

Specyfikacja funkcjonalna programu Grafo-Siekacz

Aiman Ghanim, Cyprian Ciesielski

12.03.2025

Cel projektu

Program **gs** ma na celu przeprowadzić podział grafu na określoną liczbę części. Ma on przyjmować wszystkie parametry z linii poleceń, umożliwiając użytkownikowi wskazanie liczby części, na które ma zostać podzielony graf, oraz marginesu procentowego dla różnicy w liczbie wierzchołków w powstałych częściach. W domyślnym ustawieniu (brak wskazania przez użytkownika) graf dzielony jest na 2 części przy marginesie błędu 10%.

Dodatkowo, aplikacja powinna dążyć do minimalizacji liczby przeciętych krawędzi pomiędzy częściami grafu, zapewniając równomierny podział wierzchołków w zadanych częściach.

Dane wejściowe

Plik opisujący graf może składać się z wielu linii. Poniżej opisane jest znaczenie poszczególnych linii:

1. Maksymalna możliwa liczba węzłów w wierszu (w grafie nie musi znajdować się wiersz o takiej liczbie węzłów)
2. Indeksy węzłów w poszczególnych wierszach - liczba wszystkich indeksów odpowiada liczbie węzłów grafu
3. Wskaźniki na pierwsze indeksy węzłów w liście wierszy z punktu 2
4. Grupy węzłów połączone przy pomocy krawędzi
5. Wskaźniki na pierwsze węzły w poszczególnych grupach z punktu 4. Ta sekcja może występować w pliku wielokrotnie, co oznacza, że plik zawiera więcej niż jeden graf.

Argumenty wywołania programu

Program `gs` akceptuje następujące argumenty wywołania:

- `--help / -h` wyświetla pomoc dotyczącą obsługi programu;
- `--prioritize-balance / -b` zmusza algorytm do skupienia się na minimalizacji procentowej różnicy wielkości części kosztem potencjalnie większej liczby przeciętych krawędzi.
- `--output filename / -o [nazwa pliku]` określa plik, do którego zapisany zostanie wynik;
- `--max-edges-cut n / -m [n]` określa maksymalną liczbę przeciętych krawędzi;
- `--output-statistics / -o` wyświetla statystyki dotyczące podziału grafu;
- `--precompute-metrics / -p` oblicza oraz wyświetla statystyki grafu przed podziałem;
- `--out-format text|binary / -k` wybiera format pliku wyjściowego (tekstowy, lub binarny);
- `--verbose / -v` wybiera format pliku wyjściowego (tekstowy, binarny);
- `--force / -f` dzieli graf mimo braku spełnienia marginesu podziału;
- `--graf n / -g [n]` pozwala wybrać graf do analizowania przez algorytm (jak jest ich więcej niż jeden);
- `--algorytm / -a` umożliwia wybór algorytmu na podstawie, którego będzie dzielony graf

Przykładowe wywołania programu:

- `./gs 3 20% grafy.txt --max-edges-cut 10 --precompute-metrics --output odp.txt`
- Efektem będzie podział grafu z pliku `grafy.txt` na 3 części przy marginesie błędu 20% oraz przy maksymalnym przecięciu 10 krawędzi. Wyświetlą się również statystyki grafu przed podziałem, a grafy wynikowe zostaną zapisane do pliku `odp.txt`.

Funkcjonalność programu

Projekt polega na podziale grafu na określoną liczbę części, przy czym liczba wierzchołków w każdej części nie może różnić się o więcej niż ustalony margines procentowy, liczony względem średniej liczby wierzchołków na część.

Główne cele projektu

- Równomierny podział
- Minimalizacja krawędzi między częściami

Podejście do rozwiązania

Ze względu na NP-trudność problemu (czyli brak znanego algorytmu rozwiązującego problem optymalnie), stosuje się metody przybliżone oraz heurystyki, które pozwalają uzyskać satysfakcjonujące rozwiązania w rozsądnym czasie, nawet dla dużych grafów. Kluczowe elementy podejścia to:

- **Analiza danych wejściowych:** Algorytm analizuje strukturę grafu oraz rozkład wierzchołków, co pozwala na określenie średniej liczby wierzchołków na część.
- **Balansowanie względem średniej:** Ustalony margines (np. 20%) definiuje zakres akceptowalnych wielkości części. Jeśli średnia wynosi 5 wierzchołków, to każda część musi zawierać od 4 do 6 wierzchołków.
- **Minimalizacja przecięć:** Równocześnie algorytm dąży do zminimalizowania liczby krawędzi łączących różne części.

Komunikaty błędów

Program `gs` stara się kontynuować pracę, choć w niektórych przypadkach jest to niemożliwe jak np. błędne dane, lub wymagania niemożliwe do spełnienia.

1. Nieprawidłowy format pliku wejściowego lub wyjściowego:

"Błąd: Nieprawidłowy format pliku wejściowego/wyjściowego.
(Program przyjmuje tylko formaty txt i bin)" np. użytkownik
podał plik 123.png [kod błędu: -1]

2. Brak możliwości dokonania podziału zgodnego z wymaganym marginesem lub wymaganą liczbą podziałów:

"Ostrzeżenie: Podział grafu nie spełnia założonego marginesu równowagi. Sprawdź parametry wejściowe lub spróbuj innej liczby części." np. mamy graf składający się z 5 node'ów, a chcemy podzielić go na 10 części z marginesem 20%. [kod błędu: -2]

3. Błąd podczas zapisu/odczytu plików:

"Błąd: Nie udało się otworzyć pliku 'nazwa_pliku'. Sprawdź, czy plik istnieje i czy masz odpowiednie uprawnienia."

np. nieistniejący plik [kod błędu: -3]

4. Błędne parametry przekazane przez użytkownika:

Błąd: Margines błędu musi zawierać się w przedziale 0-100% a liczba podziałów musi być większa od 0, popraw te dane, a następnie wprowadź je ponownie np. użytkownik podał margines błędu -10% a liczbę podziałów równą -550. [kod błędu: -4]