O jogo Sudoku, pré-coloração estendida em hipergrafos e algoritmos de enumeração implícita

Lázaro Albuquerque, Rosiane de Freitas
[1]Instituto de Computação - Universidade Federal do Amazonas (IComp - UFAM)
Manaus - Amazonas - Brasil
{lta,rosiane}@icomp.ufam.edu.br

RESUMO

Este trabalho é resultado de um projeto de pesquisa do Programa Jovens Talentos para a Ciência do CNPq, período 2012/2013, projeto ainda em vigência nos meses finais, do qual o autor foi selecionado e é bolsista. O jogo japonês Sudoku é investigado, consistindo de um quebra-cabeças, ou passatempo, numérico em uma grade de 9×9 células, onde algumas já estão preenchidas contendo um valor de 1 a 9. O objetivo é completar o preenchimento de todo o grid, de tal forma que cada linha, cada coluna e cada sub-grid de 3x3 células, contenha os valores de 1 a 9 exatamente uma vez. A teoria matemática e computacional envolvida é discutida, onde o mesmo pode ser modelado como um problema de coloração em grafos. Sendo assim, tal teoria em grafos é apresentada, a prova da NP-completude e proposto um modelo baseado em um problema de pré-coloração estendida em um hipergrafo 9-colorido, e algoritmos de resolução, principalmente algoritmos exatos de enumeração implícita, aplicando manipulação de bits na estrutura de representação do grafo, de tal forma a eliminar a verificação de um grande sub-conjunto de casos possíveis, possibilitando a resolução de versões difíceis do jogo Sudoku.
Palavras-chave: algoritmos, coloração em grafos, jogo matemático, teoria de grafos.
Área principal: Otimização Combinatória.

ABSTRACT

This work is the result of a research project of "Young Talent for Science Program"of CNPq brazilian agency, 2012/2013, project still in effect in its final months, of which the author is a fellow. Sudoku Japanese puzzle is investigated, consisting of a 9×9 grid in which some of the entries of the grid have a number from 1 to 9. One is then required to complete the grid in such a way that every row, every column, and every one of the nine 3×3 sub-grids contain the digits from 1 to 9 exactly once. The mathematical and computational theory involved is discussed, where it can be modeled as a graph coloring problem. Thus, the graph theory is presented, the proof of NP-completeness and is proposed a model based on a precoloring extension problem on a 9-colored hypergraph, with some algorithms, mainly exact algorithms of implicit enumeration, applying bit manipulation of the graph structure, in such a way as to eliminate the verification of a large subset of possible cases, enabling the resolution of any version of Sudoku game, including the hardest levels in an acceptable execution time.

Keywords: algorithms, graph coloring, graph theory, logical puzzle.
Main area: Combinatorial Optimization.

1 Introdução

O jogo Sudoku, apesar de ser um quebra-cabeças de lógica numérica, tornou-se muito popular em todo o mundo em um período de tempo relativamente curto, podendo ser encontrado facilmente nas bancas de revistas, bem como em sites de entretenimento e jogos educativos. O interessante é que entender como se joga o Sudoku é fácil, mas, por trás desta aparente simplicidade está um complexo problema matemático computacional, com bastante conhecimento estabelecido mas muito ainda por desvendar. Neste trabalho, o lado teórico do Sudoku é investigado, onde suas regras e níveis de dificuldade são apresentados, sua correlação com teoria dos grafos estudada e sendo proposto um modelo de coloração de vértices em hipergrafos com o desenvolvimento de um algoritmo exato de enumeração implícita que resolve os Sudokus de nível mais alto de dificuldade em um tempo computacional aceitável.

O restante deste artigo está organizado como segue. Na Seção 2.1, o jogo matemático Sudoku é apresentado e definido formalmente. Na Seção 3, mostra-se a relação entre o Sudoku, quadrados mágicos e latinos, bem como sua correlação com problemas de coloração de vértices em grafos, que são modelos bastante utilizados na literatura para o mesmo, com a proposta de um novo modelo. Na Seção 4, são apresentados dois dos algoritmos desenvolvidos para a resolução do problema, um algoritmo de enumeração explícita aplicando a técnica de *backtracking*, e uma versão refinada do mesmo, envolvendo manipulação de bits, que possibilita a diminuição de um grande número de verificações e que, consequentemente, diminui o tempo de execução e possibilita a resolução de versões do Sudoku com níveis mais difíceis em um tempo computacional aceitável, mesmo usando laptops de uso pessoal com configuração básica. Na Seção **??**, mostra-se um sumário das contribuições feitas neste trabalho. Por fim, na Seção **??**, são dadas considerações finais sobre o trabalho e os próximos passos para futuras pesquisas sobre o tema.

2 O jogo Sudoku

O jogo matemático Sudoku é formado por um quadrado bidimensional, possuindo, em sua versão clássica, um total de $9 \times 9 = 81$ células ou casas (nove linhas e nove colunas), agrupadas, por sua vez, em nove quadrados menores (caixas ou subgrades) com nove casas cada um. O jogo começa com algumas casas já preenchidas por números, algarismos de 1 a 9, onde o jogador deve completar as casas restantes também usando algarismos de 1 a 9, de tal forma que nenhum número se repita na mesma coluna ou linha, nem dentro da mesma subgrade. A Figura 1 apresenta exemplos do jogo Sudoku, onde no primeiro quadrado ou grade 9×9 está o jogo em seu estágio inicial e, em seguida, na grade à direita, o jogo ao final, com todas as casas preenchidas com números de 1 a 9, sem repetição em linhas, colunas ou sub-grades de 3×3.

Pode ser observado que o jogo Sudoku padrão é similar a um quadrado latino (do inglês, *Latim square*) de ordem 9, diferindo apenas por conta da exigência de que cada subgrade de nove casas contenha os números de 1 a 9. Também é interessante ressaltar que, embora envolva números, o jogo Sudoku não exige conhecimento matemático aprofundado. Para se preencher a grade, não é necessário aplicar nenhuma operação matemática, e sim apenas usar o raciocínio lógico para distribuir algarismos de 1 a 9 pelas casas, respeitando-se as regras do jogo. Entretanto, o raciocínio requerido para alcançar a solução pode ser complexo e de difícil registro por um ser humano normal, possuindo, então,

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

Figura 1. Exemplo de um jogo Sudoku clássico: quadrado bidimensional 9×9, inicialmente com apenas algumas casas preenchidas e, na figura à direita, com tudo preenchido indicando final de jogo.

importantes desafios a matemáticos e computacionais, o que o torna uma interessante fonte de pesquisa científica.

O nome Sudoku (ou separadamente "Su Doku") é um termo japonês, uma abreviação para a frase "*suuji wa dokushin ni kagiru*", que significa algo como "os dígitos devem permanecer únicos", ou somente, "único número". A restrição geral que indica que "cada algarismo aparece em cada linha, coluna e bloco"é chamada de "regra única"(do inglês, *one rule*). Os valores iniciais do jogo são conhecidos como "números dados"ou pistas (do inglês, *givens* ou *clues*), tal como uma linha é também conhecida como "banda"(do inglês, *band*) e cada coluna como "pilha"(do inglês, *stack*). Os algarismos de 1 a 9 são usados por comodidade, pois as relações aritméticas entre os numerais são irrelevantes (não requer cálculos matemáticos). Assim, qualquer combinação de símbolos distintos como letras, formas, ou cores, podem ser usadas no jogo sem alterar as regras. Por exemplo, algumas variações usam letras, como as versões conhecidas como *Scramblets*, da Penny Press, e *Sudoku Words* da Knight Features Syndicate.

Segundo a literatura, o jogo com a proposta do Sudoku foi, na verdade, criado no ocidente, na década de setenta, projetado pelo arquiteto aposentado Howard Garns, já com mais de setenta anos, que projetava jogos de quebra-cabeça e passatempos independentemente. Dada a semelhança, possivelmente tenha se baseado no quadrado latino, uma construção matemática criada pelo cientista suíço Leonhard Euler, no século XVIII. O jogo de Garns consistia em uma grade parcialmente preenchida onde deveria se preencher os quadros vazios restantes. As primeiras publicações foram na revista norte-americana "*Math Puzzles and Logic Problems*", da editora *Dell Magazines*, especializada em desafios e quebra-cabeças. A editora deu ao jogo o nome de "*Number Place*", que é usado até hoje nos Estados Unidos.

Somente na década seguinte, de oitenta, foi que a Nikoli, maior empresa japonesa de quebra-cabeças, levou o *Number Place* ao Japão. É fato que no Japão, os jogos numéricos são mais populares do que os passatempos de palavras cruzadas e caça-palavras, que não funcionam muito bem no idioma japonês. Sendo assim, em 1986, já com o nome de Sudoku, tornou-se um dos jogos mais populares e vendidos do Japão. Apesar de toda a popularidade no Japão, o Sudoku não conseguiu atrair a mesma atenção no Ocidente até o final de 2004, quando outro aposentado, o juiz de Hong Kong Wayne Gould, que também era um fã de quebra-cabeças e programador de computadores, viajou a Londres para convencer os editores do jornal *The Times* a publicar o Sudoku. Gould havia criado um programa de computador que gerava jogos de Sudoku com vários níveis de dificuldade e não estava cobrando nada por tal programa. O *The Times*, então, publicou o primeiro jogo de Sudoku no dia 12 de novembro de 2004 e, desde então, tal jogo tem se tornado cada vez mais popular nos EUA e no resto do mundo. Particularmente no Brasil, o Sudoku é publicado pelas revistas de passatempo "Coquetel", da empresa Ediouro, desde o início de 2005.

Existem inúmeras variações deste jogo, que podem ser quadrados bidimensionais com mais ou menos colunas e linhas, generalizando, com $n^2 \times n^2$ células ou casas. Alguns envolvem sobreposição de áreas, regiões irregulares, com informações parciais (indicando que o número da casa é par ou ímpar, por exemplo), dentre várias outras variações Delahaye (2006).

2.1 Complexidade computacional e prova da NP-Completude

O jogo Sudoku é um problema NP-Completo Garey (1979), cuja prova foi realizada por Takayuki Yato e Takahiro Seta, da Universidade de Tóquio Yato (2003). O desafio consiste em se propor um algoritmo que resolva Sudokus de todos os tamanhos e níveis de dificuldade, ou seja, generalizando, todo quadrado na forma $n^2 \times n^2$, e não apenas as versões $3^2 \times 3^2$ (9×9). Nenhum programa idealizado para resolver todos os quadrados funcionaria bem porque o tempo necessário para encontrar uma solução aumenta drasticamente à medida que n cresce assintoticamente (tende ao infinito).

Existe um número muito grande de soluções (permutações de números de 1 a 9, respeitando-se as regras, para preencher ou gabaritar o Sudoku). A relação pode ser feita com o quadrado latino, cujo número de soluções é um limite superior para o do Sudoku, devido ao fato de que no Sudoku há uma restrição extra que envolve a subgrade de n casas. Assim, como se sabe que existem apenas 12 quadrados latinos de ordem 3 e 575 de ordem 4, mas, que o crescimento é exponencial em relação ao crescimento linear da ordem, fazendo com que o número de quadrados latinos de ordem 9 seja 5.524.751.496.156.892.842.531.225.600. Por outro lado, há ocorrência de muitas soluções simétricas (e a teoria dos grupos afirma que um quadrado que deriva de outro é equivalente ao original) e, assim, se forem permutados todos os números de forma sistemática, ou então, se forem invertidas duas linhas ou duas colunas, os resultados finais serão, em essência, os mesmos. Por fim, eliminando-se soluções simétricas, o número de quadrados latinos de ordem 9 passa a ser de 377.597.570.964.258.816.

Para o caso específico do Sudoku, determinar o número exato de quadrados possíveis se revelou uma tarefa extretamente difícil e que é obtida de forma aproximada, estimando-se o número de quadrados de Sudoku válidos como sendo 6.670.903.752.021.072.936.960. Esse montante inclui as soluções derivadas de qualquer quadrado, por meio de operações elementares (rotações, simetrias, trocas das três primeiras linhas, das três primeiras colunas, etc). Se forem contados apenas uma vez os quadrados que podem ser reduzidos a uma configuração equivalente, o número final cai para 5.472.730.538 (mas ainda assim, apenas um pouco menor que a população da Terra, para dar um exemplo claro de ordem de grandeza). Apesar desta redução, mesmo que fosse solucionado um quadrado por minuto e se vivesse cem anos, só seria possível se resolver cerca de 1% do jogo.

Vale ressaltar que se inicialmente não houver nenhuma casa preenchida, ou um número pequeno demais, a solução do jogo fica bem mais fácil pois, na verdade, existem várias possíveis permutações de números que resultem em uma solução válida. Portanto, a quantidade de casas preenchidas, e a distribuição das mesmas na grade, define o nível de dificuldade do jogo Sudoku. Ainda, a quantidade de casas que devem estar inicialmente preenchidas para que exista apenas uma única solução não está definida formalmente, ou seja, ninguém ainda conseguiu apresentar uma prova formal válida. Além disso, uma grade inicial só tem interesse teórico se for mínima e não puder ser reduzida, ou seja, se a remoção de um elemento apenas implicar que a solução não é mais única. O problema ainda em aberto, portanto, consiste em determinar o número mínimo de casas iniciais a serem preenchidas para que a solução seja única. A suposição é que seja 17 (e neste caso o nível do jogo é considerado difícil, tal como o exemplo apresentado na Figura 1 acima). Por outro lado, já se sabe que o problema oposto, qual é o número máximo de elementos do quadrado inicial para o qual não há garantia de solução única, é 77, dentre 81 casas. Pode-se observar de maneira relativamente fácil que com 80, 79 ou 78 elementos ou casas preenchidas, se houver solução, ela será única.

Para a prova da NP-Completude, provar que o jogo de Sudoku pertence a NP, basta se propor um algoritmo que reconheça se um dado jogo totalmente preenchido é realmente uma solução do Jogo. Isto pode ser feito em tempo polinomial, varrendo-se as linhas, colunas e blocos (subgrades ou caixas) verificando-se se cada número no intervalo de 1 a 9 aparece exatamente uma vez, devolvendo a resposta se "sim"é solução ou "não", não é solução, caso as regras de construção não sejam todas satisfeitas.

Para a segunda parte da prova da NP-Completude, deve-se provar que o jogo do Sudoku é NP-Difícil realizando-se uma redução polinomial de um problema já provado ser NP-Difícil a ele. E Yato e Seta Yato (2003) provaram que o Sudoku é um problema NP-Difícil justamente fazendo a redução polinomial do quadrado latino para o Sudoku, ou seja, mostraram como tornar um quadrado latino $n \times n$ incompleto e criar, através dele, um Sudoku $n^2 \times n^2$ que, se resolvido, fornece támbém a resposta para o quadrado latino. Desta forma, demonstraram que resolver o jogo do Sudoku é tão difícil quanto completar um quadrado latino. Para uma prova mais robusta, pode-se aplicar a redução do problema da Satisfatibilidade (SAT) para o problema UNIQUE-SAT (caso especial onde se tem apenas uma única atribuição de variáveis que satisfaça a expressão booleana na forma normal envolvida), uma prova probabilística, na verdade, que mostra que a garantia de unicidade de solução dificilmente permitiria a resolução do problema SAT em tempo polinomial, fato que pode ser dito também sobre o jogo de Sudoku.

Um outro problema clássico poderia ser usado na prova da NP-Completude do jogo do Sudoku, que seria uma variação do problema clássico de coloração de vértices em grafos, o qual será abordado na próxima seção.

3 O Sudoku como um problema de coloração em grafos

O jogo do Sudoku pode ser modelado como um problema de coloração de vértices em grafos, sendo esta abordagem geral encontrada na literatura. Assim, no grafo que representa o jogo do Sudoku, cada célula ou casa é um vértice do grafo, existindo $n^2 = 81$ vértices no total, dispostos na forma de uma grade bidimensional, onde vértices na mesma linha ou na mesma coluna ou na mesma subgrade (subregião ou caixa), são adjancentes (possuem arestas entre si). Sendo assim, cada vértice é parte de uma linha que inclui outros oito vértices, uma coluna que inclui outros oito, e uma subgrade (região ou caixa) que inclui outros oito (dos quais quatro já foram contados na coluna ou na linha). Assim, cada um dos 81 vértices estão conectados a 20 outros $(8+8+4)$, o que totaliza 1.620 vértices (considerando muitas repetições) que compartilham uma aresta com o adjacente e, consequentemente, o número de arestas é 810 (1.620 dividido por 2, pois cada aresta é compartilhada por 2 vértices). Este grafo descrito é um grafo simples (sem arestas especiais, tal como laços, arestas paralelas ou hiperarestas).

O problema em coloração de grafos passa a ser, considerar cada número de 1 a 9 como uma cor distinta e, então, determinar uma coloração (o grafo 9-colorido, dado que já se sabe por definição que o número cromático deste grafo é 9, ou seja, são necessárias 9 cores para colorir o grafo respeitando-se as restrições), de tal forma que vértices adjacentes possuam cores distintas. Esta correlação faz com que possa ser usado no entendimento e resolução do jogo de Sudoku, todo conhecimento teórico existente para problemas de coloração de vértices em grafos,

Existem inúmeras variações do problema. Alguns vértices podem necessitar de mais de uma cor, sendo que os mesmos possuem um peso w_i que equivale à quantidade de cores necessárias do vértice i. Esta variação é conhecida como **multicoloração**. É interessante notar que um problema de multicoloração pode ser transformado em um problema de k-coloração replicando-se o vértice i no grafo w_i vezes. Os vértices também podem ter restrições de *quais* cores os mesmos podem assumir. Na **coloração com listas**, cada vértice possui uma lista de cores possíveis que podem ser usadas. Uma simplificação desta ideia é considerar cada vértice i tendo apenas um limite *superior* μ_i para a sua cor, ou seja, k vértice pode usar qualquer cor em $[0,\mu_i]$. Esta variação é conhecida por **μ-coloração**. Os vértices também podem possuir limites *inferiores*, de forma que cada vértice i possa usar apenas cores no intervalo $[\gamma_i,\mu_i]$, onde γ_i é o limite inferior. Esta variação é conhecida como **(γ,μ)-coloração** (Bonomo (2006)). Por fim, os vértices já podem estar parcialmente coloridos e o que se deseja é estender esta pré-coloração parcial para uma coloração própria completa para o grafo. Neste caso, então, tem-se um problema de pré-coloração estendida. Todas estas variações de coloração continuam sendo problemas NP-completos (em suas versões de decisão), uma vez que são generalizações do problema clássico de coloração de vértices. Tais coloração especiais de vértices podem ser usados conjuntamente para modelar outros problemas teóricos e práticos, tal como pode ser observado a seguir.

3.1 Sudoku como um problema de pré-coloração estendida em um hipergrafo 9-colorido

Neste trabalho, propõe-se um modelo de coloração para o jogo do Sudoku, através de um hipergrafo 9-colorido e a variação do problema de coloração, pré-coloração estendida, envolvendo listas de cores limitadas iguais. A Figura 2 apresenta o hipergrafo do Sudoku em duas versões (com hiperarestas na forma de semi-retas e na forma de conjuntos).

Um hipergrafo é uma generalização de um grafo, com suas arestas ligando quaisquer quantidades positivas de vértices (e não somente dois vértices, como para grafos tradicionais). Formalmente, define-se um hipergrafo $H = (V,\mathfrak{E})$ como sendo um conjunto V de elementos unitários (exatamente como o conjunto de vértices de um grafo) chamados vértices e um conjunto $\mathfrak{E}$ de subconjuntos não-vazios chamados hiperarestas. Em notação matemática, $\mathfrak{E} \subseteq (S(V)$
$emptyset)$.

A coloração de vértices clássica adaptada para hipergrafos da seguinte forma: seja $H = (V,\mathfrak{E})$ um hipergrafo, com $V = n$. Diz-se que $C = \{c_1, c_2, ..., c_n\}$ é uma coloração própria de H, se e somente se, para toda aresta $\mathfrak{e} \in \mathfrak{E}$, exista pelo menos um par de vértices $v_1, v_2 \in \mathfrak{e}$ tal que $c_1 \neq c_2$.

Desta forma, pode-se enunciar o modelo de pré-coloração estendida, envolvendo listas de cores limitadas iguais. Retomando o exemplo da Figura 2, tem-se que o mesmo pode ser modelado como um hipergrafo contendo 81 vértices e 9 hiperarestas na vertical, 9 hiperarestas na horizontal e 9 hiperarestas para as subgrades ou caixas, uma para cada, ficando um total de 27 hiperarestas (e não mais 810, como para o modelo em grafos simples). Obter uma pré-coloração estendida neste hipergrafo, portanto, consiste gerar uma coloração própria de H tal que para toda aresta $\mathfrak{e} \in \mathfrak{E}$, todo o par de vértices $v_1, v_2 \in \mathfrak{e}$ possua $c_1 \neq c_2$, respeitando-se as cores atribuídas inicialmente a alguns vértices (pré-coloração).

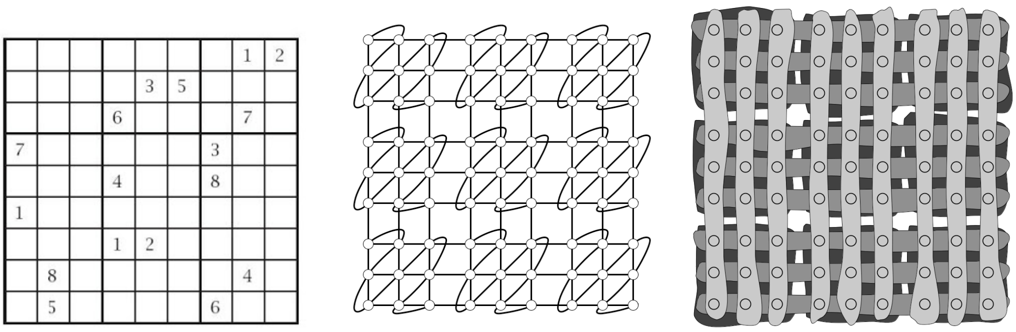

Figura 2. Exemplo do jogo Sudoku modelado como um hipergrafo: (a) formato tradicional em grade do Sudoku; (b) hipergrafo do Sudoku com hiperaestas verticais, horizontais e de curva livre; (c) hipergrafo do Sudoku com representação tradicional na forma de conjuntos.

3.2 Algoritmos exatos de enumeração implícita

Foram desenvolvidos alguns algoritmos para a resolução do jogo Sudoku, sendo um deles baseado no modelo de pré-coloração estendida no hipergrafo 9-colorido do Sudoku, tendo como base o método recursivo de enumeração explícita (ou de busca exaustiva) *backtracking*, mas, aplicando manipulação de bits para evitar a verificação exaustiva da violação das regras, de tal forma que a verificação da ocorrência de um valor (ou cor) em determinada linha, coluna ou bloco é realizada de maneira mais eficiente. O detalhamento do algoritmo está omitido devido a falta de espaço, mas, o pseudo-código é apresentado a seguir.

Tal algoritmo de *Backtracking* implícito, com manipulaçao de bits para verificação implícita de ocorrência de repetição de cores, segundo as regras do jogo Sudoku, resolveu todos os níveis do jogo Sudoku, para as instâncias encontradas na literatura. Pretende-se sistematizar tais experimentos computacionais e realizar análise estatísticas envolvendo generalizações do jogo Sudoku (não somente grades de 9×9), para melhor avaliação dos resultados.

4 Considerações Finais

Neste trabalho, o jogo Sudoku foi investigado, enfatizando seus aspectos teóricos matemáticos computacionais, principalmente sua correlação com coloração em grafos, onde fora proposto um refinamento na modelagem de tal forma que o Sudoku possa ser visto como um problema de pré-coloração estendida em hipergrafos 9-colorido com listas limitadas iguais. Isto possibilitou o desenvolvimento de um algoritmo de enumeração implícita recursivo, um backtracing com podas, de tal forma que o número de verificações das regras

Algorithm 1 SudokuImplicityBacktracking (sudoku, linha, coluna)

se (coluna > 8) entao {linha = linha + 1; coluna = 0;}
se (linha > 8) entao {retorna RESOLVIDO;}
se (sudoku.matriz[linha][coluna] > 0) entao {retorna ResolveSudoku2(sudoku, linha, coluna + 1);}
{bloco = calculaBloco(linha, coluna)};
for c **from** 1 **to** 9 **do**
  se (nao (sudoku.linhas[linha][valor] == 1 OU sudoku.colunas[coluna][valor] == 1 OU sudoku.blocos[bloco][valor] == 1)) entao {sudoku.matriz[linha][coluna] = valor; sudoku.linhas[linha][valor] = 1; sudoku.colunas[coluna][valor] = 1; sudoku.blocos[bloco][valor] = 1; }
  se (ResolveSudoku2(sudoku, linha, coluna + 1) == RESOLVIDO) entao {retorna RESOLVIDO;}
  {sudoku.matriz[linha][coluna] = 0; sudoku.linhas[linha][valor] = 0; sudoku.colunas[coluna][valor] = 0; sudoku.blocos[bloco][valor] = 0;}
end for
{retorna IMPOSSIVEL;}

para cada hiperaresta fosse reduzida, possibilitando a resolução de jogos de Sudoku com níveis difíceis de dificuldade

Como pesquisa em andamento, está o desenvolvimento de outros tipos de algoritmos de enumeração implícita, incluindo programação dinâmica e programação linear inteira baseada em uma formulação de PI $0-1$ em construção. Isto para que experimentos computacionais sistemáticos possam ser estruturados e uma análise mais aprofundada do problema e seu comportamento quanto ao número de vértices e arestas envolvidos, bem como variações de coloração e tipo de grafo considerado, possa ser feita. Este trabalho é resultado de um projeto de pesquisa do Programa Jovens Talentos para a Ciência do CNPq, período 2012/2013, projeto ainda em vigência nos meses finais, do qual o autor foi selecionado e é bolsista. A pesquisa terá continuidade em um PIBIC, projeto de iniciação científica convencional, já submetido, uma vez que o Programa Jovens Talentos é somente para graduandos dos primeiros períodos e serve justamente como um preparatório para o PIBIC.

Agradecimentos: Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Referências

Bonomo, F., Durán, G., e Marenco, J. (2006). Exploring the complexity boundary between coloring and list-coloring. *Electronic Notes Discrete Mathematics*, 25:41–47.

Delahaye, J. P. (2006). A ciência do século. *Scientific American Brasil*, 50.

Garey, M. R. e Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.

Yato, T. (2003). Complexity and completeness of finding another solution and its applications to puzzles. *Thesis - Graduate School of Science, University of Tokyo*.