アプリケーション説明書

提出日：2021年12月26日

日本大学　文理学部　情報科学科

5419063 工藤正和

1. アプリケーション名

Graph drawer

1. 想定シーン

　入力したグラフに対して、force-directedとStress Majorizationをそれぞれ用いてグラフを描画するシーン。

1. 並行して行う理由

　Stress Majorizationは全頂点最短経路や行列計算と言った時間コストがかかる処理が複数あり、force-directedも計算量O(n-2)掛かる処理が複数存在するので、それらを並行して行うことで処理時間を削減するため。

1. 実装上の制約

本アプリでは，一部の実装が便宜上簡略化されている。本来の実装と簡略された実装の関係は次のようになる。

|  |  |
| --- | --- |
| 本来の実装 | 代用の実装 |
| Jsonやcsvファイルなどによるグラフ入力 | csvファイルのみ |
| ある連立方程式を解き続けることによって得られるStress関数の大域的最適解を使用 | Stress関数の局所的最適解を使用 |
| EadesのForce-directed algorithmを使用 | 速度verlet法を用いた簡易的なForce-directed algorithmを使用 |
| グラフの描画 | 頂点の座標とエッジを標準出力 |
|  |  |

1. クラス・メゾットの一覧と説明

|  |  |
| --- | --- |
| クラス名 | 説明 |
| Main | アプリケーションの制御を行うクラス。 |
| Edge | グラフのエッジに関するクラス |
| StressMajorization | Stress Majorizationに関する処理を行うクラス |
| ForceDirected | Stress Majorizationに関する処理を行うクラス |
| GraphDrawer | グラフの描画を行うクラス |

1. 並行処理を実現するために工夫した点