The problem P vs NP

Efficient computation, Internet security,
And the limits of the human knowledge

Linda Pagli
Dipartimento di Informatica
Università di Pisa

Clay Math Institute
Millennium problems $1M each

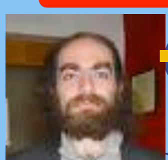
- Birch and Swinnerton-Dyer Conjecture
- Hodge Conjecture
- Navier-Stokes Equations
- P vs NP

- ~~Conjecture di Poincaré~~
- Riemann Hypothesis
- Yang-Mills Theory

Clay Math Institute
Millennium problems $1M each

- Birch and Swinnerton-Dyer Conjecture
- Hodge Conjecture
- Navier-Stokes Equations
- P vs Np

  Most recent 1971

  Easiest to explain
-  ~~Congettura di Poincaré~~
- Riemann Hypothesis
- Yang-Mills Theory

Introduction

Computers are very fast.
But certain problems still take too long!

We begin with a simple example...

A simple example

$7 \times 13 = ?$

Multiplication problem
(Answer is 91)

Another simple example

$? \times ? = 91$

"Factoring problem"

(Answer is: 7×13)

A bigger moltiplication example

1.634.733.645.809.253.848
443.133.883.865.090.859.
841.783.670.033.092.312.
181.110.842.389.333.100.
104.508.151.212.118.167.
511.579

X

1.900.871.281.664.822.113.
126.851.573.935.413.975
471.896.789.968.515.493.
666.638.539.088.027.103.
802.104.498.957.191.261.
465.571

= ?

The answer is:

3.107.418.240.490.043.721.350.750.035.888.567.930.037.346.022.842.727.
545.720.161.948.823.206.440.518.081.504.556.346.829.671.723.286.782.
437.916.272.838.033.415.471.073.108.501.919.548.529.007.337.724.822.
783.525.742.386.454.014.691.736.602.477.652.346.609

Took less than a second of computer time to find

A bigger factoring example

? X ? = 3.107.418.240.490.043.721.350.750.035.888.567.930.037.
346.022.842.727.545.720.161.948.823.206.440.518.081.
504.556.346.829.671.723.286.782.437.916.272.838.033.
415.471.073.108.501.919.548.529.007.337.724.822.
783.525.742.386.454.014.691.736.602.477.652.346.609

The answer is:

1.634.733.645.809.253.848
443.133.883.865.090.859.
841.783.670.033.092.312.
181.110.842.389.333.100.
104.508.151.212.118.167.
511.579

X

1.900.871.281.664.822.113.
126.851.573.935.413.975
471.896.789.968.515.493.
666.638.539.088.027.103.
802.104.498.957.191.261.
465.571

Took more than 20 computer years of effort to find

For $30.000 find factors:

74037563479561712828046796097429573142593188889231289084
93623263897276503402826627689199641962511784399589433050
212758537011896809828673317327310893090055250511687706329
9072396380786710086096962537934650563796359

212 digit number: RSA-704.

See the RSA Factoring Challenge for details and payments

Competition closed in 2007: nobody won the prize.

Factoring is an ingredient in modern cryptography

To open this locker few seconds are enough if you know the 4 digits PIN

Without PIN, 10.000 attempts in the worst case to open it.

Given 2 prime numbers p, q

Computing $n = p \times q$ " easy"

Computing p, q from n " difficult"

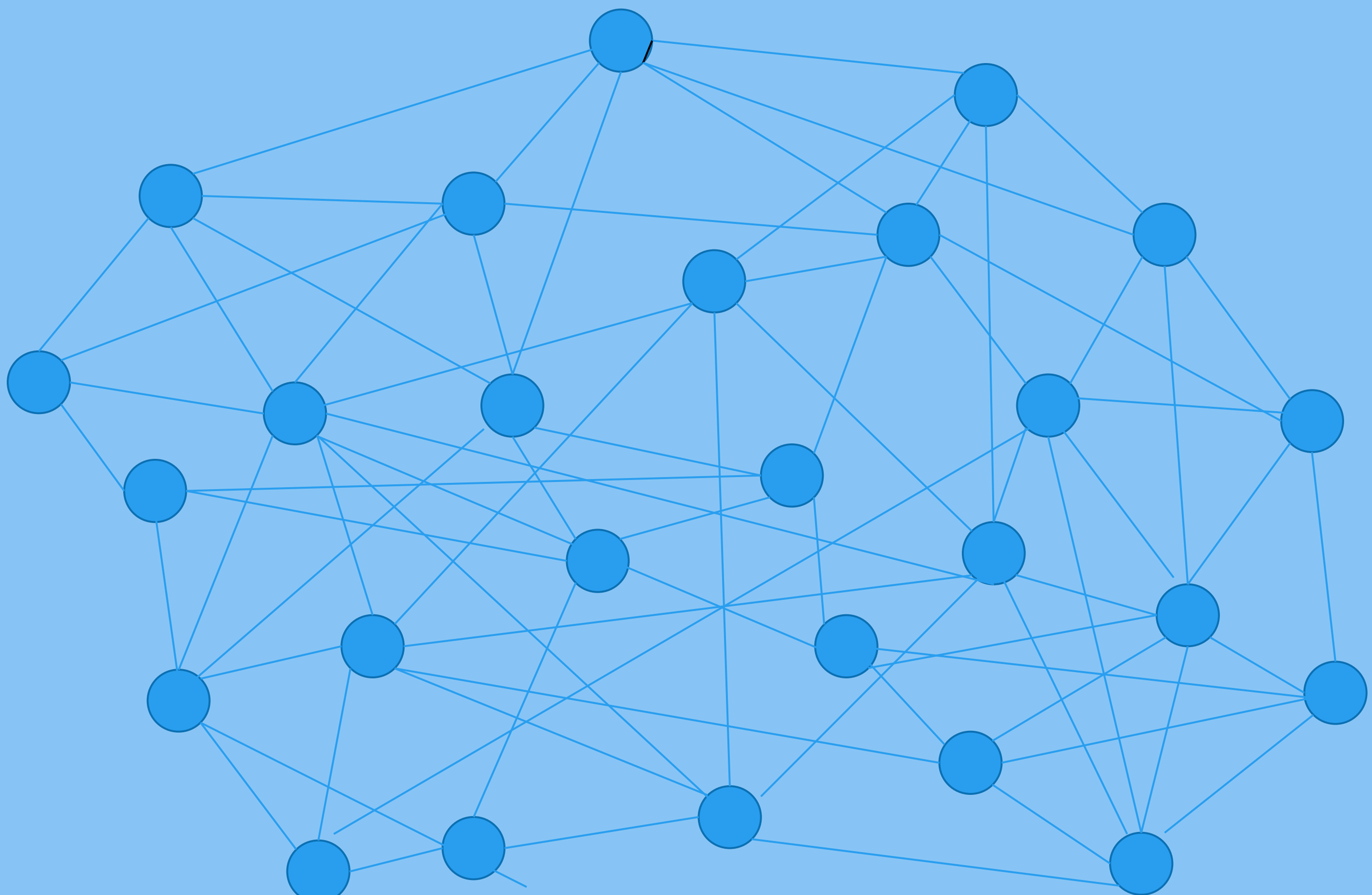
we have to try with all factors of n that, as for the locker are exponential in the number n of digits.

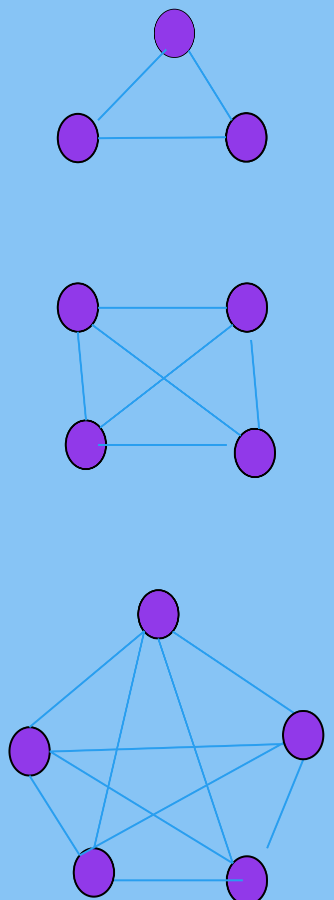
Brute Force Search: very slow when the search space is huge

Is searching necessary?

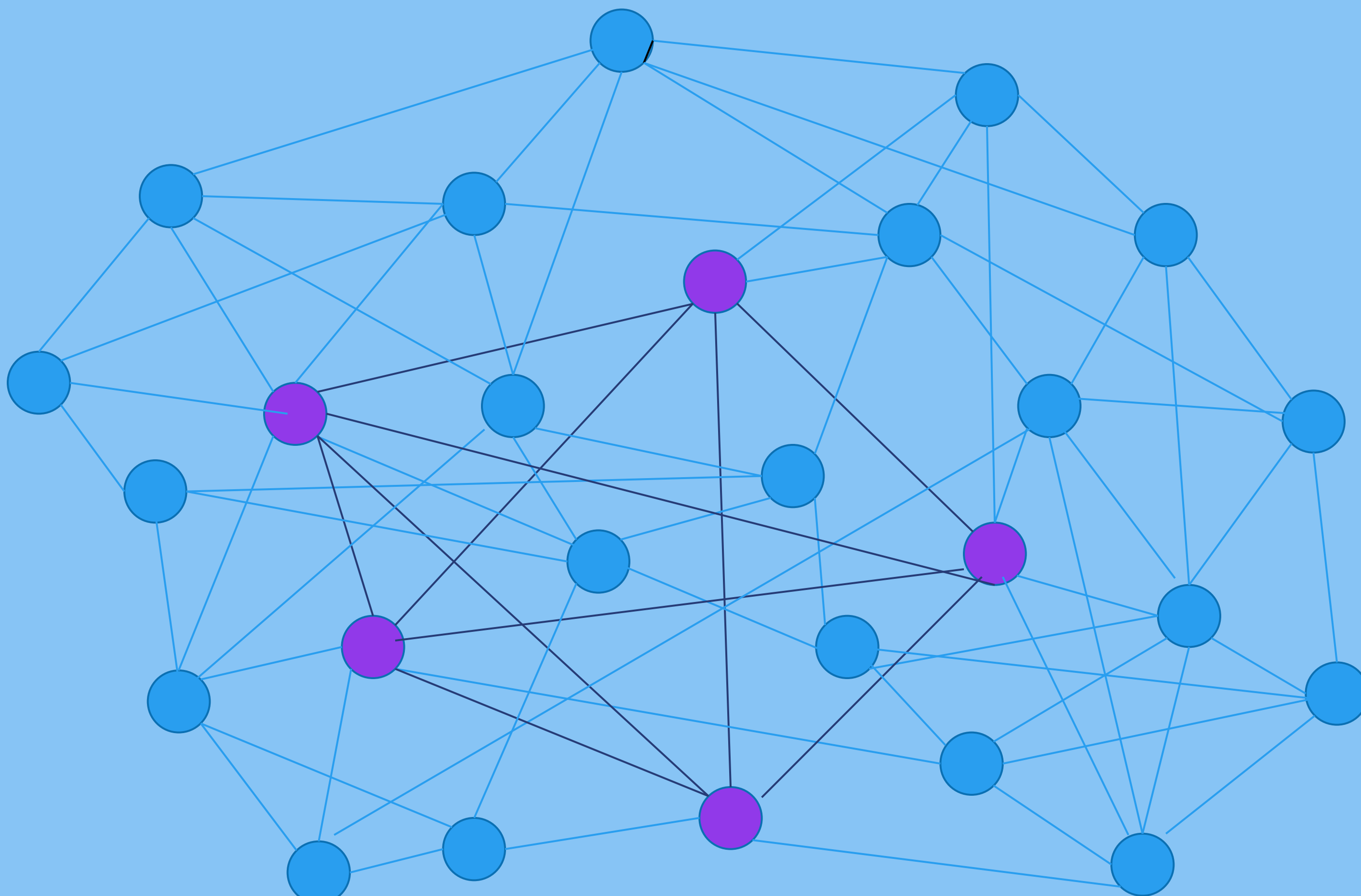
We are not able to answer.

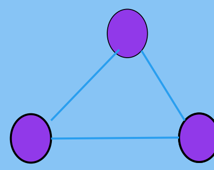
CLIQUE problem

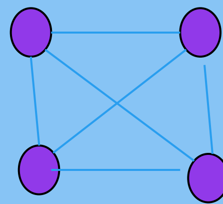


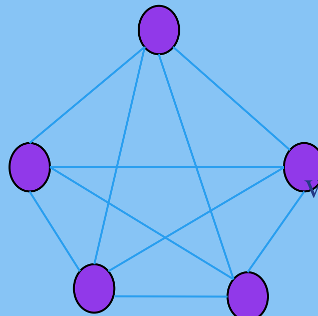


Problema della CLIQUE



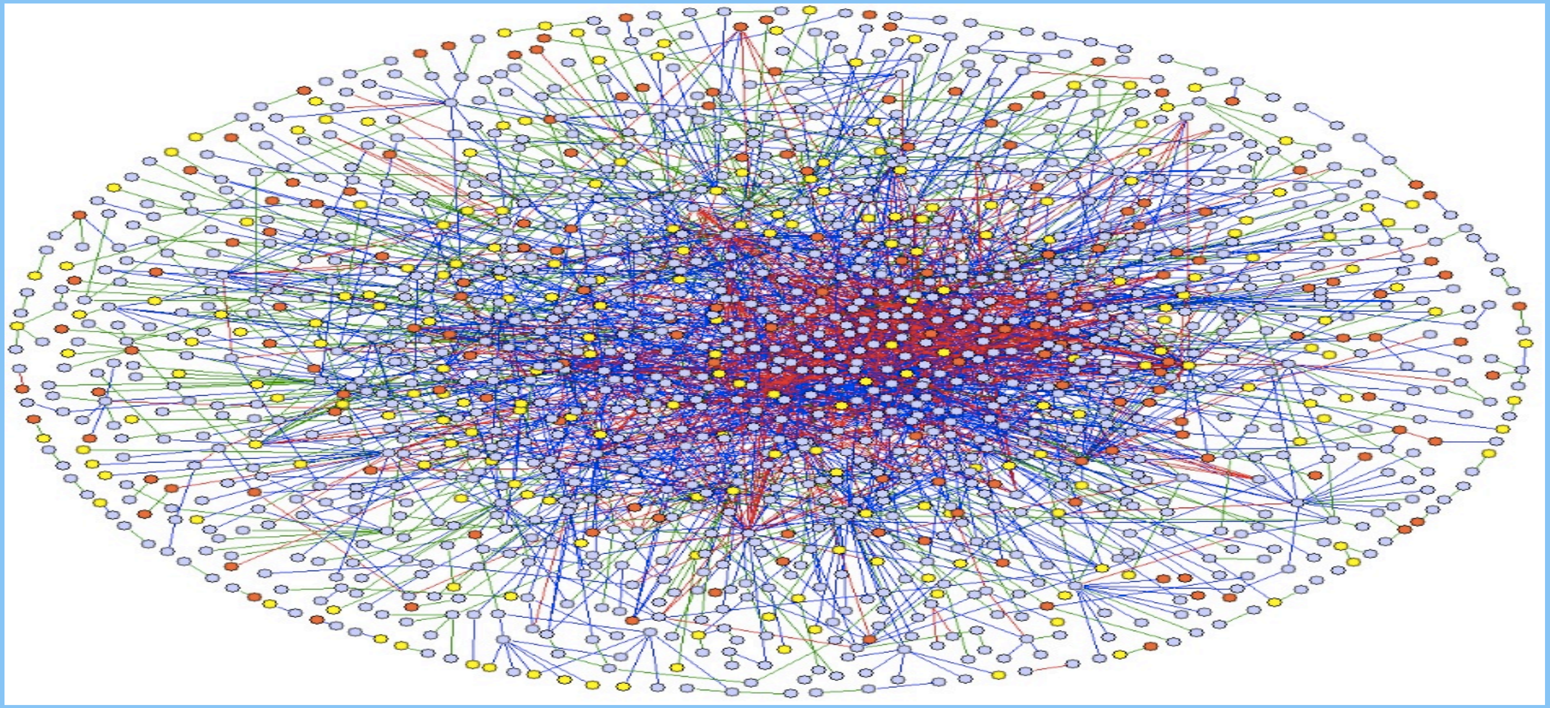






A bigger CLIQUE problem

Finding the largest clique in a big graph may take centuries of computing time!



La ricerca esaustiva è necessaria ?

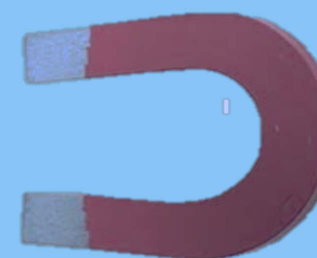
Non lo sappiamo.

Needle in Haystack problem

Found it! Took only ten days!

Finding the needle.....

Is searching necessary?



No, if we have a magnet

Other search problems

- Scheduling
- Map coloring
- Protein folding
- Graphs hysomophysm
- Puzzles (Sudoku)
- Traveling salesman
- Many others….

The P versus NP question

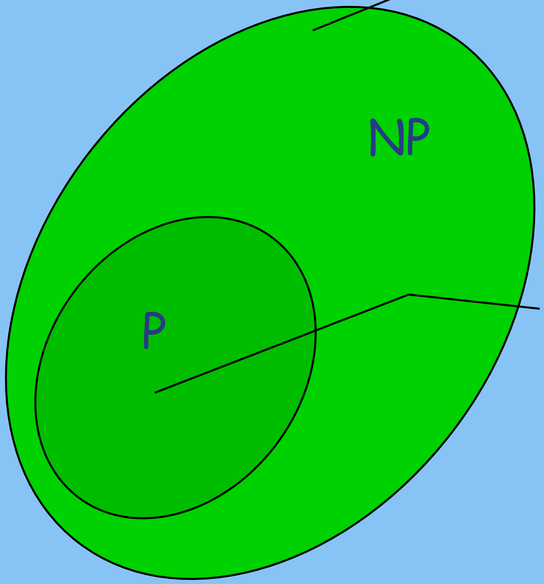
Can we solve the search problems without searching?

P and NP

- P "Polynomial time"
  Quickly solvable problems
- NP "Non deterministic Polynomial time"
  Quickly verifiable problems

  includes the search problems

Le classi P e NP



Easily verifiable problems

- Factoring
- Clique

Easily solvable problems

- Multiplication
- Sorting

P = NP

or

P ≠ NP

Recent history of the question P vs NP

- 1960 Dawn of complexity theory
  - Rabin, Blum, Hartmanis, Edmonds

- 1970 The question P vs NP; NP-completeness

  Cook, Levin, Karp

- 1956 Gödel writes to Von Neuman

  (discovered in the 90)

  - Remarkable letter forshadows P vs NP

Sometimes brute-force search can be avoided

A strange way to test primality

Old theorem. For a prime p and $a < p$:

$$a^{p-1} = 1 \pmod{p}$$

Examples:

p=7, a=2: $2^6 = 64 = 1 \pmod{7}$

p=15, a=2: $2^{14} = 16.384 = 4 \neq 1 \pmod{15}$

15 is not prime

NP-completeness

If Clique is in P then P = NP

NP-completeness

NP-complete problems :
If one is easy all are easy!
If one is difficult all are difficult!

Clique:	NP-complete
Map coloring:	NP-complete
Factoring:	open

Plenty of problems NP-complete known in Mathematics, Biology, Phyisics, Economy,….

Protein Engineering vol. 7 no. 9 pp. 1059-1068, 1994

The protein threading problem with sequence amino acid interaction preferences is NP-complete

Richard H. Lathrop

Economic Theory vol. 23, no. 2 , pp. 445-454, 2004

Finding a Nash equilibrium in spatial games is NP-complete

R. Baron, J. Durieu, H. Haller and P. Solal

[math.GR] arXiv:0802.3839v1

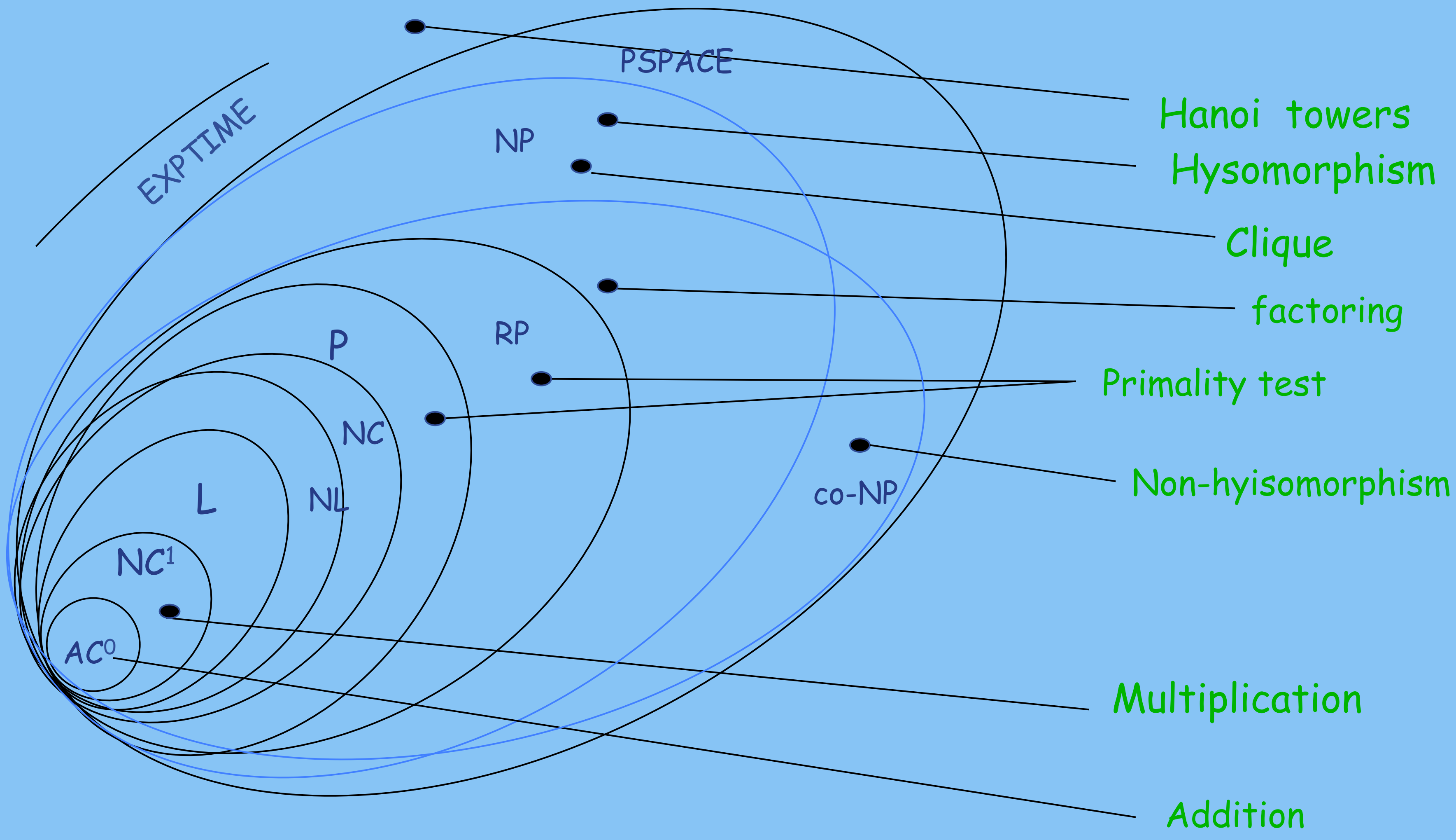
Quadratic equations over free groups are NP-complete

O. Kharlampovich, I.G. Lysenok, A G Myasnikov N. Touikan

NP-completeness: stamp of difficulty

Potential guide towards better models and theories

Complexity classes Problemi:



How to prove $P \neq NP$?
Why is so hard to prove it

- Algorithms are very sophisticated
- We should prove that all the possible solution strategies fail!

- Possible ways

  Limiting the capabilities of the machine

  Discover difficult inputs

  very large inputs

What happens in nature?

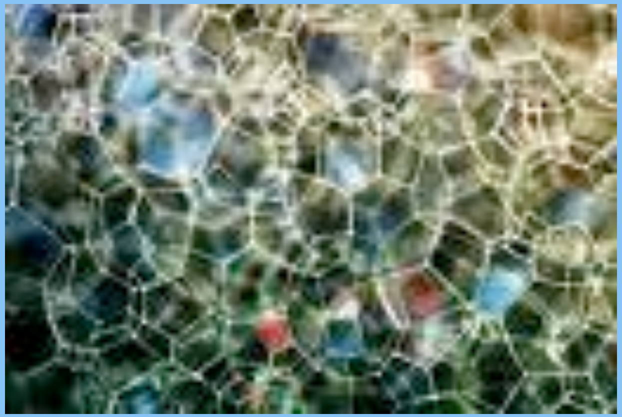
NP-complete problems "solved" by the nature

Biology: Protein Folding

Minimum energy

Fisica: Foam

Minimum surface



Economy: Nash equilibrium di Nash in strategic games

Possibilities:

wrong model, or special inputs, or ⋯ P=NP

News: Natural Sciences ⟷ Informatics

Positive consequences of P≠NP

P≠NP Some of the problems that we want to solve are difficult.

Are difficult problems useful?

Crittografia: If factoring is difficult:

- Coding
- Digital signature
- Secure E-mail
- Electronic commerce
- Shopping on-line
- Poker on-line

Will it ever be solved ?

We need new ideas


THANK
YOU
return