

従属故障ネットワークシステムに対するコ ピュラを用いた信頼性解析

KAGEYAMA, Takao / 影山, 孝夫

(開始ページ / Start Page)

1

(終了ページ / End Page)

21

(発行年 / Year)

2015-03-24

(学位授与年月日 / Date of Granted)

2015-03-24

(学位名 / Degree Name)

修士(工学)

(学位授与機関 / Degree Grantor)

法政大学 (Hosei University)

2014 年度 修士論文

従属故障ネットワークシステムに対する コンピュータを用いた信頼性解析



法政大学大学院理工学研究科
システム工学専攻

13R6207 影山 孝夫

Takao KAGEYAMA

指導教員 木村 光宏 教授

概 要

本研究では、従属故障する複数の素子により成り立つシステムの信頼性解析に、コピュラと呼ばれる関数を用い、高い精度での解析が可能かどうかを考察する。ここで扱う従属故障とは、ある素子の故障が他の素子の瞬間故障率を切り替える（上昇させる）場合のことを言う。

具体的には、2素子並列システムと5素子ネットワークシステムにおいて、故障時刻に関する完全データが得られた場合に、コピュラを用いたいくつかの解析モデルにより既知のモデルよりも精度の高い信頼度関数が得られるかどうかを、シミュレーションに基づき検討する。

Abstract

In this study, we try to investigate whether copula models contribute to the good reliability estimation for multi-component systems in some dependent environment. We assume that a failure occurs on the one component of multi-components, and the surviving component's hazard rate switches to the worse.

Specifically, we consider the 2-component parallel system and the 5-component network system. Also we presume that we can obtain the complete data sets consist of failure occurrence time of each component. We estimate reliability function by the copula models and the traditional methods based on computer simulation, and compare these performances.

目次

1	はじめに	1
1.1	研究の背景と目的	1
1.2	先行研究	1
1.3	論文構成	1
2	コピュラについて	3
2.1	コピュラの定義	3
2.2	Maximal コピュラ	3
2.3	正規コピュラ	3
3	2 素子並列システム	5
3.1	システム構成とデータ生成	5
3.2	解析モデル	6
3.2.1	単純解析モデル	6
3.2.2	推定パラメータ固定モデル	6
3.2.3	Maximal コピュラモデル	8
3.2.4	正規コピュラモデル	8
3.3	実験結果と考察	9
4	5 素子ネットワークシステム	11
4.1	システム構成とデータ生成	11
4.2	仮想的並列システムの構築	13
4.3	解析モデル	14
4.3.1	単純解析モデル	14
4.3.2	Maximal コピュラモデル	14
4.3.3	正規コピュラモデル 1	15
4.3.4	正規コピュラモデル 2	16
4.4	実験結果と考察	17
5	おわりに	19
5.1	まとめと結論	19
5.2	今後の課題	19

1 はじめに

1.1 研究の背景と目的

近年では、IT（情報技術）の目覚ましい進歩と共に高度な情報化社会が大きく広がり、業務支援や高度なシステム機能の実現には、コンピュータシステムが必要不可欠な存在となっている。特にその中でも、インターネットに代表されるような情報ネットワークシステムは様々な場面で利用される。知りたい情報を検索し入手する、航空機や鉄道などの座席を予約する、災害など緊急時の連絡網を作るなど、社会インフラとしてのその重要性は増している。そしてそれに従い、それらのシステムが、ユーザーが求める機能をきちんと提供できるかを正確に見積もる必要も高まっている。

実際に、信頼性工学という学問の一部として、コンポーネント（素子）を組み合わせで出来るシステムが故障を生じること無く正常に動作する確率を高い精度で評価する研究が数多くなされている。しかし、システムの構成素子が独立に故障する場合の研究が多くされているのに対して、従属的に故障する場合の研究はそれほど多くないのが現状といえる。これは、確率変数間に生じると想定される何らかの依存性をしっかりと見極めることが大変難しいことを意味している。だが現実的には、このようなネットワークシステムの故障を考える上で、それを構成する素子間の依存関係を考慮することは非常に重要だと思われる。例えば通信システムにおいて、あるサーバがダウンしたために他のサーバへの通信量・負荷が増大して、そのサーバの故障率が増大するといったシチュエーションは容易に想像がつく。

そこで本研究では、多素子からなる従属故障ネットワークシステムに対して、コピュラ（接合関数）を利用して信頼性解析を試み、従来の手法よりも精度の良い解析ができるか否かについて、シミュレーションに基づいた検討を行う。具体的には、2素子並列システムと5素子ネットワークシステムを取り上げ、シミュレーションにより作成したシステムの各構成素子の故障時刻（瞬間故障率を工夫し、従属故障を表現している。）を元に、観測者がシステムについての精度の高い信頼度関数を得る手段について考察を行う。

1.2 先行研究

従属故障する2素子並列システムにおける信頼性解析について、影山ら [1]、太田ら [2] は、コピュラを用いた信頼性解析を試み、一定の成果を出している。コピュラとは多変数の同時分布とその周辺分布とをつなぎ合わせるような関数のことであり、その中でも Maximal コピュラと呼ばれるコピュラを用いて、実験ならびに考察を行っている。ただ、単純なワイブル解析を行う場合と比べての誤差改善率は低く、また2素子並列システム以外のもっと複雑なシステムに対しての実験は行われていない。

よって本研究では、先行研究による手法を5素子からなるネットワークシステムに適用し効果を確認すること、また誤差改善率のさらなる向上を目指し、得られるデータに工夫を施した上で正規コピュラを用いるというモデルについて実験と検証を行うことを目標とする。

1.3 論文構成

1章で研究背景・目的を明らかにし、2章でコピュラに関する定義を説明する。3章では、2素子並列システムにおけるシステム構成・データ生成方法を説明してから、解析モデルとそれを用いた実験結果を書く。解析モデルとしては、既知の手法である単純解析モデル、推定時にパラメータを固定する推定パラメータ固定モデル、Maximal コピュラによるモデル、正規コピュラによるモデルをそれぞれ考えていく。4章は、3章の2

素子並列システムの拡張として，5 素子ネットワークシステムを考える．先ほどと同様，システム構成・データ生成方法ならびに解析モデルとそれを用いた実験結果を述べる．ここで，最小パスカットと構造関数を用いた仮想的並列化についても言及する．解析モデルは，単純解析モデル，Maximal コピュラによるモデルに正規コピュラによるモデル 2 種類を加えた計 4 つを考える．そして 5 章で，結論と今後の課題をまとめる．

2 コピュラについて

本章では、コピュラと呼ばれる関数について説明を行う．まず定義を示し、その後で本研究で用いる Maximal コピュラならびに正規コピュラについて特に詳しく書き加える．

2.1 コピュラの定義

コピュラとは、多変数の分布関数とその周辺分布関数との関係を示す関数のことである [3]．周辺分布関数 $F_1, F_2, \dots, F_n$ を持つ連続な n 変量同時分布関数を F とすると、

$$\begin{aligned} \Pr(X_1 \leq x_1, X_2 \leq x_2, \dots, X_n \leq x_n) \\ = F(x_1, x_2, \dots, x_n) = C(F_1(x_1), F_2(x_2), \dots, F_n(x_n)), \end{aligned} \quad (2.1)$$

を満たす関数 C が一意に存在し、その C をコピュラ（接合関数）と呼ぶ（スクラーの定理）．コピュラ C は無数に存在する．典型例をいくつか挙げると、変量間の相関構造を行列で表現するコピュラとして正規コピュラやtコピュラ、変量間の相関構造を1種類のパラメータで表現するコピュラとして、クレイトン・コピュラやガンベル・コピュラ、FGM コピュラなどがある．また、その他にもフレシェ - ヘフディング境界によるコピュラなどが存在する．本研究では、これらのコピュラのうち、Maximal コピュラと正規コピュラの2種類を用いて解析を行う．次節より、この2つのコピュラについて詳しく説明を加える．

2.2 Maximal コピュラ

任意のコピュラ C 、および任意の $(F_1(t_1), F_2(t_2), \dots, F_n(t_n)) = (u_1, u_2, \dots, u_n)$ に対してフレシェ - ヘフディング境界が存在し、次の不等式が成り立つ．

$$W(u_1, u_2, \dots, u_n) \leq C(u_1, u_2, \dots, u_n) \leq M(u_1, u_2, \dots, u_n). \quad (2.2)$$

ここで関数 W は、フレシェ - ヘフディング下界と呼ばれ、次の式を満たす．

$$W(u_1, u_2, \dots, u_n) = \max(1 - n + \sum_{i=1}^n u_i, 0). \quad (2.3)$$

また関数 M は、フレシェ - ヘフディング上界と呼ばれ、次の式を満たす．

$$M(u_1, u_2, \dots, u_n) = \min(u_1, u_2, \dots, u_n). \quad (2.4)$$

関数 M の境界を超えるコピュラは存在しないことから、本研究ではこれを Maximal コピュラと呼び、 $M = C_M$ とする．

2.3 正規コピュラ

正規コピュラとは、多変量正規分布と同じ依存構造を持つコピュラであり、以下のように構成される．確率変数 $X_1, X_2, \dots, X_n$ が、相関行列を Σ とする n 変量標準正規分布に従うとき、その分布関数を $\Phi_n(x_1, x_2, \dots, x_n; \Sigma)$ と表すことにする． n 変量標準正規分布の周辺分布は、1変量標準正規分布であるから、

正規コピュラはスクラーの定理より，

$$\begin{aligned} & \Pr(X_1 \leq x_1, X_2 \leq x_2, \dots, X_n \leq x_n) \\ &= \Phi_n(x_1, x_2, \dots, x_n; \Sigma) = C_N(\Phi_1(x_1), \Phi_2(x_2), \dots, \Phi_n(x_n)), \end{aligned} \quad (2.5)$$

となる C_N をもって定義される．

正規コピュラを用いて，周辺分布を 1 変量標準正規分布とすれば多変量正規分布を得ることができる．また，周辺分布を 1 変量標準正規分布以外の分布とすることで、多変量正規分布とは異なる同時分布を得ることもできる．本研究では，周辺分布をワイブル分布として正規コピュラを用いている．注意点として，実際に正規コピュラを用いるためには，相関行列 Σ を推定する必要がある．その点を含めた具体的な使用法については，後で詳しく述べることにする．

3 2素子並列システム

本章では、本研究における2素子並列システムのシステム構成とシミュレーションデータ生成に関する説明をし、続けて、提案する4つの解析モデルの具体的内容とそれらによる実験結果についてをまとめる。

3.1 システム構成とデータ生成

まず単純な2素子並列システムを考える。システムは並列の2素子(S_1, S_2 とする)からなり、片方の素子が故障するともう片方の故障率が上昇すると仮定する。そして両方の素子が故障した時に、システム全体が故障したと見なす。この時、観測者が故障時刻データセット(素子1の故障時刻、素子2の故障時刻、システムの故障時刻)を n 組得ることができると仮定し、それを利用した解析を試みる。

ここで、シミュレーションデータの生成について説明を加える。このデータ生成では、瞬間故障率の定義を利用して素子の故障を判定している。具体的には、「微小時刻 Δt において素子が故障する確率は瞬間故障率と Δt の積に等しい[4].」というものである。

2素子並列システムのシミュレーションデータは以下のようなアルゴリズムによって生成される。

アルゴリズム

手順1 2つの素子は正常動作しているとする。また、時刻を0とし、故障率の初期値を与える。初期故障率 $\lambda_1(t)$ は、 $\lambda_1(t) = a_1 t^{m_1-1}$ とする。これは指数型を含むワイブル型の故障率であり、 a, m をパラメータとしている。

手順2 時刻を微小時刻 Δt だけ進める。正常な素子について、 $[0, 1]$ の一樣乱数を作り、故障率と Δt の積と比較する。もし、一樣乱数よりも故障率と Δt の積の方が大きければ、その素子は故障したとみなし、時刻も記録する。

手順3 もし2つの素子がともに故障していれば、システム全体の故障時刻とし、時刻を記録して終了する。

手順4 もし手順2により片方の素子が故障したものの、残りの素子が正常動作している場合は、残っている素子の故障率を $\lambda_2(t)$ へと切り替える。 $\lambda_2(t) = a_2 t^{m_2-1}$ とする。ただし、故障率が連続的になるよう、 $\lambda_2(t) = a_2 t^{m_2-1} - (a_2 t_f^{m_2-1} - a_1 t_f^{m_1-1})$ という操作を施す。(t_f は先に故障した素子の時刻とする。)

図1, 図2に故障率変化の具体例を載せる。

手順5 手順2へ戻る。

これにより1組のデータセット(素子1の故障時刻、素子2の故障時刻、システムの故障時刻)が入手できる。これを n 回繰り返すことにより、 n 組のデータセットが入手できる。

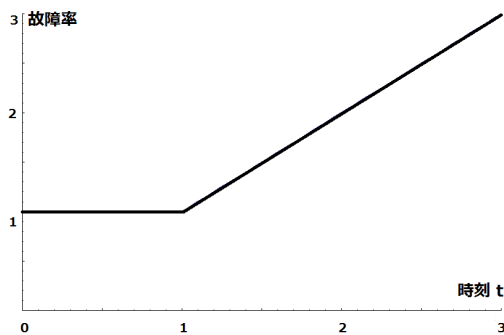


図1 $\lambda_1(t) = 1, \lambda_2(t) = t$ の場合の故障率変化。

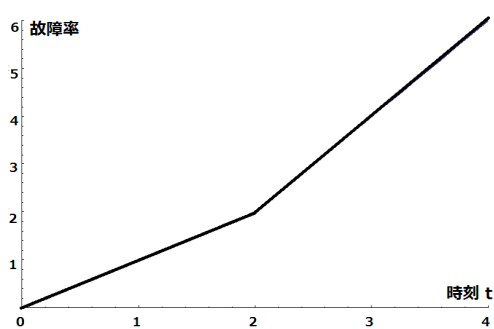


図2 $\lambda_1(t) = t, \lambda_2(t) = 2t$ の場合の故障率変化。

3.2 解析モデル

3.2.1 単純解析モデル

はじめに単純な解析手法から考える．観測者は，システムの故障時刻 ($= t_s$) を知ることができるので，その集合を最尤法によりワイブル分布へと当てはめ，システムの故障時刻の分布関数 ($= u_s$) とすることが可能である．ゆえに，このモデルにおける信頼度関数 $R_T(t)$ は次の式で求まる．

$$R_T(t) = 1 - u_s(t) . \quad (3.1)$$

この解析法は簡素で既知の手法であり，本研究では単純解析モデルとして用いて，他の手法との精度比較を行う．

3.2.2 推定パラメータ固定モデル

本小節と次小節にて，2つのワイブル型分布関数の交差に着目したモデルを考える．以下に詳しく説明する．

i 番目のデータにおける，素子 1, 素子 2 の故障時刻を t_1^i, t_2^i とすると，先に故障した素子の時刻 t_f の集合は以下のように書ける．

$$t_f^i = \min(t_1^i, t_2^i), \quad i = 1, 2, \dots, n, \quad \text{以下これを先故障群と呼ぶ} . \quad (3.2)$$

また，後で故障した素子の時刻 t_l の集合は以下のように書くことができる．

$$t_l^i = \max(t_1^i, t_2^i), \quad i = 1, 2, \dots, n, \quad \text{以下これを後故障群と呼ぶ} . \quad (3.3)$$

このように，2素子の故障時刻を先故障群と後故障群へと分類して，それぞれの群を最尤法によりワイブル分布へと当てはめる．先故障群によるワイブル型分布関数を u_f ，後故障群によるワイブル型分布関数を u_l とする．ここで，2素子並列システム特性より， u_l はシステムの故障時刻の分布関数 u_s と等しい．

さて，一般に2つのワイブル型分布関数が存在するとき，形状・尺度パラメータの値が異なるとすると，図3のように，横軸が0から無限大，縦軸が0から1までの間に必ず交差が起こる．

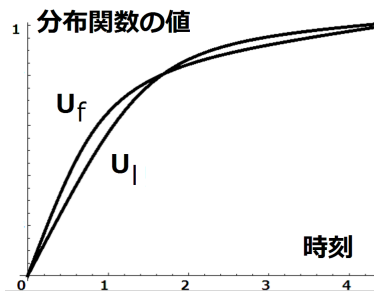


図3 2つのワイブル型分布関数の交差のイメージ図．

証明として，2つのワイブル型分布関数 $1 - \exp(-(\frac{t}{\lambda_1})^{m1})$, $1 - \exp(-(\frac{t}{\lambda_2})^{m2})$ の交点の座標を考える．次の等式を交点 ($= t$) の式へと直す．

$$\begin{aligned}
1 - \exp\left(-\left(\frac{t}{\lambda_1}\right)^{m_1}\right) &= 1 - \exp\left(-\left(\frac{t}{\lambda_2}\right)^{m_2}\right) \\
\exp\left(-\left(\frac{t}{\lambda_1}\right)^{m_1}\right) &= \exp\left(-\left(\frac{t}{\lambda_2}\right)^{m_2}\right) \\
\left(\frac{t}{\lambda_1}\right)^{m_1} &= \left(\frac{t}{\lambda_2}\right)^{m_2} \\
m_1 \log\left(\frac{t}{\lambda_1}\right) &= m_2 \log\left(\frac{t}{\lambda_2}\right) \\
m_1(\log t - \log \lambda_1) &= m_2(\log t - \log \lambda_2) \\
m_1 \log t - m_2 \log t &= m_1 \log \lambda_1 - m_2 \log \lambda_2 \\
\log t &= \frac{m_1 \log \lambda_1 - m_2 \log \lambda_2}{m_1 - m_2} \\
t &= \exp\left(\frac{m_1 \log \lambda_1 - m_2 \log \lambda_2}{m_1 - m_2}\right). \tag{3.4}
\end{aligned}$$

ここで形状・尺度パラメータの値が異なれば、実数の t は必ず存在するので交差がおこる。

ゆえに、前述の推定した先故障群・後故障群のワイブル型分布関数においても交差が起こる。このとき、交差前は先故障群の分布関数 u_f が後故障群の分布関数 u_l の上にあるが、交差後は逆転する。だがこれは不自然であるといえる。なぜなら問題設定より、先に故障した素子の時刻 t_f と後に故障した素子の時刻 t_l は、確率変数であり、任意の時刻 t において、次のような式が成り立つからだ。

$$\begin{aligned}
\Pr(t_l \leq t) &= \Pr(t_f + t_d \leq t) \\
&= \Pr(t_f \leq t - t_d) \quad (t_d > 0). \tag{3.5}
\end{aligned}$$

分布関数は性質として単調非減少であるため、この式より関数 u_f が u_l より常に上に位置すべきであることが確認できる。

ここで真の分布関数（推定ではない。）を考えると、先故障群のみがワイブル型であり、後故障群についてはワイブル型ではないと思われる。しかし、推定の都合上、先故障群の分布関数にも後故障群の分布関数にもワイブル分布を当てはめているため、このような不自然な現象が起こってしまう。よって、後故障群のワイブル型分布関数について修正を行い、精度を向上させることを考える。

精度向上への1つの試みとして、交差をしない分布関数を求める。具体的には、先故障群のワイブル型分布関数 u_f を推定した時に、その形状パラメータの値（ $= m_f$ とする。）を記録し、後故障群のワイブル型分布関数 u_{l2} の推定で、その形状パラメータに m_f を用い、尺度パラメータのみを推定する。これにより、求まった2つのワイブル型分布関数 u_f, u_{l2} は、同じ形状パラメータを持つため、交差はおこらない。この u_{l2} をシステムの故障時刻の分布関数とし、信頼度関数 $R_E(t)$ は次の式で求める。

$$R_E(t) = 1 - u_{l2}(t). \tag{3.6}$$

このモデルを推定パラメータ固定モデルと呼ぶこととする。

3.2.3 Maximal コピュラモデル

次に，Maximal コピュラを用いた解析手法について説明する．前節と同じく交差の解消が目的となる．具体的な解析は以下のように行う．

故障時刻の分類をした後，先故障群と後故障群の2つのワイブル型分布関数 u_f, u_l を最尤法により求める．それらを Maximal コピュラによりつなぎ，それをシステムの分布関数とする．

$$F_M(t_f, t_l) = C_M(u_f, u_l), \quad (3.7)$$

Maximal コピュラの定義より，下側の分布関数を取り出されるために，交差はなくなる．これより，Maximal コピュラを用いた場合のシステム信頼度関数 $R_M(t)$ は，次の式で与えられる

$$R_M(t) = 1 - F_M(t, t). \quad (3.8)$$

3.2.4 正規コピュラモデル

続いて正規コピュラを用いた解析手法について説明する．この手法は，正規コピュラにより従属性を考慮した同時分布を作り，推定に役立てようというものである．具体的な解析の流れは以下ようになる．

はじめに，2素子のうち後で故障した素子の時刻から先に故障した素子の時刻をひいた差分時刻 t_d の集合を，以下のように定義する．

$$t_d^i = \max(t_1^i, t_2^i) - \min(t_1^i, t_2^i), \quad i = 1, 2, \dots, n, \quad \text{以下これを差分時刻群と呼ぶ.} \quad (3.9)$$

そして，2素子の故障時刻を先故障群と差分時刻群へと分類し，それぞれの群を最尤法によりワイブル分布へ当てはめる．先故障群によるワイブル型分布関数を u_f ，差分時刻群によるワイブル型分布関数を u_d とする．次に，正規コピュラを用いるために必要な相関行列 Σ の推定を行う．先故障群と差分時刻群のデータを，適用した周辺分布（ワイブル分布）ごとに， $[0, 1]$ の一様分布へと変換する．

$$a_1^i = u_f(t_f^i), \quad a_2^i = u_d(t_d^i). \quad (3.10)$$

さらに a_1^i, a_2^i を逆関数法により，標準正規分布に従うデータ ω_1^i, ω_2^i へと変換する．そして

$$\frac{1}{n} \sum_{i=1}^n \omega_j^i \omega_k^i \quad (j, k = 1, 2), \quad (3.11)$$

を計算し，これを相関行列 Σ の j 行 k 列成分の推定量 $\hat{\Sigma}_{j,k}$ とする．

このようにして相関行列 $\hat{\Sigma}$ を求めた後，先故障群と差分時刻群を正規コピュラによりつなぎ，同時分布関数

$$F_N(t_f, t_d) = C_N(u_f, u_d), \quad (3.12)$$

を求める．これは先故障群と差分時刻群についての同時分布関数である．そこで，上式の同時分布関数に従う2次元乱数を，十分多い数（ここでは10000個）発生させる．この2次元乱数は， $\{\text{先故障時刻}, \text{差分時刻}\}$ の確率変数となっており，この乱数の和がシステム故障時刻の確率変数となっている．10000個ある「乱数の和」をシステムの故障時刻とみなし，昇順にソートして，カプラン・マイヤー法 [4] により経験分布 $F_{NE}(t)$ を作る．具体的には，昇順にソートした故障時刻のうち i 番目のものを t_i とおくと，時刻 t_i の時の経験分布関数の値は $i/10000$ となり，それらの点を直線でつないだものが $F_{NE}(t)$ である．

これより，正規コピュラを用いた場合のシステム信頼度関数 $R_N(t)$ は，次の式で与えられる．

$$R_N(t) = 1 - F_{NE}(t). \quad (3.13)$$

3.3 実験結果と考察

はじめに，実験の設定についての要点をまとめる．実験は，故障率のパラメータセットを変えて，10 回行う．1 回の実験の流れは以下のようになる．

手順 1 真の分布の代用の分布を作り，その信頼度関数を求める．($=R'(t)$ ，10000 組のデータをシミュレーションにより作り，それを経験分布として，カプラン・マイヤー法により作成する．)

手順 2 100 組のデータをシミュレーションにより作成し，それを単純解析モデル，推定パラメータ固定モデル，Maximal コピュラモデル，正規コピュラモデルでそれぞれ解析して 4 つの信頼度関数を作る．そして，上記の信頼度関数 $R'(t)$ と，誤差 2 乗和をとって比較する．また参考として，先述した先故障群の分布関数と後故障群の分布関数の交差以降に，システムが故障したデータの数記録する．(交差以降にシステムが故障した数が多ければ交差の解消に意味があると言える．) また，それぞれのモデルによる MTTF (平均故障時刻) も求め， $R'(t)$ の MTTF と比べて早い遅いを記録する．

手順 3 ばらつきを取り除くため，手順 2 を 10 回繰り返し，誤差 2 乗和の合計，交差以降のシステム故障データ数の合計，MTTF の早い遅いの平均も記録する．

このような実験を行い，表 1，表 2 のような結果が得られた．

表 1 2 素子並列システムにおける実験結果 1．

実験 No	$\lambda_1(t)$	$\lambda_2(t)$	td SSE	epf SSE	mc SSE	nc SSE	データ数
1	1	t	92.5718	1041.8150	92.5718	68.7516	0
2	1	4t	157.4536	1617.7102	157.4330	139.0691	11
3	2	t	120.6395	793.1533	120.6395	100.3291	0
4	2	4t	107.3692	1146.9459	107.3595	126.7336	1
5	t	2t	214.2358	947.2098	214.2358	218.9211	0
6	t	4t	157.9451	611.9271	157.9451	108.3252	0
7	t	8t	92.9225	436.9538	92.9224	46.5475	0
8	2t	3t	116.5011	774.6396	116.5011	135.9168	0
9	2t	6t	149.5089	776.0715	149.5089	131.6753	0
10	2t	9t	115.5758	613.0614	115.5757	102.5625	1
All			1324.7232	8759.4876	1324.6928	1178.8318	13

表中の，td,epf,mc,nc, はそれぞれ単純解析モデル，推定パラメータ固定モデル，Maximal コピュラモデル，正規コピュラモデルを表す．また，データ数は交差以降のシステム故障データ数を表す．

SSE は誤差 2 乗和であり，All にはそれぞれ総和が入る．

10 種類のパラメータにおいて実験を行ったが，パラメータの特徴による結果の違いはあまり見受けられなかった．故障率が指数型からワイブル型になるもの，ワイブル型からさらに強いワイブル型になるものなどを試したが，結果はばらばらで，傾向は読み取れなかった．

誤差 2 乗和を比較すると，7 種類のパラメータで正規コピュラモデルが 1 番勝っており，全パラメータ合計では，単純解析モデルと比較して約 11.0130% ほど誤差 2 乗和を減らすことができている．これにより，正規コピュラモデルは有益であると考えられる．ただ No. 4，No. 5，No. 8 においては逆に誤差 2 乗和が増えちゃっており，結果のばらつきが大きいとも考えられる．また，推定パラメータ固定モデルに関しては，誤

表 2 2 素子並列システムにおける実験結果 2 .

実験 No	MTTF	td MTTF	epf MTTF	mc MTTF	nc MTTF
1	1.1640	0.0175	0.0081	0.0175	0.0149
2	0.9369	-0.0160	-0.0162	-0.0160	-0.0211
3	0.6613	0.0131	0.0089	0.0131	0.0138
4	0.5792	-0.0111	-0.0154	-0.0111	-0.0130
5	1.4794	-0.0104	-0.0958	-0.0104	-0.0134
6	1.3422	0.0325	-0.0412	0.0325	0.0311
7	1.2412	0.0083	-0.0553	0.0083	0.0061
8	1.0850	-0.0078	-0.0661	-0.0078	-0.0077
9	0.9896	0.0055	-0.0511	0.0055	0.0004
10	0.9331	0.0042	-0.0482	0.0042	0.0027
All		0.0036	-0.0372	0.0036	0.0014

表中の MTTF は、平均故障時刻を表し、All には平均が入る .

差 2 乗和が大きく増加してしまっており、モデルとしては使えそうにない .

次に単純解析モデルと Maximal コピュラモデルの比較では、No . 2 など、ごく小さな差で Maximal コピュラが勝っており、結果が悪くなるパラメータはない . 交差以降のデータ数と重ねてみても、データ数の多い No . 2 において Maximal コピュラの効果が一番大きく出ていることから、2 つのワイブル型分布関数の交差を取り除くというアイデアの成果があったと確認できる . 全パラメータ合計で見ると、0.0023% ほど誤差 2 乗和が減っており、これは本当にごく小さな改善であるが、Maximal コピュラを用いることで結果が悪くなることは無いということが強みといえるのではないだろうか .

続けて MTTF の観点から、各モデルを考える . 真の分布の代用の分布の MTTF (以下、真の MTTF と略す .) と各モデルの MTTF を比較する . 単純解析モデルの MTTF と Maximal コピュラモデルの MTTF はすべてのパラメータで等しく、6 つのパラメータにおいて真の MTTF よりも遅い値を取っている . 全パラメータ平均でも、0.0036 ほど遅れている . 正規コピュラモデルの MTTF も、6 つのパラメータで真の MTTF より遅い値を取っているが、全パラメータ平均での遅れは 0.0014 であり、遅れの幅は小さい .

信頼性工学においては、フェールセーフという考え方がある . 簡潔に言うと、システムを安全に制御することを第一とした考え方である . この考え方を基本として見ると、MTTF (平均故障時刻) を遅く見積もることは良くないと言える . よって MTTF の見積もりから考えても、遅れの幅の小さい正規コピュラモデルは用いる価値が高いと言えるのではないだろうか .

4 5 素子ネットワークシステム

本章では、本研究における5素子ネットワークシステムのシステム構成とシミュレーションデータ生成に関する説明をし、続けて4つの解析モデルの具体的内容と、それらによる実験結果についてをまとめる。

4.1 システム構成とデータ生成

前章の2素子並列システムをより一般的に拡張した、5素子からなるネットワークシステムに対しても解析を行う。システムは素子 $S_1 \sim S_5$ によって成り立っており、構成は図4のようにになっている。

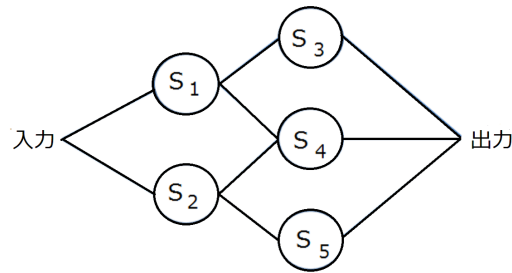


図4 5素子ネットワークシステムの構成図

ある1つの素子が故障すると他の素子に負荷がかかり、それらの素子の故障率が変化（主に上昇）していくと仮定する。例えば、 S_1 が故障すると、 S_2 は大きな影響を受けると考えるのが自然である。よってこの場合、 S_2 の負荷、故障率は大きく上昇する。またそれに伴い S_4, S_5 の負荷、故障率も上昇する。（ただし、 S_3 は孤立するため逆に負荷、故障率は下降し0となる。）

入力から出力まで到達することが出来なくなった場合をシステム全体の故障とみなす。この時、観測者が故障時刻データセット（素子1の故障時刻、素子2の故障時刻、...、素子5の故障時刻、システムの故障時刻）を得ることができると仮定し、それを利用した解析を試みる。

ここで、シミュレーションデータの生成について説明を続ける。5素子ネットワークシステムのシミュレーションデータは以下のようなアルゴリズムによって生成される。

アルゴリズム

前準備 システムの全素子の故障率を全素子の正常動作・故障の状況別に作っておく。（表3参照。）

手順1 すべての素子は正常動作しているとする。また、時刻を0とし、故障率の初期値を与える。初期故障率 $\lambda_1(t)$ は、 $\lambda_1(t) = a_1 t^{m_1-1}$ とする。これは指数型を含むワイブル型の故障率であり、 a, m をパラメータとしている。

手順2 時刻を微小時刻 Δt だけ進める。それぞれの素子について、 $[0, 1]$ の一様乱数を作り、故障率と Δt の積と比較する。もし、一様乱数よりも故障率と Δt の積の方が大きければ、その素子は故障したとし、時刻も記録する。

手順3 もし、入力から出力まで到達できないならば、システム全体の故障時刻とし、時刻を記録して終了する。

手順4 もし手順2によりある素子が故障したものの、入力から出力まで到達可能ならば、システムは動作しており、残っている素子の故障率を状況により、 $\lambda_0(t) = 0$ 、 $\lambda_2(t) = a_2 t^{m_2-1}$ 、 $\lambda_3(t) = a_3 t^{m_3-1}$ 、

$\lambda_4(t) = a_4 t^{m_4-1}$ のどれかへと変化させる．また，2 素子並列システムの場合と同じようにして，故障率が連続するように工夫する．図 5，図 6 に故障率変化の一例を載せる．

手順 5 手順 2 へ戻る．

これにより 1 組のデータセット（素子 1 の故障時刻，素子 2 の故障時刻，…，素子 5 の故障時刻，システムの故障時刻）が入手できる．これを n 回繰り返すことにより， n 組のデータセットが入手できる．

表 3 素子状態と素子故障率の対応表．

状態 No	素子状態 S_1, S_2, S_3, S_4, S_5	素子故障率 S_1, S_2, S_3, S_4, S_5
1	1, 1, 1, 1, 1	$\lambda_1(t), \lambda_1(t), \lambda_1(t), \lambda_1(t), \lambda_1(t)$
2	0, 1, 1, 1, 1	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_2(t), \lambda_2(t)$
3	1, 0, 1, 1, 1	$\lambda_3(t), \lambda_0(t), \lambda_2(t), \lambda_2(t), \lambda_0(t)$
4	1, 1, 0, 1, 1	$\lambda_1(t), \lambda_1(t), \lambda_0(t), \lambda_2(t), \lambda_2(t)$
5	1, 1, 1, 0, 1	$\lambda_1(t), \lambda_1(t), \lambda_2(t), \lambda_0(t), \lambda_2(t)$
6	1, 1, 1, 1, 0	$\lambda_1(t), \lambda_1(t), \lambda_2(t), \lambda_2(t), \lambda_0(t)$
7	0, 1, 0, 1, 1	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_2(t), \lambda_2(t)$
8	0, 1, 1, 0, 1	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_0(t), \lambda_4(t)$
9	0, 1, 1, 1, 0	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_4(t), \lambda_0(t)$
10	1, 0, 0, 1, 1	$\lambda_3(t), \lambda_0(t), \lambda_0(t), \lambda_4(t), \lambda_0(t)$
11	1, 0, 1, 0, 1	$\lambda_3(t), \lambda_0(t), \lambda_4(t), \lambda_0(t), \lambda_0(t)$
12	1, 0, 1, 1, 0	$\lambda_3(t), \lambda_0(t), \lambda_2(t), \lambda_2(t), \lambda_0(t)$
13	1, 1, 0, 0, 1	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_0(t), \lambda_4(t)$
14	1, 1, 0, 1, 0	$\lambda_1(t), \lambda_1(t), \lambda_0(t), \lambda_4(t), \lambda_0(t)$
15	1, 1, 1, 0, 0	$\lambda_3(t), \lambda_0(t), \lambda_4(t), \lambda_0(t), \lambda_0(t)$
16	0, 1, 0, 0, 1	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_0(t), \lambda_4(t)$
17	0, 1, 0, 1, 0	$\lambda_0(t), \lambda_3(t), \lambda_0(t), \lambda_4(t), \lambda_0(t)$
18	1, 0, 0, 1, 0	$\lambda_3(t), \lambda_0(t), \lambda_0(t), \lambda_4(t), \lambda_0(t)$
19	1, 0, 1, 0, 0	$\lambda_3(t), \lambda_0(t), \lambda_4(t), \lambda_0(t), \lambda_0(t)$

素子状態における 1 は正常動作，0 は故障を表す．

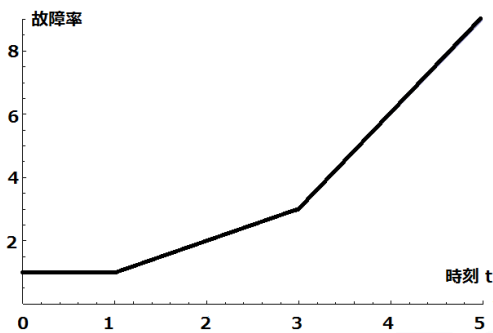


図 5 $\lambda_1(t) = 1, \lambda_2(t) = t, \lambda_4(t) = 3t$ の場合の故障率変化．

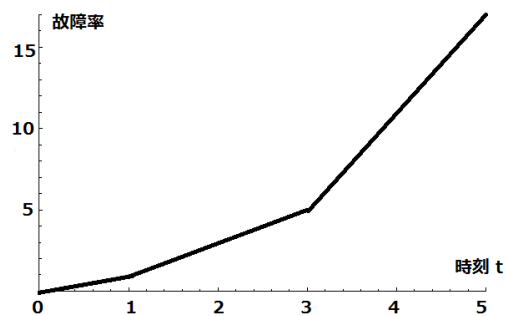


図 6 $\lambda_1(t) = t, \lambda_2(t) = 2t, \lambda_4(t) = 6t$ の場合の故障率変化．

4.2 仮想的並列システムの構築

この5素子ネットワークシステムにおいて，本研究の理論を適用するためには，ネットワークシステムを並列システムへと仮想的に置き換える必要がある．そのために最小パスセットと呼ばれる集合を考える．パスセットとは，それに属する要素がすべて正常ならシステムが正常となる部分集合のことであり，あるパスセットの部分集合にパスセットとなるものが存在しないとき，そのパスセットを最小パスセットと呼ぶ [5]．この最小パスセットを利用して構造関数（システムの状態（正常: 1，故障: 0）を素子の状態を表す変数の和と積によって表したもの）を構築し，仮想的な並列システムを作り出す．今回のネットワークであれば，最小パスセットは $(S_1, S_3), (S_1, S_4), (S_2, S_4), (S_2, S_5)$ の4つであるから，システムの状態を z ，素子 i の状態 x_i とおいた時の，このネットワークにおける構造関数は，

$$z = 1 - (1 - x_1 \cdot x_3) \cdot (1 - x_1 \cdot x_4) \cdot (1 - x_2 \cdot x_4) \cdot (1 - x_2 \cdot x_5), \quad (4.1)$$

となる．これは，図7のような並列システムと等しいと考えることができる．よって，次節より解析を考える際には，この仮想的な並列システムを用いて解析を進める．

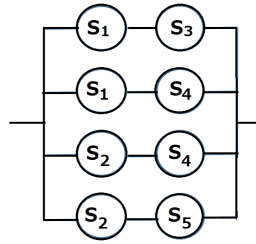


図7 仮想的並列システムの構成図．

4.3 解析モデル

4.3.1 単純解析モデル

2 素子並列システムの時に考えた単純解析モデルをそのまま適用できる．すなわち，システムの故障時刻 ($= t_s$) の集合を最尤法によりワイブル分布へと当てはめ，システムの故障時刻の分布関数 ($= u_s$) として，信頼度関数 $R_T(t)$ を次のように求める．

$$R_T(t) = 1 - u_s(t) . \quad (4.2)$$

ここでも，このモデルを単純解析モデルとして用いて，他の手法との精度比較を行う．

4.3.2 Maximal コピュラモデル

Maximal コピュラを用いた解析手法について説明する．先に述べた仮想的並列化を利用することになる．具体的な解析の流れは次のようになる．

仮想的並列化によって，図 7 のように 4 本の枝ができあがった．この枝についての故障時刻を考える． i 番目のデータにおける素子 1 から素子 5 までの故障時刻をそれぞれ $t_1^i, \dots, t_5^i$ とすると， i 番目のデータの 4 本の枝の故障時刻 $b_1^i, b_2^i, b_3^i, b_4^i$ はそれぞれ次のようになる．

$$\begin{aligned} b_1^i &= \min(t_1^i, t_3^i) , \\ b_2^i &= \min(t_1^i, t_4^i) , \\ b_3^i &= \min(t_2^i, t_4^i) , \\ b_4^i &= \min(t_2^i, t_5^i) . \end{aligned} \quad (4.3)$$

さらに， $b_1^i, b_2^i, b_3^i, b_4^i$ を並び替えて，1 本目故障時刻 r_1^i ，2 本目故障時刻 r_2^i ，3 本目故障時刻 r_3^i ，4 本目故障時刻 r_4^i を作る．すなわち，

$$\begin{aligned} r_1^i &= (b_1^i, b_2^i, b_3^i, b_4^i) \text{ のうち 1 番目に小さい時刻} , \\ r_2^i &= (b_1^i, b_2^i, b_3^i, b_4^i) \text{ のうち 2 番目に小さい時刻} , \\ r_3^i &= (b_1^i, b_2^i, b_3^i, b_4^i) \text{ のうち 3 番目に小さい時刻} , \\ r_4^i &= (b_1^i, b_2^i, b_3^i, b_4^i) \text{ のうち 4 番目に小さい時刻} , \end{aligned} \quad (4.4)$$

となる．このような並べ替えの作業を n 個のデータ全てについて行い，1 本目故障時刻群， $\dots$ ，4 本目故障時刻群を作る．

続いて，それぞれの群を最尤法によりワイブル分布へ当てはめる．1 本目故障時刻群によるワイブル型分布関数を u_1 ，2 本目故障時刻群によるワイブル型分布関数を u_2 ，3 本目故障時刻群によるワイブル型分布関数を u_3 ，4 本目故障時刻群によるワイブル型分布関数を u_4 とする．そして，この 4 つのワイブル分布関数を Maximal コピュラによりつなぎ，それをシステムの分布関数とする．

$$F_M(r_1, r_2, r_3, r_4) = C_M(u_1, u_2, u_3, u_4) . \quad (4.5)$$

これより，Maximal コピュラを用いた場合のシステム信頼度関数 $R_M(t)$ は，次の式で与えられる

$$R_M(t) = 1 - F_M(t, t, t, t) . \quad (4.6)$$

4.3.3 正規コピュラモデル 1

正規コピュラを用いた解析を 2 種類試みる．ここでは，まず 1 つ目を説明する．

前節で用いた仮想的並列化による 4 本の枝をここでも用いる． i 番目のデータの 4 本の枝の故障時刻 $b_1^i, b_2^i, b_3^i, b_4^i$ はそれぞれ先ほどの定義と同じであり，それらを並べ替えてできる， $r_1^i, r_2^i, r_3^i, r_4^i$ も同様である．ここで，1 本目故障時刻と 2 本目故障時刻との差分時刻 t_{d12} ，ならびに 2 本目故障時刻と 3 本目故障時刻との差分時刻 t_{d23} ，3 本目故障時刻と 4 本目故障時刻との差分時刻 t_{d34} を，それぞれ次のように定義する．

$$\begin{aligned} t_{d12}^i &= r_2^i - r_1^i, & i &= 1, 2, \dots, n, & \text{以下これを差分時刻群 12 と呼ぶ.} \\ t_{d23}^i &= r_3^i - r_2^i, & i &= 1, 2, \dots, n, & \text{以下これを差分時刻群 23 と呼ぶ.} \\ t_{d34}^i &= r_4^i - r_3^i, & i &= 1, 2, \dots, n, & \text{以下これを差分時刻群 34 と呼ぶ.} \end{aligned} \quad (4.7)$$

そして，1 本目故障時刻群，差分時刻群 12，差分時刻群 23，差分時刻群 34 の 4 つの群を，それぞれ最尤法によりワイブル分布へ当てはめる．1 本目故障時刻群によるワイブル型分布関数を u_1 ，差分時刻群 12 によるワイブル型分布関数を u_{d12} ，差分時刻群 23 によるワイブル型分布関数を u_{d23} ，差分時刻群 34 によるワイブル型分布関数を u_{d34} とする．

続いて，正規コピュラを用いるために必要な相関行列 Σ の推定を行う．4 つの群のデータを適用した周辺分布（ワイブル分布）ごとに， $[0, 1]$ の一様分布へと変換する．

$$a_1^i = u_1(r_1^i), \quad a_2^i = u_{d12}(t_{d12}^i), \quad a_3^i = u_{d23}(t_{d23}^i), \quad a_4^i = u_{d34}(t_{d34}^i), \quad (4.8)$$

さらに $a_1^i, a_2^i, a_3^i, a_4^i$ を逆関数法により，標準正規分布に従うデータ $\omega_1^i, \omega_2^i, \omega_3^i, \omega_4^i$ ，へと変換する．そして

$$\frac{1}{n} \sum_{i=1}^n \omega_j^i \omega_k^i \quad (j, k = 1, 2, 3, 4), \quad (4.9)$$

を計算し，これを相関行列 Σ の j 行 k 列成分の推定量 $\hat{\Sigma}_{j,k}$ とする．

このようにして相関行列 $\hat{\Sigma}$ を求めた後，4 つの群を正規コピュラによりつなぎ，同時分布関数

$$F_{N1}(r_1, t_{d12}, t_{d23}, t_{d34}) = C_{N1}(u_1, u_{d12}, u_{d23}, u_{d34}), \quad (4.10)$$

を求める．これは 1 本目故障時刻群と差分時刻群 12，差分時刻群 23，差分時刻群 34 についての同時分布関数である．

そこで，上式の同時分布関数に従う 4 次元乱数を，十分多い数（10000 個）発生させる．この 4 次元乱数は， $\{1 \text{ 本目故障時刻}, \text{差分時刻 12}, \text{差分時刻 23}, \text{差分時刻 34}\}$ の確率変数となっており，この乱数の和がシステム故障時刻の確率変数となっている．10000 個ある「乱数の和」をシステムの故障時刻とみなし，ソートして，カプラン・マイヤー法により経験分布 $F_{N1E}(t)$ を作る．前章と同様に，昇順にソートした故障時刻のうち i 番目のものを t_i とおくと，時刻 t_i の時の経験分布関数の値は $i/10000$ となり，それらの点を直線でつないだものが $F_{N1E}(t)$ となる．これより，正規コピュラを用いた場合の 1 つ目のシステム信頼度関数 $R_{N1}(t)$ は，次の式で与えられる

$$R_{N1}(t) = 1 - F_{N1E}(t). \quad (4.11)$$

4.3.4 正規コピュラモデル 2

正規コピュラを用いた解析の 2 つ目を説明する．仮想的並列化による 4 本の枝の各定義は同様である．

ここで，1 本目故障時刻と 4 本目故障時刻との差分時刻 t_{d14} を次のように定義する．

$$t_{d14}^i = r_4^i - r_1^i, \quad i = 1, 2, \dots, n, \quad \text{以下これを差分時刻群 14 と呼ぶ.} \quad (4.12)$$

そして，1 本目故障時刻群と差分時刻群 14 の 2 つの群を，それぞれ最尤法によりワイブル分布へ当てはめる．

1 本目故障時刻群によるワイブル型分布関数を u_1 ，差分時刻群 14 によるワイブル型分布関数を u_{d14} とする．

続いて，正規コピュラを用いるために必要な相関行列 Σ の推定を行う．2 つの群のデータを適用した周辺分布（ワイブル分布）ごとに， $[0, 1]$ の一様分布へと変換する．

$$a_1^i = u_1(r_1^i), \quad a_2^i = u_{d14}(t_{d14}^i), \quad (4.13)$$

さらに a_1^i, a_2^i を逆関数法により，標準正規分布に従うデータ ω_1^i, ω_2^i へと変換する．そして

$$\frac{1}{n} \sum_{i=1}^n \omega_j^i \omega_k^i \quad (j, k = 1, 2), \quad (4.14)$$

を計算し，これを相関行列 Σ の j 行 k 列成分の推定量 $\hat{\Sigma}_{j,k}$ とする．

このようにして相関行列 $\hat{\Sigma}$ を求めた後，2 つの群を正規コピュラによりつなぎ，同時分布関数

$$F_{N2}(r_1, t_{d14}) = C_{N2}(u_1, u_{d14}), \quad (4.15)$$

を求める．これは 1 本目故障時刻群と差分時刻群 14 についての同時分布関数である．

そこで，上式の同時分布関数に従う 2 次元乱数を，十分多い数（10000 個）発生させる．この 2 次元乱数は， $\{1 \text{ 本目故障時刻}, \text{差分時刻 14}\}$ の確率変数となっており，この乱数の和がシステム故障時刻の確率変数となっている．10000 個ある「乱数の和」をシステムの故障時刻とみなし，ソートして，カプラン・マイヤー法により経験分布 $F_{N2E}(t)$ を作る．これも先ほどと同様，昇順にソートした故障時刻のうち i 番目のものを t_i とおくと，時刻 t_i の時の経験分布関数の値は $i/10000$ となり，それらの点を直線でつないだものが $F_{N2E}(t)$ となる．これより，正規コピュラを用いた場合のシステム信頼度関数 $R_{N2}(t)$ は，次の式で与えられる．

$$R_{N2}(t) = 1 - F_{N2E}(t). \quad (4.16)$$

4.4 実験結果と考察

実験の設定についての要点を確認する．実験は，故障率のパラメータセットを変えて，13 回行う．1 回の実験の流れは，2 素子並列システムの時とほぼ同じで，以下になる．

- 手順 1 真の分布の代用の分布を作り，その信頼度関数を求める．($=R'(t)$ ，10000 組のデータをシミュレーションにより作り，それを経験分布として，カプラン・マイヤー法により作成する) を一つ決める．
- 手順 2 100 組のデータをシミュレーションにより作成し，それを単純解析モデル，Maximal コピュラモデル，正規コピュラモデル 1，正規コピュラモデル 2 でそれぞれ解析して 4 つの信頼度関数を作る．そして，上記の信頼度関数 $R'(t)$ と，誤差 2 乗和をとって比較する．また，それぞれのモデルによる MTTF (平均故障時刻) も求め， $R'(t)$ の MTTF と比べて早い遅いを記録する．
- 手順 3 ばらつきを取り除くため，手順 2 を 10 回繰り返す，誤差 2 乗和の合計を計算する．また，MTTF の早い遅いの平均も記録する．

このような実験を行い，表 4 と表 5 のような結果が得られた．

表 4 5 素子ネットワークシステムにおける実験結果 1．

No	$\lambda_1(t)$	$\lambda_2(t)$	$\lambda_3(t)$	$\lambda_4(t)$	td SSE	mc SSE	nc1 SSE	nc2 SSE
1	1	t	$2t$	$3t$	150.0141	150.0141	1026.7973	148.6113
2	1	$3t$	$6t$	$9t$	196.4564	196.4560	947.5648	147.0158
3	2	t	$2t$	$3t$	68.3800	68.3800	835.7418	79.8826
4	2	$3t$	$6t$	$9t$	118.8058	118.8058	779.4786	92.1506
5	t	$2t$	$4t$	$6t$	128.8188	128.8188	1158.9987	172.0344
6	t	$3t$	$6t$	$9t$	141.5126	141.5125	1105.7171	127.0972
7	t	$4t$	$8t$	$12t$	177.2328	177.2328	912.6022	178.5158
8	$2t$	$3t$	$5t$	$7t$	134.8641	134.8641	1456.6278	180.7745
9	$2t$	$6t$	$8t$	$10t$	132.9294	132.9294	993.4217	94.7706
10	$2t$	$9t$	$11t$	$13t$	71.0065	71.0065	919.2989	89.6148
11	t	t^2	t^2	$2t^2$	74.4536	74.4536	1672.5345	48.1754
12	t	$2t^2$	$3t^2$	$4t^2$	193.6730	193.6727	1672.4960	199.5436
13	t	$4t^2$	$6t^2$	$8t^2$	141.9287	141.9139	1375.0735	141.0973
All					1730.0760	1730.0602	14856.3529	1699.2839

表中の，td,mc,nc1,nc2 はそれぞれ単純解析モデル，Maximal コピュラモデル，

正規コピュラモデル 1，正規コピュラモデル 2 を表す．

また SSE は誤差 2 乗和を表し，All にはそれぞれ総和が入る．

13 種類のパラメータにおいて実験を行ったが，2 素子並列システムの時と同様，パラメータの特徴による結果の違いはあまり見受けられなかった．似たようなパラメータによる結果も，大きく違うパラメータによる結果も，ばらばらであり傾向は読み取れなかった．

誤差 2 乗和を比較すると，7 種類のパラメータで正規コピュラモデル 2 が 1 番勝っており，全パラメータ合計では，単純解析モデルと比較して約 1.7798% ほど誤差 2 乗和を減らすことができています．これにより，正

表 5 5 素子ネットワークシステムにおける実験結果 2 .

No	MTTF	td MTTF	mc MTTF	nc1 MTTF	nc2 MTTF
1	0.7103	0.0011	0.0011	0.5794	-0.0004
2	0.5759	0.0185	0.0185	0.4193	0.0164
3	0.4171	-0.0076	-0.0076	0.2489	-0.0090
4	0.3662	0.0124	0.0124	0.2363	0.0112
5	1.0853	-0.0254	-0.0254	0.5871	-0.0233
6	1.0209	0.0040	0.0040	0.6454	0.0014
7	0.9749	-0.0044	-0.0044	0.5322	-0.0062
8	0.8196	-0.0165	-0.0165	0.4560	-0.0168
9	0.7478	0.0034	0.0034	0.3632	0.0017
10	0.7112	-0.0101	-0.0101	0.3126	-0.0108
11	1.1960	0.0135	0.0135	0.7740	0.0136
12	1.0620	-0.0057	-0.0057	0.6282	-0.0051
13	0.9651	-0.0044	-0.0044	0.5614	-0.0049
All		-0.0016	-0.0016	0.4880	-0.0025

表中の、MTTF は平均故障時刻を表し、All にはそれぞれ平均が入る .

規コピュラモデル 2 はある程度の成果があると言えるだろう . ただ他の 6 種類のパラメータにおいては逆に誤差 2 乗和が増えてしまっており , ばらつきが大きく確実な効果を期待することはできないと思われる .

次に , 単純解析モデルと Maximal コピュラモデルの比較では , No . 2 , No . 6 , No . 12 , No . 13 においてごく小さな差で Maximal コピュラが勝っており , 他のパラメータにおいては同じ値を取っている . 全パラメータ合計で見ると , 0.0009% ほど誤差 2 乗和が減っている . これもまた , 2 素子並列システムの時と同じく , 改善の幅は小さいものの , Maximal コピュラを用いることで結果が悪くなることは無いということが強みといえるだろう . また , 正規コピュラモデル 1 については全てのパラメータにおいて大幅に誤差 2 乗和が増加しており , モデルとしては使えそうにはない . 理由としては 4 本の「枝の故障時刻」の構築時に , 同じ値が数多く入ってしまっているからではないかと思われる . (例えば 1 番最初に素子 1 が故障すると , 枝 b_1, b_2 の故障時刻は同じになる . このようなことが多々ある .)

続けて MTTF の観点から , 各モデルを考える . 真の MTTF と各モデルの MTTF を比較する . 単純解析モデルの MTTF と Maximal コピュラモデルの MTTF はすべてのパラメータで等しく , 7 つのパラメータにおいて真の MTTF よりも早い値を取っている . 全パラメータ平均でも , 0.0016 ほど早くなっている . 正規コピュラモデル 2 の MTTF は , 8 つのパラメータで真の MTTF より早い値を取っており , 全パラメータ平均でも , 0.0025 ほど早くなっている . これは真の MTTF と比べ少し大きめのずれであるが , このずれは故障時刻を早めに見積もっていることを意味している . 先ほどのフェールセーフの考え方を考慮すると , それほど問題があるとは言えないのではないだろうか .

5 おわりに

5.1 まとめと結論

本研究では、2 素子並列システムと5 素子ネットワークシステムを題材に、従属故障する複数の素子からなるシステムの信頼性解析にコピュラを取り入れることで解析の精度が向上するか否かの検討を行った。どちらの場合も、正規コピュラを用いたモデルに一定の成果が得られた。ただし、5 素子の場合は正規コピュラモデル2 の方で成果が得られた反面、正規コピュラモデル1 は役に立たなかった。これは、例えば最初に素子1 が故障した場合に、2 本の枝 b_1, b_2 に同じ故障時刻が記録されるということが極端な結果につながったのではないだろうか。正規コピュラを用いた解析では、全体をいくつに区切るかということも考えていかななくてはならない。また、100% 成果があるわけではなく、特に5 素子の場合、逆に誤差が増えてしまう場合も多くあった。

それに対し、Maximal コピュラを用いたモデルでは、成果は非常に小さいものの誤差が増えるケースは見られなかった。ただ、MTTF を比較すると、正規コピュラモデルでは故障率を早めに見積もっており、フェールセーフの考えには良く適合すると言えるだろう。

システムの信頼性解析が求められる状況は様々である。もう少し詳しく正規コピュラモデルの特徴がつかめれば、それによる解析が良いと思われるが、高い確実性を求められる場合（病院の医療システム等）では、Maximal モデルを用いて解析したほうが良い場合も考えられる。よって、状況に応じて、正規コピュラモデルと Maximal コピュラモデルを使い分けていくことが有益であると考えられる。

5.2 今後の課題

これからの課題としては、正規コピュラモデルの特徴をもっとはっきりつかむことや、もっと大きなシステムにおいて成果があるか確認することが挙げられる。

また、Maximal コピュラや正規コピュラにこだわらず、他のコピュラを用いたモデルも検討すべきかもしれない。実際に、今回の研究の途中で、他の様々なコピュラも解析に用いることも考えた。その選定基準として、データから経験コピュラと呼ばれるコピュラを作り、その経験コピュラと最も適合するコピュラを用いるという方針だった。しかし計算量がとても多く時間がかかりすぎるために断念することを余儀なくされた。

もう一つのポイントとしては、実験を行う際にシミュレーションによる生成データではなく、現実起こった問題に関する実測データを用いることができれば、より実務への応用を考えやすくなると思われる。

参考文献

- [1] 影山孝夫, 太田修平, 木村光宏, 「コンピュータによる 2 素子並列システムの従属故障に関する一考察」, 電子情報通信学会技術研究報告 (信頼性), R2013-79, pp. 1-5 (2013).
- [2] 太田修平, 影山孝夫, 木村光宏, 「2 素子並列システムのカスケード故障に関するコンピュータ解析」, 電子情報通信学会 2014 年総合大会論文集, A-9-4 (2014).
- [3] 戸坂凡展, 吉羽要直, 「コンピュータの金融実務での具体的な活用方法の解説」, 日本銀行金融研究所, 金融研究, pp. 115-162 (2005).
- [4] 市田嵩, 鈴木和幸, 「信頼性の分布と統計」, 日科技連出版社, (2005).
- [5] 野中保雄, 島岡淳, 「冗長系 理論と実際」, 日科技連出版社, (1990).
- [6] R. T. Clemen and T. Reilly, “Correlations and copulas for decision and risk analysis”, *Management Science*, Vol. 45, Issue 2 (1999).
- [7] I. Dobson, B. A. Carreras, and D. E. Newman, “A loading-dependent model of probabilistic cascading failure”, *Probability in the Engineering and Informational Sciences*, Vol. 19, pp. 15–32 (2005).
- [8] 弓削哲史, 丸山恵, 柳繁, 「システムの共通原因故障率の推定について」, 信学技法, Vol. R2013-68, pp. 25–30 (2013).
- [9] 松岡猛, 「確率論的安全評価における従属故障と外的事象の取り扱い」, システム / 制御 / 情報, Vol. 36, No. 3, pp. 158–170 (1992).

謝辞

本研究，および本論文の執筆をするにあたり，温かいご指導ご鞭撻を頂きました木村光宏教授に心より感謝致します．また快く副査を引き受けて頂いた田村信幸准教授にも感謝致します．また，様々な助言や温かい言葉を頂いた信頼性工学研究室の緒先輩方，同輩ならびに後輩諸氏に感謝の意を表します．