

# 「モデリングとシミュレーション実験」

## 2019 年度期末レポート課題

締切:2020/2/10

### 1 ネットワーク上の酔歩

頂点 (vertex) と頂点をつなぐ辺 (edge) からなるシステムをネットワーク (network) と呼ぶ。グラフ理論では、グラフ (graph) とネットワークを区別する場合があるが、ここでは区別しない。このネットワーク上を粒子が酔歩する問題を考える。

#### 1.1 ネットワークの定義

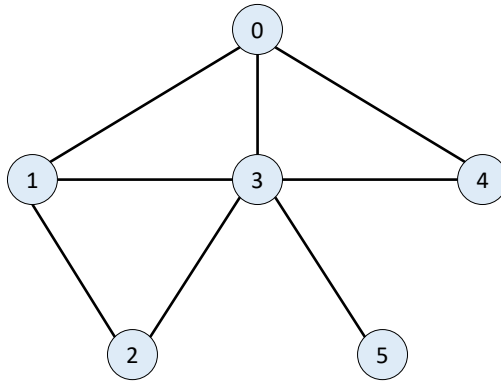


図1 簡単なグラフの例

$N$  個の頂点  $\{v_i\}$  からなる無向ネットワーク、つまり辺に向きのないネットワークを考える。頂点の連結の様子を隣接行列 (adjacency matrix)  $A$  で表すことにする。行列  $A$  の要素は

$$A_{ij} = \begin{cases} 1 & \text{頂点 } i \text{ と } j \text{ が結ばれている} \\ 0 & \text{それ以外} \end{cases} \quad (1.1)$$

である。簡単なグラフ図 1 に対する隣接行列は式 (1.2) で表される。

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (1.2)$$

## 1.2 ネットワーク上の酔歩

$M$  個の粒子がネットワーク上をランダムに移動することを考える。各時刻で、各粒子は隣接する頂点の一つを等確率で選び、必ず移動する。ただし、粒子はお互いに独立に移動するものとする。つまり、 $k$  個の隣接頂点がある頂点に粒子が居る場合には、その粒子は次の時刻に、隣接する頂点の一つを確率  $1/k$  で選び、移動する。図 1 の場合では、 $v_5$  に居る粒子は必ず  $v_3$  へ移動し、 $v_2$  に居る粒子は、 $v_1$ 、 $v_3$  のいずれかへ等しい確率  $1/2$  で移動する。

## 2 クラス設計

ネットワーク上の酔歩をシミュレートするために、以下のようなクラスを作成しよう。

- ネットワークの構造は、隣接行列クラス `AdjacencyMatrix` で記述する。シミュレーションに必要なメソッドも準備する。
- 酔歩する粒子のクラス `Particle` を作成する。次に移動できる頂点つまり隣接頂点のリストを引数として渡すと、そのなかから行先をランダムに選び移動するメソッドを準備する。
- シミュレーションを実行するクラス `Simulation` を作成する。

**課題 1** ネットワーク上の酔歩のためのプロジェクト `RWNetwork` を作成しなさい。さらに、その下に、各クラスを配置するためのパッケージ `model` を作成しなさい。

### 2.1 AdjacencyMatrix クラス

ネットワークの構造を表すクラスを `AdjacencyMatrix` とする。コンストラクタの仕様を表 1 に示す。配列  $d$  は次元配列であり、図 1 の場合では、 $n=6$  であり、 $d$  はソース

コード 1 のようになる。AdjacencyMatrix クラスは、その private フィールドとして、配列  $d$  及び  $n$  を保持する。また、最低限必要なメソッドの仕様を表 2 に示す。

表 1 AdjacencyMatrix クラスのコンストラクタ

コンストラクタと説明
AdjacencyMatrix(int d[], int n) 隣接行列の要素 $d$ を与えて初期化する。 $d$ のサイズは $n \times n$ である。

ソースコード 1 図 1 に対応した配列  $d$

```

1 int d[] = {
2     0,1,0,1,1,0
3     1,0,1,1,0,0
4     0,1,0,1,0,0
5     1,1,1,0,1,1
6     1,0,0,1,0,0
7     0,0,0,1,0,0
8 };

```

表 2 AdjacencyMatrix クラスのパブリックメソッド

型	メソッドと説明
int	get(int i, int j) $d[i*n + j]$ を返す
List<Integer>	nextNodes(int j) 頂点 $v_j$ から移動できる頂点のリストを返す
int	inDegree(int i) 頂点 $v_i$ に入ってくる辺の数を返す

課題 2 仕様に基づき、クラス AdjacencyMatrix を作成しなさい。また、その動作を確認しなさい。動作確認の内容も説明しなさい。

## 2.2 Particle クラス

酔歩する粒子を表すクラスを Particle とする。コンストラクタを表 3 に、メソッドを表 4 に示す。Particle クラスは、private フィールドとして、粒子を識別する番号  $id$  及び現在居る頂点の番号  $currentNode$  を有する。

表3 Particle クラスのコンストラクタ

コンストラクタと説明
<b>Particle(int x,int id)</b> 初期ノード $x$ と、各粒子を区別する一意の $id$ を与えて初期化する。

表4 Particle クラスのパブリックメソッド

型	メソッドと説明
int	<b>move(List&lt;Integer&gt; neighbours)</b> 次の時刻に移動可能な頂点のリスト $neighbours$ を与え、その中からランダムに移動先を決定し、移動する。移動した先の頂点番号を返す。
int	<b>getCurrentNode()</b> 現在いる頂点の番号を返す。

課題3 仕様に基づき、クラス Particle を作成しなさい。また、その動作を確認しなさい。動作確認の内容も説明しなさい。

## 2.3 Simulation クラス

シミュレーションを実行するクラスを Simulation とする。Simulation クラスは、隣接行列を表す次元配列  $d$ 、頂点数  $n$ 、及び粒子総数  $population$  を与えて、表5に示す仕様で初期化する。Simulation クラスは、private フィールドとして、ネットワークを表現する AdjacencyMatrix クラスのインスタンス  $matrix$ 、頂点数  $n$ 、及び粒子のリストである List<Particle>のインスタンス  $particles$  を有する。また、最低限必要なメソッドを表6に示す。

表5 Simulation クラスのコンストラクタ

コンストラクタと説明
<b>Simulation(int d[], int n, int population)</b> 隣接行列に対応した配列 $d$ 、頂点数 $n$ 、粒子数 $population$ を与えて初期化する。

課題4 仕様に基づき、クラス Simulation を作成しなさい。また、その動作を確認しなさい。動作確認の内容も説明しなさい。

表 6 Simulation クラスのパブリックメソッド

型	メソッドと説明
void	<code>update()</code> 全ての粒子を 1 ステップ移動する。
int	<code>inDegree(int i)</code> 頂点 $v_i$ に入ってくる辺の数を返す
int []	<code>evalDistribution()</code> 各頂点 (配列のインデクス) に居る粒子数を返す。

### 3 シミュレーションの実行

**課題 5** 図 1 に対応したシステムで、粒子数 10,000 個に対して、時間が 10,000 ステップ経過した後の粒子の分布を調べなさい。各頂点  $v_i$  に対して、入ってくる辺の数  $k_i$ 、その頂点に居る粒子数  $n_i$  とし、 $(k_i, n_i)$  をプロットしなさい。

**課題 6** 頂点数 10 のネットワークを作成し、その上の酔歩のシミュレーションを実施しなさい。粒子数 10,000 個に対して、時間が 10,000 ステップ経過した後の粒子の分布を調べなさい。各頂点  $v_i$  に対して、入ってくる辺の数  $k_i$ 、その頂点に居る粒子数  $n_i$  とし、 $(k_i, n_i)$  をプロットしなさい。

**課題 7** シミュレーションの結果を分析し、理論的予想と比較しなさい。

## 4 レポート提出

レポートは以下のように作成し、締切日の 12 時までに提出すること。

- 課題に沿って、内容を記載すること。ただし、単に課題の答えを記載するだけでは不十分である。課題の記述・理解、方法なども課題に前後して記載することで、レポートそのもので問題設定から解決までが読み取れる形式とすること。
- 「考察」として、レポートを通じて得たことを記載すること。
- 正しい日本語または英語で記述すること。Word や L<sup>A</sup>T<sub>E</sub>X などを使って、適切に組版すること。必要なプログラム、スクリプト、図も入れ込むこと。単なるテキストファイルの PDF は受け付けない。
- 他人のプログラム、レポートを写したと判断される場合には、0 点とする。
- 書籍や Web ページ等を参考としている場合には、必ずその出典を明示すること。
- レポートは PDF 形式とし、教務システムのポータルより提出すること。
- 提出前に、必ず、PDF の内容を十分に確認すること。