

オープンソースソフトウェア開発における ネットワーク成長モデルに基づく ソーシャルコミュニティの形成法の提案

The Suggestion of the Formation Method of the Social Community Based on the Network Growth Model in the Open-Source Software Development

上本 拓也

Takuya Uemoto

県立広島大学大学院総合学術研究科経営情報学専攻

Email: yslius7221@gmail.com

市村 匠

Takumi Ichimura

県立広島大学経営情報学部経営情報学科

Email: ichimura@pu-hiroshima.ac.jp

Abstract—The social community in open source software developers has complex structure of network. In fact, we investigated the network structure from node and arcs that represents the relation of SourceForge developers. As a result, we found some characteristic between the project and the number of agents joined in the project in the growing process of software development. By such observations, we developed a agent based model with the grow in process of network structure.

I. はじめに

オープンソースソフトウェア開発におけるコミュニティ形成過程を再現するシミュレーションシステムが Madey らによって開発された [1]. シミュレーション結果より, Madey らはネットワークの成長を再現可能なモデルの提案を行った.

オープンソースソフトウェア開発におけるネットワークコミュニティ形成のシミュレーションシステムでは, エージェントは開発者を表しており, エージェントはソフトウェア開発のためのプロジェクトを生成する. そのプロジェクトには他のエージェントが参加する. エージェントは複数のプロジェクトに参加することで, プロジェクトごとに形成されたコミュニティをつなぐ役割を果たす. このように, オープンソースソフトウェアのネットワークコミュニティが形成されていく様子をシミュレーションし, その過程を観察する.

本研究では, シミュレーションにソフトウェアのタスク量やエージェントのスキルを取り入れ, 形成されたコミュニティとエージェントの行動の関連性を調査する. また, エージェントの行動に利他行動 [4] を入れることで, ソフトウェア開発により最適なコミュニティ形成を提案する.

本論文ではエージェントベースのネットワークシミュレーションを開発し, 形成されたコミュニティの数や開発のタスク量などの関係性について特徴を発見した.

II. SOURCEFORGE.NET

SourceForge.net はオープンソースソフトウェアのソースレポジトリであり, ソフトウェア開発の管理の役割を担う. ソフトウェア開発を行うプロジェクトや開発者などの情報もデータベース [3] に保管されている.

このデータベースより, プロジェクトとそれに参加している開発者のデータを抽出し, プロジェクトに参加している開発者の数 (回数) について調査を行った. 2004 年 9 月時のデータを図 1 に示す. 横軸はプロジェクトの回数, 縦軸はその回数における累積度数をそれぞれ対数をとった値である. 図によると, ある点までは線形的に減少し, それ以降は 2 つの直線で表される幅を持った分布となった. このデータをもとに, 本研究で提案するシミュレーションシステムにおけるパラメータなどを定義するために使用される.

III. シミュレーションシステムの概要

本研究で提案するシミュレーションシステムでは, メジャーエージェントとマイナーエージェントを定義する. メジャーエージェントはプロジェクトの生成を行い, マイナーエージェントがプロジェクトに参加・辞退を行うことでソーシャルネットワークを構築する. メジャーエージェントはプロジェクトマネージャ, マイナーエージェントは開発者を表す. エージェントはプロジェクトに参加すると, 開発タスクを担う.

SourceForge.net はホームページ [2] で 10 個のカテゴリでソフトウェアを分類している. メジャーエージェントはソフトウェアを生成する際, 10 個のカテゴリの中から 1 つカテゴリを選択する. ソフトウェア開発にお

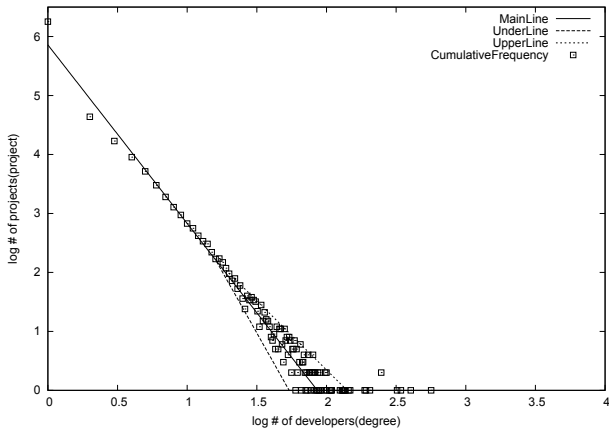


図 1. ソフトウェア開発に参加している開発者の数とその累積度数に関する, SourceForge.net の分布

いて, ソフトウェアには複数のタスクがあり, カテゴリ特有の 1 種類のタスクと全カテゴリに共通する 3 種類のタスクとする. 全カテゴリ共通のタスクとは, ネットワーク・通信処理, データベース処理, グラフィック処理であり, 特定のカテゴリに限らず全てのソフトウェア開発に共通するタスクである.

マイナーエージェントは複数のタスクに関するスキルを持ち, スキルとは単位時間あたりにどの程度のタスクを処理できるかを表す. エージェントがプロジェクトに参加する際, そのプロジェクトのカテゴリのスキルを持たないエージェントは参加することができない. 1つのプロジェクトに複数のエージェントが参加している場合, タスクは均等に与えられる.

A. パラメータ

プロジェクトのタスク量とエージェントのスキルを決定するパラメータを定義する. プロジェクト i に与えられるパラメータは P_i^k であり, エージェント j に与えられるパラメータは S_j^k である.

1) プロジェクトのパラメータ: プロジェクトはソフトウェア開発の中の 1つ以上の開発タスクを割り当てられる. P_i^k はプロジェクト i の開発タスク k の開発のタスク量であり, $[0, 2]$ で与えられる. プロジェクト i のタスク量 \mathbf{P}_i はベクトルであり, 要素は割り当てられた開発タスクのタスク量 P_i^k を用いて $\mathbf{P}_i = \{P_i^1, P_i^2, \dots, P_i^n\}$ と表される.

2) エージェントのパラメータ: S_j^k はエージェントの開発タスクごとの単位時間あたりに処理が可能なタスク量であり, 初期値は $[0, 1]$ でランダムに与えられる. エージェントに割り当てられたタスク量を S_j^k で割ることで開発に関わる時間が求められる. 参加しているプロジェクトごとに開発時間を求め, その和 T_j はエージェント j が開発に費やす時間の合計を表す.

$$T_j = \sum_i T_{ij} = \sum_i \frac{W_{ij}^k}{S_j^k} \quad (1)$$

ここで, W_{ij}^k はエージェント j に割り当てられたプロジェクト i の開発タスク k のタスク量である. 1ステップにおける時間は有限であり, T_j はパラメータ T_{limit} を用いて $0 \leq T_j \leq T_{limit}$ ($0 < T_{limit} \leq 24$) という制約が与えられる.

IV. エージェントの行動

文献 [1] において, エージェントの行動は 4 つある. 本研究で提案するシミュレーションモデルではメジャーエージェントとマイナーエージェントを定義し, 4 つの行動を 2 つのエージェントに割り当てる.

- メジャーエージェント
 - プロジェクトを生成する
- マイナーエージェント
 - プロジェクトに参加する
 - プロジェクトを辞退する
 - 何も行動を起こさない

マイナーエージェントは 1 ステップで 3 つの行動から 1 つを選択し, それぞれの行動アルゴリズムに従って行動する.

A. プロジェクトを生成する

メジャーエージェントがプロジェクトを生成する状況は 2 つある.

- メジャーエージェント j が生成したプロジェクトにおいて, エージェントの負担が大きい場合, そのプロジェクトのタスクを担うサブプロジェクトを生成する.
- 新しいソフトウェアを開発するためのプロジェクトを生成する.

メジャーエージェント j がプロジェクトを生成する行動のアルゴリズムは以下の通りである.

- 1) プロジェクトを生成する確率 New と Sub を求める.
- 2) ランダム値を発生させ, New と比較を行い, 新しいソフトウェアを開発するためのプロジェクトを生成する.
- 3) ランダム値を発生させ, Sub と比較を行い, サブプロジェクトを生成するかどうかを決定する.

メジャーエージェントは, エージェント 1 個あたりの負荷を計算し, しきい値と比較してサブプロジェクトを生成するかどうかを決定する. 負荷の計算は各タスクごとに計算を行う.

$$Sub_i^k = \frac{P_i^k}{N_i^k} \quad (2)$$

$\left\{ \begin{array}{l} IF \ Sub_i^k \geq \ Sub_{threshold} \ THEN \ CreateProject \\ IF \ Sub_i^k < \ Sub_{threshold} \ THEN \ NotCreate \end{array} \right.$

Sub_i^k は, プロジェクト i の開発タスク k におけるエージェント 1 個あたりのタスク量であり, P_i^k はプロジェクト i の開発タスク k に関して割り当てられたタスク

量, N_i^k はそのプロジェクトに参加しているエージェントの数, $Sub_{threshold}$ はタスクを分散するためのしきい値である. 各開発タスクに関して判別を行い, プロジェクトのどのタスクを分散するかを決定する. 決定した分散するタスクを持つサブプロジェクトを生成し, そのタスクのスキルを持つエージェントに参加させる.

B. プロジェクトに参加する

マイナーエージェント j がプロジェクトに参加する行動のアルゴリズムは以下の通りである.

- 1) エージェント j は, 自分が参加していないプロジェクトを1つ選択する.
- 2) 選択されたプロジェクトに対して参加する確率を求める.
- 3) ランダム値を発生させ, 手順2)で求めた確率と比較を行い, 参加するかどうかを決定する.

参加することを決定した場合でも, 開発に費やす時間 T_j を再計算し, T_j が制約である T_{limit} を越える場合, プロジェクトに参加することをやめる.

1) プロジェクトに参加する確率: プロジェクトに参加する確率は, プロジェクトのタスク量とエージェントが持つスキルよりタスクの達成度が求められる. この時, プロジェクトのカテゴリに関するエージェントのスキルが0の場合は参加する確率は0とし, スキルを持っている場合はその値に基づいて計算を行う. プロジェクト i に参加する確率は以下の通りである.

$$J_{ij} = \frac{\sum_{k=1}^n (P_i^k - \sum_{j=1} S_j^k)}{\sum_{k=1}^n P_i^k} \quad (3)$$

$$\begin{cases} IF J_{ij} \geq J_{threshold} THEN JointProject \\ IF J_{ij} < J_{threshold} THEN NotJoin \end{cases}$$

$\sum_{k=1}^n (P_i^k - \sum_{j=1} S_j^k)$ は, プロジェクト j のタスク量からエージェントのスキルを引くことで, プロジェクトのタスクをどの程度達成できるかを求め, $\sum_{k=1}^n P_i^k$ で割ることで割合を求める. この値がしきい値 $J_{threshold}$ を越えるとエージェントはプロジェクトに参加する.

C. プロジェクトを辞退する

マイナーエージェント j がプロジェクトに参加する行動のアルゴリズムは以下の通りである.

- 1) エージェント j は, 自分が参加しているプロジェクトを1つ選択する.
- 2) 選択されたプロジェクトに対して辞退する確率を求める.
- 3) プロジェクトを辞退するしきい値と比較を行い, 辞退するかどうかを決定する.

手順2)で求められる確率は以下の式(4)で求められる.

$$L_{ij} = \sum_{k=1}^n \left(\frac{P_i^k}{N_i^k} - S_j^k \right) \quad (4)$$

$$\begin{cases} IF L_{ij} \geq L_{threshold} THEN LeaveProject \\ IF L_{ij} < L_{threshold} THEN NotLeave \end{cases}$$

$\frac{P_i^k}{N_i^k}$ はエージェント1個あたりのタスク量であり, そのタスク量とエージェント j のスキルがどの程度満たしているかを求める. $L_{threshold}$ はしきい値であり, エージェントの負荷 (L_{ij}) がこの値を越えるとプロジェクトを辞退する.

V. ネットワーク成長モデル

本研究では, エージェントの行動にネットワーク成長モデルである動的適応度を持つ Barabasi-Albert(BA)モデル [5] を適用する. 動的適応度を持つ BA モデルとは, 次数が高い点に高い確率で辺が繋がれる優先的選択を取り入れた BA モデルに動的適応度を加えたものである. ソフトウェアに参加するエージェントをソフトウェアの次数とし, エージェントが参加するソフトウェアを決定する際にこのモデルを適用する. ソフトウェア j を選択する確率は以下の通りである.

$$S_j = \frac{\eta_j k_j}{\sum \eta_j k_j} \quad (5)$$

ここで, η_j は適応度, k_j は次数である. 適応度 η_j は時間経過とともに減少する. 動的適応度により, コミュニティの成長・安定・減退を表すことができる.

VI. シミュレーション

ソフトウェアのタスクやエージェントのスキルを取り入れることでコミュニティ形成にどのような影響が見られるか観察するためにシミュレーションを行った.

メジャーエージェントは1,000個, マイナーエージェントは20,000個で行う. マイナーエージェントの行動ロジックに用いられるパラメータに様々な値を入れて複数回のシミュレーションを行い, 全体として見られる傾向を調査した.

VII. シミュレーション結果

シミュレーションによって形成されたコミュニティに関して, 参加しているエージェントの数(次数)とその累積度数の対数をとった値の分布を図2に示す. 図には分布の他に分布を近似する曲線が描かれており, あらゆるパラメータにおいてこのような近似曲線が描かれることがわかった.

近似曲線は, 最小二乗法により求めた直線に対し, 極座標によって描かれるスパイラルを足した曲線である. また, 横軸のある点を境に幅 δ を持った分布を示している. 極座標方程式, および幅 δ は角度 θ を用いて以下の式で求められる.

$$r = a(\theta) \times \theta \quad (6)$$

$$a(\theta) = 0.07 - 0.07 \times \frac{\theta}{2\pi}$$

$$\delta = 1.6 \times \frac{\theta - \theta_1}{2\pi} \quad (\theta_1 \leq \theta) \quad (7)$$

ここで、 θ_1 とは分岐を開始する点を表す角度を表す。どのようなパラメータにおける結果でも、式(6)と式(7)で表される曲線で近似されることが分かった。

次に、ソフトウェアの次数とマイナーエージェントの行動について考察する。図3はソフトウェアの次数ごとに、ソフトウェアに参加したエージェントの数と辞退したエージェントの数を表している。横軸はソフトウェアの次数、縦軸はそれぞれの行動のエージェントの数を表す。参加と辞退、どちらにおいても次数が高いソフトウェアほど値が線形的に上昇している。加えて、ソフトウェアの次数ごとのタスク量と、生成時に与えられる適応度の平均値について図4に表す。タスク量と適応度が高いほどソフトウェアの次数が高くなることを表す。式(5)より、適応度が高いほどエージェントに選択されやすくなり、かつ式(3)よりタスク量が多いほど参加しやすくなるためだと考えられる。

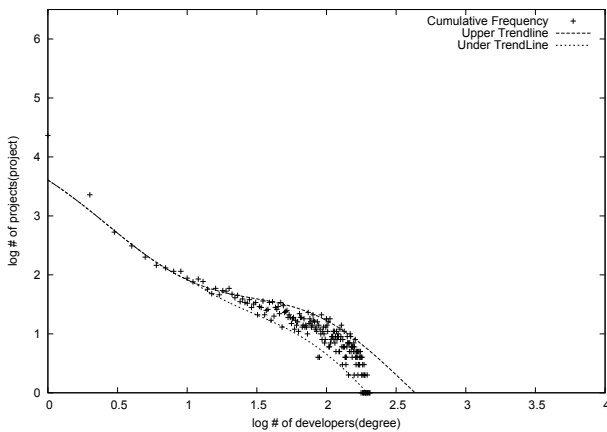


図2. 動的適応度を持つBAモデルに基づいた選択：ソフトウェア開発に参加しているエージェントの数の次数分布

VIII. おわりに

ソフトウェア開発におけるコミュニティ形成について、ソフトウェアのタスクやエージェントのスキルを取り入れてシミュレーションを行った。エージェントがソフトウェアを選択する際には動的適応度を持つBAモデルを取り入れた。ソフトウェアのタスク量と適応度が高くなるほどソフトウェアに参加するエージェントの数が上昇することが分かった。適応度が高いソフトウェアほどエージェントが選択する確率が上昇し、タスク量が多いほど達成されていないタスク量が増加するため参加する確率が高くなるためである。

今後の課題として、エージェントの行動に利他行動を取り入れ、軍隊アリシミュレーションで獲得した知見を活用した、開発においてより最適なコミュニティの形成法を提案することを考えている。

謝辞

本研究の一部は、JSPS 科研費 25330366 の助成を受けたものである。

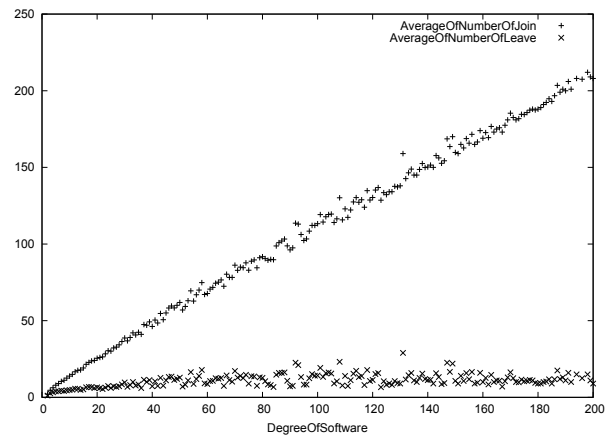


図3. 動的適応度を持つBAモデルに基づいた選択：次数別、ソフトウェアの参加したエージェントの数と辞退したエージェントの数の平均

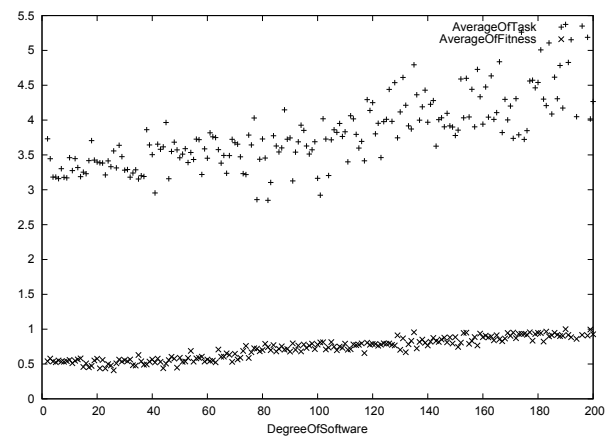


図4. 動的適応度を持つBAモデルに基づいた選択：次数別、ソフトウェアの総タスク量と適応度の初期平均値

参考文献

- [1] Madey, G., Gao, Y., Freeh, V., Tynan, R., Hoffman, C.: 'Agent-based Modeling and Simulation of Collaborative Networks', Proc. of Americas Conference on Information Systems(AMCIS2002) pp.1839-1842(2003)
- [2] <https://sourceforge.net/> (2015/06/22)
- [3] http://srda.cse.nd.edu/mediawiki/index.php/Making_Queryes (2015/06/22)
- [4] T, Ichimura, T, Uemoto, A, Hara.: 'Emergence of Altruism Behavior for Multi Feeding Areas in Army Ant Social Evolutionary System', Proc of IEEE SMC 2014, San Diego, CA, pp170-175(2014)
- [5] T, Gao, G, Madey, V, Freeh.: "Modeling and Simulation of the Open Source Software Community", Agent-Directed Simulation Conference, San Diego, CA, April(2005)

問い合わせ先

〒734-8558

広島県広島市南区宇品東一丁目1番71号
 県立広島大学経営情報学部経営情報学科
 市村 匠