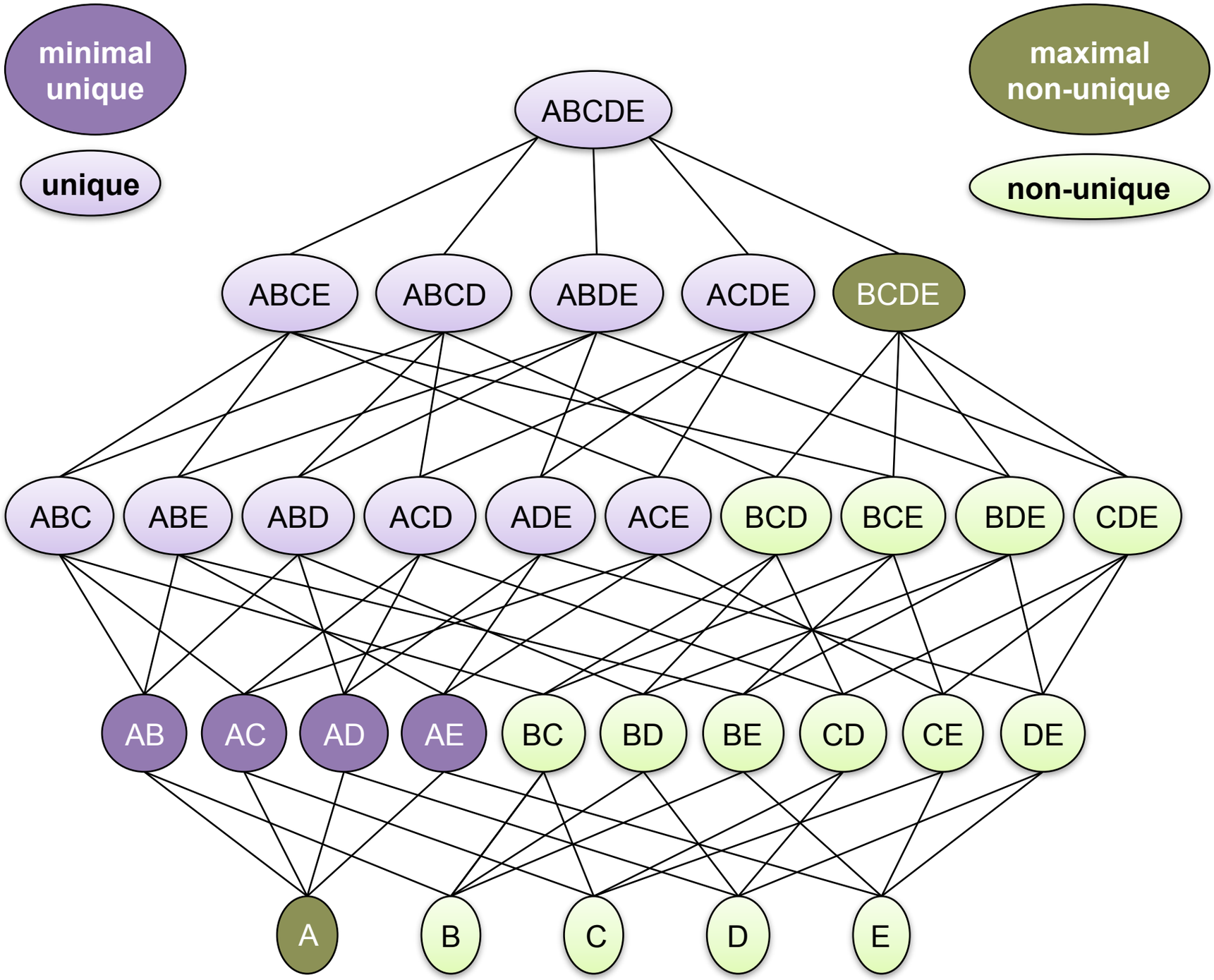
PUCC

Graph Coloring Problem

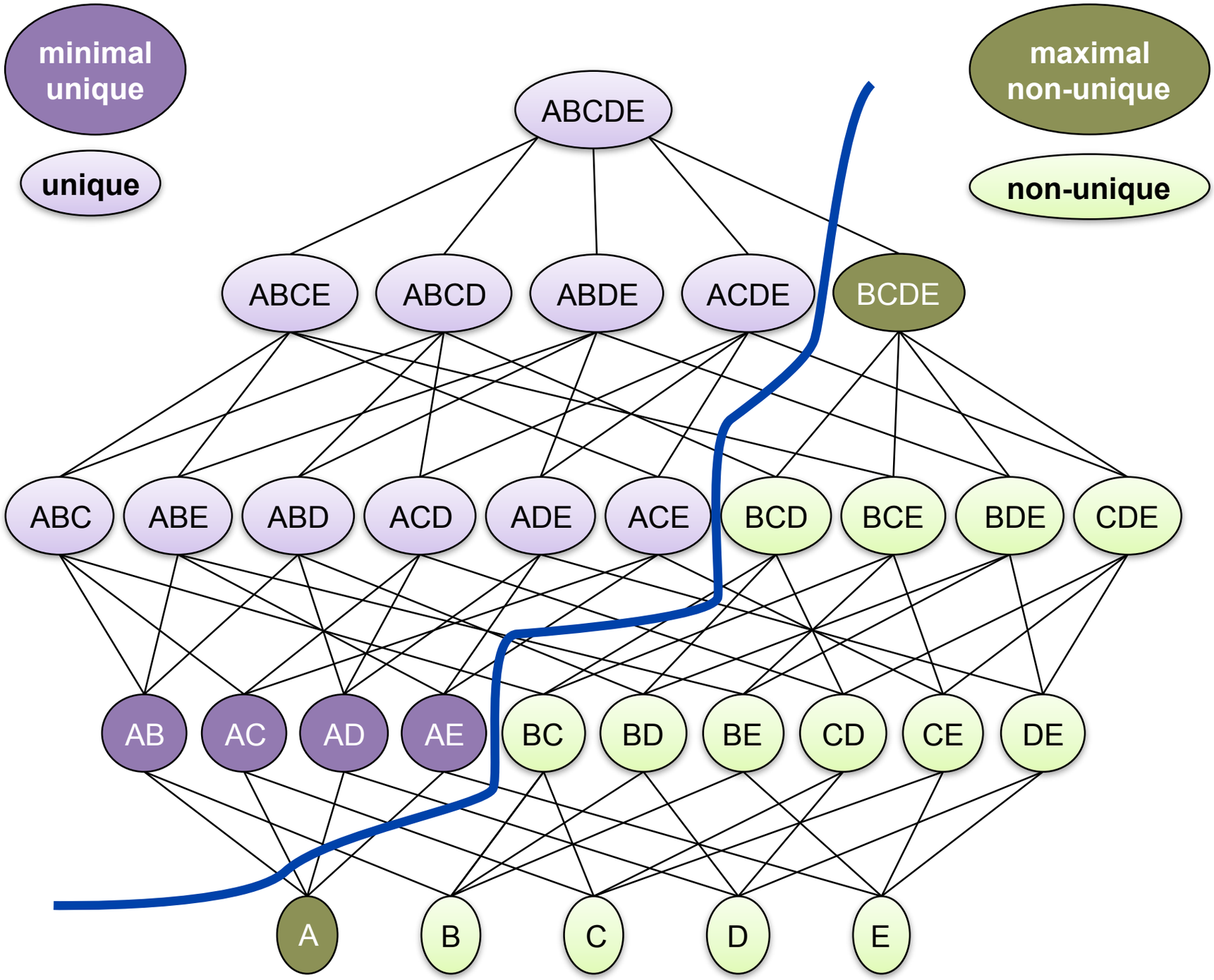
Identifying the Border Line



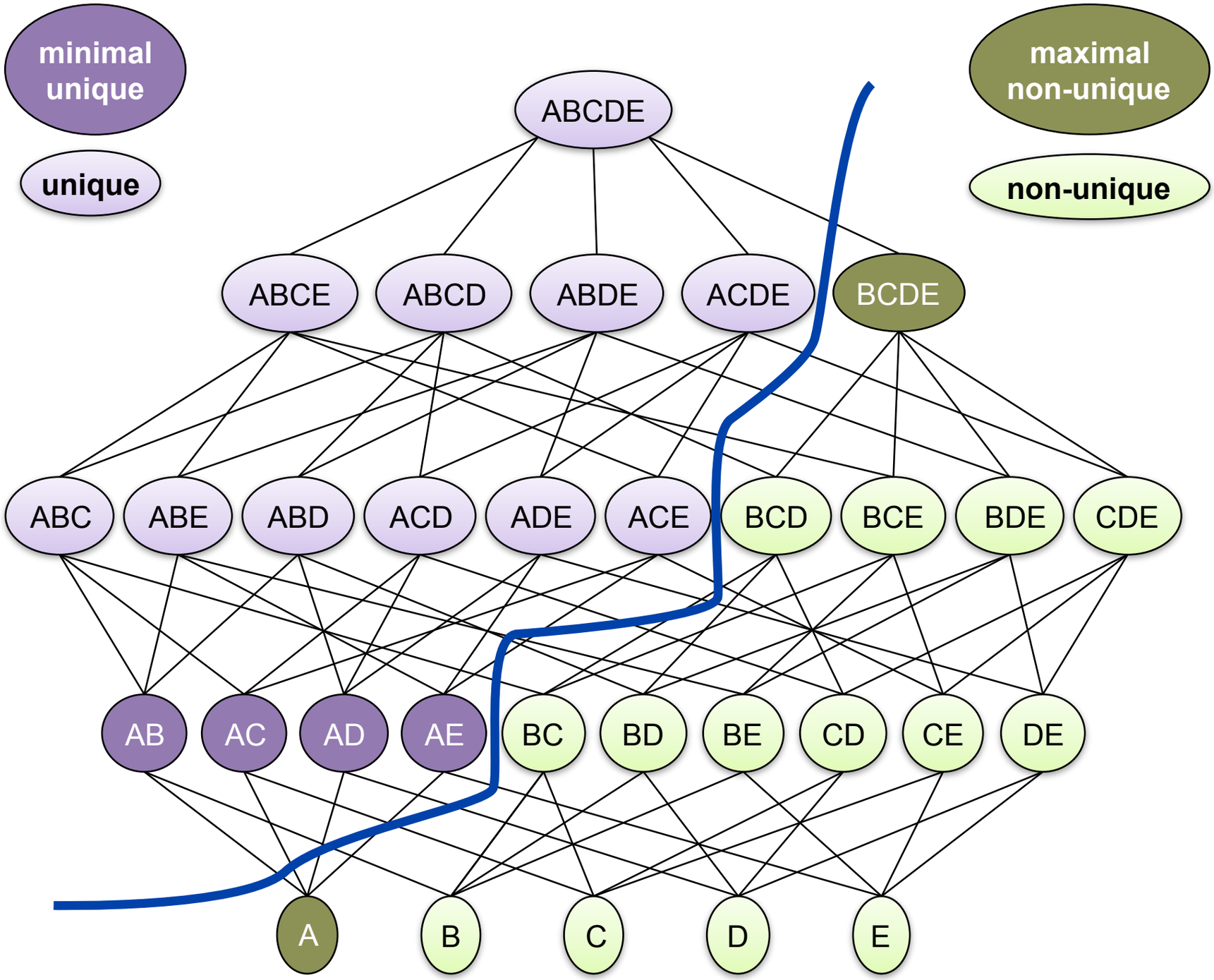
Identifying the Border Line



Identifying the Border Line

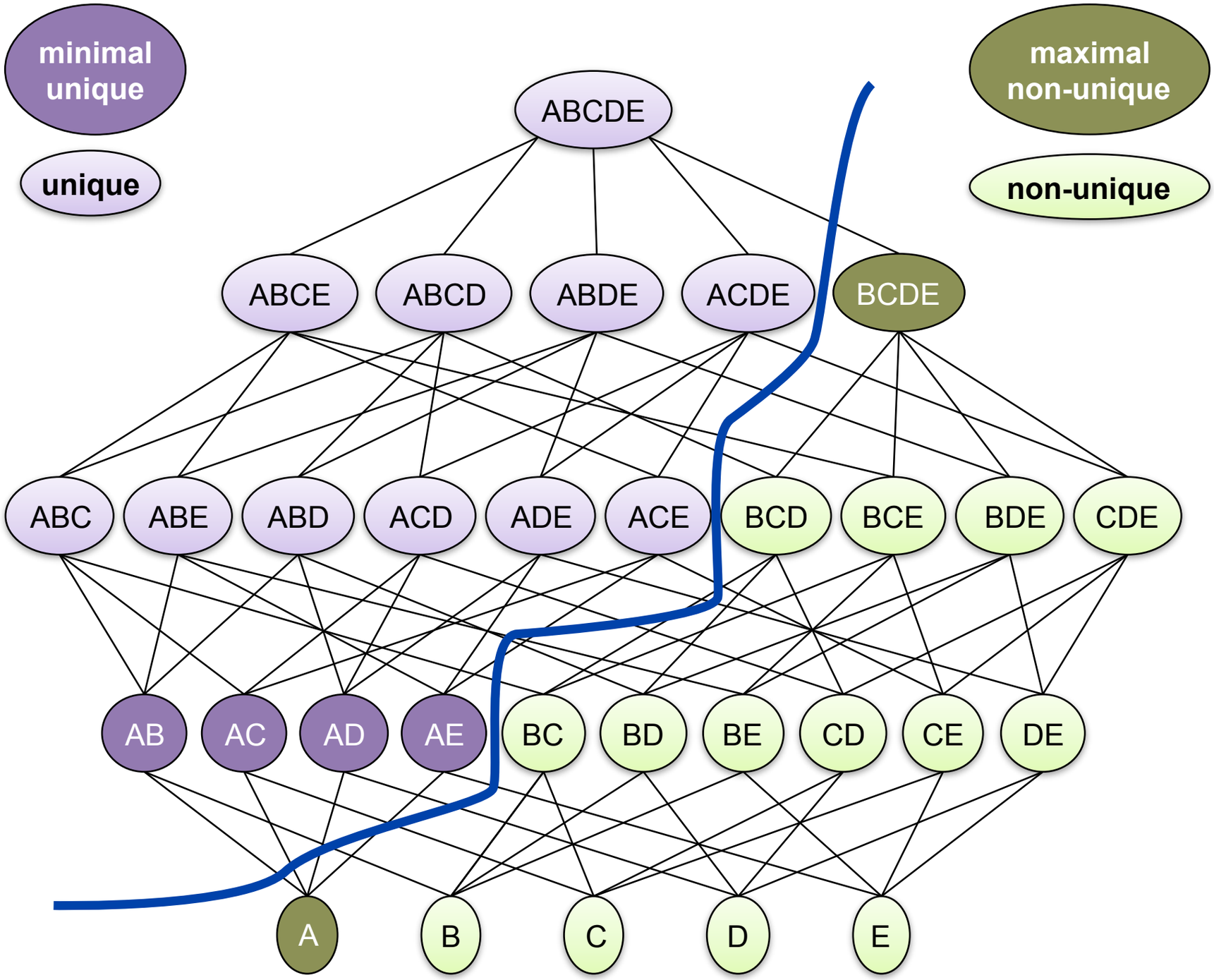


Identifying the Border Line



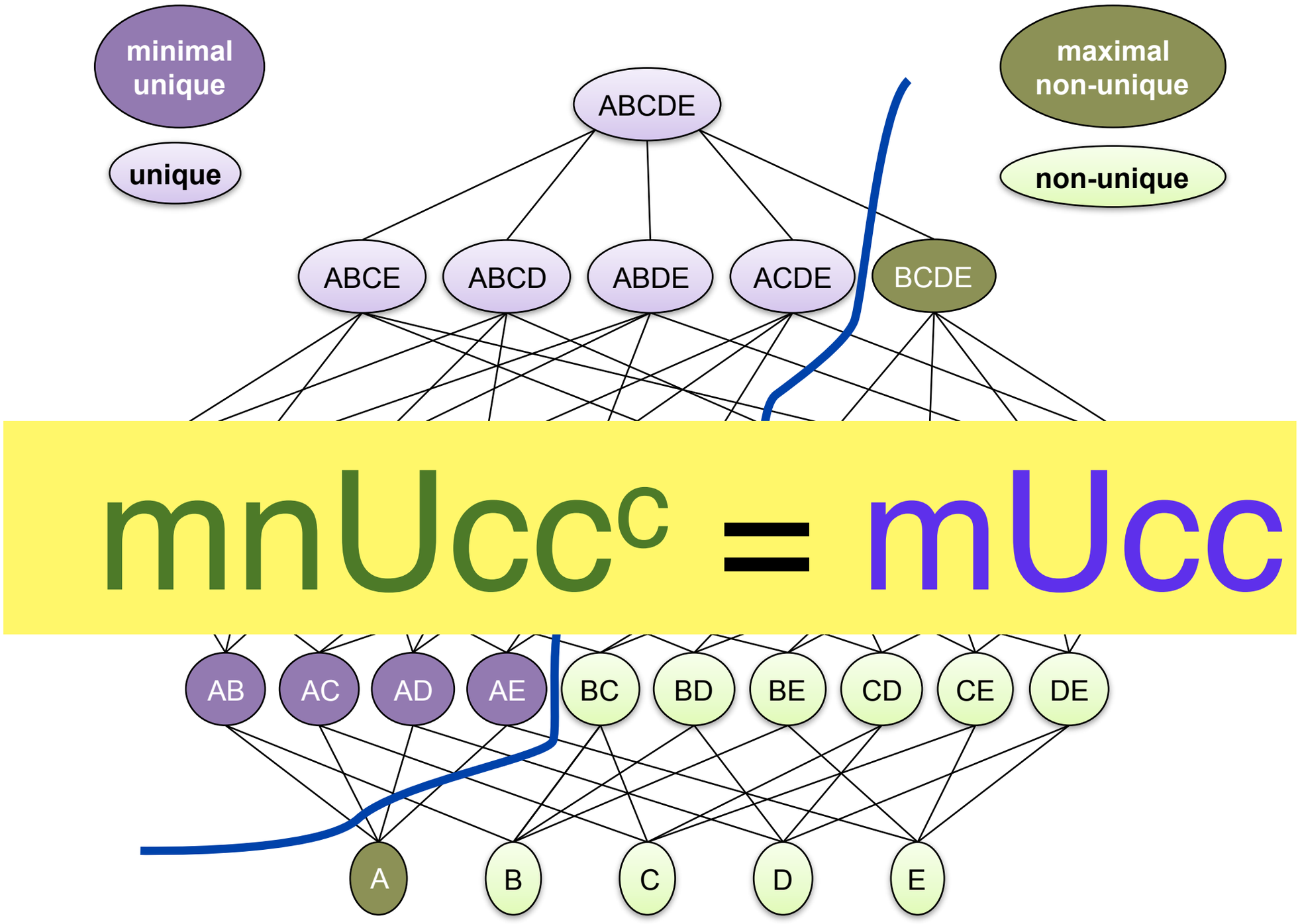
- Ducc finds **both** minimal uniques and maximal non-uniques

Identifying the Border Line



- Ducc finds **both** minimal uniques and maximal non-uniques
- Ducc then applies an **aggressive** pruning

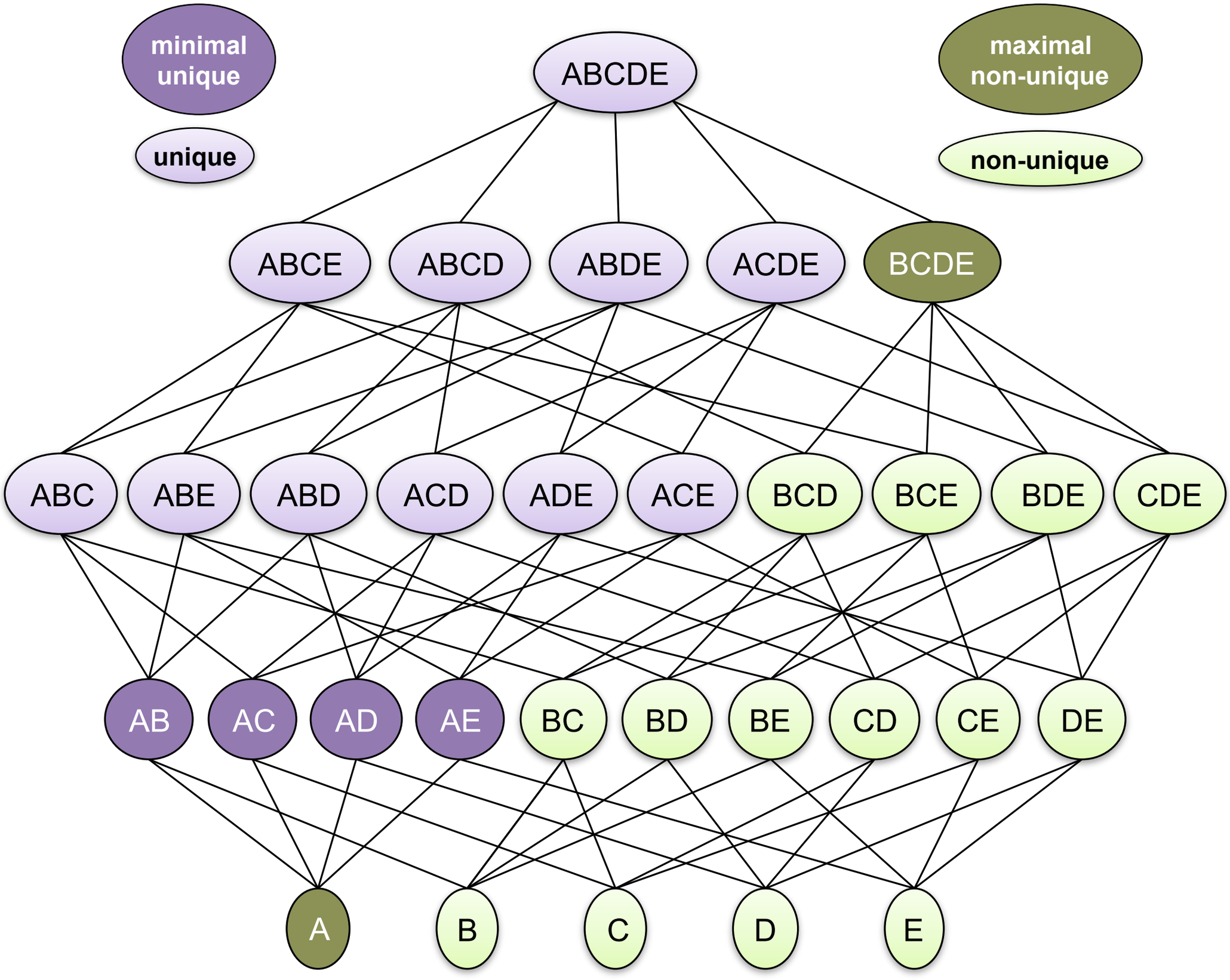
Identifying the Border Line



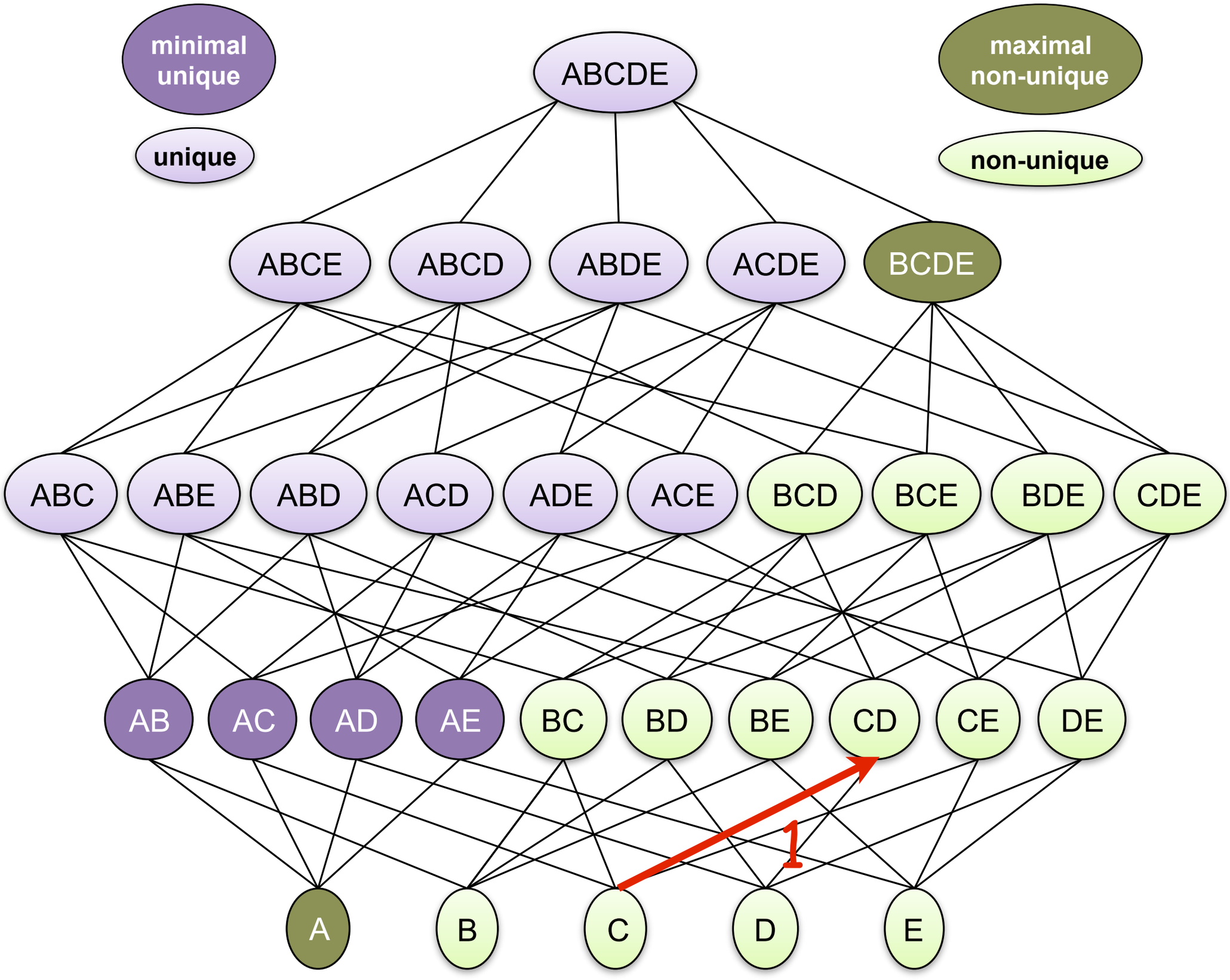
- Ducc finds **both** minimal uniques and maximal non-uniques
- Ducc then applies an **aggressive** pruning

Hybrid Graph Traversal

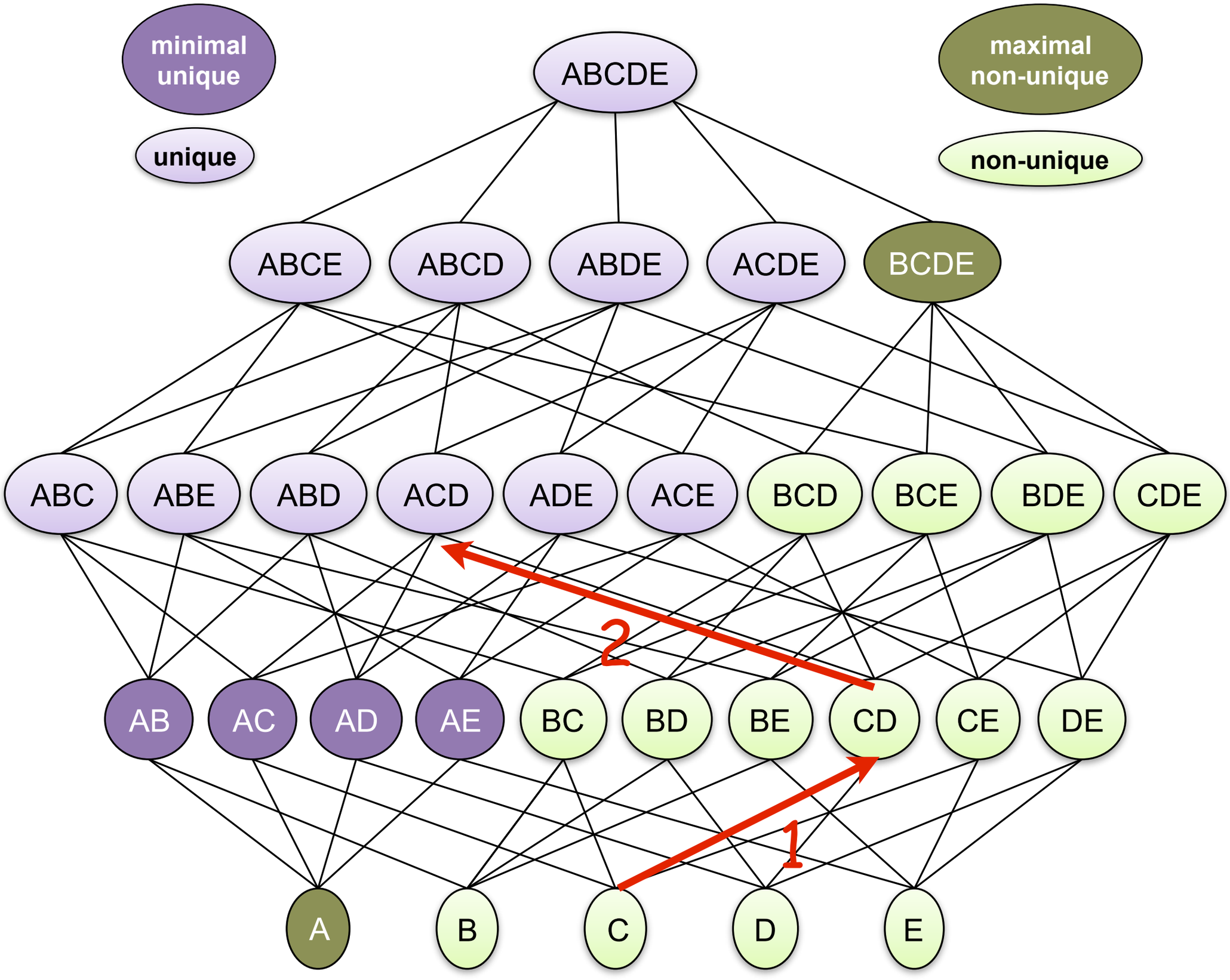
Depth-First + Random Walk



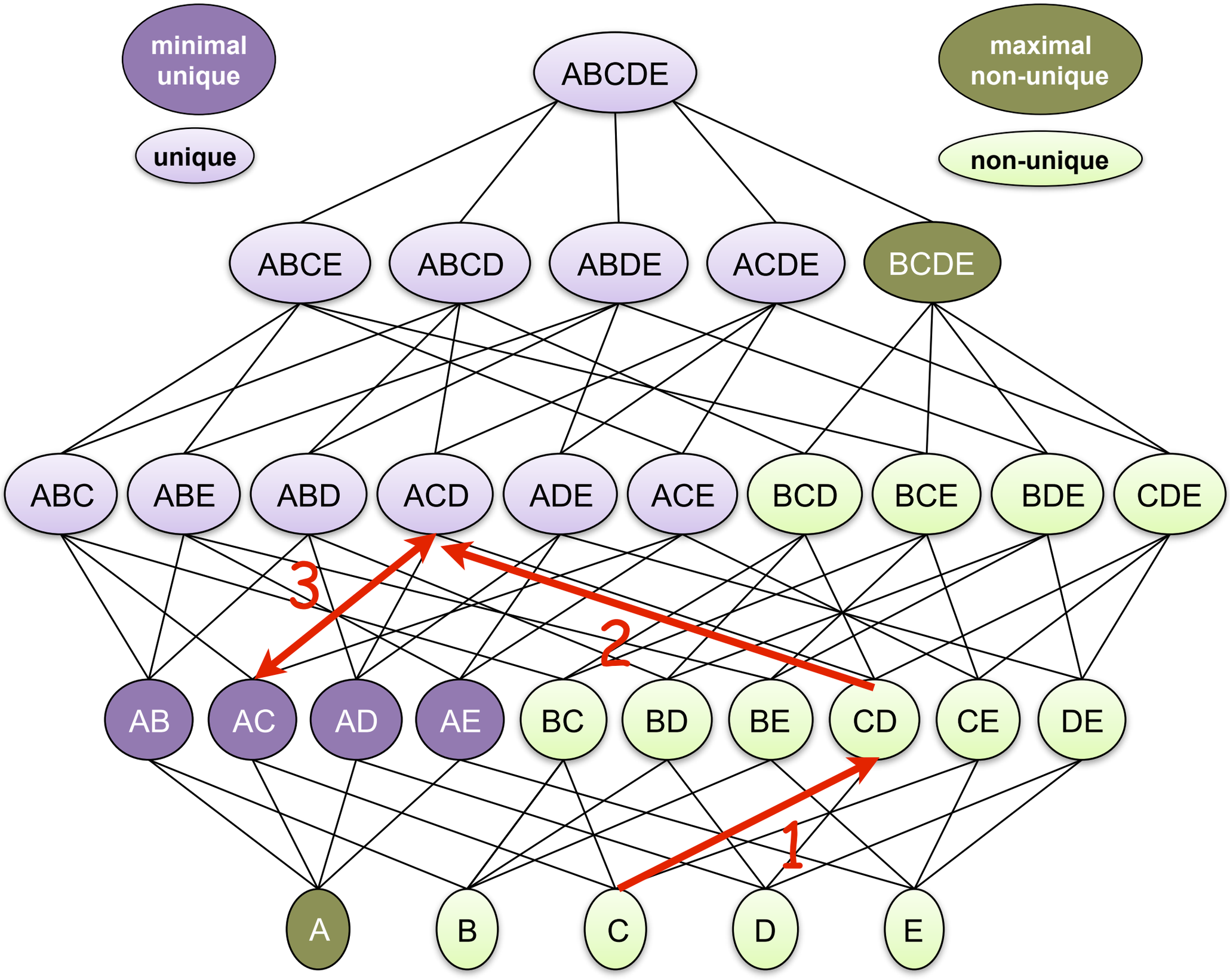
Depth-First + Random Walk



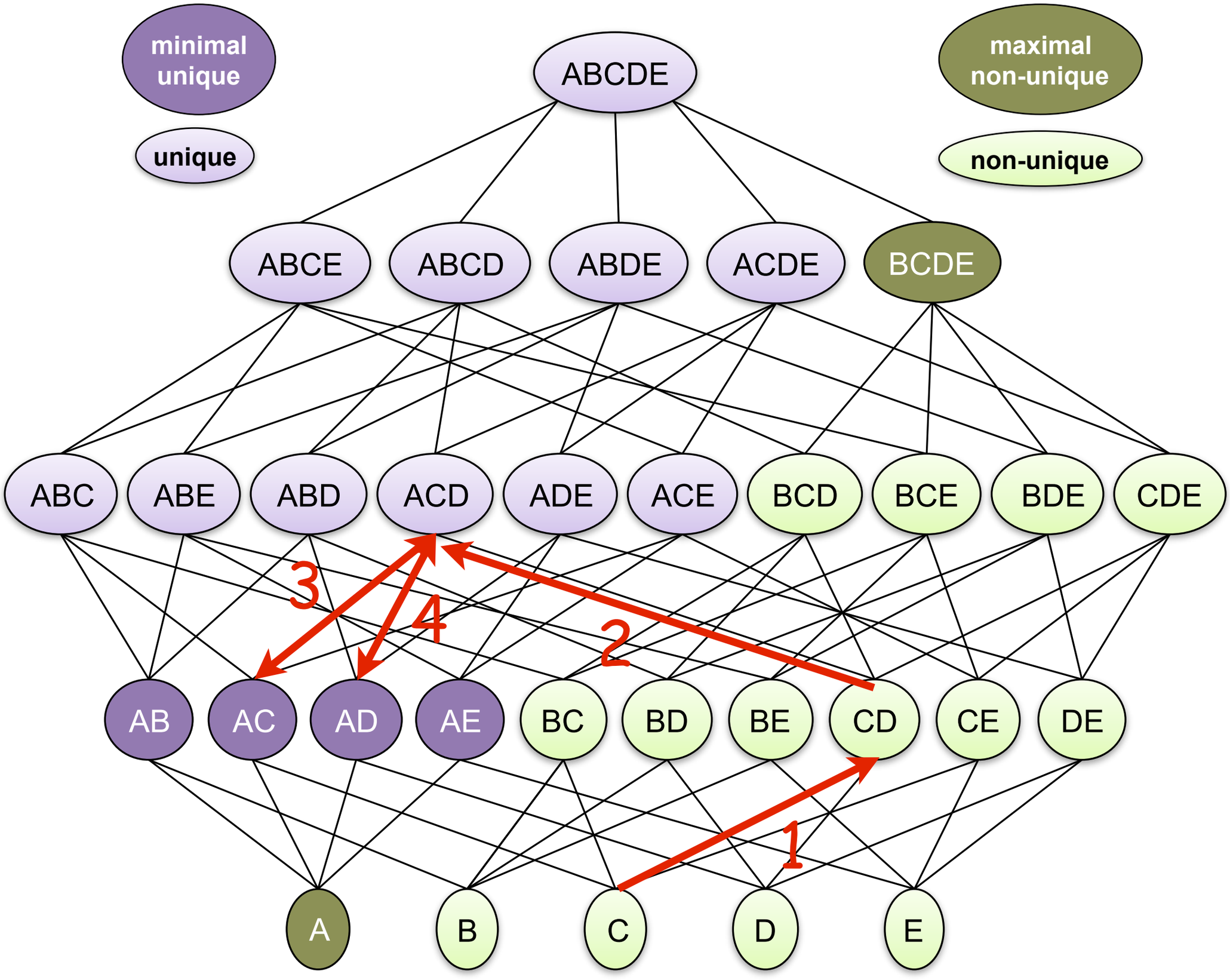
Depth-First + Random Walk



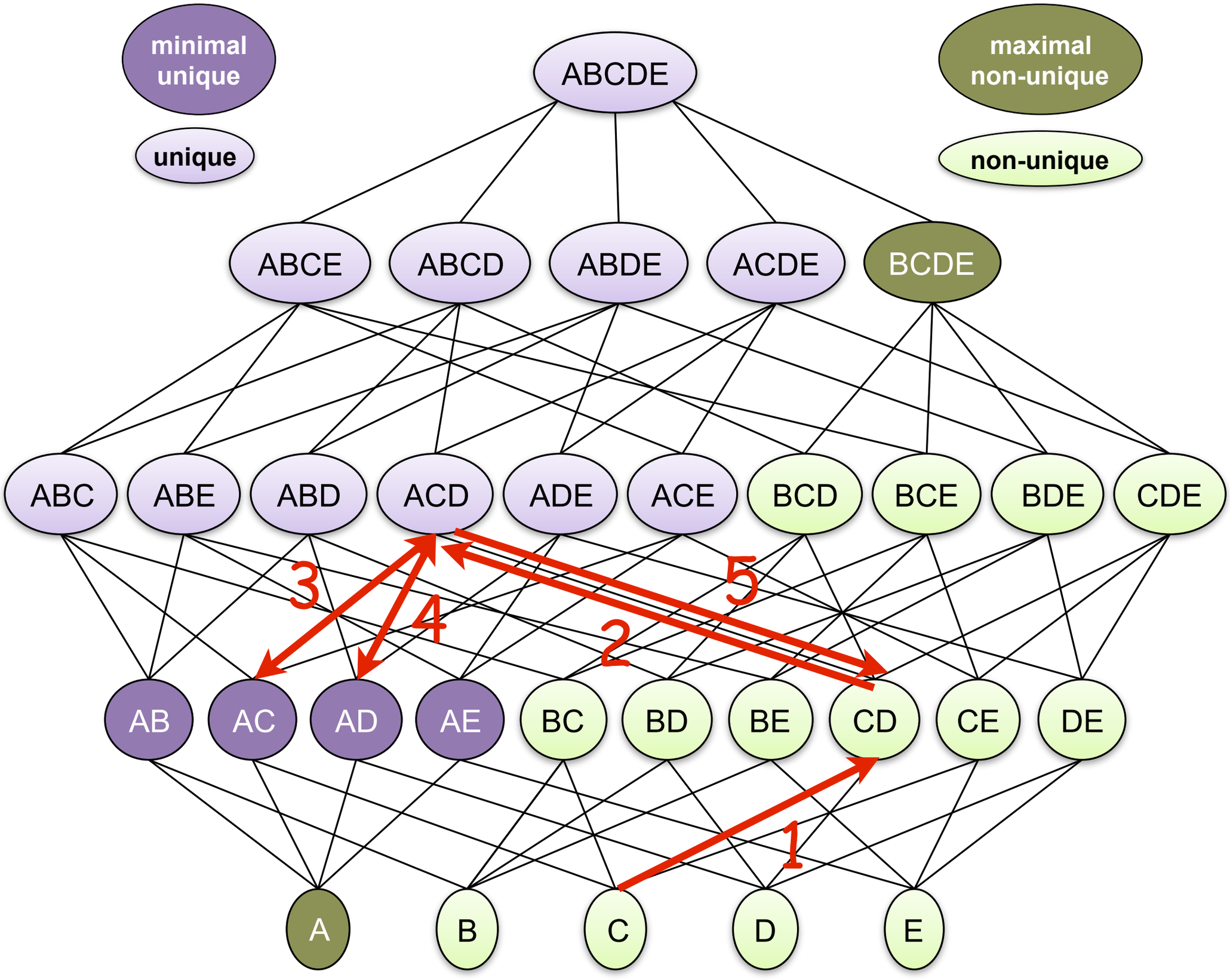
Depth-First + Random Walk



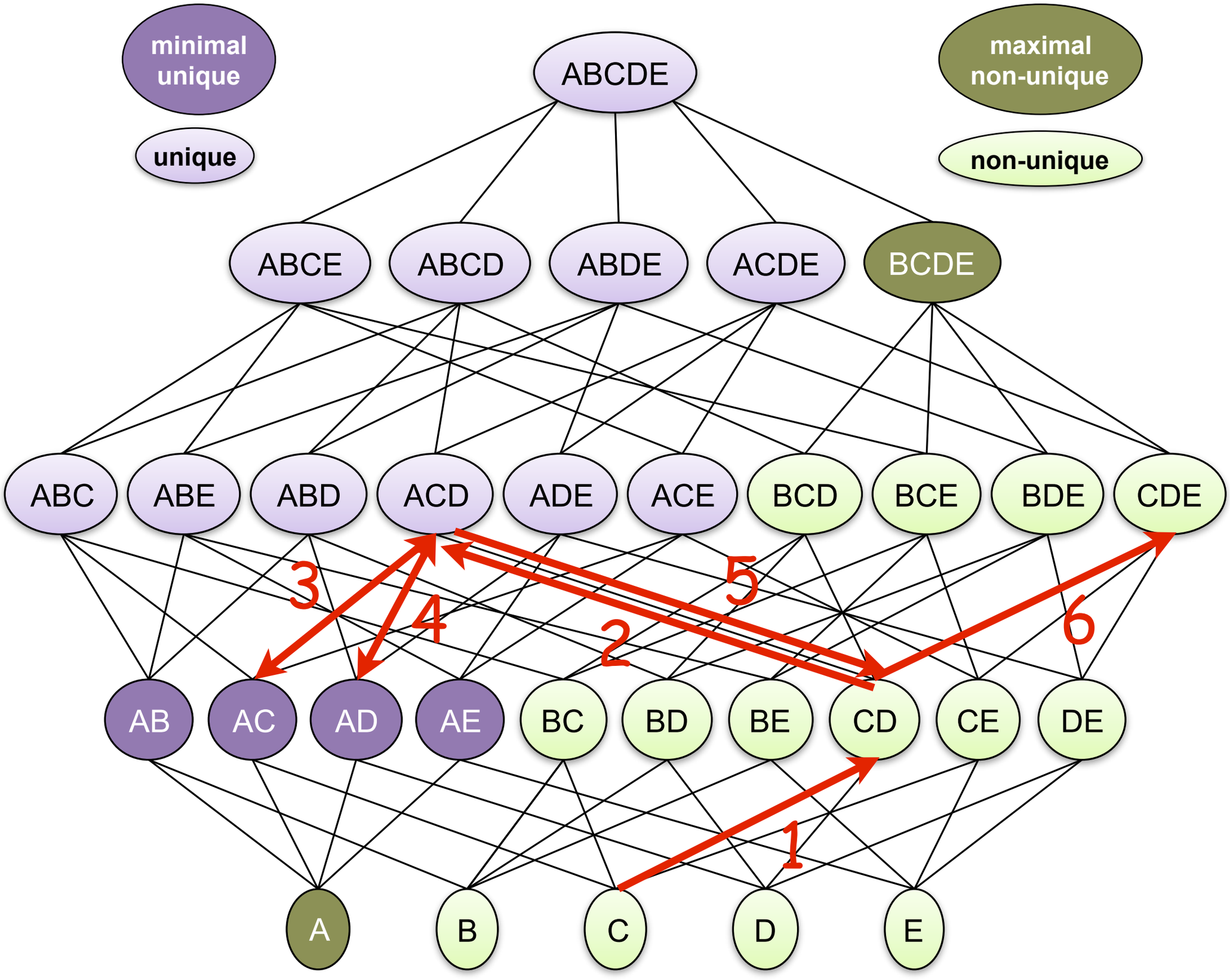
Depth-First + Random Walk



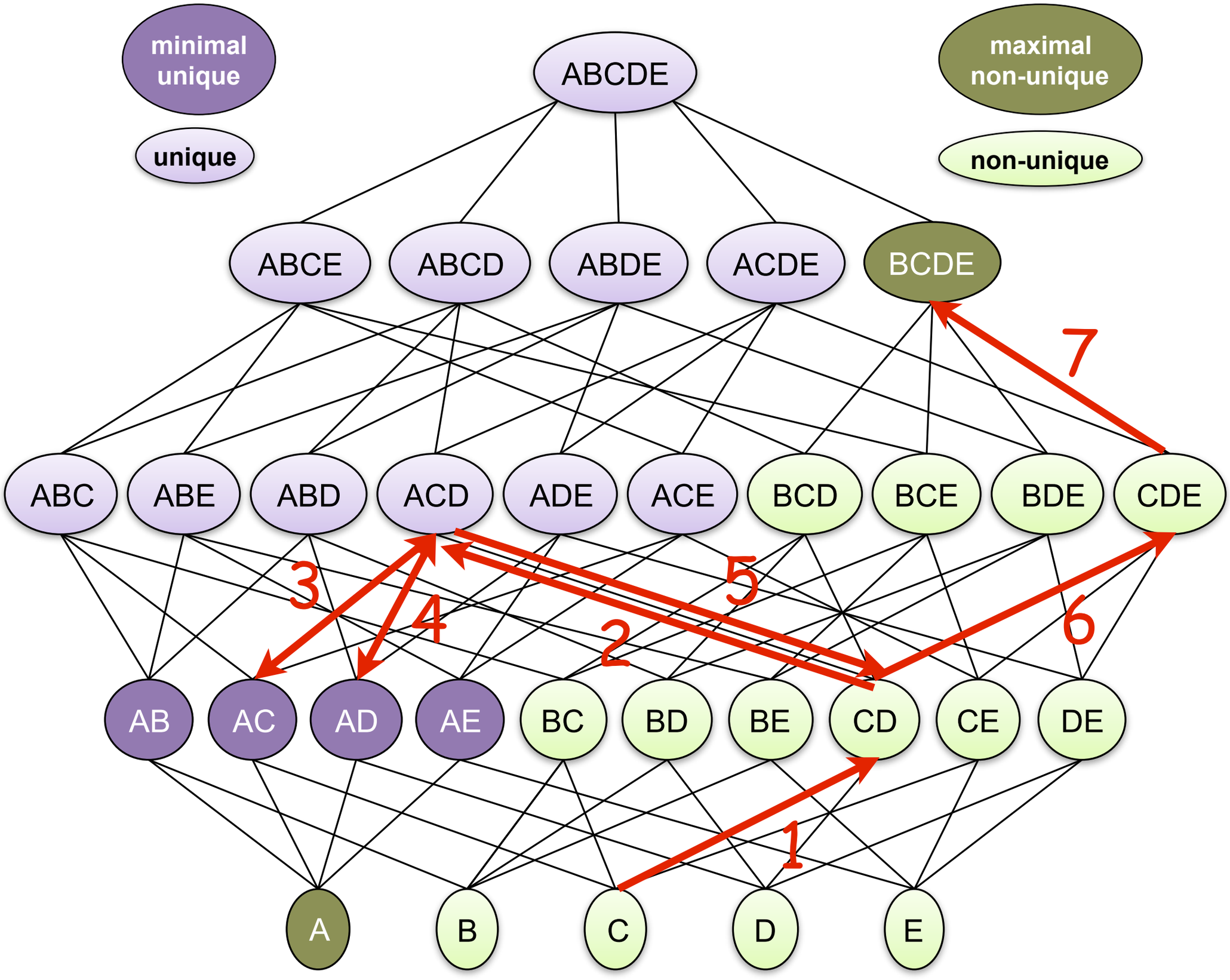
Depth-First + Random Walk



Depth-First + Random Walk



Depth-First + Random Walk



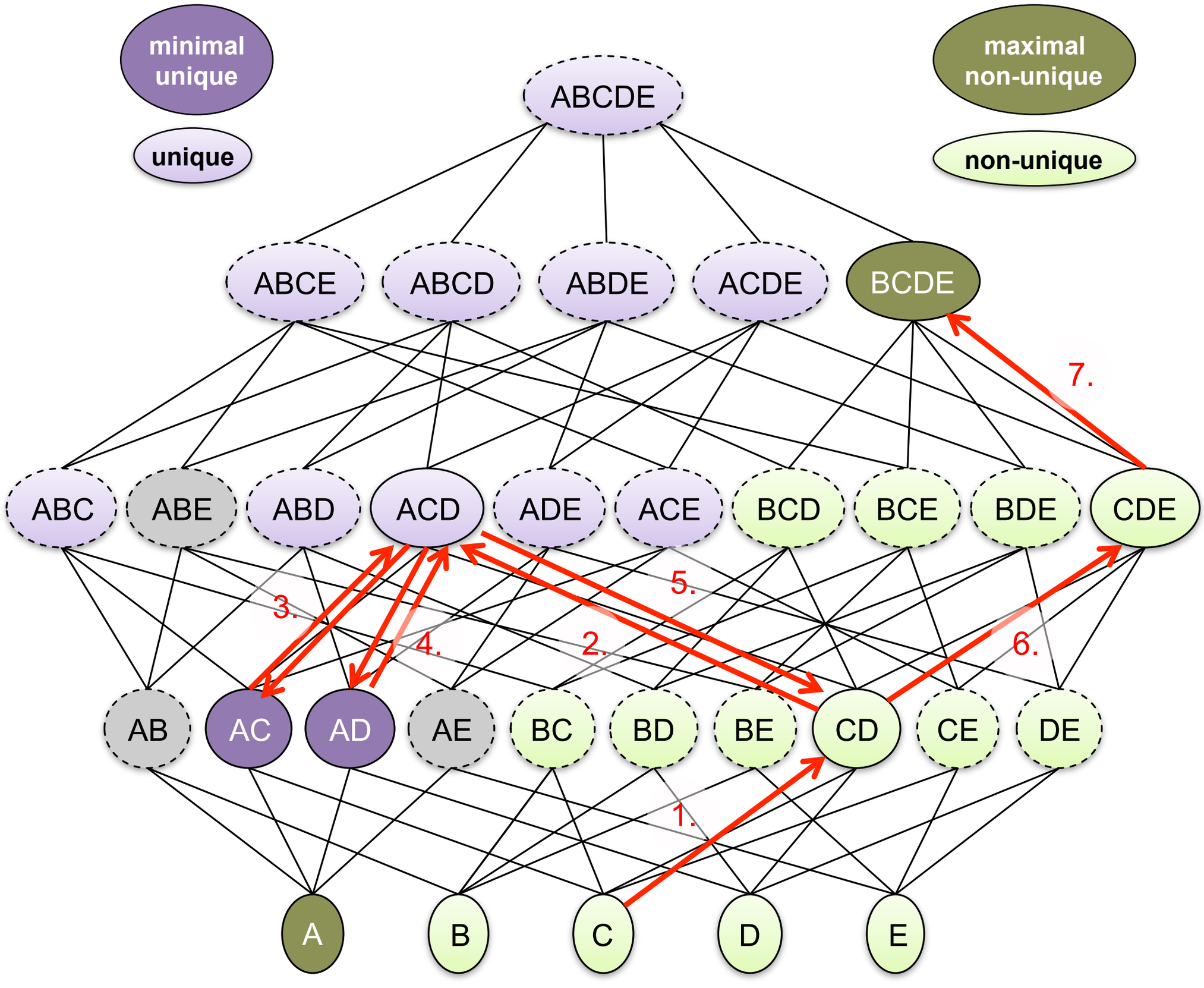

Ladies Fashion

Apply

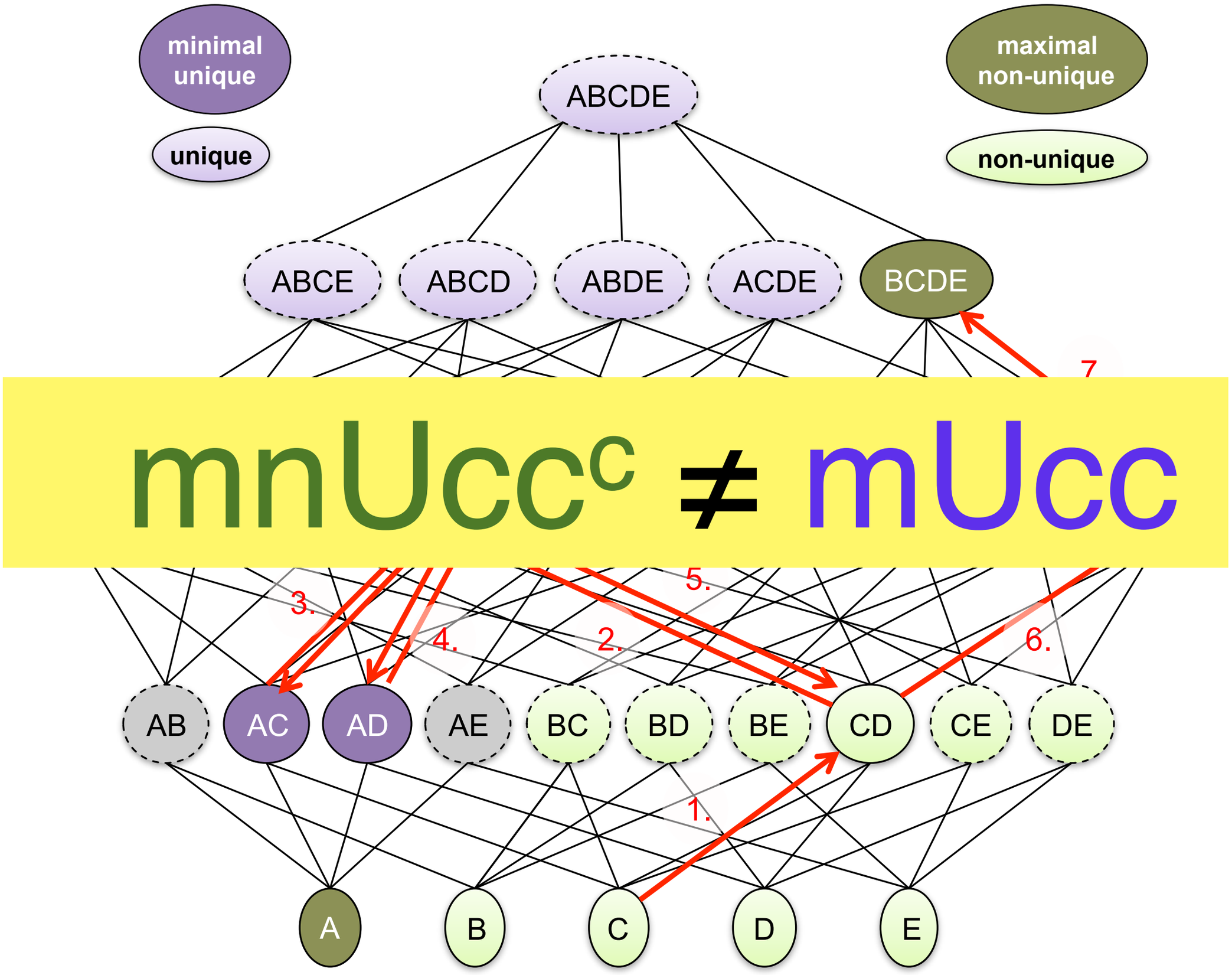
Random Walk

Graph Holes

Effects of Aggressive Pruning



Effects of Aggressive Pruning



Fast Checking

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...
- **First:** {{r1, r2}, {r3, r4}}
- **City:** {{r1, r2, r4}}
- ...

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...
- **First:** {{r1, r2}, {r3, r4}} = $\{A_1, A_2\}$
- **City:** {{r1, r2, r4}} = $\{B_1\}$
- ...

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...
- **First:** {{r1, r2}, {r3, r4}} = $\{A_1, A_2\}$
- **City:** {{r1, r2, r4}} = $\{B_1\}$
- ...

build (First)

$\{r1, A_1\}$
$\{r2, A_1\}$
$\{r3, A_2\}$
$\{r4, A_2\}$

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...
- **First:** {{r1, r2}, {r3, r4}} = $\{A_1, A_2\}$
- **City:** {{r1, r2, r4}} = $\{B_1\}$
- ...

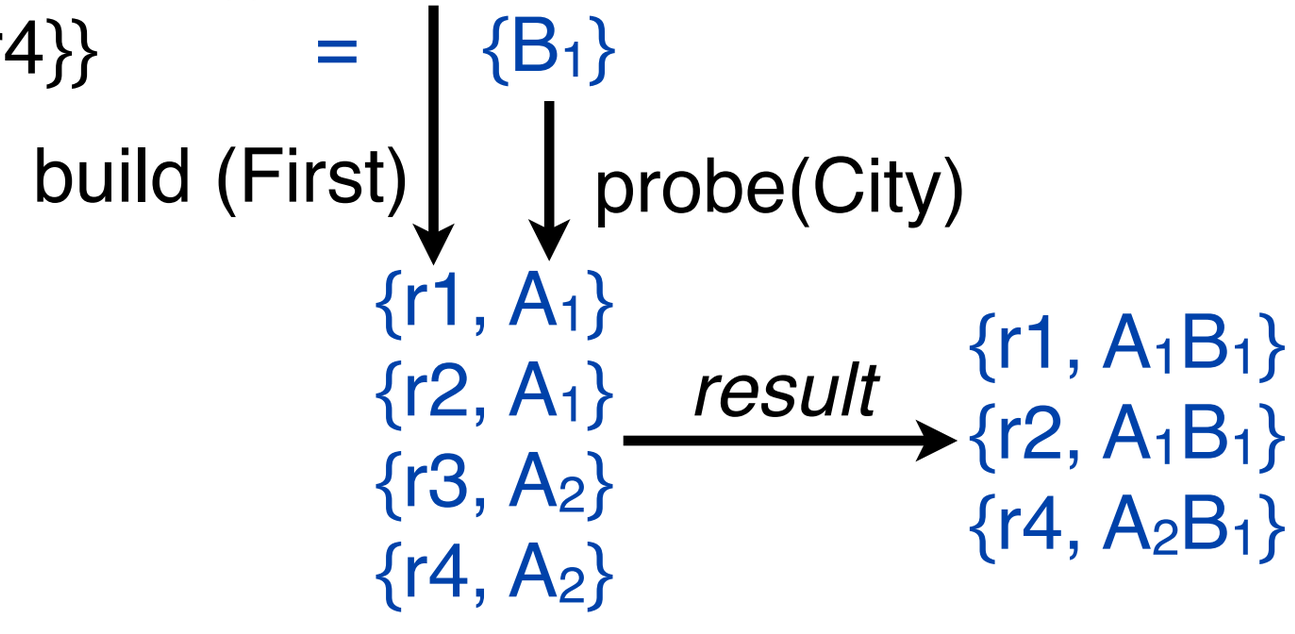
build (First) probe(City)

$\{r1, A_1\}$
$\{r2, A_1\}$
$\{r3, A_2\}$
$\{r4, A_2\}$

Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

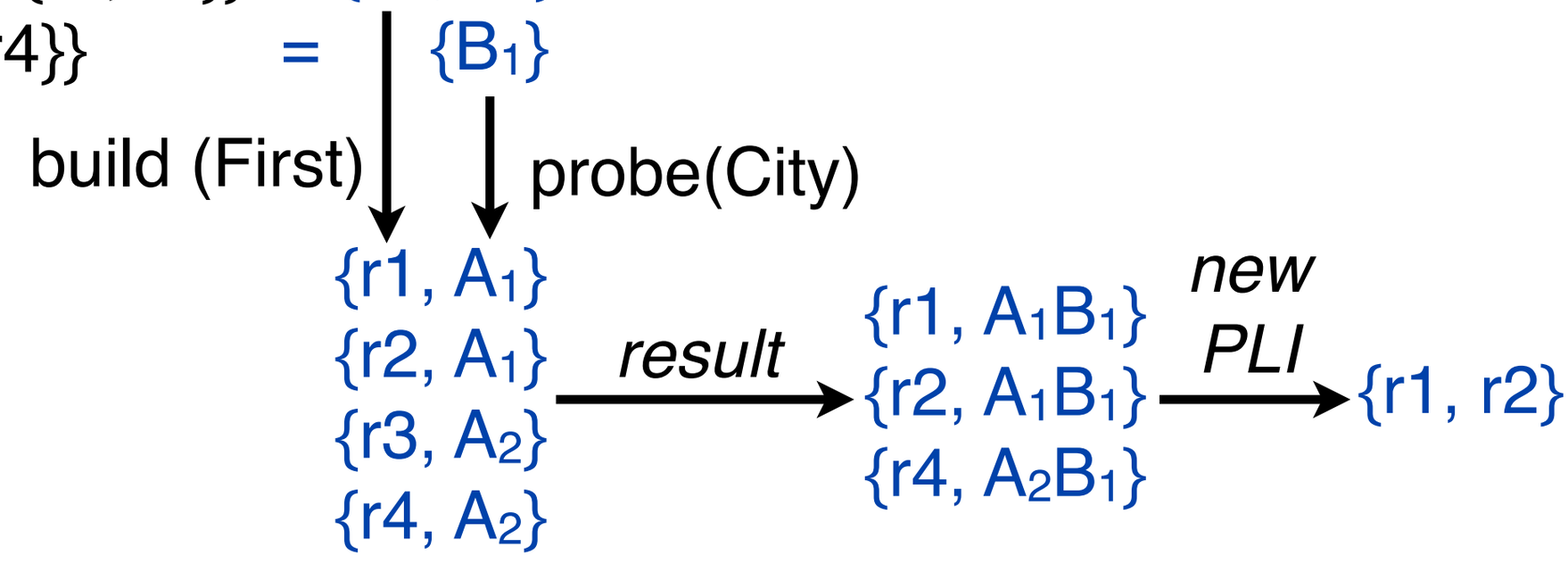
- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...
- **First:** {{r1, r2}, {r3, r4}} = $\{A_1, A_2\}$
- **City:** {{r1, r2, r4}} = $\{B_1\}$
- ...



Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

- **First:** {Al -> {r1, r2}, Peggy -> {r3, r4}}
- **City:** {Chicago -> {r1, r2, r4}}
- ...
- **First:** {{r1, r2}, {r3, r4}} = $\{A_1, A_2\}$
- **City:** {{r1, r2, r4}} = $\{B_1\}$
- ...

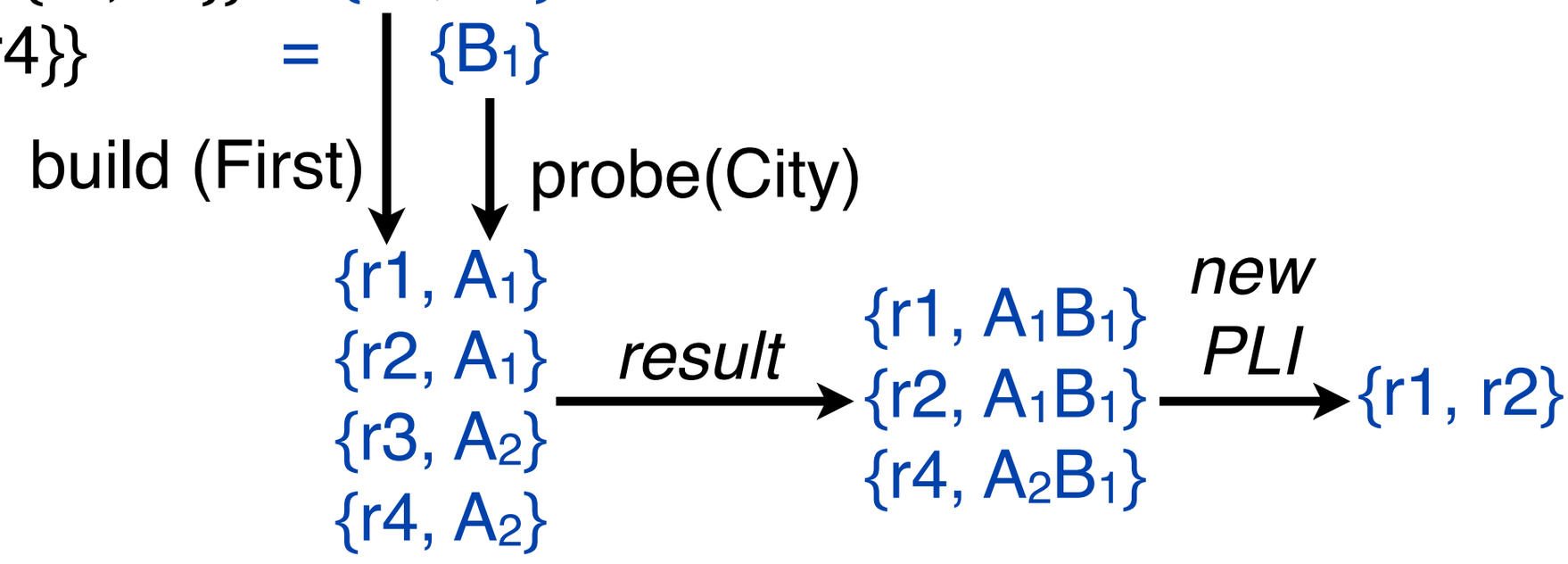


Position List Index

First (A)	Last (B)	Age (C)	Phone (D)	City (E)
Al	Bundy	20	3333	Chicago
Al	Power	21	1234	Chicago
Peggy	Power	21	5555	Hamburg
Peggy	Bundy	20	1010	Chicago

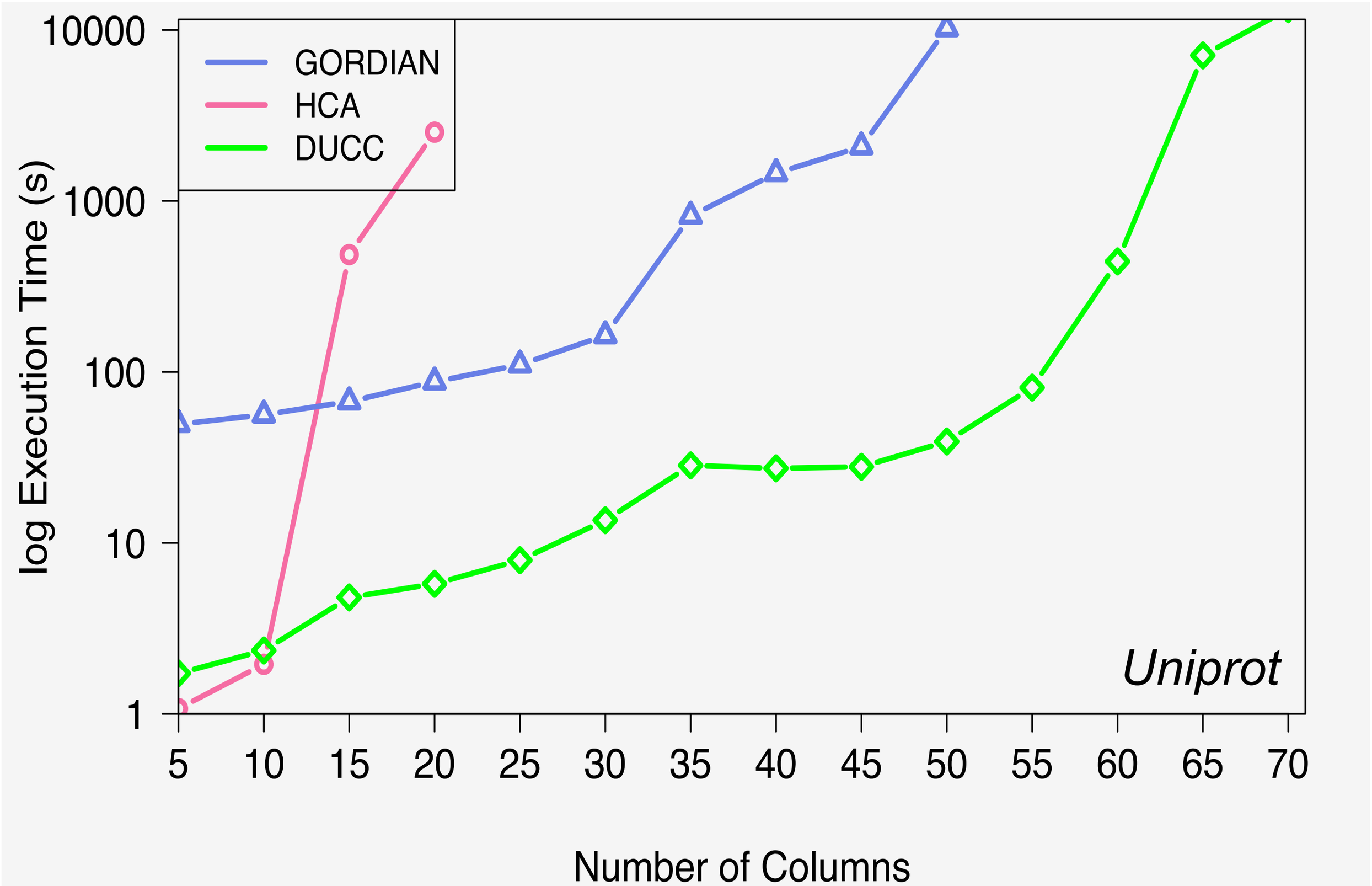
checks in < 1ms

- **First:** {{r1, r2}, {r3, r4}} = $\{A_1, A_2\}$
- **City:** {{r1, r2, r4}} = $\{B_1\}$
- ...

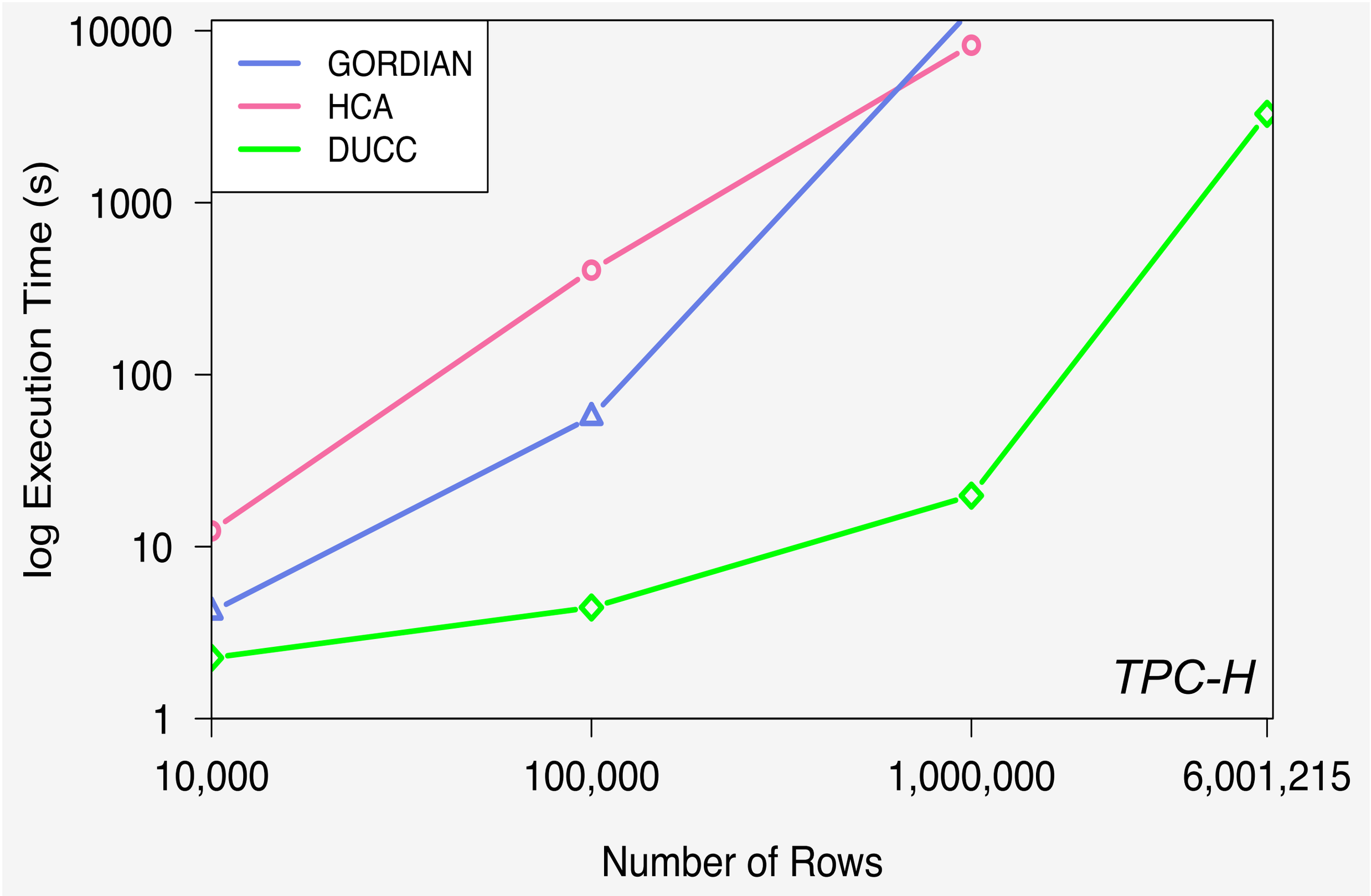


Results

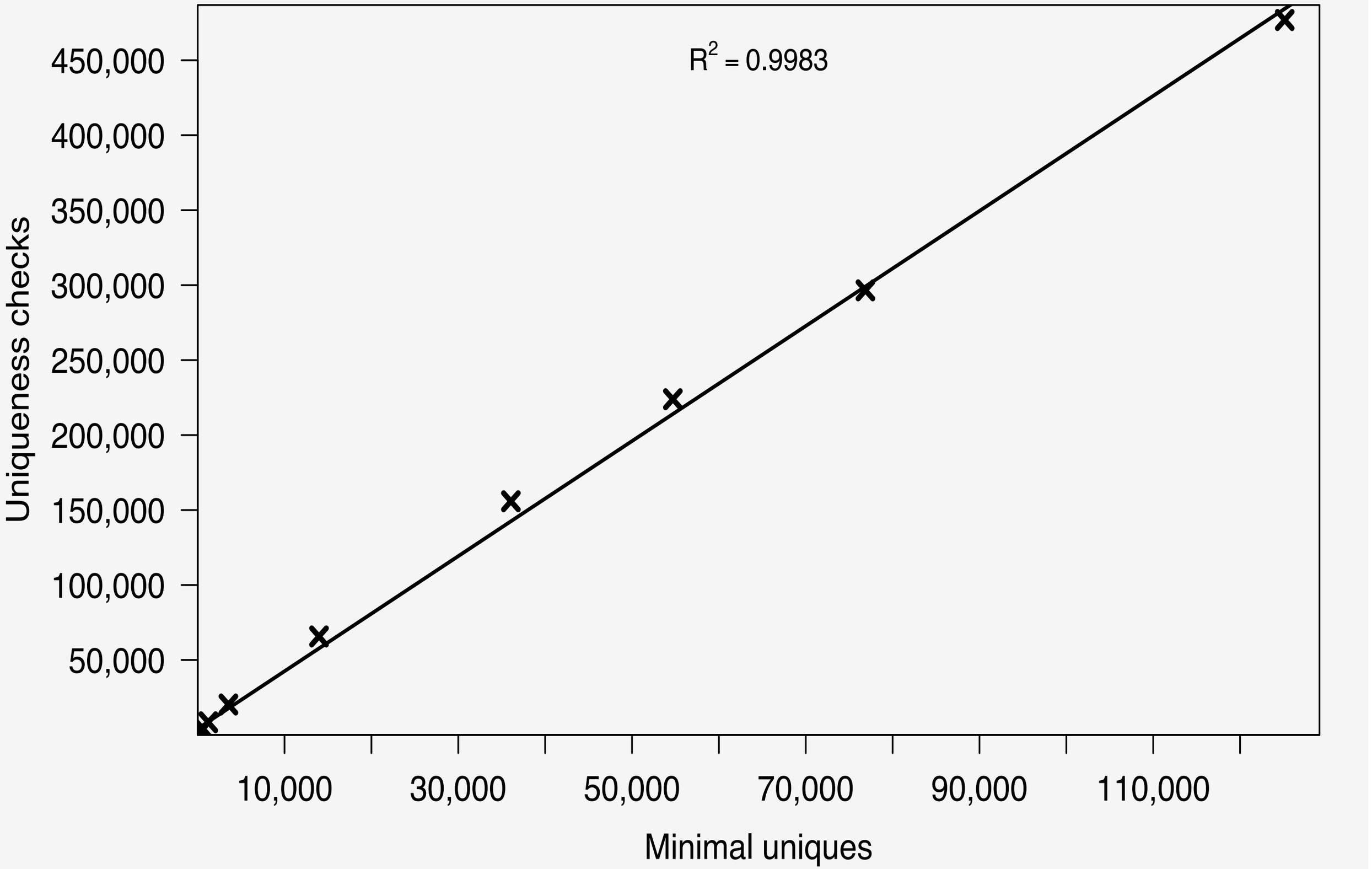
Scaling the Number of Columns



Scaling the Number of Rows (w. 15 cols)

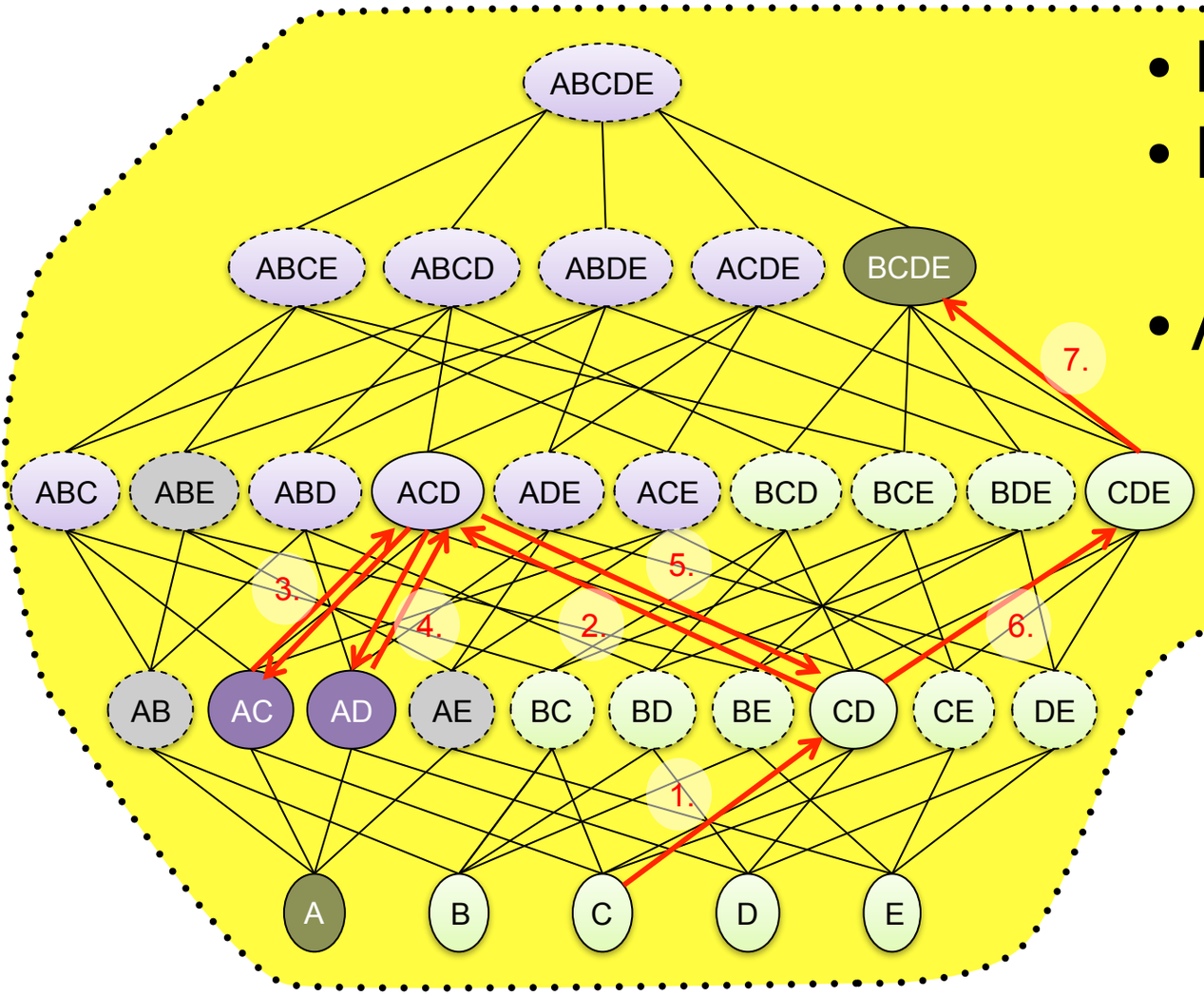


Solution Space and Checks



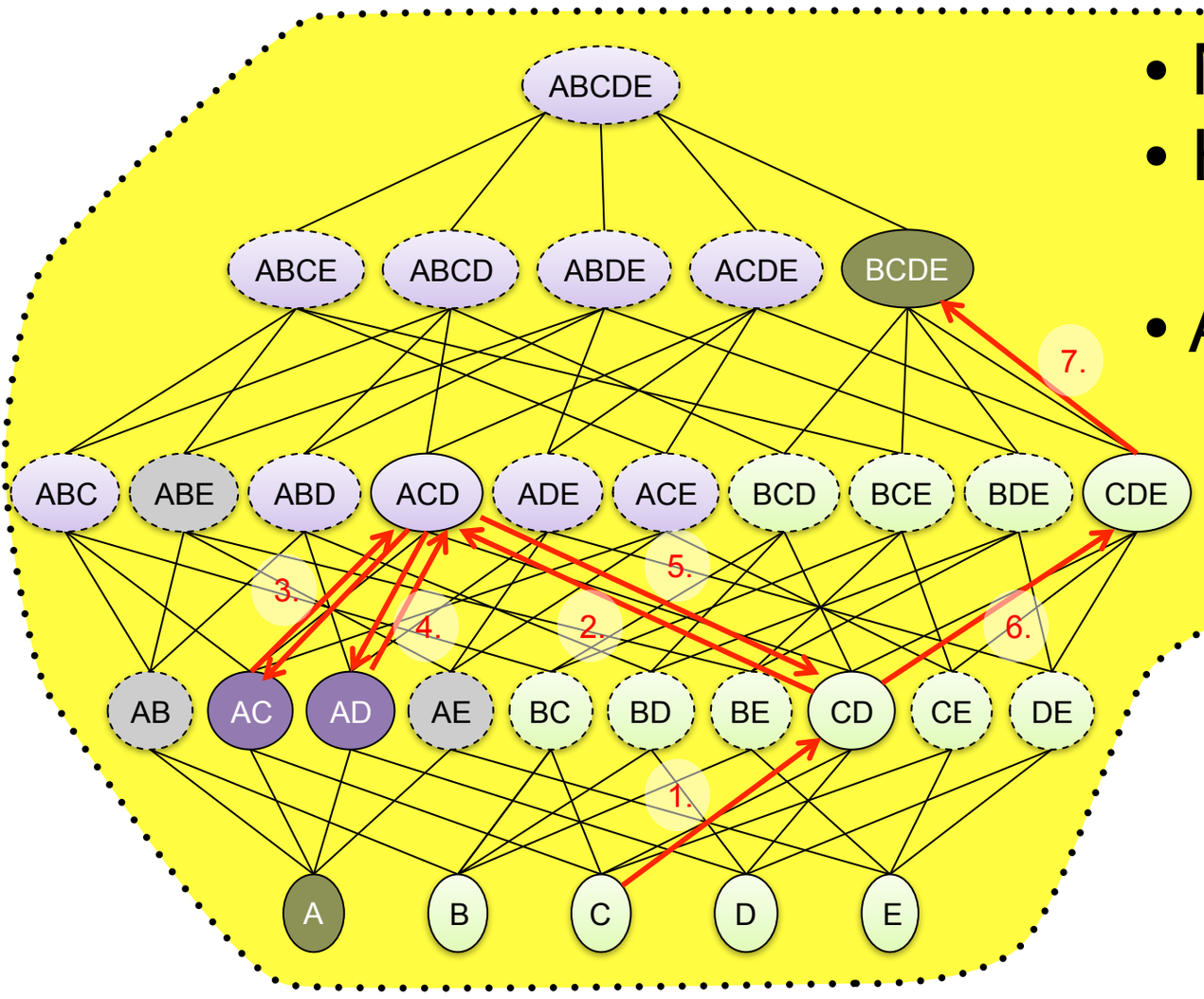
DUCC: Discovery of Unique Column Combinations

DUCC: Discovery of Unique Column Combinations



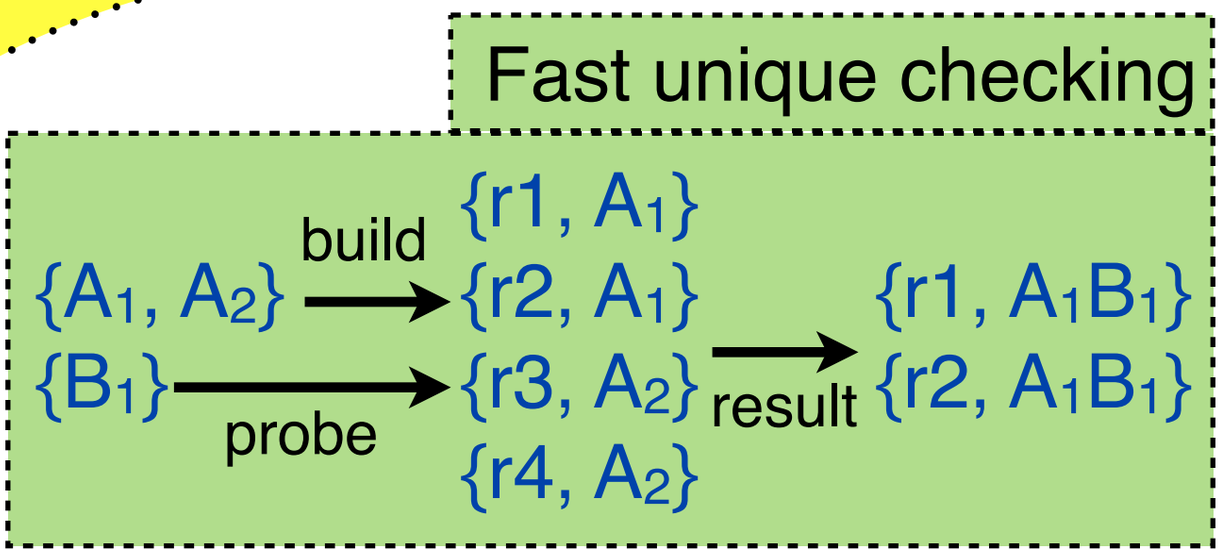
- Modeled as a graph coloring problem
- Hybrid traversal
  - (dept-first + random walk)
- Aggressive pruning technique
  - Technique to find and remove graph holes ($mnUcc^c \neq mUcc$)

DUCC: Discovery of Unique Column Combinations

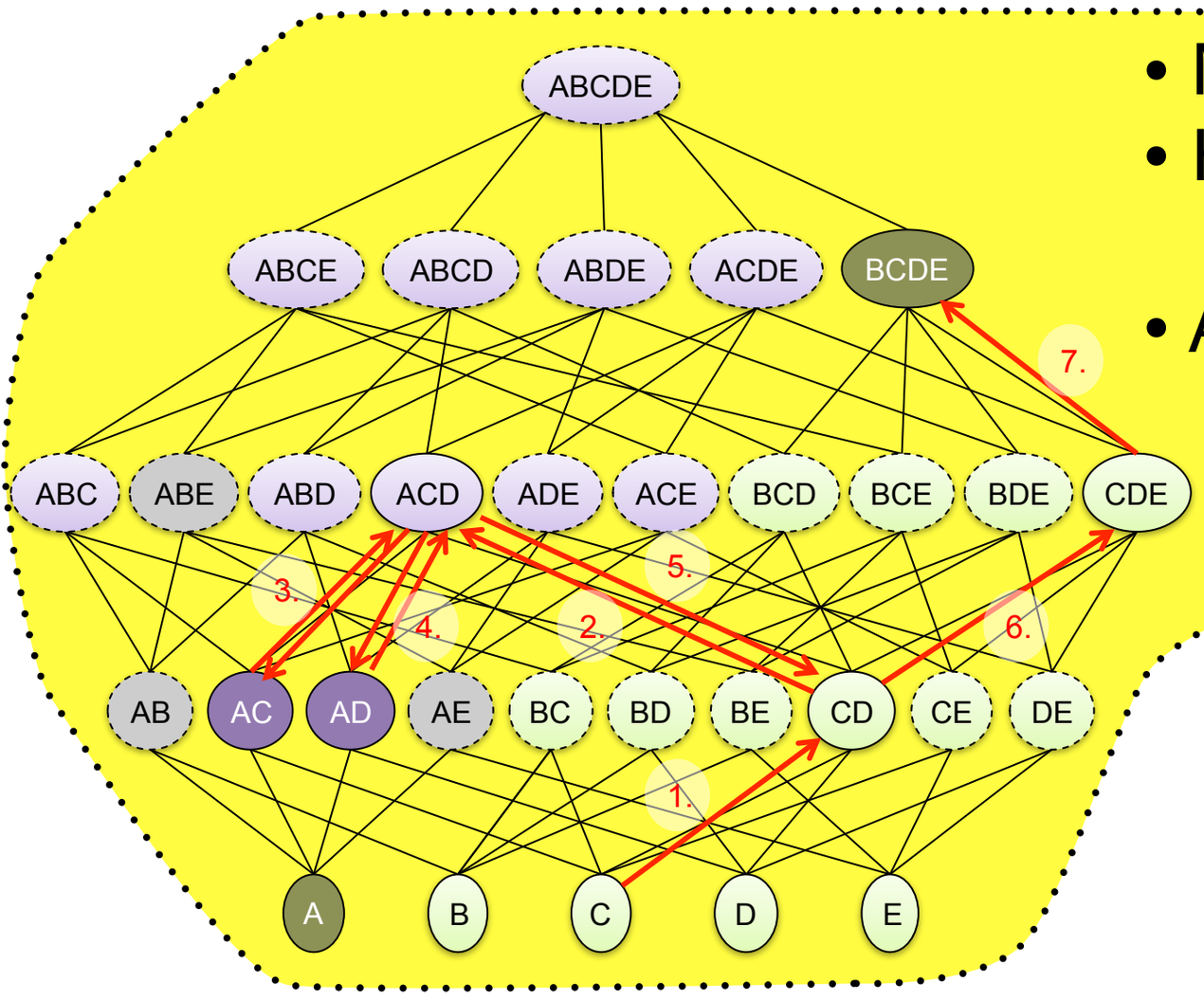


- Modeled as a graph coloring problem
- Hybrid traversal
  - (dept-first + random walk)
- Aggressive pruning technique
  - Technique to find and remove graph holes ($mnUcc^c \neq mUcc$)

Fast unique checking

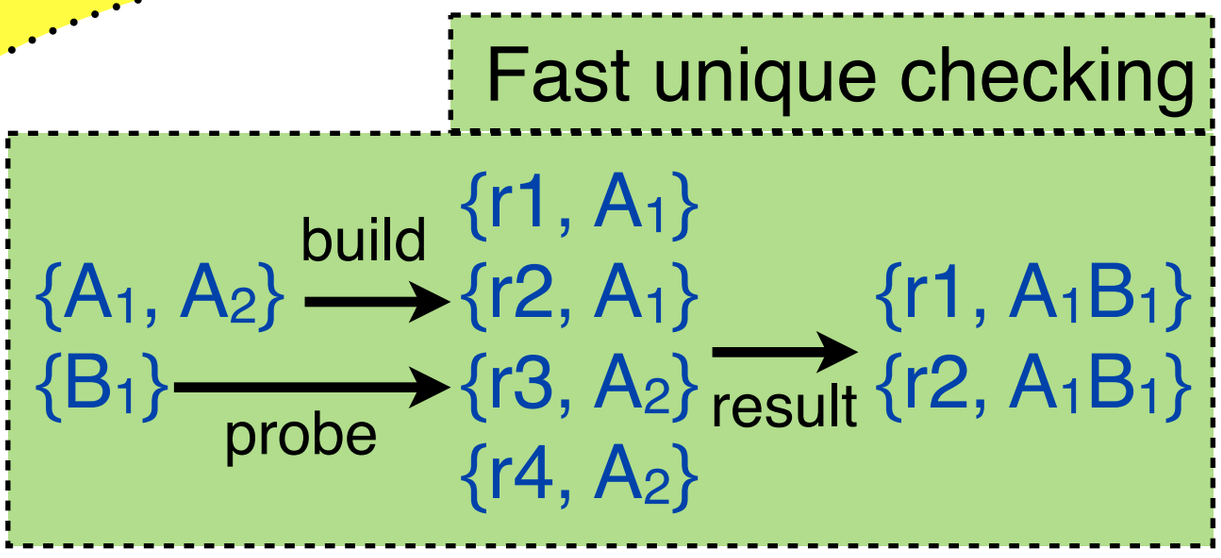


DUCC: Discovery of Unique Column Combinations



- Modeled as a graph coloring problem
- Hybrid traversal
  - (dept-first + random walk)
- Aggressive pruning technique
  - Technique to find and remove graph holes ($mnUcc^c \neq mUcc$)

Fast unique checking



- DUCC is up to more than **2 orders of magnitude** faster than state-of-the-art
- DUCC efficiently follows the border line (**solution set size dependent**)