Prova do Teorema de Brooks

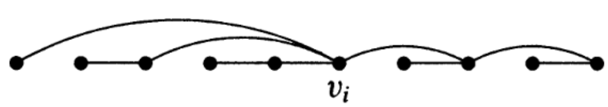
Theorem 1. *Para todo grafo conexo G que não é um ciclo ímpar nem um grafo completo, $\chi(G) \leq \Delta(G)$.*

Demonstração. Seja G um grafo conexo e seja $k = \Delta(G)$. Podemos assumir que $k \geq 3$, dado que G é um grafo completo quando $k \leq 1$ e G é um ciclo ímpar ou um grafo bipartido quando $k = 2$, caso no qual o resultado é verdadeiro.

Nosso objetivo é ordenar os vértices de G de modo que cada vértice tenha no máximo $k-1$ vizinhos anteriores a ele na ordem dada. Assim, o algoritmo de coloração gulosa aplicado sobre esta ordenação produzirá uma coloração própria de vértices com o limitante superior desejado, que é $\chi(G) \leq \Delta(G)$. A seguir, consideramos dois casos, dependendo se G é regular ou não.

Caso 1. G não é k-regular.

Quando G não é k-regular, nós podemos escolher um vértice de grau menor do que k para ser o último vértice v_n. Como G é conexo, conseguimos construir uma árvore geradora de G a partir do vértice v_n (usando BFS, por exemplo), atribuindo índices em ordem decrescente a cada vértice alcançado na busca. Com exceção do vértice v_n, todos os demais vértices na ordenação $v_1, \ldots, v_n$ tem um vizinho com índice maior que ele ao longo do caminho que o conecta ao vértice v_n na árvore de busca em largura. Portanto, cada vértice v_i tem no máximo $k-1$ vizinhos com índices menores que i e, assim, a coloração gulosa usa no máximo $k = \Delta(G)$ cores para colorir $V(G)$.



Caso 2. G é k-regular.

Caso 2.1 G possui um vértice de corte.

Neste caso, seja x um vértice de corte de G e sejam $G_1, G_2, \ldots, G_s$ as componentes conexas de $G - x$. A partir de cada componente conexa G_i construa o subgrafo H_i formado pela componente conexa G_i mais o vértice x e todas as arestas que ligam x a vértices de G_i em G. Note que $H_i \subset G$. Além disso, o grau de x em uma componente H_i arbitrária é menor do que k. Assim, o método do Caso 1 acima fornece uma k-coloração própria de vértices para H_i, $1 \leq i \leq s$. Permutando os nomes das cores nos subgrafos H_i, nós podemos fazer com que as colorações atribuam a mesma cor a x em cada um dos subgrafos H_i, completando assim uma k-coloração própria de vértices de G.

Caso 2.2. G é 2-conexo (não tem vértice de corte).

Como G é k-regular, em qualquer ordenação dos vértices de G, o último vértice terá k vizinhos antes dele na ordenação. No entanto, nem tudo está perdido. A coloração gulosa ainda pode funcionar se nós conseguirmos arranjar para que dois vizinhos de v_n recebam a mesma cor.

Em particular, suponha que algum vértice v_n tem vizinhos v_1 e v_2 tais que v_1 não é adjacente a v_2 e $G - \{v_1, v_2\}$ é conexo. Neste caso, nós indexamos os vértices da árvore geradora do grafo $G - \{v_1, v_2\}$ usando os números $3, 4, \ldots, n$ tal que os rótulos aumentem ao longo dos caminhos em direção à raiz v_n. Como antes, cada vértice anterior a v_n na ordenação tem no máximo $k - 1$ vizinhos com índices menores. A coloração gulosa aplicada a essa ordenação de $G - \{v_1, v_2\}$ usa no máximo $k - 1$ cores nos vizinhos de v_n, dado que v_1 e v_2 recebem a mesma cor, e o resultado segue.

Então, a fim de concluir a demonstração, basta mostrar que todo grafo k-regular 2-conexo com $k \geq 3$ tem uma tripla de vértices v_1, v_2, v_n com as propriedades listadas acima.

Escolha um vértice $x \in V(G)$.

Subcaso 2.2.1: $\kappa(G - x) \geq 2$. Neste caso, defina $v_1 = x$ e defina v_2 como um vértice à distância 2 de x. Tal vértice v_2 existe pois G é regular e não é um grafo completo. Seja v_n um vizinho comum de v_1 e v_2.

Subcaso 2.2.2: $\kappa(G - x) = 1$. Nesta caso, faça $v_n = x$. Como G não tem vértices de corte, x tem um vizinho em todo bloco folha de $G - x$. Vizinhos v_1 e v_2 de x nestes dois blocos não são adjacentes (caso contrário os blocos aos quais eles pertencem seriam um único bloco, o que não acontece). Além disso, $G - \{x, v_1, v_2\}$ é conexo, dado que blocos não possuem vértices de corte. Como $k \geq 3$, o vértice x tem outro vizinho, e $G - \{v_1, v_2\}$ é conexo. $\square$