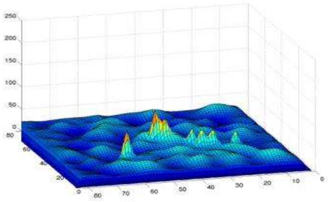
Average Case Analysis of NP-complete Problems: Maximum Independent Set and Exhaustive Search Algorithms

Cyril Banderier, Hsien-Kuei Hwang,
Vlady Ravelomanana & Vytas Zacharovas

LIPN – UMR CNRS 7030, Institut Galilée,
Université de Paris 13.
{cb,vlad}@lipn.univ-paris13.fr
&
Institute of Statistical Sciences,
Academia Sinica – Taipei.
{hkhwang,vytas}@stat.sinica.edu.tw

Complexity landscape of algorithms



Worst-case complexity: $\text{Max}_x T(x)$ Average-case complexity: $\text{E}_x[T(x)]$

Smoothed complexity: $\text{Max}_x \text{E}_\epsilon[T(x+\epsilon)]$
$\longrightarrow$ what are the difficult regions?

Smoothed Analysis of Algorithms (~20 articles so far):
Why The Simplex Algorithm Usually Takes Polynomial Time [Spielman & Teng 01]
Smoothed analysis of three combinatorial problems [Banderier & Beier & Mehlhorn 03]

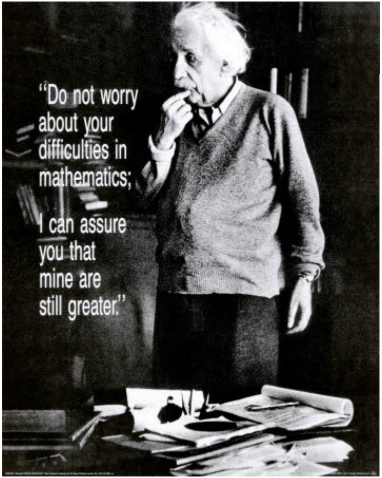
Constraint Satisfaction Problems

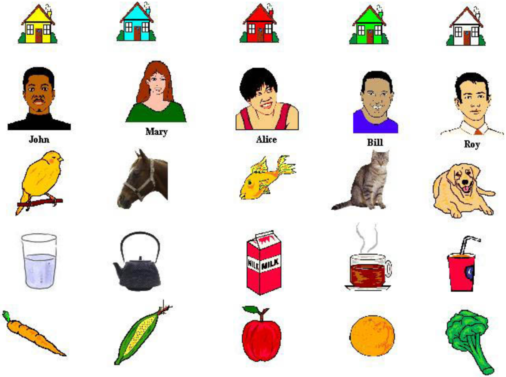
We consider a whole class of problems known as **Max–CSP** which play a key rôle in Computer Science.

CSP

- n variables $x_1, x_2, \cdots x_n$ belonging to finite domains.
- A set of constraints (clauses) over these variables: a constraint is a relation between the variables defining *authorized* combinations of them.
- **Decision problem**: does it exist an assignment (or a solution) of the variables satisfying all the constraints?
- **Optimization problem**: maximize the number of satisfied constraints.

Constraint Satisfaction Problem: Einstein Problem



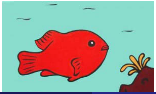


Constraint Satisfaction Problem: Einstein Problem

The author of this problem is Albert Einstein who said that 98% of the people in the world couldn't solve it.

Facts:
1. There are 5 houses (along the street) in 5 different colors: blue, green, red, white and yellow.
2. In each house lives a person of a different nationality: Brit, Dane, German, Norwegian and Swede.
3. These 5 owners drink a certain beverage: beer, coffee, milk, tea and water, smoke a certain brand of cigar: Blue Master, Dunhill, Pall Mall, Prince and blend, and keep a certain pet: cat, bird, dog, fish and horse.
4. No owners have the same pet, smoke the same brand of cigar, or drink the same beverage.

Hints:
1. The Brit lives in a red house.
2. The Swede keeps dogs as pets.
3. The Dane drinks tea.
4. The green house is on the left of the white house (next to it).
5. The green house owner drinks coffee.
6. The person who smokes Pall Mall rears birds.
7. The owner of the yellow house smokes Dunhill.
8. The man living in the house right in the center drinks milk.
9. The Norwegian lives in the first house.
10. The man who smokes blend lives next to the one who keeps cats.
11. The man who keeps horses lives next to the man who smokes Dunhill.
12. The owner who smokes Blue Master drinks beer.
13. The German smokes Prince.
14. The Norwegian lives next to the blue house.
15. The man who smokes blend has a neighbor who drinks water.

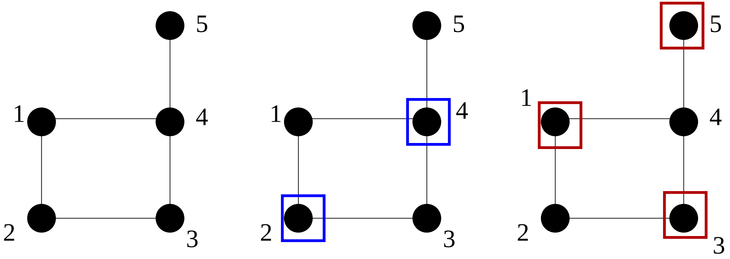


Who keeps the fish?

Examples of Max-2-CSP problems

Max-2-CSP (Each constraint concerns at most 2 variables)
is a **very general paradigm**,
includes most of the classical **graph theory problems**:

- Maximum bipartite subgraph (or Max-2-COL)
- Max-CUT
- Max-2-SAT, Max-2-XORSAT
- graph colorability...
- Maximum Independent Set (MIS).



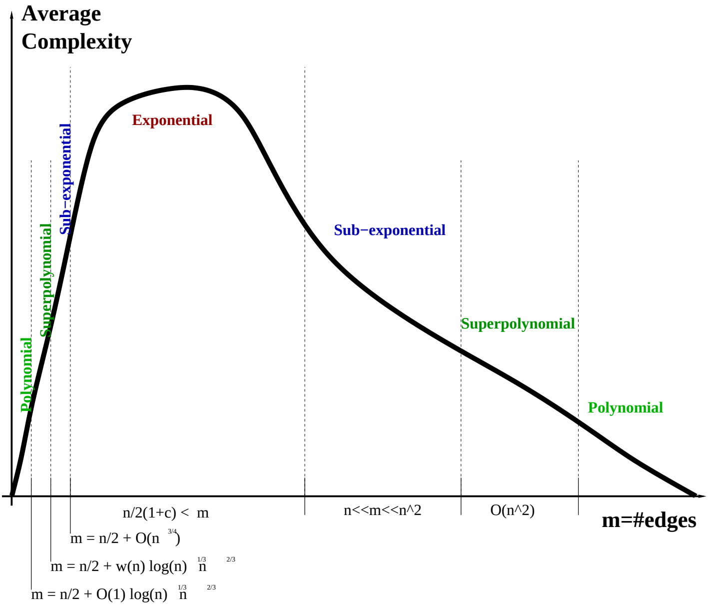
$\{2, 4\}$ is an IS

$\{1,3,5\}$ is an IS of MAXIMUM size

Motivations

1. Numerous applications (coloration, finances, codes, distributed systems, ...).
2. Typical NP-complete decision problem (cf. Karp's original list [KARP 72]).
3. Typical NP-HARD approximation problem [HAASTAD 99] .

Overview of our Results



Main ideas of the proof

1. **EAST SIDE** (=dense graphs).
   We analyze an exhaustive algorithm.
2. **WEST SIDE** (=sparse graphs).
   We analyze a very general algorithm working on all MAX-2-CSP. We use reductions (transformations of the underlying graph) mainly on vertices of degree < 3. We "**`branch`**" only on vertices of degree ≥ 3.

Remark.

In both cases, the algorithms are **exact** and **ALWAYS RETURN** the MIS. As input, the underlying graphs of the constraints are random $\mathbb{G}(n, m)$ and $\mathbb{G}(n, p)$. Our results quantify the average number of global iterations of the algorithms.

An exhaustive algorithm for the MIS

Procedure $\textsc{MaxIS}(G = (V, E) : \texttt{labeled graph})$
Pick the least vertex v in V ;

(⋆ EITHER v is not in the MIS $\longrightarrow$ remove it ⋆)
$V_1 := V \setminus \{v\}$;
$E_1 := E \setminus \{$ incident edges of $v\}$;
$\text{maxis}_1 := \text{Card}(\textsc{MaxIS}(G_1))$;

(⋆ OR v is in the MIS $\longrightarrow$ remove v and all its neighbors ⋆)
$V_2 := V \setminus \Gamma(v)$; (⋆ $\Gamma(v) = \{v\} \bigcup$ its neighbors ⋆)
$E_2 := E \setminus \{$ incident edges of the vertices in $\Gamma(v)\}$;
$\text{maxis}_2 := 1 + \text{Card}(\textsc{MaxIS}(G_2))$;

Return: $\max(\text{maxis}_1, \text{maxis}_2)$

Let $M_n =$ be the mean number of iterations of the algorithm whenever the input are graphs from $\mathbb{G}(n,p)$. We have

$$\mathbf{M_n = M_{n-1} + \sum_{i=0}^{n-1} \binom{n-1}{i} p^i (1-p)^{n-1-i} M_{n-1-i}}$$

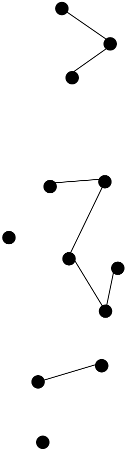
Theorem.

$$\mathbf{M_n} \sim \frac{\mathbf{e}^{(\rho/\mathbf{2})(\log(\mathbf{1}/\mathbf{r}))^{\mathbf{2}}}\mathbf{G}(\rho\log(\mathbf{1}/\mathbf{r}))}{\mathbf{r}^{\rho+\mathbf{1}/\mathbf{2}}\sqrt{\mathbf{2}\pi\rho\log(\mathbf{1}/\mathbf{r})}} \sim n^{O(\log n)},$$

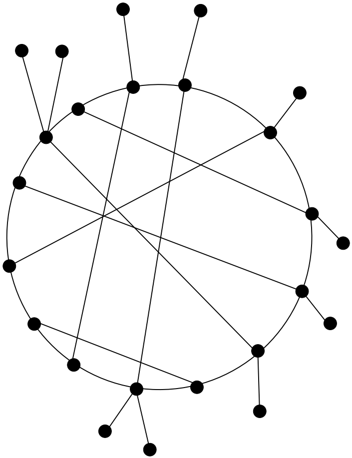
where $\rho = 1/\log(1/(1-p))$, $r = \frac{W(n/\rho)}{n/\rho}$ and G is continuous and 1-periodic function:

$$G(u) = q^{(\{u\}^2+\{u\})/2} \sum_{-\infty<j<\infty} \frac{q^{j(j+1)/2}}{1+q^j q^{-\{u\}}} q^{-(j+1)\{u\}}.$$

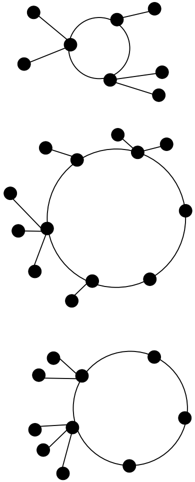
Sparse graphs: analysis of an algorithm for MAX-2-CSP



Set of trees

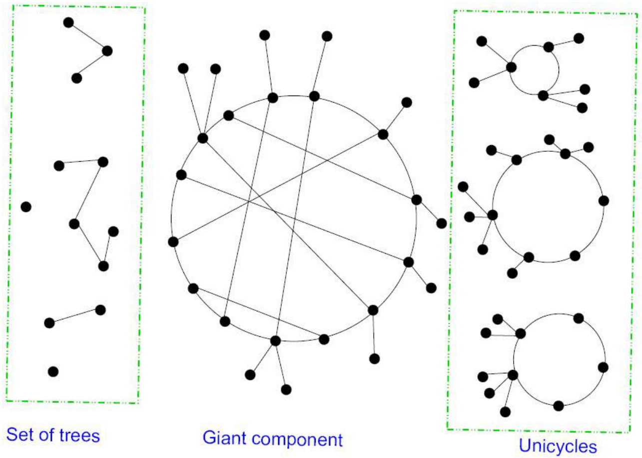


Giant component

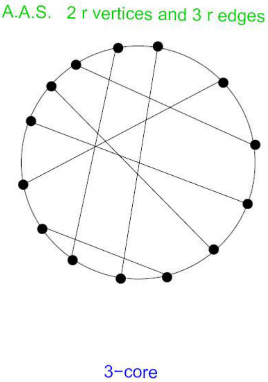


Unicycles

Sparse graphs: input graph before the reductions



- Reduce all vertices of degree 1 and 2. (as in [Tarjan 77]).
- These reductions cost polynomial time.
- Branch on vertices of degree ≥ 3.



- On such graphs, the cost is $< nK^r$ [Scott, Sorkin 03]
- The average cost is therefore upper bounded by

$$\sum_{r=0}^{m} nK^r \underbrace{\texttt{Proba}\,[\,\text{graph } n \text{ vertices}\,,\, m \text{ edges has excess } r\,]}_{\mathbf{p_r(n,m)}}\,.$$

Lemma. The proba. that a graph with n vertices and $m = \frac{n}{2} + \mu n^{2/3}$ edges has a connected component of excess $r = \frac{16}{3}\mu^3 + O(\mu^{3/2})$ is

$$\mathbf{p_r(n,m)} \sim \left(\frac{3}{160\pi\mu^3}\right)^{1/2} \exp\left(-\frac{3}{160}\frac{\left(r - \frac{16}{3}\mu^3\right)^2}{\mu^3}\right)$$

(with uniform error terms as $r,\ \mu,\ n \to \infty$ et $|r - \frac{16}{3}\mu^3| \leq O(\mu^{3/2})$, $r \leq O(n^{1/4})$ and $\mu \leq O(n^{1/12})$).

Proof. Generating functions ! (below $T \equiv T(z) = ze^{T(z)}$)

$$p_r(n,m) \sim \frac{n!}{\binom{\binom{n}{2}}{m}} [z^n] \underbrace{\frac{(T(z) - T(z)^2/2)^{n-m}}{(n-m)!}}_{\text{trees}} \underbrace{\frac{e^{-T(z)/2 - T(z)^2/4}}{(1-T(z))^{1/2}}}_{\text{unicycles}} \underbrace{W_r(z)}_{\text{giant comp}},$$

where W_r is Wright's GF for graphs of excess r

(+**saddle-point method** [KNUTH-JANSON-ŁUCZAK-PITTEL 93, FLAJOLET-SEDGEWICK 09]).

Sparse graphs: results

Recall that the desired average cost of the reduction algorithm is

$$\sum_{r=0}^{m} nK^r \underbrace{\texttt{Proba}\,[\text{ graph } n \text{ vertices },\, m \text{ edges has excess } r]}_{\mathbf{p_r(n,m)}} \; .$$

Theorem. Let $m = \frac{n}{2} + \mu n^{2/3}$ with $\mu = o(n^{1/3})$. The average cost of all MAX-2-CSP problems (and not only MIS) on $\mathbb{G}(n, m)$ is at most

$$\exp\left(\frac{\mathbf{2}\log\mathbf{2}}{\mathbf{3}}\,(\mathbf{5}\log\mathbf{2}+\mathbf{4})\,\mu^{\mathbf{3}}\right)\left(\mathbf{1}+\mathbf{o}(\mathbf{1})\right).$$

Robustness of our classifications

- These very **simple** algorithms are amongst the fastest ones.
- Our classifications are **robust**:
  1. Most of existing algorithms are "**Tarjan-Chvàtal-like**":

$$\mathbf{T(n) \leq T(n-6) + T(n-4) \longrightarrow O((1+\varepsilon)^n)}\,.$$

  2. Even if one day a genious finds a fast algorithm leading to a recurrence of the type

$$\mathbf{T(n) = T(n-A) + T(\lceil cn \rceil)}, \qquad A \text{ fixed and } c < 1$$

(cf Hardy-Ramanujan work on partitions, or AIMD protocol for TCP/IP) or

$$\mathbf{T(n) = T(n-A) + \sum_{i=0}^{n-A-1} \binom{n-A-1}{i} p^i (1-p)^{n-A-1} T(i)}\,,$$

all of them are expected to lead to a superpolynomial complexity $n^{\ln n}$!!!!

Open problems

1. A WELL-KNOWN and LONGSTANDING PROBLEM (cf. [KARP] and [FRIEZE] 70++):

   Is there a POLYNOMIAL TIME algorithm that finds the MIS in $\mathbb{G}(n, p = 1/2)$?

2. OUR WORK suggests an intermediate CHALLENGE ($$$):

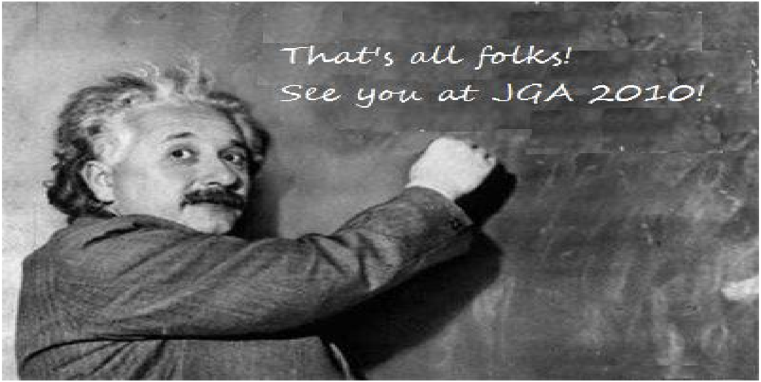
   Is there a $n^{o(\log n)} = n^{\text{LITTLE-OH}(\log n)}$ algorithm which finds the MIS in $\mathbb{G}(n, p = 1/2)$?

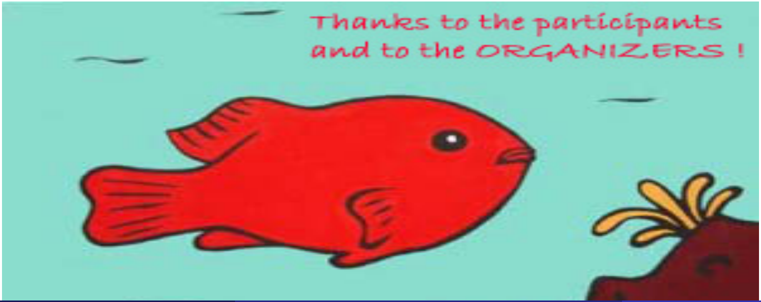
   Recall that the Metropolis algorithm [JERRUM] "works" in $n^{O(\log n)} = n^{\text{BIG-OH}(\log n)}$.

Conclusion

Nice application of enumerative/analytic approaches on decision/optimization problems and phase transitions

- 2-SAT [BOLLOBÀS, BORGS, CHAYES, KIM, WILSON 01]
- 2-XORSAT [DAUDÉ, RAVELOMANANA 09]
- Airy distribution [BANDERIER, FLAJOLET, SCHAEFFER, SORIA, 01]
- links with statistical mechanics [MONASSON 96-09, BRAUNSTEIN, MÉZARD, ZECCHINA 05, MOORE, KIRKPATRICK]
- ... your favorite algo/problems ?!


That's all folks!
See you at JGA 2010!


Thanks to the participants
and to the ORGANIZERS !