Next 10 years of Constraint Programming

The Science of Constraints

Carla P. Gomes
and
Bart Selman

Cornell University

CP 2006

Progress in the last 10 years

- CP has proved itself!
  - CP has found a niche in which its techniques excel
    - →Highly combinatorial problems;
    - →OR is (slowly) recognizing the role of CP
  - CP solvers have moved into the real-world arena (e.g., ILOG Solver, EXPRESS/CHIP)
  - CP has developed sophisticated techniques by bridging across different disciplines e.g.:
    - Global constraints (use of network flow algorithms, dynamic programming, automata, etc)
    - Hybrid approaches (integration with OR)
    - Modeling language (e.g., OPL, COMET,ZINC)

Progress in the last 10 years

CP as a community has attained a "critical mass"

Quality and reputation of CP conferences and journals has grown steadily

Challenges/Opportunities

Challenges and Opportunities: Technical

Beyond pure discrete models
Beyond satisfaction → optimization

Integration with different theories is key to combining discrete and continuous reasoning and going beyond satisfaction

Good models for hybridization

- OR / SAT / Non-Linear Models / Global and Local Search / Other theories (e.g., probabilistic reasoning, machine learning) more later… (note: OR is very bad at integration…)
- Explanations between theories (Nogood learning / cut generation, etc) – catalog of "explanations" for different global constraints
- Incrementality, propagators, frequency of communication between theories

Challenges and Opportunities: Technical

- Learning (cf. e.g., SAT)
  - Clause learning and conflict analysis
  - Problem vs. solver features (machine learning)
  - Structural problem characteristics to guide search – problems vs. solvers
- Randomization, Restarts, and Algorithm Portfolios (cf. e.g., SAT)

- Fully exploiting the increasing computational power – e.g., clusters (distributed, parallel, portfolios → machine learning may play a critical role…)

Side comment: SAT community much more receptive and reactive to new ideas… why? More later…

Challenges and Opportunities: Technical

Blackbox Solvers

- cf. SAT and MIP community;
- Tradeoff between expressiveness and complexity; restricted languages (not really formally but in practice)
- Source code available
- Easy use

CP-Solvers

for

Dummies

Gecode, Eclipse –

Minion

Comet

but not really blackbox solvers … maybe we should have blackbox versions of these…

Also benchmarks, competitions, etc

Challenges and Opportunities: Technical

- Beyond satisfaction and optimization:

  Quantified CSP,

  Model Counting,

  Probabilistic Constraint Reasoning,

  Preferences

Challenges and Opportunities: Technical

- Understanding, explaining, and exploiting constraint structures as they occur in the real-world. The main challenge in our opinion

  More later …

Challenges and Opportunities: Applications

Question to the community:

- Application beyond traditional (OR) applications – "killer application" providing fundamentally new ideas to approach them (RELEVANT)

  (cf. SAT (verification); Machine Learning / Data Mining – different from OR and Statistics)

  - Computational Biology – e.g. protein folding (non-linear models)?
  - Software and Hardware verification?
    - Test pattern generation
  - Natural Language Understanding?
  - Security?
  - Combinatorial auctions?

Challenges and Opportunities: Scientific Questions

CP has made tremendous progress in terms of:

- Techniques
- Traditional Applications (still looking for “killer” application…)
- Industry

Question to the community:

What are the core scientific questions that we are pursuing?

Challenges and Opportunities: Scientific Questions

What are the core scientific questions?

- E.g. Theoretical Computer Science – understanding the fundamental limits of computation with emphasis on space and time tradeoffs;
- E.g. Artificial Intelligence – understanding, modeling, and replication of human intelligence;
- Machine Learning – understanding the processes behind human learning and scientific discovery
- Astronomy – origins of universe

Challenges and Opportunities: Scientific Questions

Related issue:

Engineering vs. Academia
Industry vs. Academia

Similar issues happen in OR – OR is known for its techniques and applications (ORIE) but not really as a major academic and scientific player – is that what we want for CP?

Challenges and Opportunities: CP Uniqueness

Question to the community:

What makes CP unique, different from OR and algorithms?

- Techniques? Declarative?
- Applications?

Intelligent Information Systems Institute

CORNELL UNIVERSITY

Algorithm development: the battle between CS&OR vs. CP

Traditional Algorithmic Design (CS)

Traditional algorithmic design:

- Driven by worst-case or (formal) average case analysis
- Effective approach since early 60s **but** shortcomings are becoming more apparent:
  - Worst-case too pessimistic. Any NP-complete problem: 2^n (can't do any better if P=/=NP; "end of alg. design")
  - Average-case: Requires concrete probability distribution on instances. Quite likely infeasible to get a match with "real-world" problem distribution.

Computational tasks are studied
as if they were a **formal mathematical object**.

Intelligent Information Systems Institute

CORNELL UNIVERSITY

Our proposed alternative

Viewing CP's as natural phenomena

– Study computational constrained problems (CP) as **naturally occurring objects / phenomena**. That is,

View CP algorithm design as a problem of the natural sciences instead of "only" a mathematical problem.

– Advantage:

Constrained Problems are worst-case NP-complete but (real-world) structure likely allows for practically effective and scalable solution strategies.

– Let's make this more concrete.

Viewing CP's as natural phenomena

Central question:

- Understanding, Exploiting, and Explaining Constrained Structures as Occurring in Real-World – not only as abstract mathematical objects but also as "natural" phenomena

Methodology:

- Scientific Method –
  - observe the phenomena;
  - develop formal models and theories to explain the phenomena;
  - check the validity of the model in real-world problems (validation or refutation)

Approaches

- Develop solution strategies exploiting structural properties as found in the real world

Viewing CP's as a natural phenomena

- **Key advantage:**
  - Avoids worst-case and (formal) average-case **road block.**

So, let's look for interesting properties found in empirical studies of CSPs, propose formal models and validate those models

CP's as a natural phenomena

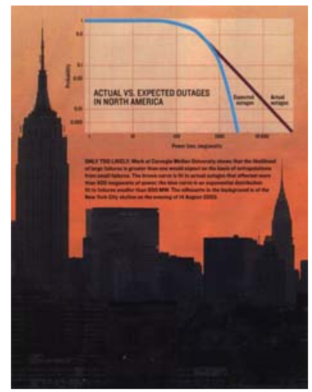
- **Examples:** **[Apologies for bias towards our own work.]**
  - **Phase transition phenomena** (empirical start followed by rigorous mathematical modeling; became active subfield with mathematicians, physicists, and computer scientists.)
  - **Heavy-tailed phenomena in backtrack search** (empirical start followed by mathematical modeling; led to randomization and rapid restarts (used in SAT solvers) and backdoors; dialog with different communities: pervasiveness of heavy-tails in real-world settings in economics, science, and engineering.

The Pervasiveness of Heavy-Tailed Phenomena in Science: Economics, Engineering, and Computation

AMERICAN ASSOCIATION FOR THE ADVANCEMENT OF SCIENCE

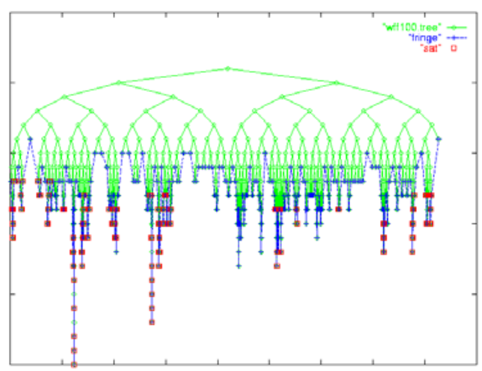
Annual meeting (2005).b

Tsunami 2004



Blackout of August 15th 2003 > 50 Million People Affected

Financial Markets with huge crashes



Backtrack search

… there are a few billionaires

CPs as natural phenomena.

- **Examples, cont.:**
  - **Backbone and backdoor variables**

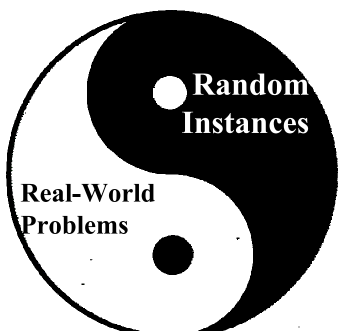
    initially a formal notion to explain heavy-tails, validated empirically; boosts understanding of CSP/SAT solvers on large real-world problems; explain why randomization is so effective.
  - **XOR Streamlining**

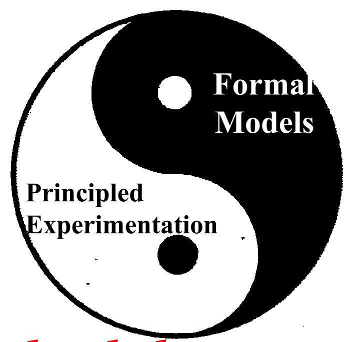
    Inspired by a real-world problem, motivated by a need for a general approach and by the intriguing theoretical properties of XOR constraints; led to knew model counting and sampling strategies; boosts CSP search on real-world instances.

[Apologies for bias towards our own work.]

CSPs as natural phenomena.

- It's unlikely that pure mathematical thinking / modeling would have led us to the discovery of these phenomena.
- In fact, to understand real-world CPs (which are everywhere!), we argue that the empirical study of phenomena followed by rigorous models and analysis is a *sine qua non* for the advancement of the field.





- *Fortunate side-benefit: scientific methodology sets us apart from standard approach towards algorithm development / optimization in CS & OR. And, we believe this methodology will be very fruitful over the next decade(s).*

CPs as natural phenomena.

Other examples of how this view has (implicitly) molded the CP field – at all levels including techniques (e.g., several types of global constraints and of course applications).

Gracefulness in graphs

Can we use CP to prove (or disprove) the
un-gracefulness of a class of graphs?
(Barbara Smith)
Connections to crystallography,
astronomy, etc.

Challenge/Opportunities: Challenge Problems

- Community should discuss and formulate a few (5-10) “killer” challenge problems:

  - should be quite different from what other fields (OR / algorithms) are doing

  - should be highly visible problems that show a unique angle

Successful examples that have boosted other areas:

AI – Deep Blue (1997)

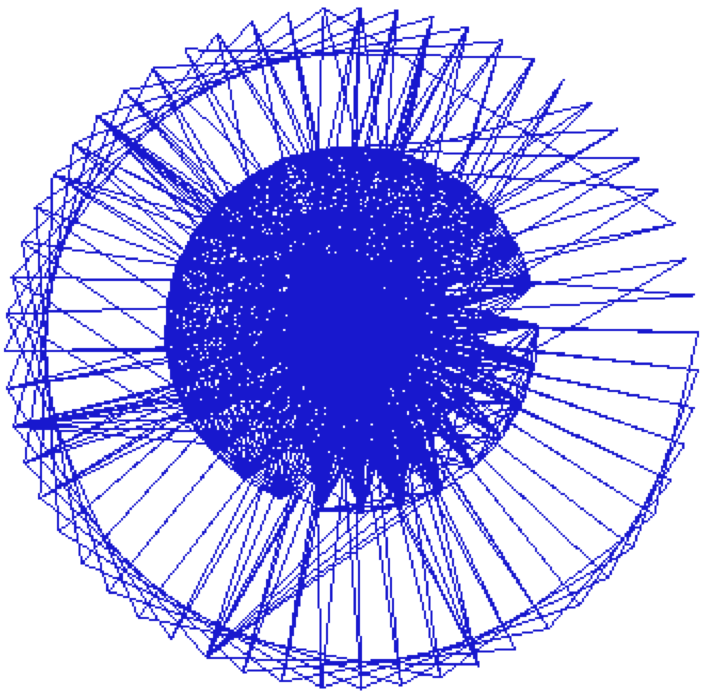
Theorem Proving – Proof of Robbins Conjecture (1996)

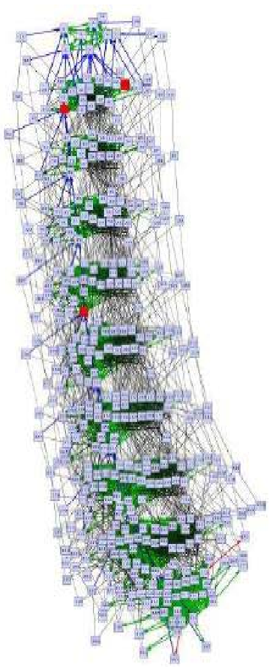
Robotics – DARPA Grand Challenge Race (2005)

The End

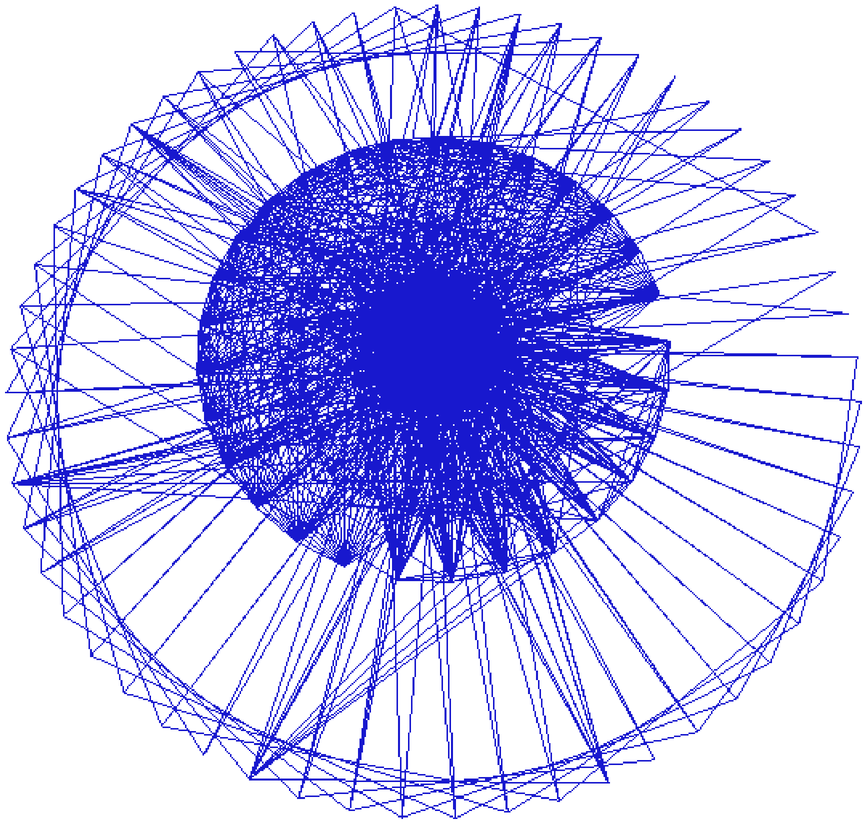
www.cs.cornell.edu/gomes

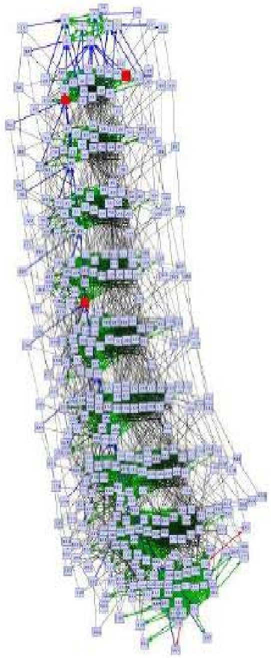
Map Top: running without backdoor



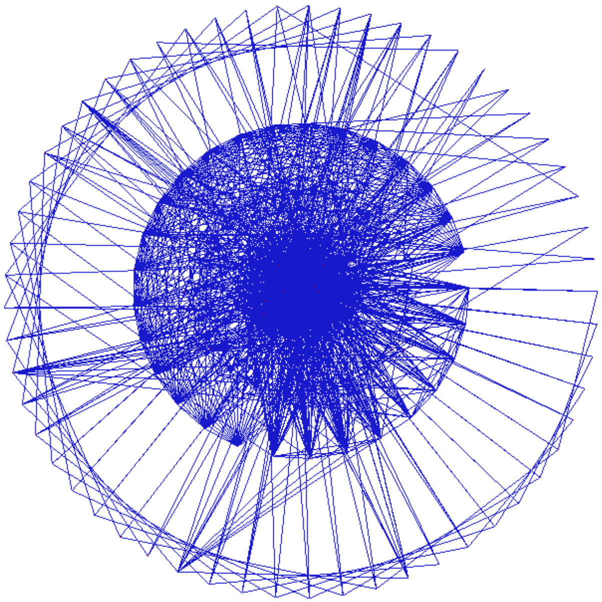


Map Top: running with “a” backdoor (size 9 – not minimum)

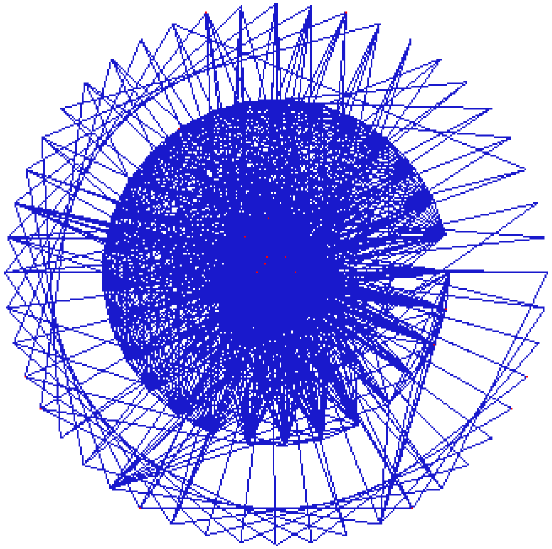




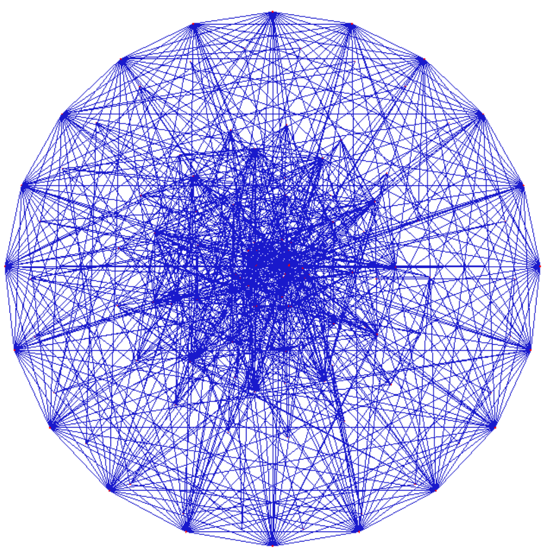
Map Top: running with backdoor (minimum – size 3)



Initial Graph



After setting one backdoor



After setting two backdoors

Instance is Unsatisfiable

After setting three backdoors