Doing applied (multidisplinary) research

A matter of balance

Thomas Schiex

August 27 2023

CP2023 Doctoral Program, Toronto, Canada

A Short History

Quick CV

1986	Generalist engineering school	
1991	PhD in AI	Lisp, λ-calculus, compilation, denotational semantics
1991	Software industry	
1991	Applied research at ONERA	French Aerospace Research Agency
1994	Applied research at INRA	Science for people, life and earth
2023	Still there!	startup project

Extra facts

- CP: theory and algorithms for the Weighted CSP — Cost Function Networks & toulbar2
- This is based on my French scientist experience, your case may sometimes differ
- I love Science.

Why should I consider doing applied (multidisciplinary) research?

- Because I apparently have to…
- Funding
- Access to Problems and Data
- Beef up your CV
- Collaborations
- Software
- Impact!
- Learn and do other types of exciting Science
- Have Fun!

Because I have to?

Do I really have to?

- Set a balance between Applied and more Basic Research
- Don't overestimate your "Applied obligations" (research institutes)
- Look to extreme profiles in your department, be convincing

Because I have to?

Do I really have to?

- Set a balance between Applied and more Basic Research
- Don't overestimate your "Applied obligations" (research institutes)
- Look to extreme profiles in your department, be convincing

Appreciate it!

- Applied Research can be inspiring for Basic research
- Useful to guide you in areas that are/can become significant

Funding, collaborative projects

Funding in multidisciplinary projects

- Applied research very present in national/supra national calls for projects
- Computer Science is rather cheap compared to experimental sciences
- Competition for money: shoot high — Hofstadter's law[a] + money for Basic research
- Computer scientist are rare in applied research, so desirable
- This may be for bad reasons (makes the project more sexy — the AI hype makes it worse)

[a] A project always takes longer than expected, even when the law is taken into account.

Funding, collaborative projects

Funding in multidisciplinary projects

- Applied research very present in national/supra national calls for projects
- Computer Science is rather cheap compared to experimental sciences
- Competition for money: shoot high — Hofstadter's law[a] + money for Basic research
- Computer scientist are rare in applied research, so desirable
- This may be for bad reasons (makes the project more sexy — the AI hype makes it worse)

[a] A project always takes longer than expected, even when the law is taken into account.

Risks!

- Risk of losing focus/perspective, jumping from AR projects to more AR projects.
- No maintained methodological plans for future
- Loss of perception of the "Front of research" in Basic research
- Computer Science changes quickly, you may become obsolete
- This is a weakness even if you intend to do only applied research

A gradient of project types, from funded to informal

Large (supra-national) funded projects

- Often require to have "ready-to-use" technology
- People may be there just for money
- Collaborations may be very loose (cover for ongoing work)
- But excellent for networking and getting visible!
- Money for informal/Basic projects
- Opportunities for higher-impact publications (partners, project)

A gradient of project types, from funded to informal

Large (supra-national) funded projects

- Often require to have "ready-to-use" technology
- People may be there just for money
- Collaborations may be very loose (cover for ongoing work)
- But excellent for networking and getting visible!
- Money for informal/Basic projects
- Opportunities for higher-impact publications (partners, project)

Informal projects

- Almost a guarantee that all partners are genuinely interested
- No deadline: can trade time for quality
- Ideal for long-term plans or software development
- All my durable software started inside informal collaborations
- They did help to access supra-national projects

Access to Problems and Data

Beyond random and crafted problems

- Real impactful problems with real data are different (easier but heavier)
- They can challenge your methodology or software and point out new directions for research
- Try to choose your Problems
  - Fit with your own skills
  - Match with your long-term Basic research plans
  - Important problem (for you too), hot, original, with hints of possible progress
  - Prefer repeatable problems over one-shot or tiny niche problems
- Can give access to visible high-impact factor publications
- Can be contributed to (a/your) repository + data paper.

Radio Link Frequence Assignment Problem (1992-1994)

Data still used in 2006 and later

- Not really multidisciplinary
- Funded by EUCLID (Military European Framework)
- Partners: a client (CELAR) + discrete optimization teams with various technologies
  $\rightarrow$ ILP, CP, Simulated annealing, Genetic algorithms, Dynamic programming,…

Radio Link Frequence Assignment Problem (1992-1994)

Data still used in 2006 and later

- Not really multidisciplinary
- Funded by EUCLID (Military European Framework)
- Partners: a client (CELAR) + discrete optimization teams with various technologies
  $\rightarrow$ ILP, CP, Simulated annealing, Genetic algorithms, Dynamic programming,…

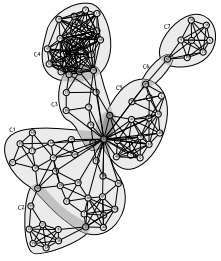
NP-hard problem already formalized by CELAR

- A set of fixed radio stations with some pairwise **links**
- Assign a frequency to each link from a finite set of available ones
- Avoid interferences (if possible, else MINimize interference)
- Minimize spectrum usage (number of (CARD) or maximum (SPAN) frequency used)
- Real data with actual technological constraints from CELAR

Mid and long term impact on my research

Project results

- All technologies could solve most instances (optimality)
- Interference MINimization problems remained open
- Very specific constraint graph structures
- One MIN instance solved by CP with a specific graph decomposition



Post-project impacts

- Data: lead to a well-cited data paper[a], real binary WCSPs! — XCSP,CFNlib,CSPLib
- Basic: importance of **Weighted** CSP (pre-toolbar) — Valued CSP[4]
- Basic: motivated research on Treewidth + Branch and Bound — BTD, HBFS[1,3]
- Funding: ANR (French Funding Agency) white project (Basic)

[a] B. Cabon et al. "Radio Link Frequency Assignment". In: *Constraints Journal* 4 (1999), pp. 79–89.

Beef up your CV

Open-minded and sensisitive to real-world problems

- Shows to recruiters that you are not a CP/CS-nerd
- Possibly get amazing number of citations DEE
- Make a game-changing contributions outside of CP recognition

Multidisciplinary collaborations?

Different from disciplinary ones

- Competences are naturally complementary
- CS desirable: if possible, choose collaborators with care (CV, publications,…)
- Don't assume they know what a Computer Scientist is.
  $\longrightarrow$ Optimization, Statistician, Data Scientist, Software Engineer, System Enginner, Excel expert
- Not interested in how/why it works, only what it can do for them
- Scientists have large egos, be prepared to defend your point (Basic)
- Check that they really need you: make them pay first (data cleaning)
- Software is great to durably inject a contribution

The good

- Developping software is fun (for most people)
- And often useful to encapsulate scientific contributions
- It can enable participation to funded projects and give access to (hot) data (publications!)

The good

- Developping software is fun (for most people)
- And often useful to encapsulate scientific contributions
- It can enable participation to funded projects and give access to (hot) data (publications!)

The long term cost

- Applying and maintaining software is demanding (doc, bugs, API changes)
- Don't engage in it for fun: can you make a difference?
- What do you intend to do with it?
  - Testing your own algorithms: Ok!
  - For the CP community (+users): competitors, originality, killer app, developpers, users?
  - For applied problems: possibly even more competitors (more users too!)
- Be ready to throw away the baby (missed target, competition, lack of visibility)

Visibility, reliability, accessibility, documentation…

How can I make my software tool a success?

- People won't use it just because it contains new fancy algorithms
- Working documented code (with examples)
- Caseshowed in as many publications/projects as you can (proof of usefulness)
- Play with extreme data (volume, hardness,…) and publish
- Win competitions in as many areas as you can: Winner takes all!
- Submit tutorials to conferences, schools…
- Accumulate benchmarks as examples
- Make it accessible (package, scripting API,…)

Valorization

From a scientific software to a company

- It's hard to successfully transmit it to an existing company
- Need to have excellent knowledge of its strengths,weaknesses, technology
- More doable by a Post-Doc, a PhD student or even a senior scientist
- Obvious: you need a business model, know your market and competitors
- A whole new story, 75% of startups die in their first years
- Very likely a great experience, ask Guillaume Fages!

From a research tool to a SOTA CFN and stochastic GM solver

- CFN = Weighted CP (replace Booleans in CP by integer costs)
- Born in 1999 for pure research purposes (benchmarking Soft AC)
- Javier Larrosa (Spain) and myself (in C)
- Completely rewritten by S. de Givry in C++: toulbar2 lead dev.
- Crucial for publication but no multidisciplinary application before 2006
- Participates in all WCSP/Stochastic GMs competition (last UAI'22)
  - → ToulBar2 variants were superior to the CPLEX variants in all our tests",AAAI'20, S. Haller et al.
- Documentation, Debian, accumulation of benchmarks (CFNLib)
- Engineering: enhanced file formats (JSON, decimal point numbers), Pytoulbar2
- We have tutorials, participated to schools
- Several contributors (MIT licence, France, Hong-Kong, Spain, Germany)

Applications in the lab and beyond

In the lab

- Farm animals pedigree debugging
- Assistance for complex genome sequence assembly
- Bayesian network structure and parameters estimation
- Spatio temporal layout of agro-forestry crops
- Protein Design (our killer app)

  → "The Toulbar2 package for WCSPs significantly improved the state-of-the-art efficiency for protein design"

  Com. ACM-20, B. Donald et al.

Applications in the lab and beyond

In the lab

- Farm animals pedigree debugging
- Assistance for complex genome sequence assembly
- Bayesian network structure and parameters estimation
- Spatio temporal layout of agro-forestry crops
- Protein Design (our killer app)

  → "The Toulbar2 package for WCSPs significantly improved the state-of-the-art efficiency for protein design"

  Com. ACM-20, B. Donald et al.

And beyond

- Musical composition,…
- Probabilistic ML (image analysis, MDPs)
- Used by at least one startup

For Impact!

Go beyond direct impact in CP

- Solve open problems in other disciplines: visibility for you/CP
- Many scientists don't have the slightest idea of what modern computing can do
- Save the world!

Learn other types of Science

Demanding but rewarding

- It takes time to understand their problem(s)
- To understand the fundamental "body of knowledge" behind it
- To perceive their "front of research" (originality)
- Every Science area has its "do and don't"
  - CS: between math (hypothetico-deductive) and experimental (benchamrking),…
  - We like theorems, properties, proofs and universal answers
  - Experimental sciences are ruled by the real world
  - Things that work vs. things that are beautifully crafted, or just hard to produce
- All this is needed to see where you can have the most impact (and publish)

How can I do that?

You need to

- Talk to people (possibly inconsistent)
- Attend to scientific talks
- Invite people or visit other labs
- Read introductory books (more than one)
- Read survey papers
- Then read technical papers, several of them
- Really enjoyable if you like to learn more new Science
- Also expands your methodological knowledge (existing solutions)
- Enhances self-criticism on CP and improves significance

Getting knowledgeable in other science areas

Is it that simple?

- You cannot change your "application area" every month or year
- It took me 3 to 5 years to become fluent enough to discuss with biologists
- I dive into a specific biological problem for a long period (decade)
- The ratio of time I spend on Applied vs. Basic research oscillates with a long period (years)

Choose you application domain carefully (when possible)

Publication

- Prefer a young but maturing interdisciplinary domain
  - With a gradient of applied CS-related journals (CABIOS, Bioinformatics, Journal of computer and Chemistry,…)
  - You need a critical mass of active scientists to be reviewed
- Interdisciplinary Bonus
  - Methodological (AI/CP/OR: lead author)
  - Interdisciplinary (Bioinformatics: often lead author)
  - Applied science area (Biology: rarely lead author)

Anything disagreeable then?

Well…Yes

- It takes a lot of time and effort
- You'll have to abandon attractive Basic research trails (others will explore them, frustration)
- It's harder to attract methodological students/partners (biases)
- Keep your Applied/Basic balance!

Where?

Two classes of positions

- In an applied multidisciplinary institute that has a computer/math lab/department
- In a pure biology/physics/ecology/…lab that wants a CS person
- I have always been in the first type of position
- My feeling is that it's much better to preserve the "Holy Balance".

Interdisciplinary with Software (1999)

Gene prediction

- Nearby lab starts a “genome sequencing project” on its favorite organism
- Internationally recognized specialists on this organism
- Need to identify *coding genes* in DNA (two stranded linear molecule)
  - Very few software to analyze the DNA sequence
  - Existing software showing very bad performance on their data
  - Very hot topic, only few genomes sequenced at this time

bibliography

Interdisciplinary with Software (1999)

Gene prediction

- Nearby lab starts a "genome sequencing project" on its favorite organism
- Internationally recognized specialists on this organism
- Need to identify *coding genes* in DNA (two stranded linear molecule)
  - Very few software to analyze the DNA sequence
  - Existing software showing very bad performance on their data
  - Very hot topic, only few genomes sequenced at this time bibliography

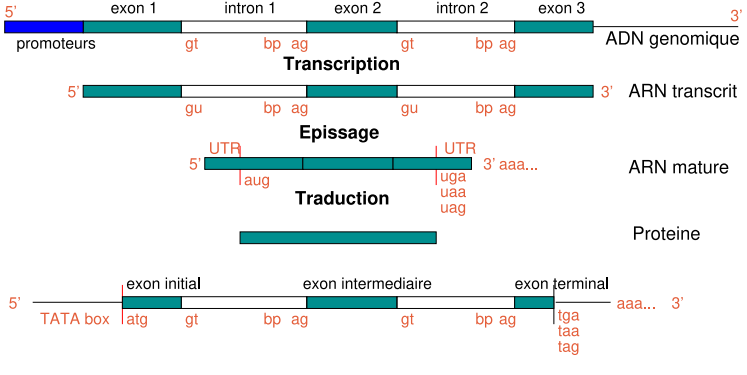
What CP/OR/AI can do here?

- Prediction is more Machine Learning than Weighted CP…but
- DNA is a discrete object (4 letters), being part of a gene (no, forward, reverse, frame) too
- The problems seems expressible as a discrete optimization problem CFN
- Looks Fun and significant!

More problem details

Problem facts

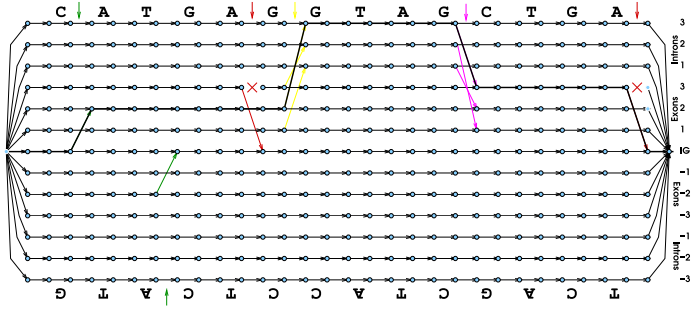
- Each coding region is bounded by signals (fuzzy patterns detectable by existing ML tools)
- Coding regions have specific statistical biases (3 periodicity, codon biases, Markov chains)
- Additional experimental evidence that some interval should be coding (bonus for coding)
- Constraints on minimum/maximum length of regions, overlaps



A (constrained) shortest path problem — linear CFN

One solution which improved over SOTA

- A weighted graph that captures all solutions — linear decision diagram/CFN
- Double dynamic programming algorithms — handles distribution lengths
- Graph weighted by a combination of the external information above
- Parameters set my maximum of empirical accuracy — GA + block coordinate descent
- Similar to "Linear Conditional Random Fields" — Lafferty, few years later
- **Established a connection between WCSP/CFN & Probabilistic Graphical Models**



From an informal project to…

Much more

- Invitations to several international/European projects — 1/organism
- Publications in Nature-type journals — 1/organism
- Better knowledge of genomics
- And of discrete probabilistic ML technology (CRF, competitors)
- An open-source C++ software (40k lines) + far more scripts for pre/post processing

From an informal project to…

Much more

- Invitations to several international/European projects 1/organism
- Publications in Nature-type journals 1/organism
- Better knowledge of genomics
- And of discrete probabilistic ML technology (CRF, competitors)
- An open-source C++ software (40k lines) + far more scripts for pre/post processing

A papers and citations factory!

- But increasing time for maintenance, enhancements, applications, publications
- Basic research activity became hard to maintain
- Decided for adoption: EuGene under the control of bioinformatics colleagues
- Dedicated engineer recruited for maintenance/evolution (bio lab)
- Still used, I asked to not appear anymore on papers
- Work spread over 20 years!

So?

Applied multidisciplinary research

- Very rewarding, lot of satisfaction
- You learn both other disciplines and related methodologies
- Widens your horizon
- Demanding but compatible with and very useful for Basic research
- A lot of pleasure and fun!
- Keep your balance!

References I

[1] David Allouche et al. "Anytime Hybrid Best-First Search with Tree Decomposition for Weighted CSP". In: *Principles and Practice of Constraint Programming*. Springer. 2015, pp. 12–29.

[2] B. Cabon et al. "Radio Link Frequency Assignment". In: *Constraints Journal* 4 (1999), pp. 79–89.

[3] S. de Givry, T. Schiex, and G. Verfaillie. "Exploiting Tree Decomposition and Soft Local Consistency in Weighted CSP". In: *Proc. of the National Conference on Artificial Intelligence, AAAI-2006*. 2006, pp. 22–27.

[4] T. Schiex, H. Fargier, and G. Verfaillie. "Valued Constraint Satisfaction Problems: hard and easy problems". In: *Proc. of the 14th IJCAI*. Montréal, Canada, Aug. 1995, pp. 631–637.