Audition MCF 141

Alexandre Vigny

https://alexandre-vigny.github.io/mcf141.pdf

May 22, 2023

Alexandre Vigny

→ 31 years old, born in 1992.

→ PhD defense September 2018 *(5 years ago)*.

→ Living in Bremen, Germany.

→ Master in University Paris Diderot, 2015.
Logique Mathématique et Fondement de l'Informatique (LMFI).

→ PhD in University Paris Diderot. (3 years)
With Arnaud Durand & Luc Segoufin

→ Post-doc in Warsaw. (1 year)
With Szymon Toruńczyk & Mikołaj Bojańczyk

→ Post-doc in Bremen. (3 years)
With Sebastian Siebertz

Summary

Area of Research

- → Graph theory
- → Logic
- → Distributed computing

Highlights

- → Publication in J.ACM
- → 2 Upcoming journal papers
- → 9 Conference papers
- → Co-organizer of a workshop

PODC-DARe: Distributed Algorithms on REalistic network models

Summary

Area of Research

- → Graph theory
- → Logic
- → Distributed computing

Highlights

- → Publication in J.ACM
- → 2 Upcoming journal papers
- → 9 Conference papers
- → Co-organizer of a workshop

  PODC-DARe: Distributed Algorithms on REalistic network models

Teaching

- → In Paris (3 years)
  $\sim$ 180h
- → In Bremen (3 years)
  $\sim$ 270h

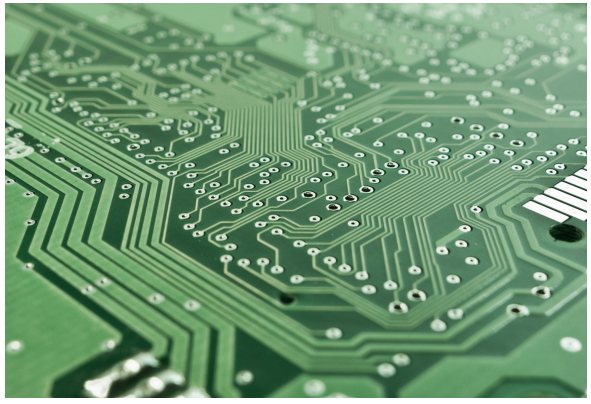
Highlights

- → Creation of syllabuses
- → Responsible for two courses
- → All level bachelor to master
- → Both in French and English

Algorithmic graph theory

Given a graph G and a property P: "Does G satisfy P?"

$\rightarrow$ Is G planar?

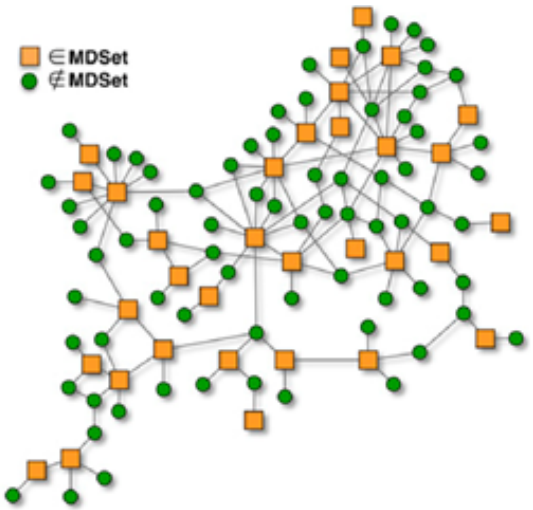


Algorithmic graph theory

Given a graph G and a property P: "Does G satisfy P?"

$\rightarrow$ Is G planar?

$\rightarrow$ Does G have a k-dominating set?



Algorithmic graph theory

Given a graph G and a property P: "Does G satisfy P?"

→ Is G planar?

→ Does G have a k-dominating set?

→ Is G connected?

Goal: Efficient algorithms ...

... at least for restricted graph classes and/or simple properties.

Logic

First-order (FO) logic

→ Can express k-independent set: *There are k vertices, that are not adjacent*

$$\exists x_1 \dots \exists x_k \bigwedge_{i<j} (\neg E(x_i, x_j) \wedge x_i \neq x_j)$$

→ Cannot express : connectivity, planarity, 2-colorability, ...

Logic

First-order (FO) logic

→ Can express k-independent set: *There are k vertices, that are not adjacent*

$$\exists x_1 \dots \exists x_k \bigwedge_{i<j} (\neg E(x_i, x_j) \wedge x_i \neq x_j)$$

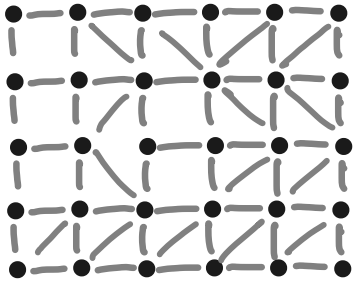
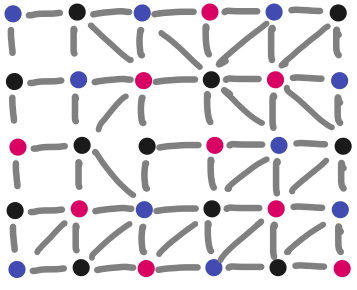
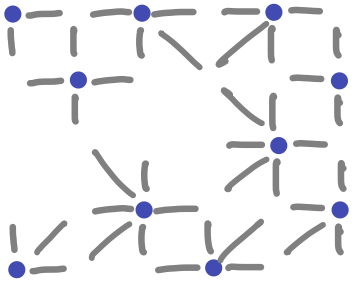
→ Cannot express : connectivity, planarity, 2-colorability, ...

Monadic second-order (MSO) logic

→ More general than FO

→ Can express : 3-colorability:

$$\exists X_1 \exists X_2 \exists X_3 \, (\forall x \bigvee_{i<3} x \in X_i) \wedge (\forall x \forall y \; E(x,y) \rightarrow \bigwedge_{i<3} (x \notin X_i \vee y \notin X_i))$$

Distributed computing

Distributed Computing : Local model

→ Different notion of efficient

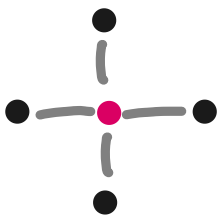
→ Time needed VS Information needed

Distributed computing

Distributed Computing : Local model

→ Different notion of efficient

→ Time needed VS Information needed

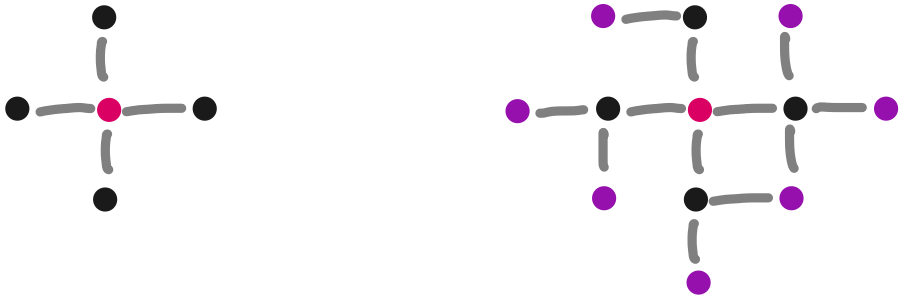


Distributed computing

Distributed Computing : Local model

→ Different notion of efficient

→ Time needed VS Information needed

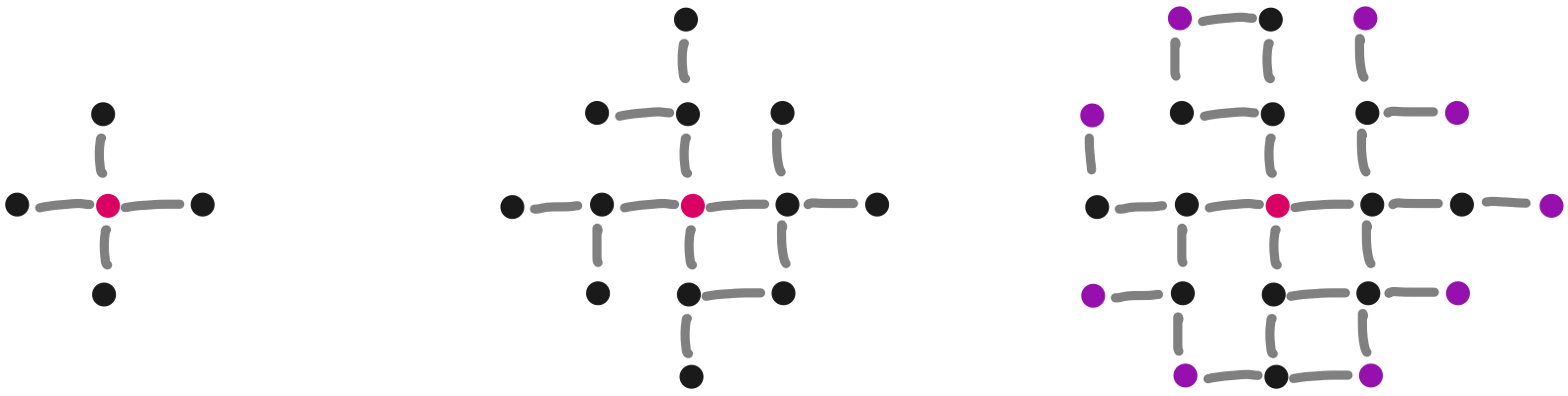


Distributed computing

Distributed Computing : Local model

→ Different notion of efficient

→ Time needed VS Information needed



→ Can you decide locally?

Logic & Meta theorems

Problems can be expressed in logic. (FO, MSO,...)

The $\mathcal{L}$, $\mathcal{C}$ model-checking problem:
Given $\varphi \in \mathcal{L}$ and $G \in \mathcal{C}$, does $G \models \varphi$?

Goal: fixed parameter tractable algorithms $O(f(\varphi) \cdot |G|^c)$

Logic & Meta theorems

Problems can be expressed in logic. (FO, MSO,...)

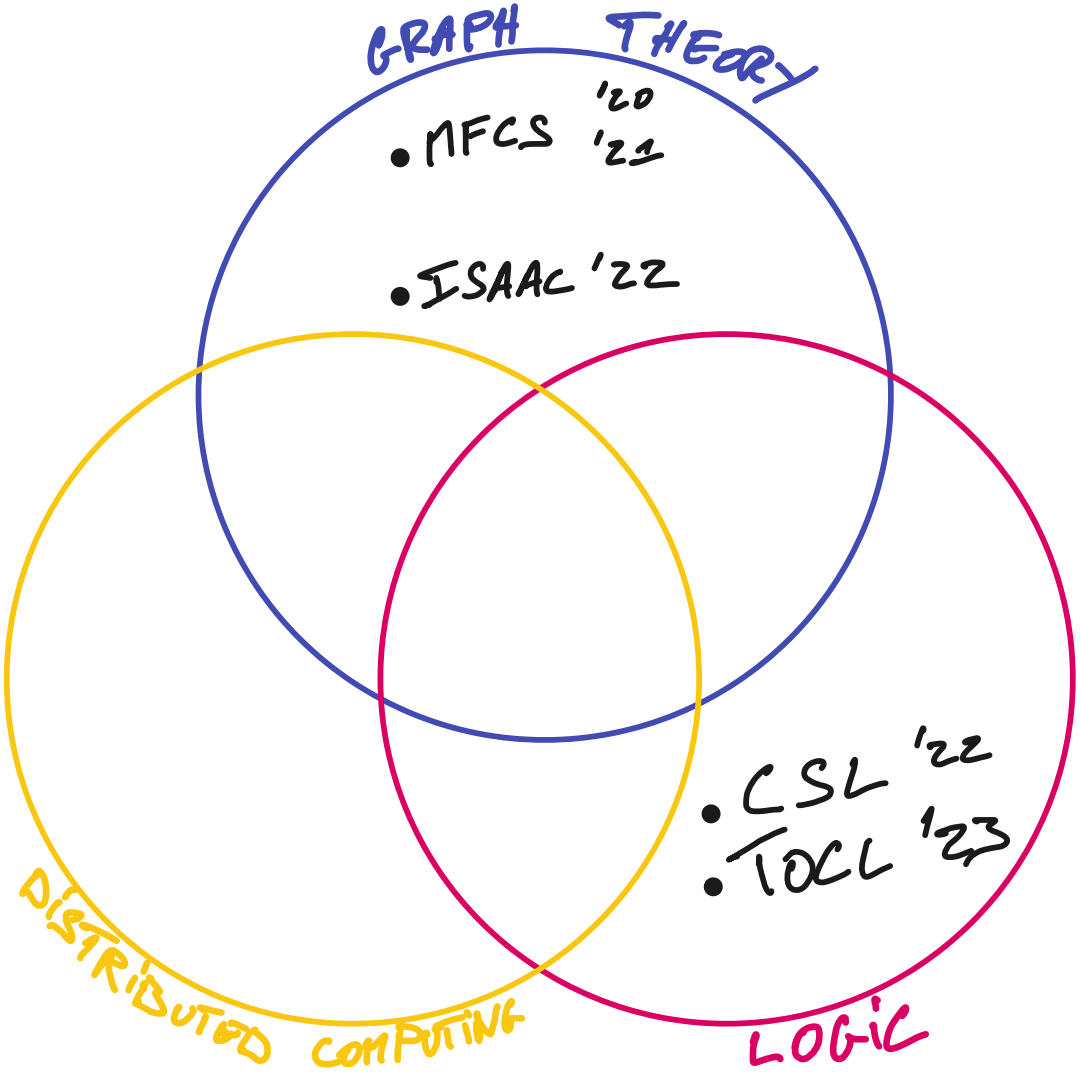
The $\mathcal{L}$, $\mathcal{C}$ model-checking problem:
Given $\varphi \in \mathcal{L}$ and $G \in \mathcal{C}$, does $G \models \varphi$?

Goal: fixed parameter tractable algorithms $O(f(\varphi) \cdot |G|^c)$

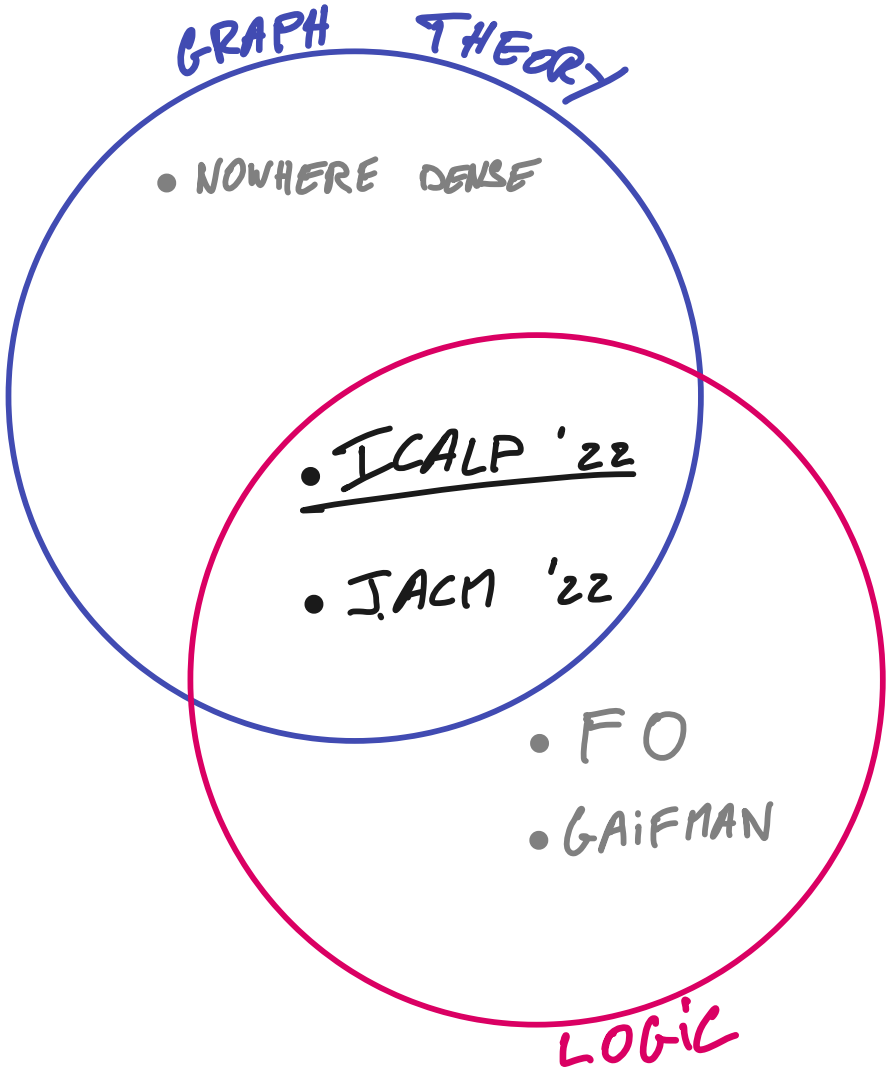
Courcelle's Theorem (1990):
for $\varphi \in \mathrm{MSO}$ and treewidth$(G) \leq k$, in time $O(f(\varphi, k) \cdot |G|)$

$\rightarrow$ Generalize many known results, ex:
Arnborg, Proskurowski 1989:
independent sets, dominating sets, graph coloring, Hamiltonian, ...
are linear on partial k-tree.

Result overview

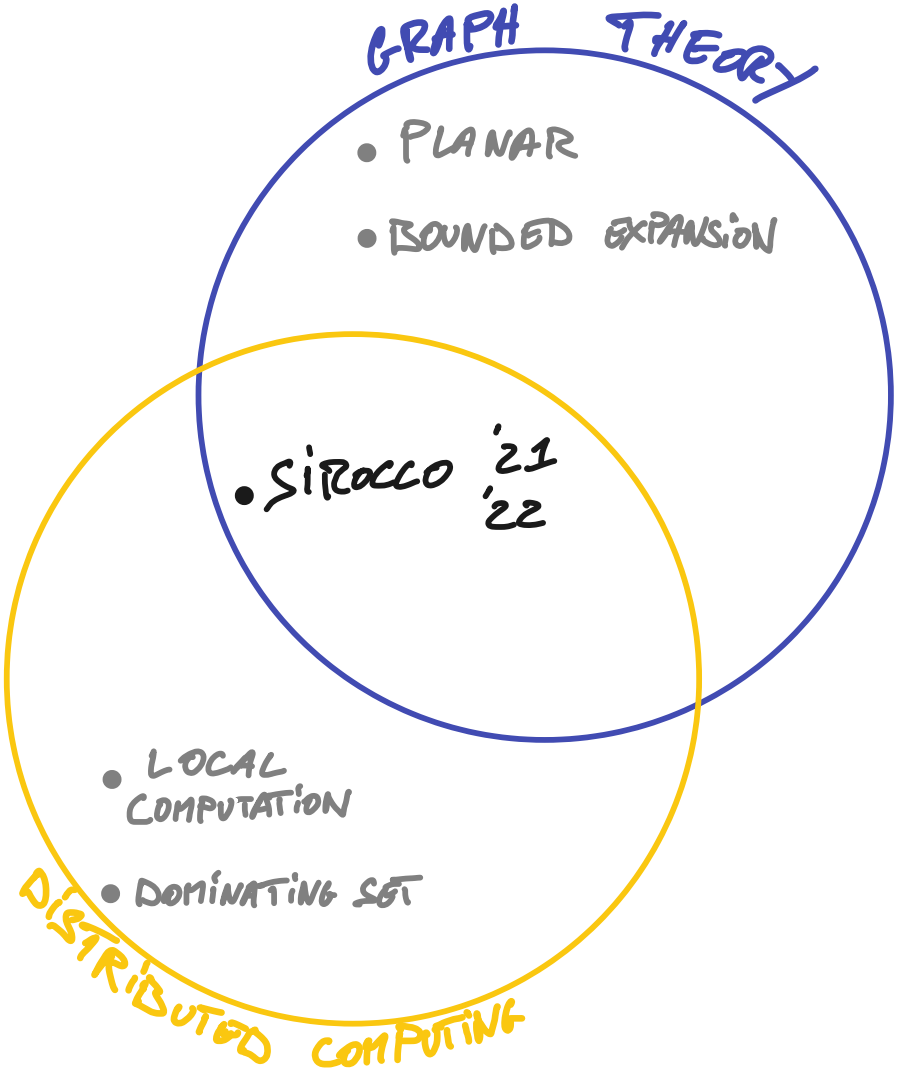


Result overview



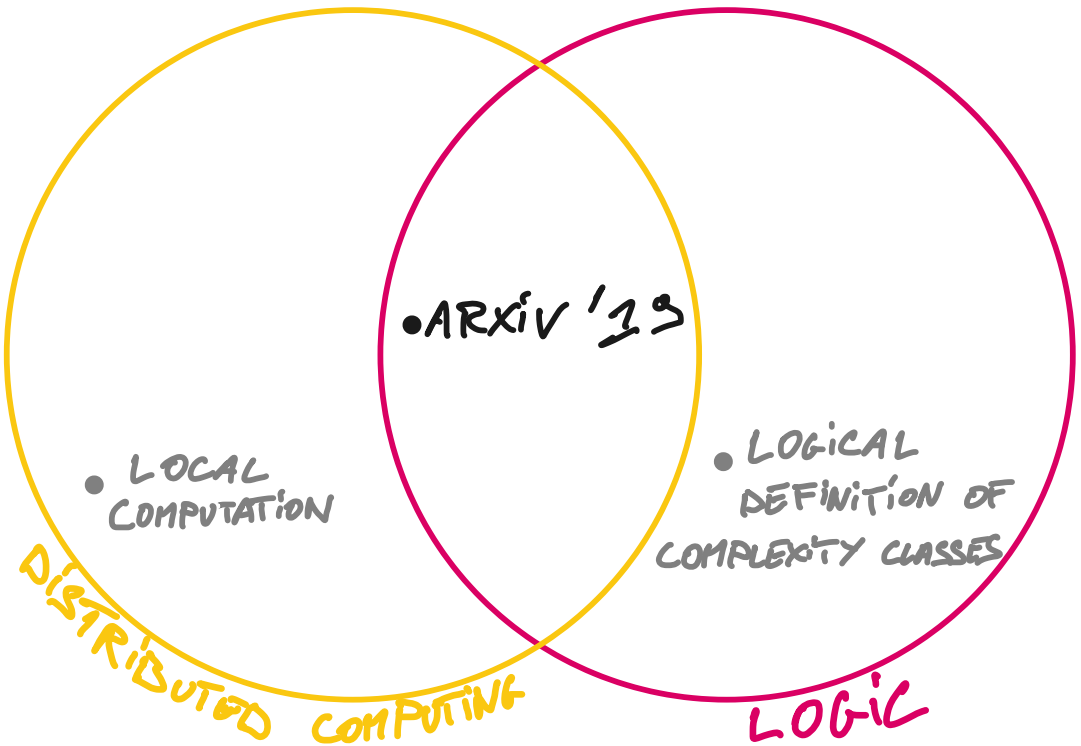
- ENUMERATION FO QUERIES
  -> PODS 2018
  JACM 2022

Result overview



- ENUMERATION FO QUERIES
  - -> PODS 2018
    SACM 2022
- DISTRIBUTED DOMINATION ON SPARSE GRAPH CLASSES
  - -> SIROCCO 2021/2022
  - -> EUR. J. COMB. (TO APPEAR)

Result overview



- ENUMERATION FO QUERIES
  -> PODS 2018
  SACM 2022
- DISTRIBUTED DOMINATION ON SPARSE GRAPH CLASSES
  -> SIROCCO 2021/2022
  -> EUR. J. COMB. (TO APPEAR)
- PARAMETERIZED DISTRIBUTED COMPLEXITY THEORY : A LOGICAL APPROACH
  -> ARXIV

Research project

Research project

First *(short term)* goal: new logics

Recently definition of a new logic: $\mathrm{FO} + \mathrm{conn}$

→ Pilipczuk, Schirrmacher, Siebertz, Torunczyk, Vigny. ICALP'22

→ more expressive than FO,

→ less than MSO.

Beyond $\mathrm{FO} + \mathrm{conn}$

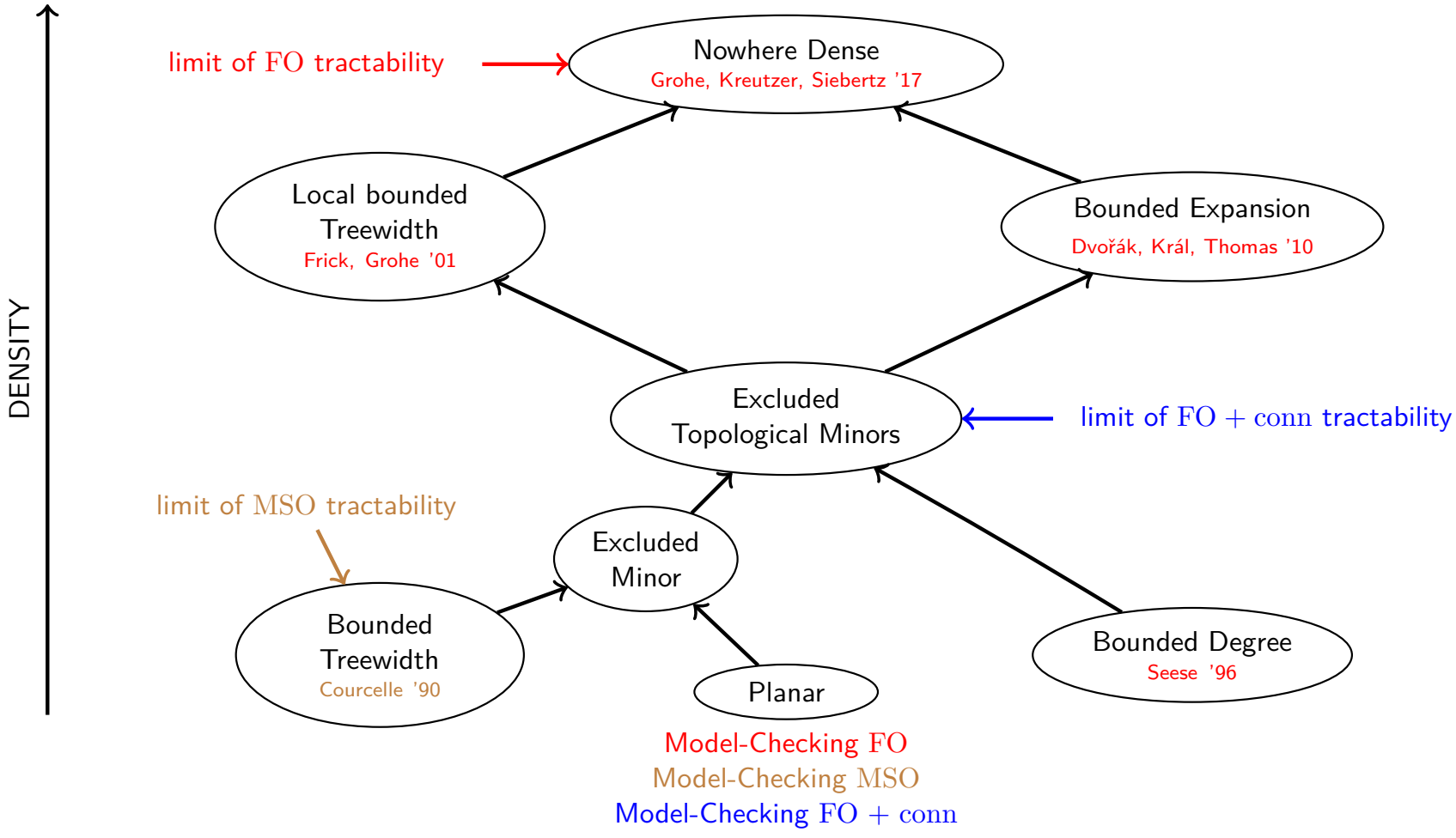
→ What can be added?

→ What do we want to express?

→ Example: a path of even length, using only blue nodes, ...

Keeping in mind algorithmic applications

Characterization of graph classes



Directed graphs

Direction on edges

- → More general
- → Problems are harder
  E.g. Directed Dominating Set is NP-complet on DAGs
- → Some problems do not care about orientation

Reconfiguration problems

- → No need to find a set
- → Here, the orientation matters!

Distributed computing & Certification

Local computing

→ Other notion of efficient

→ Still looking for meta theorems

Complexity classes

→ Hard problems may not be equally hard

→ Define complexity classes through logic

Compact certification

→ Feuilloley, Bousquet, Pierron
What Can Be Certified Compactly? Compact local certification of MSO properties in tree-like graphs. In PODC'22

→ Fraigniaud, Montealegre, Rapaport, Todinca
A Meta-Theorem for Distributed Certification. In SIROCCO'22

Integration

LINKS (Linking Dynamic Data)

- → S. Salvati, C. Paperman, I. Boneva, F. Capelli, M. Monet
- → Tree-like structures & MSO (S. Salvati)
- → Connections computation-logic (S. Salvati and C. Paperman)
- → Graph databases (I. Boneva)
- → Knowledge compilation & tree-like structures (F. Capelli, M. Monet)

- → Distributed computing is new
- → Adding a solid background on graph theory

Past teaching in Paris

Around 180h

→ Mainly Bachelor level

→ Similar to topics in Faculté des Sciences et Technologies

Subject	**Years**	**Level**	**Activity**
Initiation à la Programmation	2015-2016	L1	TP
Langages et Automates	2015-2016	L2	TD
Éléments d'Algorithmique	2016-2017	L2	TD
Base de données	2016-2017 2017-2018	L3 M1	TP & TD
Concepts Informatique	2017-2018	L1	TD

Past teaching in Bremen

Around 250h

- → Master level
- → Creation of syllabuses
- → Fully in charge of a lecture

Subject	Years	Level	Activity
Finite Model Theory	2019-2020	Master	TD
Sparsity	2019-2020	Master	TD
Parametrized Complexity	2019-2020	Master	TD
Set and Model Theory	2020-2021	Master	TD
Databases, Graphs, Algorithms	2020-2021 2021-2022	Master	Cours & TD
Set and Model Theory	2021-2022	Master	Cours & TD

Highlight

Various settings

→ In French, in English

→ All levels

→ Physical and remote

Various responsibilities

→ Creation of the content

→ Fully in charge of a lecture

Takeaway

→ Polls

→ Online white boards

Future 1

L1:

→ Outils pour l'informaticien·ne

→ Initiation à la programmation

→ Algorithmes et programmation

L2:

→ Algorithmique et les structures de données

→ Programmation orientée objet

→ Programmation en C

L3:

→ Programmation objet

→ Bases de données

→ Algorithmique et structures de données

Future 2

Master

→ Parcours Machine Learning
- Algorithmique avancée
- Bases de données avancées

Responsibilities

→ Responsable d'année

→ Stages

→ ...

Algorithmes & Graphes I, Introduction

→ Notions Élémentaires
- Modélisation par graphes
- Définitions
- Dirigés / non-dirigés
- Connexité / composante connexes
- Arbres et acyclicité

→ Représentation et implémentation
- listes d'adjacence
- matrice d'adjacence

Algorithmes & Graphes II, Dans le vif du sujet

→ Parcours de graphes
- Profondeur (Depth-First Search)
- Largeur (Breadth-First Search)
- Plus court chemin

→ Arbres couvrants
- Algorithme de Jarník-Prim-Dijsktra
- Algorithme de Kruskal

→ Flow
- Directed graphs
- Flow and Cut
- Ford-Fulkerson (max-flow min-cut))

Algorithmes & Graphes III, Ouverture

→ Graphes particuliers : Graphes planaires
- Notion de mineur
- Formule d'Euler

→ Problèmes difficiles
- 3-coloriage
- Ensemble dominant
- Notion de reduction
- NP-completude

Algorithmes & Graphes IV, Conclusion

→ Niveau : L2 / L3

→ Prérequis : Algorithmes et programmation
- Boucle
- Recursion
- Liste
- Tableau

→ Conflit avec Algorithmique et structure de données ?
- Champ d'application de ces structures
- Introduction de notions de complexité
  - $O(\log(n))$, $O(n)$, $O(n^2)$
  - Polynomial vs NP-complet

Thank you!

→ 2019-2023: Postdoc, University of Bremen. With Sebastian Siebertz

→ 2018-2019: Postdoc, University of Warsaw. With Szymon Toruńczyk & Mikołaj Bojańczyk

→ 2015-2018: Thesis, University Paris Diderot. With Arnaud Durand & Luc Segoufin

Info:

→ 1 Journal: J.ACM (TOCL & Eur. J. Comb. to appear)

→ 9 Conferences: ICDT, PODS, MFCSx2, SIROCCOx2, ISSAC, CSL, ICALP.

→ 1 Workshop (co-organizer): `https://podc-dare.github.io/`.

→ 1 Popularization: La gazette du GDR-IM.

tinyurl.com/short-polls

back