

エッジOSSアップグレードにおける工数管理の 複数シナリオを考慮した最適メンテナンス問題

山口大学大学院・創成科学研究科 田村 慶信 (Yoshinobu Tamura) [†]

[†]Graduate School of Sciences and Technology for Innovation, Yamaguchi University

鳥取大学・名誉教授 山田 茂 (Shigeru Yamada) ^{††}

^{††}Emeritus Professor, Tottori University

1 はじめに

ネットワーク環境は、クラウドコンピューティングからエッジコンピューティングへと変化しつつある。特に、事業継続性の確保や、ネットワーク遅延を避ける構造を積極的に進めることが可能となるため、IoT デバイス側にエッジサーバを設置することが積極的に行われている。IoT により、膨大なデータがクラウドサービスにより、一極集中となることを避ける必要がある。このようなエッジコンピューティングを構築する際には、ソースコードやソフトウェア設計の透明性から、オープンソースソフトウェア (Open Source Software, 以下 OSS と略す) でサービスを提供することが多い。OSS は、クラウドサービス、組込みシステム、サーバ、およびアプリケーションソフトウェアなど、様々な社会システムにおいて利活用されてきた [1, 2]。例えば、OpenStack のようなクラウド OSS が知られている。近年では、こうした OpenStack にもエッジコンピューティングを構成するためのコンポーネントも組み込まれるようになっており、容易にエッジコンピューティング環境を構築できるようになっている。

特に、OSS の場合は、注目されユーザの多いものに限って、頻繁にソフトウェアのバージョンアップグレードが行われている。中でもメジャーバージョンアップグレードは変更の幅が大きく、メジャーバージョンアップグレードが実施された場合、ソフトウェア構造は大きく変化するため、利用上のトラブルや維持管理に多くの時間とメンテナンス工数が必要とされる。一方で、OSS は、ソースコードが公開されているため、そのセキュリティや品質上の問題 [3, 4] に多くの企業が悩まされているという現状もある。

過去に、エッジコンピューティングに対する通信環境の周期性変化を考慮したジャンプ拡散過程モデルを提案した。特に、情報通信工学分野において信号の周期性を表現するために多用されているフーリエ級数に基づき、ノイズの周期的変化を包括した。エッジコンピューティング環境の場合には、ネットワーク、人的資源、地理的制約、人口動態など、様々な環境による影響を受けやすい [5]。本論文では、OSS のアップグレードに伴い工数管理に変化が生じるものと仮定して、複数シナリオを考慮した工数評価について議論する。さらに、実際のデータを利用した提案手法の数値例を示す。

2 ノイズの周期性をもつジャンプ拡散過程モデル

ソフトウェアの信頼性を評価するための数理モデルとして、これまでに数百におよぶソフトウェア信頼性モデルが提案されてきた [3, 4]。ソフトウェア信頼度成長モデルはフォールトデータに基づいてソフトウェア信頼性が評価されるが、フォールト発生の原因となるプロジェクト状態を定量的に評価することができれば、ソフトウェア開発と運用において QCD (Quality, Cost, Delivery) の観点から最適化を図ることが可能となるものと考えられる。本論文では、ソフトウェア開発工数に基づくジャンプ拡散過程モデルについて議論する [6–11]。

過去に、情報通信工学分野において信号の周期性を表現するために多用されているフーリエ級数に基づき、ノイズの周期的変化を包括することで、以下のようなエッジコンピューティングに対する通信環境の周期性変化を考慮したジャンプ拡散過程モデルを提案した [12]。投入開発工数のサンプルパスは、以下ようになる。

$$\Lambda_{je}(t) = \alpha \left[1 - \exp \left\{ -\beta t - p(t)\sigma\omega(t) - q(t) \sum_{i=1}^{M_t(\tau)} \log V_i \right\} \right], \quad (1)$$

$$\Lambda_{js}(t) = \alpha \left[1 - (1 + \beta t) \exp \left\{ -\beta t - p(t)\sigma\omega(t) - q(t) \sum_{i=1}^{M_t(\tau)} \log V_i \right\} \right]. \quad (2)$$

このとき、モデルに含まれるパラメータは以下ようになる。

$\Lambda_*(t)$: 時刻 $t = 0$ で OSS の運用が開始され、任意の時刻 $t (t \geq 0)$ までの投入開発工数,

α : OSS の特定バージョンのリリースに必要とされる開発工数,

$\omega(t)$: Gauss 過程,

σ : 定数パラメータ,

$M_t(\tau)$: $\omega(t)$ とは独立な強度パラメータ τ をもつポアソン過程であり、時刻 t までにジャンプが発生した回数を表す,

V_i : i 回目のジャンプ幅を表す独立な確率変数である,

β : 開発工数の変化率.

また、情報通信工学分野特有の、通信環境におけるノイズの変化を周期的であるものと仮定する。このとき、フーリエ級数展開に基づく以下のような重み関数を定義する。

$$p(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos \left(\frac{2\pi n t}{T} \right) + b_n \sin \left(\frac{2\pi n t}{T} \right) \right]. \quad (3)$$

ここで、 n はサイクル回数、 T は実際のデータ数を表す。このとき、各パラメータは以下のように定義される。

$$a_0 = 0, \quad a_n = 0, \quad b_n = \frac{2}{n\pi} \{1 - (-1)^n\}.$$

このとき、 $q(t)$ は、 $p(t)$ の Y 軸に対して対象となる以下のような関数として定義する。

$$q(t) = \frac{a_0}{2} - \sum_{n=1}^{\infty} \left[a_n \cos \left(\frac{2\pi n t}{T} \right) - b_n \sin \left(\frac{2\pi n t}{T} \right) \right]. \quad (4)$$

情報通信ネットワークにおける研究分野について考えた場合、フーリエ級数に基づく通信環境の数学的表現は、ネットワーク工学の観点からも周期信号や周波数を数学的に取り扱う際に多用されている。本論文では、こうした情報通信工学の分野において多用されているフーリエ級数に基づき、ネットワーク環境からの影響を周期的に変化するノイズとして考えた。

既存モデルに含まれているパラメータ α 、 β 、および σ については、通常確率モデルにおいて一般的に知られている最尤推定法を適用する。本論文におけるジャンプ拡散過程モデルについては、Wiener 過程とジャンプ拡散過程の混合確率過程であることから、パラメータ推定については、その取り扱いが難しいことも知られている。例えば、その代表的なものとしてモーメント法なども提案されているが、決定的なパラメータ推定法は提案されていない。一方、ジャンプ項に含まれるパラメータ集合 θ は一般に

は既知ではないので、実測データなどの利用可能なデータを使って値を推定しなければならない。本論文における提案モデルは、混合確率過程であることから、異なる確率過程とデータ構造の統計的独立性という観点から、ジャンプ項に含まれるパラメータ集合 θ については、実測データと提案モデルとの誤差関数を目的関数とし、この目的関数を最小化するために、遺伝的アルゴリズムを適用した [13–15]。

3 複数シナリオを考慮した最適メンテナンス問題

上述したノイズの周期性をもつジャンプ拡散過程モデルに基づく開発・保守労力の最適化問題を考える。提案モデルから、OSS の運用に伴う開発・保守労力に基づく最適メンテナンス時刻を推定することが可能となる [16, 17]。まず、OSS の開発と運用段階における総開発労力を定式化するため、以下のようなパラメータを定義する。

r_1 : 運用段階における開発工数の重要度,

r_2 : OSS の運用に必要とされる単位時間当りの運用工数,

r_3 : メンテナンス後における保守工数の重要度.

このとき、運用段階における開発工数は、以下のように定式化できる。

$$E_1(t) = r_1 \Lambda(t) + r_2 t. \quad (5)$$

また、メンテナンス後におけるソフトウェアの保守工数は次式で与えられる。

$$E_2(t) = r_3 \{\alpha - \Lambda(t)\}. \quad (6)$$

上記から、総開発工数は、以下のように定式化できる。

$$E(t) = E_1(t) + E_2(t). \quad (7)$$

式 (7) を最小にする時刻 t^* が最適メンテナンス時刻となる。

4 数値例

OpenStack プロジェクト [18] から得られた開発工数データに基づく数値例を示す。図 1 および図 2 は、 $\Lambda_{je}(t)$ および $\Lambda_{js}(t)$ の場合における推定された累積開発工数を表す。次に、 $n = 3$ の場合における重み関数 $p(t)$ および $q(t)$ の推定結果を図 3 に示す。この場合、 n の値を大きくするほど、矩形波に近づくことが分かる。

さらに、 $n = 1, n = 3$, および $n = 5$ の場合における $\Lambda_{je}(t)$ および $\Lambda_{js}(t)$ に基づく総開発工数のサンプルパスを図 4~ 図 9 に示す。図 4~ 図 9 から、指数形ジャンプ拡散過程モデルにおける推定された総開発工数はジャンプ項における影響を受けるが、S 字形ジャンプ拡散過程モデルにおける推定された総開発工数についてはジャンプ項による影響はなく安定していることが分かる。さらに、S 字形ジャンプ拡散過程モデルと S 字形ジャンプ拡散過程モデルにおいて、ノイズの形状の違いも特徴的な点として確認できる。ノイズの小さい時期を目安にマイナーバージョンアップやメジャーバージョンアップを行うことも、開発工数最小化戦略として本モデルを利用することも可能となる。

5 おわりに

ソフトウェアの信頼性は、発見されたフォールトデータや故障発生時間間隔データのような故障に直接的に関係のあるデータが多用されてきた。特に、開発工数とフォールトの関係性に着目した場合、これら両者には原因と結果のような依存関係が存在する。例えば、テスト労力依存型ソフトウェア信頼度

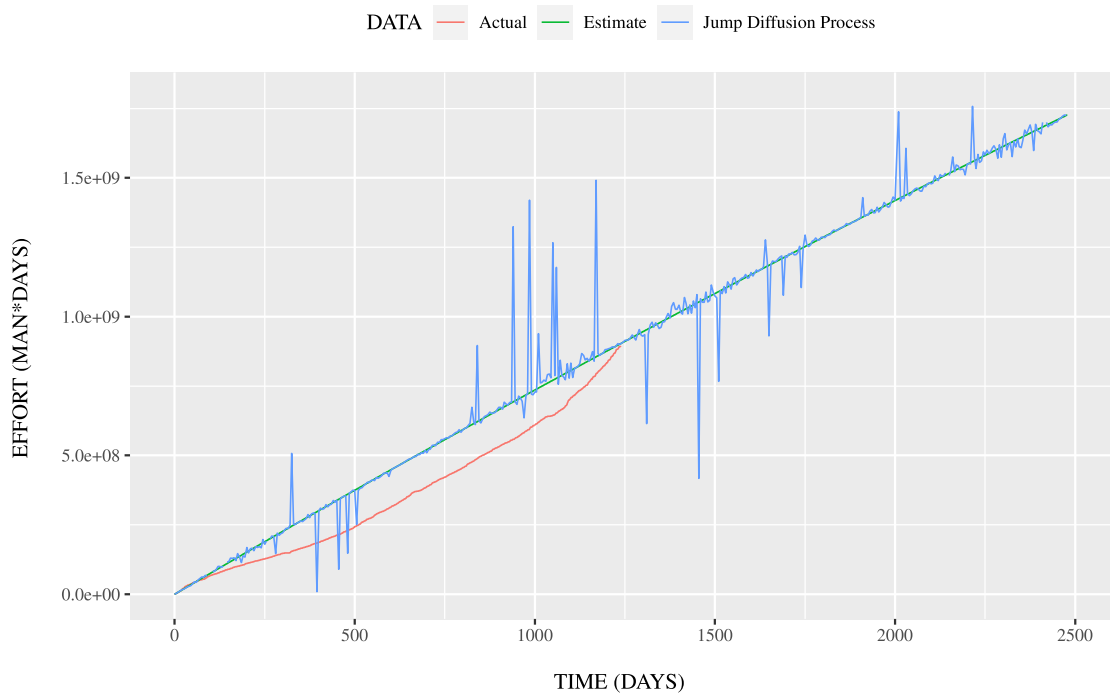


図 1 : $\Lambda_{je}(t)$ の場合における推定された投入開発工数のサンプルパス.

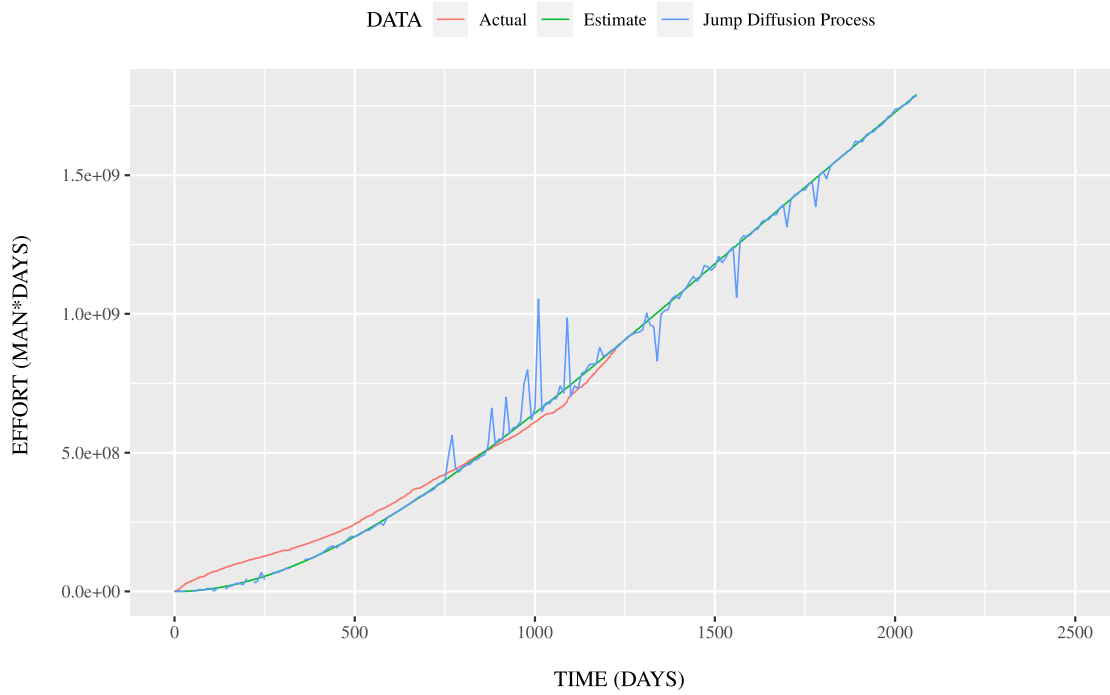


図 2 : $\Lambda_{js}(t)$ の場合における推定された投入開発工数のサンプルパス.

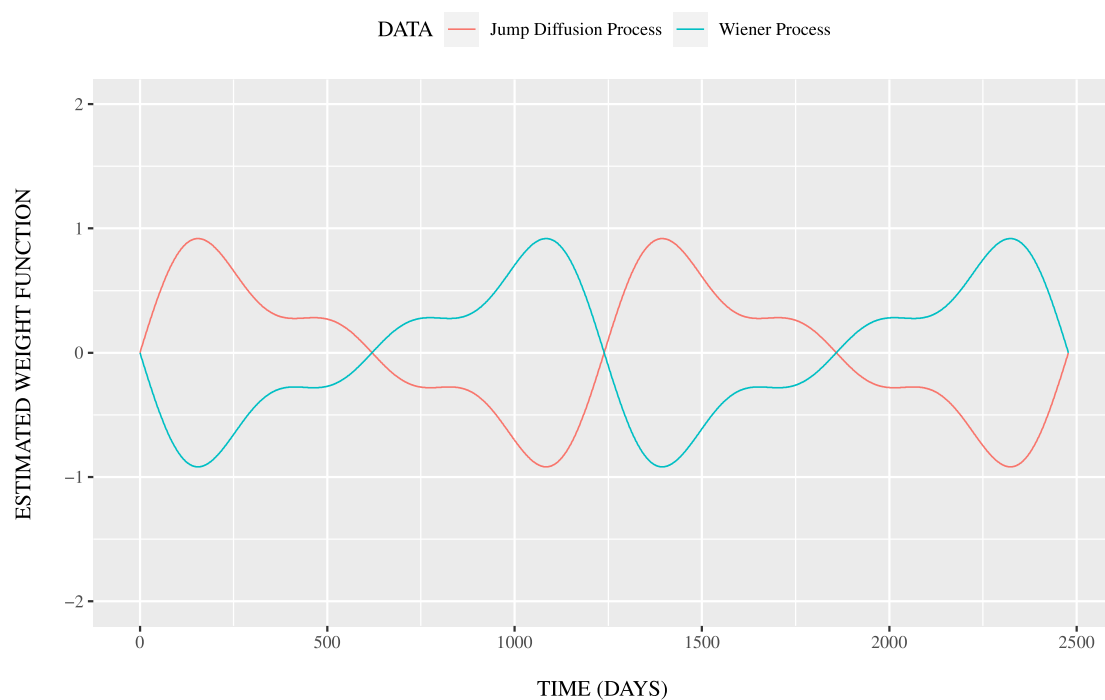


図 3 : $\Lambda_{je}(t)$ および $\Lambda_{js}(t)$ の場合における推定された重み関数 $p(t)$ および $q(t)$.

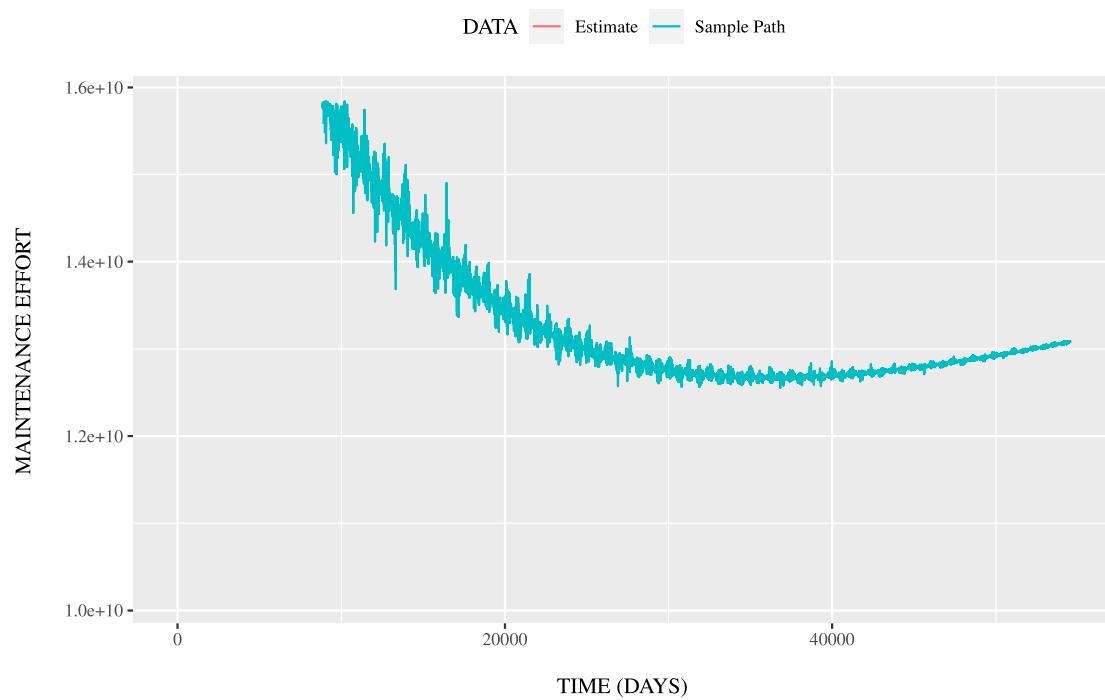


図 4 : $\Lambda_{je}(t)$ の場合における推定された総開発工数 ($n = 1$) .

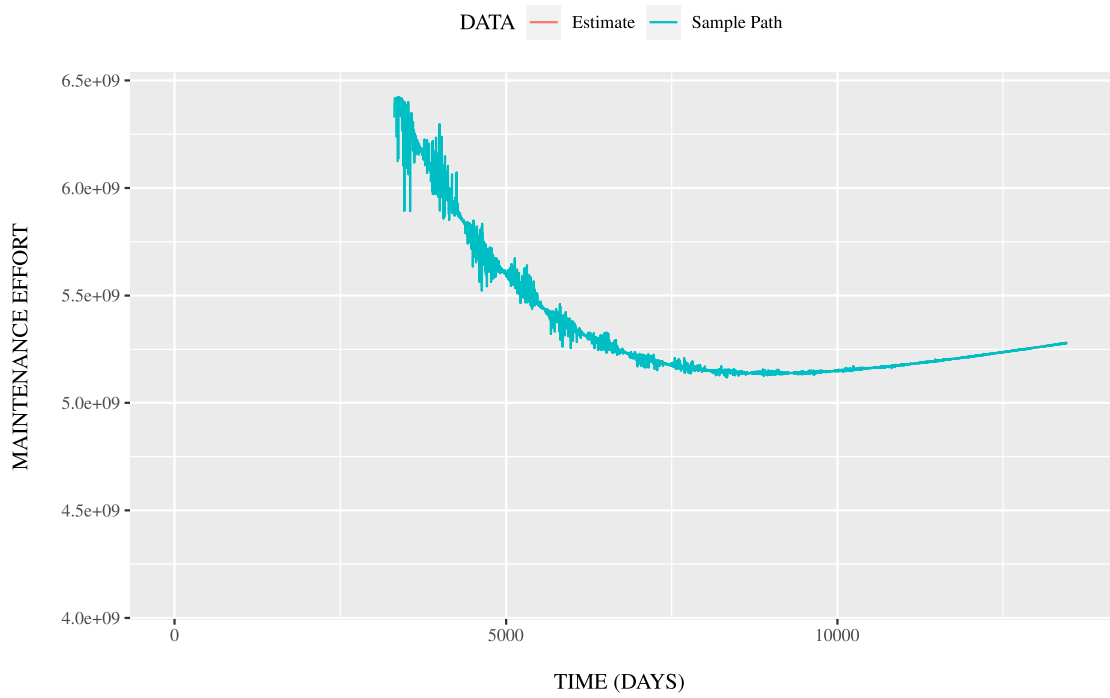


図 5 : $\Lambda_{js}(t)$ の場合における推定された総開発工数 ($n = 1$) .

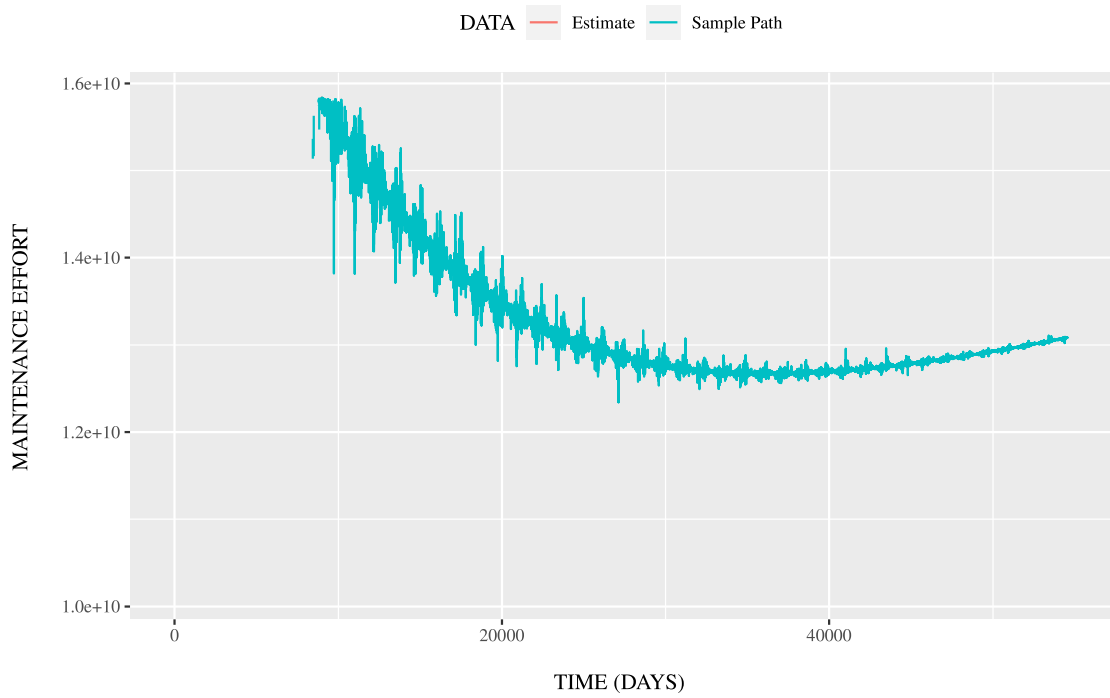


図 6 : $\Lambda_{je}(t)$ の場合における推定された総開発工数 ($n = 3$) .

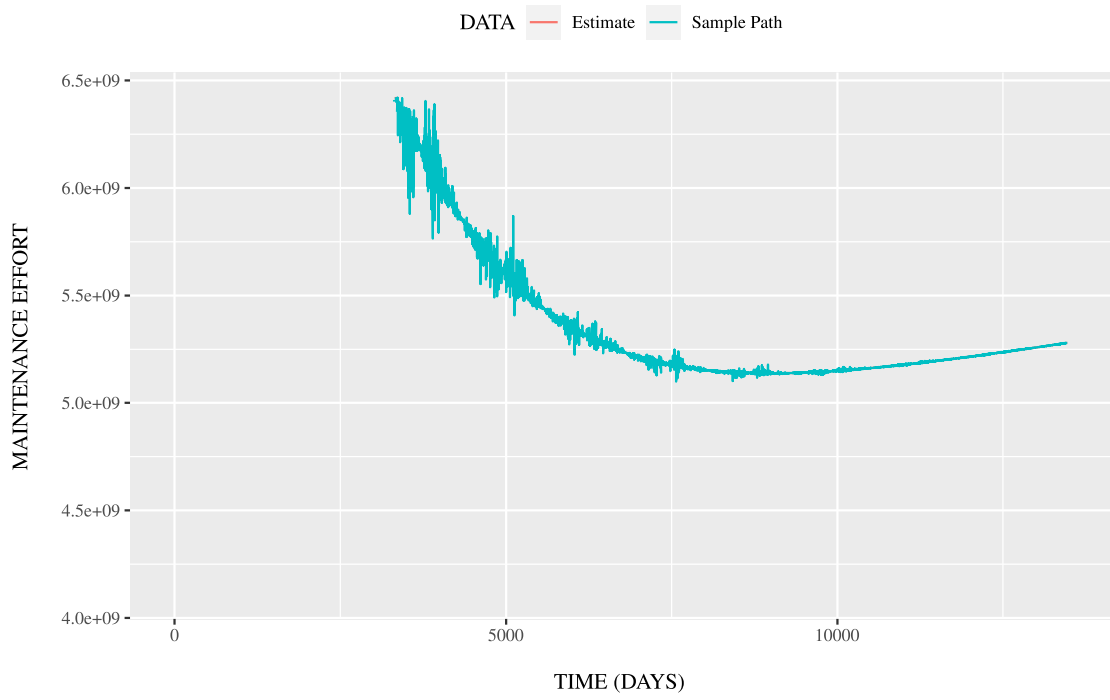


図 7 : $\Lambda_{js}(t)$ の場合における推定された総開発工数 ($n = 3$) .

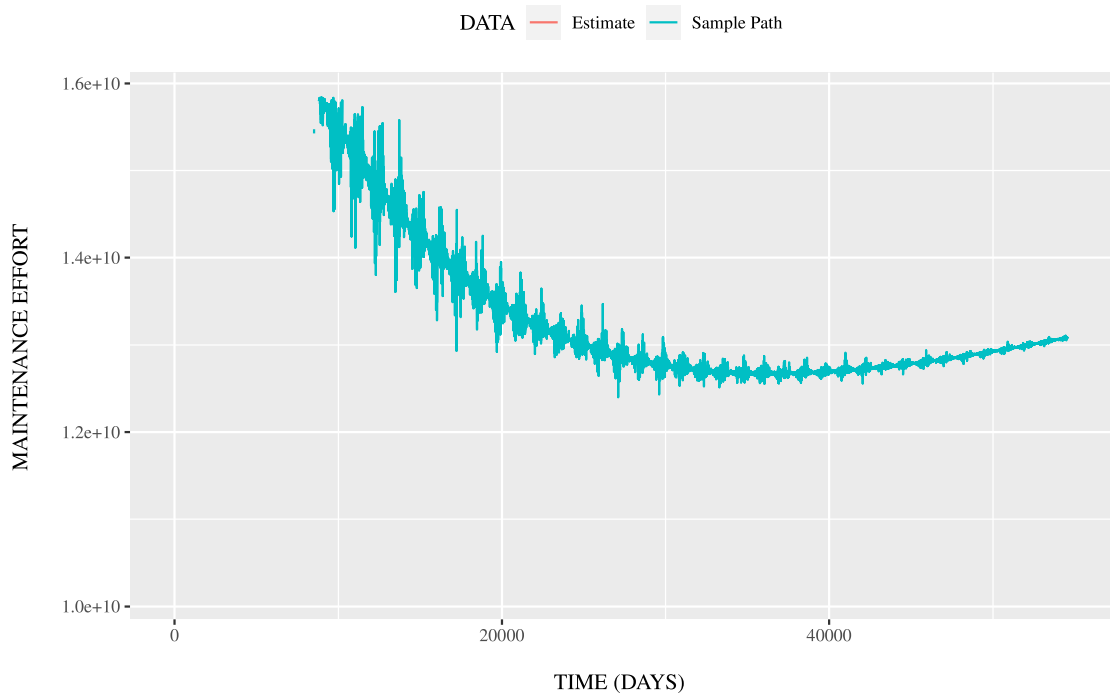


図 8 : $\Lambda_{je}(t)$ の場合における推定された総開発工数 ($n = 5$) .

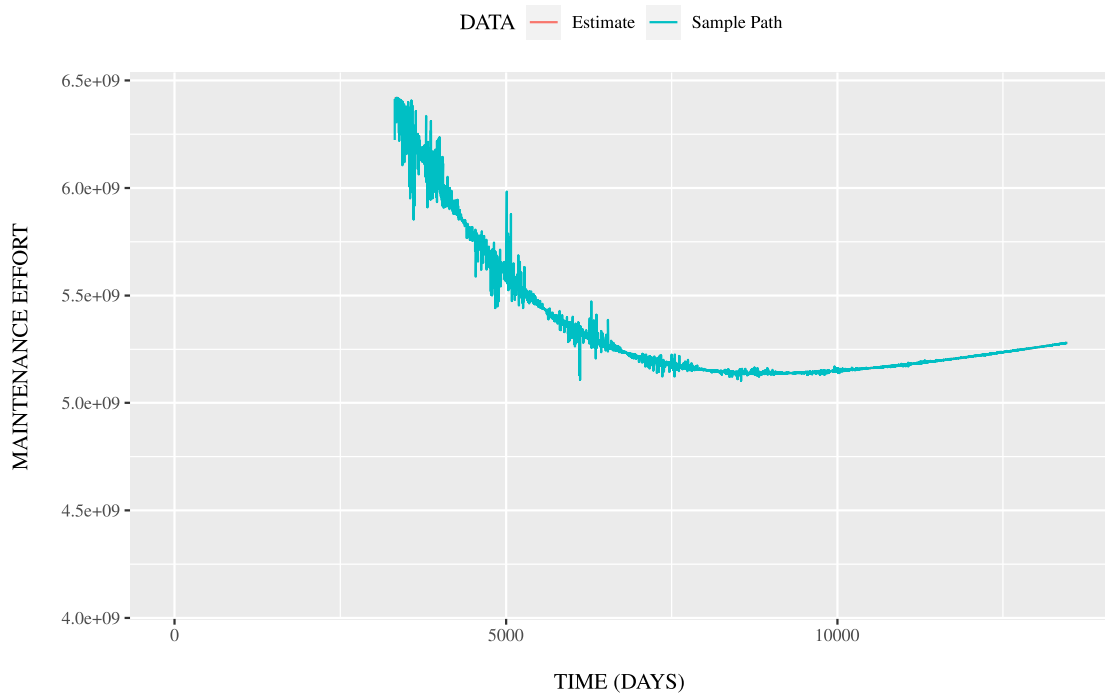


図 9 : $\Lambda_{js}(t)$ の場合における推定された総開発工数 ($n = 5$) .

成長モデルは、その典型的な一例である。開発工数を適切に管理することは、フォールト発生抑制につながるものと考えられる。

過去に、ノイズの周期性をもつジャンプ拡散過程モデルを提案した。本論文では、情報通信工学の分野において多用されているフーリエ級数に基づき、ネットワーク環境からの影響が周期的に変化するノイズとして考えた。このとき、エッジコンピューティング環境により近く、再現性のある手法を目指し、複数シナリオを考慮した最適メンテナンス問題として考慮することにより、具体的な数値例とともに提案手法について議論した。

謝辞

本研究の一部は、JSPS 科研費基盤研究 (C) (課題番号 20K11799) の援助を受けたことを付記する。

参考文献

- [1] S. Yamada and Y. Tamura, *OSS Reliability Measurement and Assessment*, Springer International Publishing, Switzerland, 2016.
- [2] J. Norris, “Mission-critical development with open source software,” *IEEE Software Magazine*, Vol. 21, No. 1, pp. 42–49, 2004.
- [3] S. Yamada, *Software Reliability Modeling: Fundamentals and Applications*, Springer-Verlag, Tokyo/Heidelberg, 2014.
- [4] P.K. Kapur, H. Pham, A. Gupta, and P.C. Jha, *Software Reliability Assessment with OR Applications*, Springer-Verlag, London, 2011.

- [5] S. Wang, Y. Zhao, J. Xu, J. Yuan, and C.H. Hsu, “Edge server placement in mobile edge computing,” *Journal of Parallel and Distributed Computing*, Vol. 127, pp. 160-168, 2019.
- [6] R.C. Merton, “Option pricing when underlying stock returns are discontinuous,” *Journal of Financial Economics*, Vol. 3, Issues 1–2, pp. 125–144, 1976.
- [7] L. Arnold, *Stochastic Differential Equations–Theory and Applications*. John Wiley & Sons, New York, 1974.
- [8] S. Yamada, M. Kimura, H. Tanaka, and S. Osaki, “Software reliability measurement and assessment with stochastic differential equations,” *IEICE Transactions on Fundamentals*, Vol. E77–A, No. 1, pp. 109–116, 1994.
- [9] Y. Tamura, H. Sone, and S. Yamada, “Productivity assessment based on jump diffusion model considering the effort management for OSS project,” *International Journal of Reliability, Quality and Safety Engineering* (World Scientific), Vol. 26, No. 5, pp. 1950022-1–1950022-22, 2019.
- [10] Y. Tamura, H. Sone, K. Sugisaki, and S. Yamada, “A method of parameter estimation in flexible jump diffusion process models for open source maintenance effort management,” *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*, Macau, China, December 15-18, 2019, CD-ROM (Reliability and Maintenance Engineering 2).
- [11] Y. Tamura, H. Sone and S. Yamada, “Flexible jump diffusion process models for open source project with application to the optimal maintenance problem,” *International Journal of Reliability, Quality and Safety Engineering*, Vol. 27, No. 6, World Scientific, pp. 2050020-1–2050020-18, 2020.
- [12] 田村慶信, 曾根寛喜, 上木 僚太, 山田茂, 「エッジコンピューティングに対する通信環境の周期性変化を考慮したジャンプ拡散過程モデル」, 京都大学数理解析研究所講究録「不確定環境下における意思決定数理の新展開」研究集会講究録, No. 2220, pp. 184-190, 2022年5月.
- [13] Y. Tamura and S. Yamada, “Maintenance effort management based on double jump diffusion model for OSS project,” *Annals of Operations Research*, DOI: 10.1007/s10479-019-03170-w, Springer US, Online First, pp. 1-16, February 2019.
- [14] J.H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, 1975.
- [15] K. Sugisaki, Y. Tamura, and S. Yamada, “OSS effort expense optimization based on Wiener process model and GA,” *Journal of Software Engineering and Applications*, Vol. 14, No. 1, pp. 11-25, 2021.
- [16] S. Yamada and S. Osaki, “Cost-reliability optimal software release policies for software systems,” *IEEE Transactions on Reliability*, Vol. R-34, No. 5, pp. 422–424, 1985.
- [17] S. Yamada and S. Osaki, “Optimal software release policies with simultaneous cost and reliability requirements,” *European Journal of Operational Research*, Vol. 31, Nol. 1, pp. 46–51, 1987.
- [18] The OpenStack project, Build the future of Open Infrastructure, <https://www.openstack.org/>