Dar uma festa de sucesso é NP-difícil

Combinando

Sabrina Teodoro

20/10/2021

Baseado em Sanjeev Arora e Boaz Barak. Computational Complexity: A Modern Approach. Cambridge University Press, 2009.

Festa

Festa

Considere uma lista de pessoas e uma lista de todos os pares delas que não se gostam.

Festa

Considere uma lista de pessoas e uma lista de todos os pares delas que não se gostam.

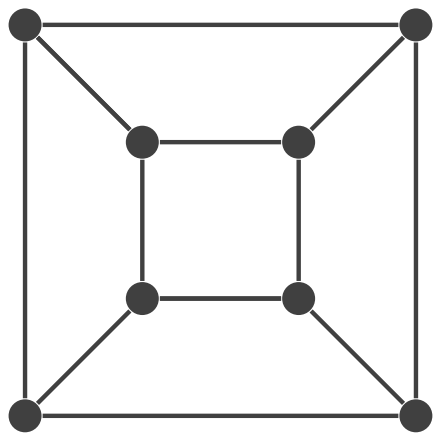
Existe um jeito fácil de encontrar o maior conjunto de pessoas que você pode convidar para uma festa de modo que quaisquer duas pessoas convidadas se gostem?

Grafos

Representando as possíveis pessoas convidadas como vértices de um grafo com arestas entre quaisquer duas pessoas que não se gostam, o problema se torna encontrar um conjunto independente máximo (CIM) no grafo.

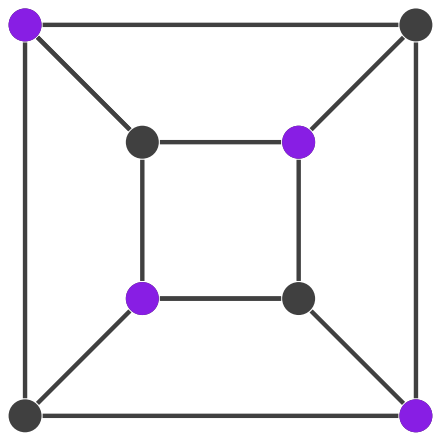
Grafos

Representando as possíveis pessoas convidadas como vértices de um grafo com arestas entre quaisquer duas pessoas que não se gostam, o problema se torna encontrar um conjunto independente máximo (CIM) no grafo.



Grafos

Representando as possíveis pessoas convidadas como vértices de um grafo com arestas entre quaisquer duas pessoas que não se gostam, o problema se torna encontrar um conjunto independente máximo (CIM) no grafo.



Linguagem formal

Dada uma função $f : \{0,1\}^* \to \{0,1\}$, a linguagem formal associada a ela é o conjunto

$$L_f = \{x \in \{0,1\}^* : f(x) = 1\}$$

Linguagem formal

Dada uma função $f : \{0,1\}^* \to \{0,1\}$, a linguagem formal associada a ela é o conjunto

$$L_f = \{x \in \{0,1\}^* : f(x) = 1\}$$

Identificamos o problema de, dado x, computar $f(x)$ com o problema de, dado x, decidir se $x \in L_f$.

NP

Uma linguagem L está em (ou é) NP se existem uma função polinomial $p : \mathbb{N} \to \mathbb{N}$ e um algoritmo de tempo polinomial M tais que para todo $x \in \{0,1\}^*$

$$x \in L \iff \exists u \in \{0,1\}^{p(|x|)} : M(x,u) = 1$$

Chamamos u de um certificado para x.

CIM é NP

Linguagem

Vamos denotar grafos por G e inteiros por k.
Considere a linguagem

$$\texttt{CIM} = \{\langle G, k\rangle : \exists S \subset V(G) : |S| \geq k \wedge \forall u, v \in S, uv \notin A(G)\}$$

CIM é NP

Linguagem

Vamos denotar grafos por G e inteiros por k.
Considere a linguagem

$$\texttt{CIM} = \{\langle G, k\rangle : \exists S \subset V(G) : |S| \geq k \wedge \forall u, v \in S, uv \notin A(G)\}$$

Um algoritmo para CIM verifica se existe um conjunto independente S de tamanho pelo menos k em G.
Não é imediato que um tal algoritmo pode ser usado para encontrar S, embora seja o caso.

CIM é NP

Prova

Considere o algoritmo M de tempo polinomial a seguir:

CIM é NP

Prova

Considere o algoritmo M de tempo polinomial a seguir: dados um par $\langle G, k \rangle$ e uma string $u \in \{0,1\}^*$, devolve 1 se, e somente se, u codifica um conjunto independente de tamanho k em G.

CIM é NP

Prova

Considere o algoritmo M de tempo polinomial a seguir: dados um par $\langle G, k \rangle$ e uma string $u \in \{0,1\}^*$, devolve 1 se, e somente se, u codifica um conjunto independente de tamanho k em G.

É claro que $\langle G, k \rangle \in \mathtt{CIM}$ se, e somente se, existe uma string u tal que $M(\langle G, k \rangle, u) = 1$.

CIM é NP

Prova

Considere o algoritmo M de tempo polinomial a seguir: dados um par $\langle G, k \rangle$ e uma string $u \in \{0,1\}^*$, devolve 1 se, e somente se, u codifica um conjunto independente de tamanho k em G.

É claro que $\langle G, k \rangle \in \text{CIM}$ se, e somente se, existe uma string u tal que $M(\langle G, k \rangle, u) = 1$. Logo CIM é NP, sendo que u serve como certificado de que $\langle G, k \rangle \in \text{CIM}$.

Redução e NP-completude

Uma linguagem L é redutível em tempo polinomial para uma linguagem L' se existe uma função $f : \{0,1\}^* \to \{0,1\}^*$ computável em tempo polinomial tal que para todo $x \in \{0,1\}^*$

$$x \in L \iff f(x) \in L'$$

Denotamos por $L \leq_p L'$.

Redução e NP-completude

Uma linguagem L é redutível em tempo polinomial para uma linguagem L' se existe uma função $f : \{0,1\}^* \to \{0,1\}^*$ computável em tempo polinomial tal que para todo $x \in \{0,1\}^*$

$$x \in L \iff f(x) \in L'$$

Denotamos por $L \leq_p L'$.

Dizemos que L' está em (ou é) NP-difícil se $L \leq_p L'$ para todo L em NP.

Redução e NP-completude

Uma linguagem L é redutível em tempo polinomial para uma linguagem L' se existe uma função $f : \{0,1\}^* \to \{0,1\}^*$ computável em tempo polinomial tal que para todo $x \in \{0,1\}^*$

$$x \in L \iff f(x) \in L'$$

Denotamos por $L \leq_p L'$.

Dizemos que L' está em (ou é) NP-difícil se $L \leq_p L'$ para todo L em NP. Se, além disso, L' for NP, dizemos que L' está (ou é) NP-completo.

3SAT

Fórmula Booleana

Uma fórmula booleana sobre $u_1, \cdots, u_n$ é uma sentença composta pelas variáveis e pelos operadores lógicos $\wedge, \vee, \neg$.

3SAT

Fórmula Booleana

Uma fórmula booleana sobre $u_1, \cdots, u_n$ é uma sentença composta pelas variáveis e pelos operadores lógicos $\wedge, \vee, \neg$.

Sejam ϕ uma fórmula booleana sobre $u_1, \cdots, u_n$ e $z \in \{0,1\}^n$. Então $\phi(z)$ denota o valor de ϕ quando atribuímos os valores de z às suas variáveis.

3SAT

Fórmula Booleana

Uma fórmula booleana sobre $u_1, \cdots, u_n$ é uma sentença composta pelas variáveis e pelos operadores lógicos $\wedge, \vee, \neg$.

Sejam ϕ uma fórmula booleana sobre $u_1, \cdots, u_n$ e $z \in \{0,1\}^n$. Então $\phi(z)$ denota o valor de ϕ quando atribuímos os valores de z às suas variáveis.

Uma fórmula ϕ é satisfatível se existe alguma valoração z tal que $\phi(z) = 1$. Caso contrário, ϕ é insatisfatível.

3SAT

Forma Normal Conjuntiva

Uma fórmula sobre $u_1, \cdots, u_n$ está na forma normal conjuntiva (CNF) se tem a forma

$$\bigwedge_i \left(\bigvee_j v_{ij} \right)$$

em que cada v_{ij} é uma variável u_k ou sua negação $\bar{u}_k := \neg u_k$.

3SAT

Forma Normal Conjuntiva

Uma fórmula sobre $u_1, \cdots, u_n$ está na forma normal conjuntiva (CNF) se tem a forma

$$\bigwedge_i \left(\bigvee_j v_{ij} \right)$$

em que cada v_{ij} é uma variável u_k ou sua negação $\bar{u}_k := \neg u_k$.
Chamamos os termos v_{ij} de literais da fórmula e $\left(\bigvee_j v_{ij} \right)$ de suas cláusulas.

Uma fórmula kCNF é uma CNF em que cada cláusula contém no máximo k literais.

3SAT

Forma Normal Conjuntiva

Uma fórmula sobre $u_1, \cdots, u_n$ está na forma normal conjuntiva (CNF) se tem a forma

$$\bigwedge_i \left(\bigvee_j v_{ij} \right)$$

em que cada v_{ij} é uma variável u_k ou sua negação $\bar{u}_k := \neg u_k$.
Chamamos os termos v_{ij} de literais da fórmula e $\left(\bigvee_j v_{ij} \right)$ de suas cláusulas.

Uma fórmula kCNF é uma CNF em que cada cláusula contém no máximo k literais.

$$(u_1 \vee \bar{u}_2 \vee u_3) \wedge (u_2 \vee \bar{u}_3 \vee u_4) \wedge (\bar{u}_1 \vee u_3 \vee \bar{u}_4)$$

3SAT

Teorema de Cook-Levin

Denotamos por SAT e 3SAT as linguagens de todas as fórmulas CNF e 3CNF satisfatíveis, respectivamente.

3SAT

Teorema de Cook-Levin

Denotamos por SAT e 3SAT as linguagens de todas as fórmulas CNF e 3CNF satisfatíveis, respectivamente.

Teorema de Cook-Levin

SAT e 3SAT são NP-completo.

3SAT

Teorema de Cook-Levin

Denotamos por SAT e 3SAT as linguagens de todas as fórmulas CNF e 3CNF satisfatíveis, respectivamente.

Teorema de Cook-Levin
SAT e 3SAT são NP-completo.

Para demonstrar isso, eles tiveram que mostrar que toda linguagem NP pode ser reduzida para SAT.

Mas, para provar a NP-completude de qualquer outra linguagem L, não precisamos trabalhar tão duro; basta reduzir SAT ou 3SAT para L.

CIM é NP-difícil

Construção

Vamos reduzir 3SAT para CIM.

CIM é NP-difícil

Construção

Vamos reduzir 3SAT para CIM.
Mais especificamente, mostrar como transformar, em tempo polinomial, cada 3CNF ϕ de m-cláusulas em um grafo G de $7m$ vértices de modo que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho pelo menos m.

CIM é NP-difícil

Construção

Seja ϕ uma 3CNF. Segue a construção de G:

CIM é NP-difícil

Construção

Seja ϕ uma 3CNF. Segue a construção de G:
Associamos 7 vértices a cada cláusula C de ϕ, que correspondem às 7 possíveis valorações parciais satisfatíveis para os 3 literais de C.

CIM é NP-difícil

Construção

Seja ϕ uma 3CNF. Segue a construção de G:
Associamos 7 vértices a cada cláusula C de ϕ, que correspondem às 7 possíveis valorações parciais satisfatíveis para os 3 literais de C.

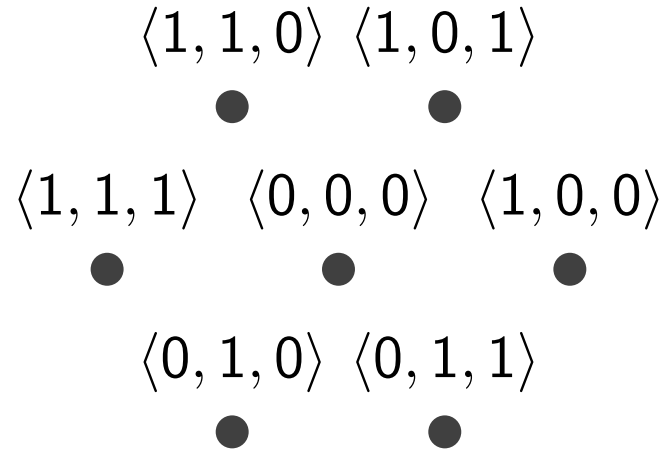


Figura: Vértices associados à cláusula $C = (u_1 \vee u_2 \vee \bar{u_3})$.
Note que a valoração $\langle u_1, u_2, u_3 \rangle = \langle 0, 0, 1 \rangle$ torna C falsa.

CIM é NP-difícil

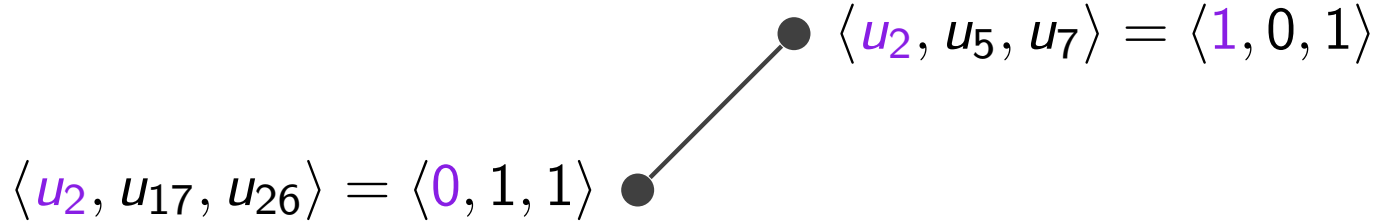
Construção

Unimos dois vértices por uma aresta se eles correspondem a valorações inconsistentes, isto é, que atribuem valores diferentes a alguma das variáveis que compartilham.

CIM é NP-difícil

Construção

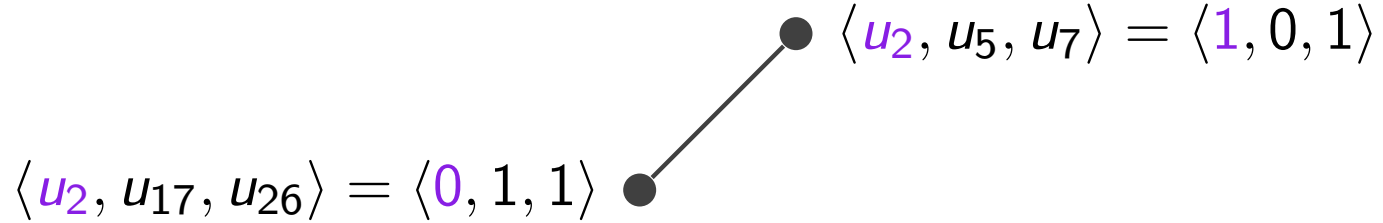
Unimos dois vértices por uma aresta se eles correspondem a valorações inconsistentes, isto é, que atribuem valores diferentes a alguma das variáveis que compartilham.



CIM é NP-difícil

Construção

Unimos dois vértices por uma aresta se eles correspondem a valorações inconsistentes, isto é, que atribuem valores diferentes a alguma das variáveis que compartilham.



Também adicionamos arestas entre vértices associados à mesma cláusula.

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial.

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial. Resta mostrar que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho m.

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial. Resta mostrar que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho m.

($\Rightarrow$) Suponha que ϕ tem uma valoração satisfatível u.

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial. Resta mostrar que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho m.

($\Rightarrow$) Suponha que ϕ tem uma valoração satisfatível u. Defina um conjunto S de m vértices de G como segue:

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial. Resta mostrar que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho m.

($\Rightarrow$) Suponha que ϕ tem uma valoração satisfatível u. Defina um conjunto S de m vértices de G como segue: para cada cláusula C de ϕ adicione a S o vértice associado à valoração u restrita a C.

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial. Resta mostrar que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho m.

($\Rightarrow$) Suponha que ϕ tem uma valoração satisfatível u. Defina um conjunto S de m vértices de G como segue: para cada cláusula C de ϕ adicione a S o vértice associado à valoração u restrita a C.

Com isso, não existem dois vértices em S que correspondam a atribuições inconsistentes.

CIM é NP-difícil

Prova

É fácil ver que podemos transformar ϕ em G em tempo polinomial. Resta mostrar que ϕ é satisfatível se, e somente se, G tem um conjunto independente de tamanho m.

($\Rightarrow$) Suponha que ϕ tem uma valoração satisfatível u. Defina um conjunto S de m vértices de G como segue: para cada cláusula C de ϕ adicione a S o vértice associado à valoração u restrita a C.

Com isso, não existem dois vértices em S que correspondam a atribuições inconsistentes. Logo S é um conjunto independente de tamanho m.

CIM é NP-difícil

Prova

($\Leftarrow$) Suponha que G tem um conjunto independente S de tamanho m.

CIM é NP-difícil

Prova

($\Leftarrow$) Suponha que G tem um conjunto independente S de tamanho m.

Definimos uma valoração u como segue:

CIM é NP-difícil

Prova

($\Leftarrow$) Suponha que G tem um conjunto independente S de tamanho m.
Definimos uma valoração u como segue: para cada $i = 1, 2, \cdots, m$, se existir um vértice em S associado a uma valoração parcial $a \in \{0, 1\}$ para u_i, faça $u_i = a$;

CIM é NP-difícil

Prova

($\Leftarrow$) Suponha que G tem um conjunto independente S de tamanho m.
Definimos uma valoração u como segue: para cada $i = 1, 2, \cdots, m$, se existir um vértice em S associado a uma valoração parcial $a \in \{0, 1\}$ para u_i, faça $u_i = a$; caso contrário, faça $u_i = 0$.

Isso está bem definido porque, como S é um conjunto independente, cada variável u_i pode assumir no máximo um valor.

CIM é NP-difícil

Prova

($\Leftarrow$) Suponha que G tem um conjunto independente S de tamanho m.
Definimos uma valoração u como segue: para cada $i = 1, 2, \cdots, m$, se existir um vértice em S associado a uma valoração parcial $a \in \{0, 1\}$ para u_i, faça $u_i = a$; caso contrário, faça $u_i = 0$.

Isso está bem definido porque, como S é um conjunto independente, cada variável u_i pode assumir no máximo um valor. Por outro lado, S contém exatamente um vértice associado a cada cláusula.

CIM é NP-difícil

Prova

($\Leftarrow$) Suponha que G tem um conjunto independente S de tamanho m.
Definimos uma valoração u como segue: para cada $i = 1, 2, \cdots, m$, se existir um vértice em S associado a uma valoração parcial $a \in \{0, 1\}$ para u_i, faça $u_i = a$; caso contrário, faça $u_i = 0$.

Isso está bem definido porque, como S é um conjunto independente, cada variável u_i pode assumir no máximo um valor. Por outro lado, S contém exatamente um vértice associado a cada cláusula. Logo, u é satisfatível para todas as cláusulas de ϕ.

Dúvidas?