

A dicotomia polinomial versus NP-completo de problemas desafiadores em teoria dos grafos

Celina Miraglia Herrera de Figueiredo

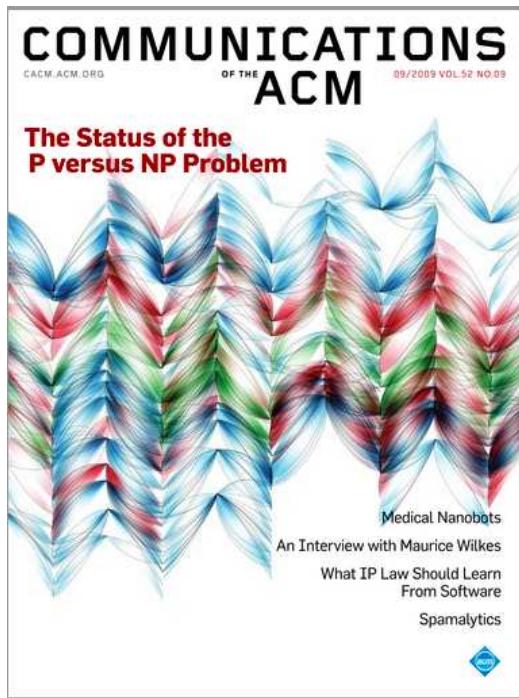
Programa de Engenharia de Sistemas e Computação
COPPE, Universidade Federal do Rio de Janeiro

Outubro 2011

O Problema do Milênio

Problema central em Teoria da Computação: P versus NP

Existe pergunta cuja resposta pode ser verificada rapidamente, mas cuja resposta requer muito tempo para ser encontrada?



setembro 2009

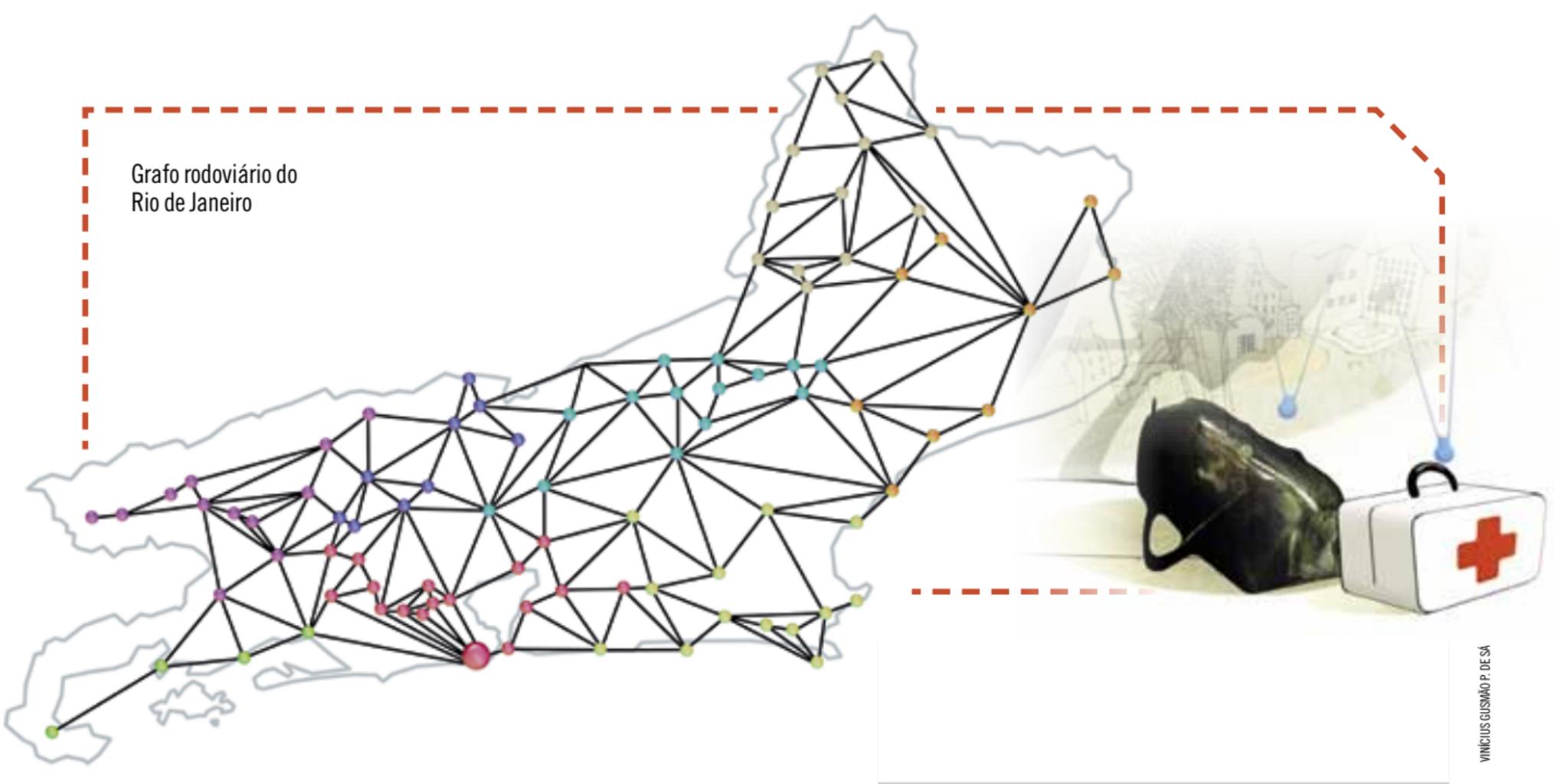
RESOLVER OU VERIFICAR?

UMA PERGUNTA QUE VALE UM MILHÃO DE DÓLARES



RESOLVER É MAIS FÁCIL QUE VERIFICAR?

Em 1903, em um congresso da Sociedade Norte-americana de Matemática, o matemático Frank Cole (1861-1926) provou que o número $2^{67} - 1 = 147.573.952.589.676.412.927$ não é primo, exibindo a fatoração $193.707.721 \times 761.838.257.287$. Quando apresentou essa fatoração, Cole fez a multiplicação desses dois números enormes no quadro e em silêncio, sendo ao final aplaudido de pé. É simples – embora tedioso, se feito manualmente – calcular $2^{67} - 1$, multiplicar $193.707.721$ por $761.838.257.287$ e verificar que dão o mesmo número. Já encontrar essa fatoração é difícil. Cole disse que ele levou três anos trabalhando aos domingos.



Complexidade computacional

A maioria dos problemas computacionais pertence à classe NP,
admitem um certificado polinomial

Em várias e diferentes áreas, procuramos objetos matemáticos:
percurso de um caixeiro viajante, atribuição de verdade,
emparelhamento máximo, coloração mínima de um grafo

O objeto matemático procurado é o certificado,
a prova de que o problema está em NP

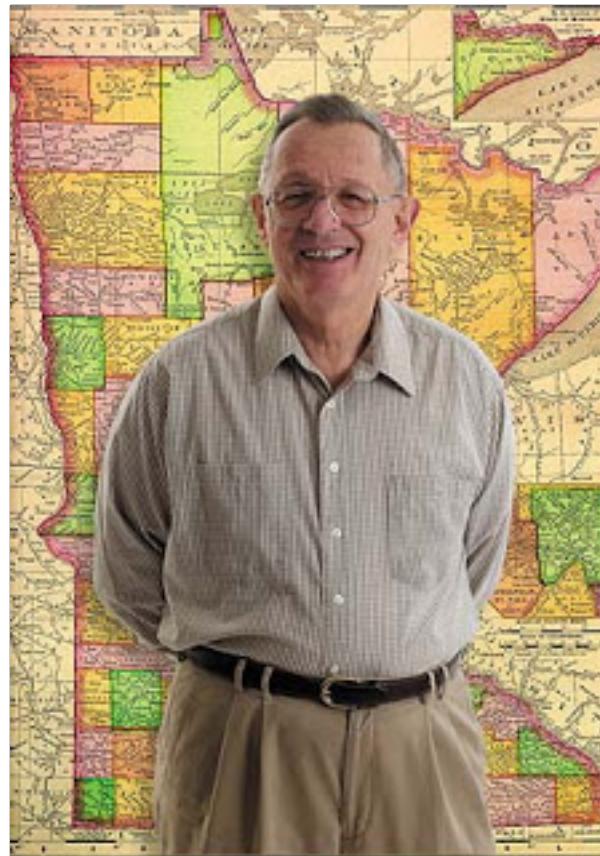
O estudo da complexidade computacional de problemas considera principalmente problemas em NP e tenta distinguir os solúveis em tempo polinomial dos não através da classe dos problemas NP-completos

C. Papadimitriou – *Computational Complexity* 1994

Bastam Quatro Cores

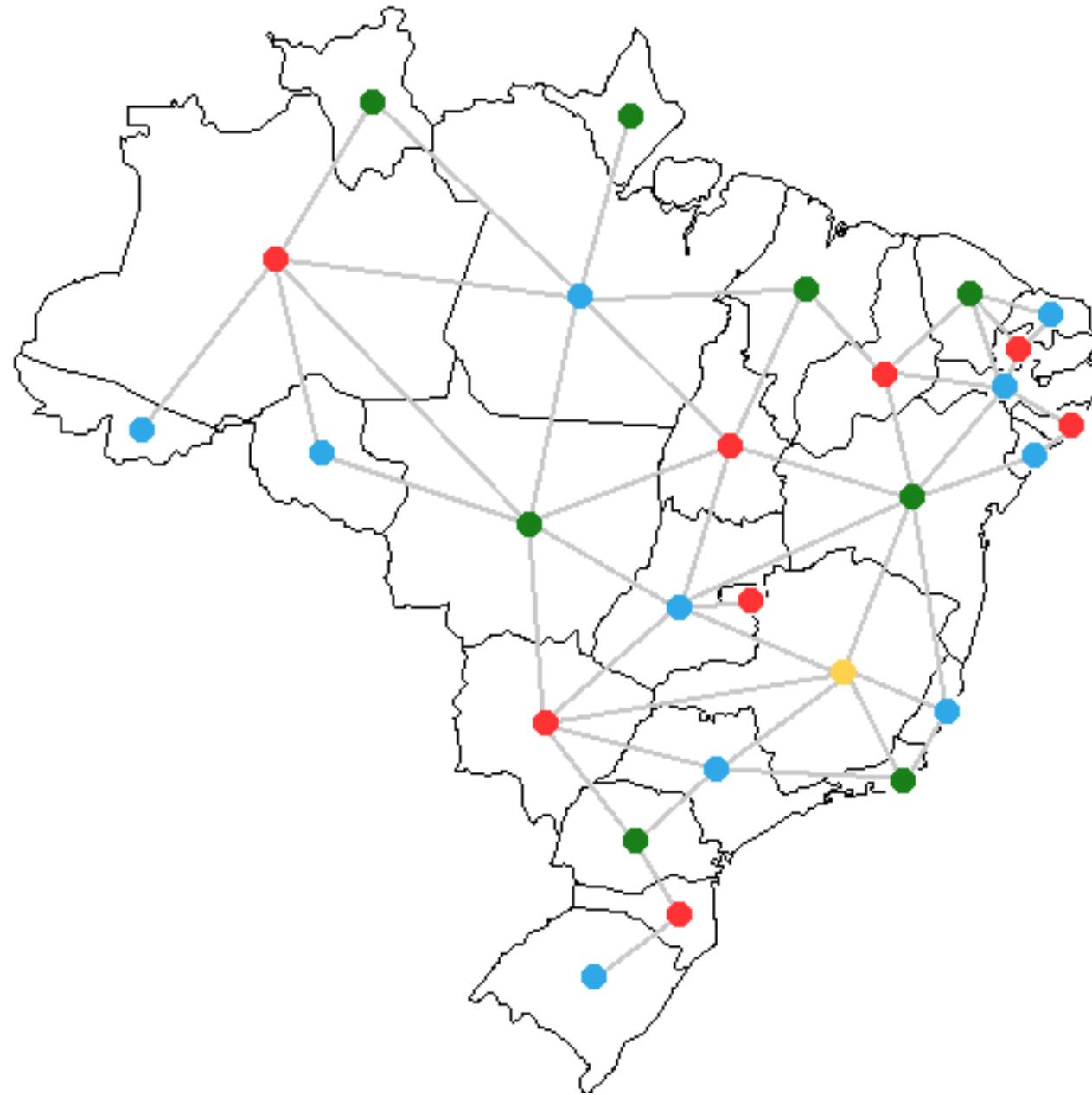
Celina Miraglia Herrera de Figueiredo
Programa de Engenharia de Sistemas e Computação





Kenneth Appel
1932–2013





Minha contribuição

Que característica de um determinado problema o classifica como polinomial ou NP-completo?

Classificação em P ou NP-completo de dois problemas históricos em teoria dos grafos

Problema cuja complexidade separa classes de grafos
Classe de grafos que separa a complexidade de problemas

Três dicotomias cheias:
todo problema é classificado em P ou NP-completo

“The P vs. NP-complete dichotomy of some challenging problems in graph theory”

Discrete Appl. Math. 2011

(plenária no Latin-American Algorithms, Graphs and Optimization Symposium 2009)

Guia de NP-completude

Identificação de problema interessante, de classe de grafos interessante

Classificação da complexidade de um problema: P ou NP-completo

Problema que separa classes de grafos

Classe de grafos que separa problemas

D. Johnson – *J. Algorithms* 1985, *ACM Trans. Algorithms* 2005

M. Golumbic, H. Kaplan, R. Shamir – *J. Algorithms* 1995

J. Spinrad – *Efficient Graph Representations* 2003

Problemas separadores e Classes de grafos separadoras

	VERTEXCOL	EDGECOL	MAXCUT
perfect	P	N	N
chordal	P	O	N
split	P	O	N
strongly chordal	P	O	O
comparability	P	N	O
bipartite	P	P	P
permutation	P	O	O
cographs	P	O	P
indifference	P	O	O
split-indifference	P	P	P

N: NP-completo P: polinomial O: aberto

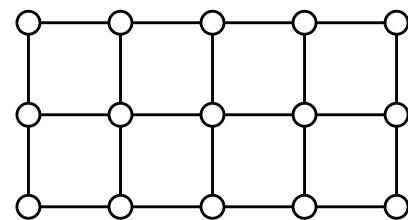
D. Johnson – *J. Algorithms* 1985

J. Spinrad – *Efficient Graph Representations* 2003

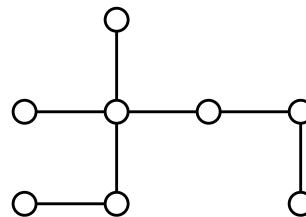
Imersão em grades

Teoria dos grafos: Reconhecimento das grades parciais é problema aberto

Desenho de grafos: Decidir se um grafo admite um VLSI layout com arestas unitárias é NP-completo



a grade $G_{3,5}$



imersão para $\{1, 2, 4\}$ -árvore

A. Brandstädt, V.B. Le, et al. – Information system on graph class inclusions.

<http://wwwteo.informatik.uni-rostock.de/isgci/>, 2002, atualizado 2010

S. N. Bhatt, S. S. Cosmadakis – *Inform. Process. Lett.* 1987

P vs. N dicotomia para grades parciais restritas por graus

D	D-grafos	D-árvores
{1}	P	P
{2}	P	—
{3}	P	—
{4}	P	—
{1,2}	P	P
{1,3}	N	N
{1,4}	P	P
{2,3}	N	—

D	D-grafos	D-árvores
{2,4}	N	—
{3,4}	P	—
{1,2,3}	N [G89]	N [G89]
{1,2,4}	N [BC87]	N [BC87]
{1,3,4}	N	N
{2,3,4}	N	—
{1,2,3,4}	N [BC87]	N [BC87]

Não há conjunto D que defina problema separador

S. N. Bhatt, S. S. Cosmadakis – *Inform. Process. Lett.* 1987

A. Gregori – *Inform. Process. Lett.* 1989

“Complexity dichotomy on partial grid recognition”

Theoret. Comput. Sci. 2011 (com Vinícius Sá, Guilherme Fonseca, Raphael Machado)

Referências

“On edge-colouring indifference graphs”

Theoret. Comput. Sci. 1997 (com João Meidanis, Célia Mello)

“Finding skew partitions efficiently”

J. Algorithms 2000 (com Sulamita Klein, Yoshiharu Kohayakawa, Bruce Reed)

“Optimizing bull-free perfect graphs”

SIAM J. Discrete Math. 2004 (com Frédéric Maffray)

“The complexity of clique graph recognition”

Theoret. Comput. Sci. 2009 (com Liliana Alcon, Luerbio Faria, Marisa Gutierrez)

“The polynomial dichotomy for three nonempty part sandwich problems”

Discrete Appl. Math. 2009 (com Rafael Teixeira, Simone Dantas)

“Chromatic index of graphs with no cycle with a unique chord”

Theoret. Comput. Sci. 2010 (com Raphael Machado, Kristina Vušković)

“Complexity dichotomy on partial grid recognition”

Theoret. Comput. Sci. 2011 (com Vinícius Sá, Guilherme Fonseca, Raphael Machado)