

2次元格子システムに対する最適配置問題とその解法

中村 太信 ^{†1}

Component Assignment Problem of Two-dimensional Lattice Systems and Its Solution

by

Taishin Nakamura

Abstract

現代社会には我々の社会や暮らしを支えている多くのシステムが存在する。これらのシステムをいかに安定的に稼働させるかが重要であり、システムが期待通りに動作し続けられる性質である信頼性の確保は課題となっている。信頼性工学 (特に、信頼性数理) の分野では、現実システムの特徴を捉えたシステムモデルが数多く提案されており、そのシステムモデルに対して、システムの性能評価方法、最適設計や最適保全方策などについての数理的手法の開発が行われている。本稿では、システムを構成する要素の集中故障を評価する“2次元格子システム”を紹介し、筆者がこれまでにを行った研究を紹介する。特に、信頼性工学分野において、困難かつ重要な問題である最適配置問題の関連研究と効率的な最適配置探索アルゴリズムについて詳しく説明する。高性能な計算機やソフトウェアが利用できる環境が整ってきているとしても、現実には手に負えない問題は数多く存在するが、問題が持つ数理的な性質を利用することで、工夫を凝らした効率的なアルゴリズムを構築することが可能となり、様々な現実問題の解決につながる。

Keywords: 信頼性工学, 2次元格子システム, 最適配置問題, アルゴリズム

1 はじめに

現代社会には我々の社会や暮らしを支えている多くのシステムが存在する。これらのシステムは障害が発生することなく機能し続けることはなく、いずれ故障し、要求された機能を発揮できなくなる。特に、原子力発電所や航空機が故障した場合には、単なる経済的損失に留まらず、人命を奪い、我々の生活に広範囲かつ甚大な被害をもたらす可能性がある。従って、システムをいかに安定的に稼働させるかが重要であり、システムが期待通りに動作し続けられる性質である信頼性の確保は課題となっている。信頼性工学が目指すのは、工学が関与しているあらゆるものづくりに関して、故障や誤作動せずに、期待どおりの機能を発揮する視点から、画一的に信頼性を付与することである [1]。

現実のシステムを効率的・効果的に運用するためには、信頼度 (正常に稼働する確率) を適切な方法で評価し、システムの設計・運用段階に何らかの対策を講

じることが必要となる。信頼性工学 (特に、信頼性数理) の分野では、現実システムの特徴を捉えたシステムモデルが数多く提案されており、そのシステムモデルに対して、システムの性能評価方法、最適設計や最適保全方策などについての数理的手法の開発が行われている。本稿では、システムを構成する要素の集中故障を評価する“2次元格子システム”を紹介し、筆者がこれまでにを行った研究を紹介する。特に、信頼性工学分野において、困難かつ重要な問題である最適配置問題の関連研究と効率的な最適配置探索アルゴリズムについて詳しく説明する。

2 2次元格子システム

2次元格子システムとは、システムの構成要素であるコンポーネントが2次元格子状に配置され、それらのコンポーネントが一定範囲に集中して故障するとシステム全体に影響を与えるシステム・現象を表現するシステムモデルである。現実には、コンポーネントがまばらに故障しても、周囲のコンポーネントが故障し

^{†1} 情報理工学部 コンピュータ応用工学科

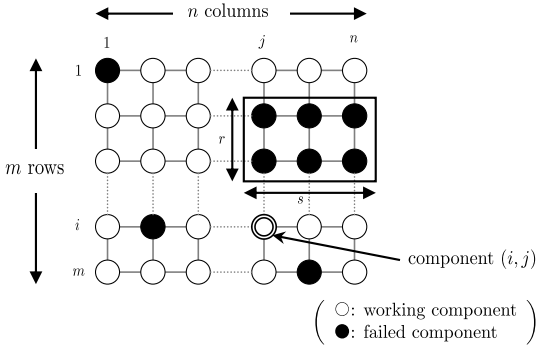


図1 長方形システム [3]

たコンポーネントの機能を補完するなどの理由により、システム全体としては正常に機能するが、コンポーネントが集中して故障するとシステムが正常に機能しなくなるという特徴を有するシステム（または、現象）が存在する。2次元格子システムは、このような「コンポーネントが一定範囲に集中して故障すると故障する」というシステムの特徴を捉えたシステムモデルである。2次元格子システムは、「システム形状」と「故障条件」により様々なバリエーションが存在する。本稿では「システム形状」の観点から分類した

- 長方形システム
- 円筒型システム
- トーラス型システム

をそれぞれ紹介し、筆者がこれまでにを行った研究を簡単に紹介する。

2.1 長方形システム

長方形 connected- $\mathbf{X}$ -out-of- $(m, n):F$ システム（以下、長方形システム）は、 m 行 n 列の格子状にコンポーネントが配置されているシステムで、 $\mathbf{X}$ はシステム故障の条件となるコンポーネントの故障形状を表す。例えば、 $\mathbf{X}$ が (r, s) の場合、図1に示すように、 r 行 s 列のコンポーネントがすべて故障する場合に故障となるシステムを表現できる [2]。また、 $\mathbf{X}$ が $(1, 2)$ -or- $(2, 1)$ の場合、「1 行 2 列のコンポーネントが故障する」もしくは「2 行 1 列のコンポーネントが故障する」（つまり、隣接した 2 つのコンポーネントが連続故障する）場合に故障となるシステムを表現できる [2]。

長方形システムは、センサが（近似的に）格子状に配置されたセンシングシステムを表現できる。センシングシステムとは、監視対象にセンサを配置することで、情報を収集し、その検出された情報に基づき、有用な情報を提供するシステムである。円形の監視範囲

を持つセンサを監視範囲が重複するように格子状に配置することで、あるセンサが故障しても、周囲のセンサが監視範囲を補うため、領域全体を監視することができる。一方、センサが集中して故障すると、監視不可範囲が生じ、センシングシステムは正常に機能しなくなる。長方形システムを用いることで、このようなセンシングシステムの信頼度評価・設計を実施することができる。その他にも、長方形システムは液晶等の表示システムや部屋の照明などに適用できる [4]。

信頼性工学の主要な問題の 1 つに信頼度算出問題がある。この問題は、コンポーネントの信頼度（正常に動作する確率）が与えられたときに、システム全体の信頼度を算出する問題である。システムの設計段階において、システム信頼度を算出し、運用に先立って問題点を知ることは非常に有用である。一般的に、 N 個のコンポーネントからなるシステム信頼度を求める場合には、コンポーネント故障が独立に生起するならば、すべてコンポーネントの状態を列挙し、それらの確率の総和によりシステム信頼度を求められる。しかし、その総数は 2^N であり、コンポーネント数が多い場合には膨大な計算時間を要する。そこで、Nakamura 他 [5] は、特別な場合の長方形システムの信頼度を求める際、再帰方程式 [6] を行列表現に書き換え、行列のべき乗計算を工夫したことで高速に信頼度が求められることを示した。

既存方法 [7, 8] により信頼度の厳密値を求めることは理論的に可能であっても、システムの規模が大きいため、計算時間や記憶容量等、計算機の制約で計算実行が困難な場合が多く、信頼度算出可能なシステムのサイズは限定的である。そこで、Nakamura 他 [9] は、長方形システムに対して、厳密方法と比べて短い時間で計算可能なシステム信頼度の上限値及び下限値を導出した。ここで、信頼度の上下限値とは、システムの信頼度がこの値以上でない、あるいはこの値以下でないことを保証する数値である。システム信頼度の厳密値は上限と下限の範囲内に収まっているため、上下限値が一致した場合にはその範囲においてシステム信頼度の厳密値とみなすことができる。

Nakamura 他 [10] は、Increasing Failure Rate (IFR) 保存問題に取り組んだ。この問題は、コンポーネントが IFR である（コンポーネントの故障率が時刻に対して単調増加する）ことがシステムにも保存されること、すなわち、システムも必ず IFR となることを明らかにする問題である。信頼性解析において、エージングは重要な概念であり、故障率関数の形状を把握することが求められる。コンポーネントが IFR であることがシステムにも保存されることが明らかにな

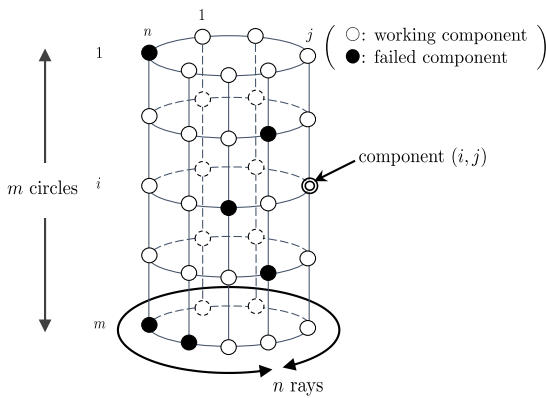


図2 円筒型システム [3]

れば、コンポーネントが任意の寿命分布を持つ場合における最適保全方策の提案 [11–13] などに寄与する。Nakamura 他 [10] では、システムの行数が 2, 3, 4 の場合 (列数は任意) において、コンポーネントの IFR が保存されるシステムが 3 種類存在することを解析的に示した。システムの行数と列数がともに 5 以上のシステムに対しては、数値実験を行い、実験の範囲内ではコンポーネントの IFR が保存されるシステムが一つも存在しないことを確認した。

筆者は、最適配置問題にも取り組んできた。最適配置問題とは、各コンポーネントの信頼度が同一でない場合に、コンポーネントをどのように配置すればシステム信頼度が最大になるかという問題である。ここで、コンポーネントを自由に入れ替えてもシステムの機能としては問題がないことを前提とする。最適配置を求めることは、信頼性を保証したシステムを経済的に設計する上で重要となる。長方形システムに対する最適配置問題に関しては 3 章で詳しく述べる。

2.2 円筒型システム

円筒型 connected- $\mathbf{X}$ -out-of- (m, n) :F システム (以下、円筒型システム) は、長方形システムの両端 2 辺を連結した構造を持ち、コンポーネントが m 円 n 放射の円筒状に配置されている (図 2)。

円筒型システムは、原子炉内の監視システムに適用できる。このシステムは円筒物の表面を覆うように温度センサが取り付けられており、領域内の温度分布を計測する。熱源の位置や形状を推定する際、熱源にある程度の大きさがあると考え、複数の隣接したセンサの情報により熱源の位置や形状を検知することができる。しかし、センサが集中して故障すると、熱源を検知することができなくなり、システム故障となる。

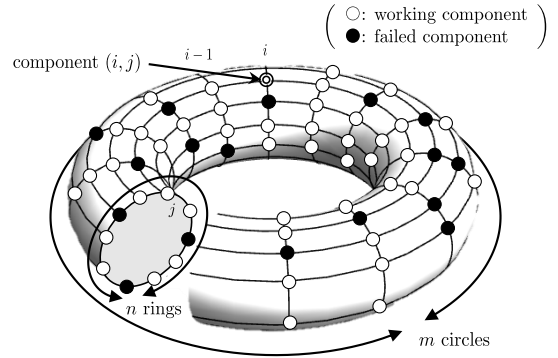


図3 トーラス型システム [3]

Nakamura 他 [14] では、Finite Markov Chain Imbedding Approach (FMCIA) を用いて、円筒型システムに対する信頼度算出問題に取り組んだ。FMCIA とは、「システム故障が発生しないように、システムサイズを大きくする過程」を「マルコフ連鎖における状態遷移」に帰着させることで、マルコフ連鎖の考え方を利用したシステム信頼度算出方法である [15, 16]。具体的には、円筒型システムの構造を基にマルコフ連鎖を構築し、推移確率行列を求めた後、チャップマン=コルモゴロフ方程式を用いることで、システム信頼度を求めた。更に、効率的な信頼度算出を可能にするために、円筒型システムの特徴に応じた 2 種類の方法を提案した。数値実験により、特に、コンポーネント信頼度が同一である場合、提案方法は効率的にシステム信頼度を算出できることがわかった。これは、コンポーネント信頼度が同一である場合、すべての推移確率行列が等しくなるためである。

2.3 トーラス型システム

Nakamura 他 [17] は従来の長方形システムや円筒型システムを拡張したコンポーネントがトーラス状に配置されたトーラス型 connected- $\mathbf{X}$ -out-of- (m, n) :F システム (以下、トーラス型システム) を提案した。このシステムは、長方形システムの上下と左右の 2 辺をそれぞれ連結した構造を持ち、コンポーネントが m 円 n 環のトーラス状に配置されている (図 2)。トーラス型の実システムには、荷電粒子加速器や、「京」「富岳」に採用されたアーキテクチャ [18] などの例がある。

Nakamura 他 [19] では、トーラス型システムに対して FMCIA を用いた信頼度算出方法を提案した。更に、Nakamura 他 [20] では、信頼度算出のための再帰方程式を導出し、それを利用したアルゴリズムを提案した。具体的には、トーラス型システムの信頼度を一部のコンポーネントの状態を与えた小さなシステムの和で表

すことで、再帰的にシステム信頼度を算出した。

2次元格子システムに関する研究は、Kuo and Zuo [21], Yamamoto and Akiba [22], Akiba 他 [4], Cui 他 [23] などにまとめられている。

3 Lin/(r, s)/(m, n):F システムに対する最適配置問題

本章では、長方形型 connected-(r, s)-out-of-(m, n):F システム (以下, Lin/(r, s)/(m, n):F システム) に対する最適配置問題に関する関連研究を紹介し、筆者が行った研究での成果を示す。ここで、Lin/(r, s)/(m, n):F システムとは、コンポーネントが m 行 n 列の格子状に配置されており、システム内の任意の位置において r 行 s 列内のコンポーネントがすべて故障するときにシステム故障となるシステムである (図 1)。また、最適配置問題とは、システム信頼度を目的関数として、それを最大化するコンポーネント配置を探索する問題である。対象とするシステムのコンポーネントを適切に並び替えることで、システム信頼度向上を実現し、安定的な運用が可能となる。3 章と 4 章では

- (a) コンポーネントとシステムは稼働か故障の 2 状態を取り、
- (b) コンポーネントは独立に故障し、
- (c) コンポーネント信頼度は与えられており、
- (d) コンポーネントは任意の位置に割り当てられることを仮定する。

3.1 最適配置問題の定式化

本節では、Lin/(r, s)/(m, n):F システムに対する最適配置問題の定式化を行う。 $i = 1, 2, \dots, m$ と $j = 1, 2, \dots, n$ に対して、 (i, j) をシステム内の i 行 j 列の位置とする。 $\pi(i, j) \in \{1, 2, \dots, mn\}$ を位置 (i, j) に割り当てられたコンポーネント番号とすると、 $m \times n$ 個のコンポーネントの配置は

$$\Pi = (\pi(i, j))_{1 \leq i \leq m, 1 \leq j \leq n} \quad (1)$$

によって表現される。コンポーネント番号を $\tau \in \{1, 2, \dots, mn\}$ とすると、コンポーネント τ の信頼度は p_τ と表され、一般性を失うことなく、 $p_1 < p_2 < \dots < p_{mn}$ とする。つまり、 p_τ は τ 番目に信頼度の低いコンポーネントの信頼度を意味している。また、ベクトル $\mathbf{p} = (p_1, p_2, \dots, p_{mn})$ を定義する。更に、 $\mathbf{p}$ が与えられており、配置が Π である Lin/(r, s)/(m, n):F システムの信頼度を $R((r, s), (m, n), \mathbf{p}; \Pi)$ とする。これらの記号を用いて、Lin/(r, s)/(m, n):F システムの最適配置問題は以下のように定式化される。

$$\Pi^* = \arg \max_{\Pi \in \Omega} R((r, s), (m, n), \mathbf{p}; \Pi) \quad (2)$$

ただし、 Ω はすべての配置 Π の集合とする。

3.2 最適配置問題に関する関連研究

本節では、Lin/(r, s)/(m, n):F システムの最適配置問題に関する関連研究を紹介する。Lin/(r, s)/(m, n):F システムの最適配置問題に関する研究は (A) 最適配置が持つ性質の調査と (B) 最適配置探索アルゴリズムの開発に大別される。

(A) 最適配置が持つ性質の調査 Zuo [24] は、 $r = m$ もしくは $s = n$ の場合における Lin/(r, s)/(m, n):F システムに対して最適配置が満足すべき必要条件を導出した。その後、Koutras 他 [25] は、Lin/(r, s)/(m, n):F システムの最適配置が満足すべき必要条件を導出した。

定理 1 (Koutras 他 [25]). Lin/(r, s)/(m, n):F システムにおいて、 Π が最適配置である必要条件是

- (a) $2 \leq j \leq \min\{s, n - s + 1\}$ ならば、 $i = 1, 2, \dots, m$ に対して、

$$\pi(i, j - 1) < \pi(i, j)$$

- (b) $\max\{n - s + 2, s + 1\} \leq j \leq n$ ならば、 $i = 1, 2, \dots, m$ に対して、

$$\pi(i, j - 1) > \pi(i, j)$$

- (c) $2 \leq i \leq \min\{r, m - r + 1\}$ ならば、 $j = 1, 2, \dots, n$ に対して、

$$\pi(i - 1, j) < \pi(i, j)$$

- (d) $\max\{m - r + 2, r + 1\} \leq i \leq m$ ならば、 $j = 1, 2, \dots, n$ に対して、

$$\pi(i - 1, j) > \pi(i, j)$$

である。

ここで、 $\pi(i, j - 1) < \pi(i, j)$ は $p_{\pi(i, j - 1)} < p_{\pi(i, j)}$ 、つまり、位置 $(i, j - 1)$ よりも位置 (i, j) に信頼度の高いコンポーネントを割り当てることを意味している。必要条件を満たさない配置は、より目的関数値 (信頼度) が大きい配置が存在するため、最適配置になりえないことがわかる。最適配置を探索する際、必要条件を用いることで劣解を排除して解探索空間を削減することができる。そのため、必要条件是効率的に最適配置を探索する上で有用である。

信頼度を目的関数とする最適配置問題には「不変性」という概念がある。不変性とは最適配置がコンポーネント信頼度の値に依存せず、各コンポーネント信頼度の大小関係のみに依存する性質である。ほとんどのシ

システムにおいては、最適配置はコンポーネント信頼度の値に依存する（不変な最適配置が認められない）ことが確認されているが、中には、不変な最適配置を認めるシステムが存在する。その場合、コンポーネント信頼度の大小関係がわかれば、計算機を用いることなく、最適配置を決定することができる。Hwang and Dinghua [26] は、以下の定理により、 $r = m$ ($s = n$) の場合における $\text{Lin}/(r, s)/(m, n):F$ システムの不変性の存在条件を明らかにしている。

定理 2 (Hwang and Dinghua [26]). (a) $r = 2$ かつ $s = n - 1$, (b) $r = 2$ かつ $s = 1$, (c) $s = n$ のいずれかの場合においては、 $\text{Lin}/(r, s)/(r, n):F$ システムは不変な最適配置を認める。

例えば、 $\text{Lin}/(2, 1)/(2, 5):F$ システム ((b) の場合) の最適配置は

$$\Pi^* = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 10 & 9 & 8 & 7 & 6 \end{pmatrix} \quad (3)$$

である。ここで、行列内の値はコンポーネント番号であり、 $p_1 < p_2 < \dots < p_{10}$ である。

(B) 最適配置探索アルゴリズムの開発 計算機の性能向上に伴い、問題例によっては多少の時間はかかるとしても、最適性が保証された解を得ることが可能となった。しかし、最適配置問題は NP 困難な組合せ最適化問題であるため、実用的な規模の問題例に対しては分枝限定法などの汎用的手法を単に適用するだけでは最適配置を得ることは難しい。そこで、与えられる入力ケースの性質を利用した効率的な最適配置探索アルゴリズムは開発されてきた。Omura 他 [27] は分枝限定法をベースとして、 $r = m - 1$ かつ $2s > n$ の場合におけるアルゴリズムを開発した。

また、対象とするシステムを取り扱いが容易なシステムに帰着させることで、最適配置探索の効率化を行った研究もおこなわれている。このことは、単純化されたシステムの最適配置を求められれば、元の $\text{Lin}/(r, s)/(m, n):F$ システムの最適配置も求まることを意味する。Omura 他 [28] は、 $r = m$ かつ $s = 2$ の場合において、 $\text{Lin}/(r, s)/(m, n):F$ システムを単純なシステムに帰着させ、そのシステムが持つ不変性を使用した最適配置探索アルゴリズムを開発した。Nakamura 他 [29] は、 $r = m$ の場合における $\text{Lin}/(r, s)/(m, n):F$ システムを単純なシステムに帰着させ、効率的なアルゴリズムを提案した。Nakamura 他 [30] は、 $r = m - 1$ かつ $s = n - 1$ の場合に特化したアルゴリズムを提案

した。基本的な考え方としては、 $\text{Lin}/(r, s)/(m, n):F$ システムが $r = m - 1$ かつ $s = n - 1$ の条件を満たすとき、定理 1 [25] を用いることで「 mn 個のコンポーネントで構成された $\text{Lin}/(r, s)/(m, n):F$ システム」は「 $2(m + n) - 5$ 個のコンポーネントが円状に配置されたシステム」に帰着可能であることを理論的に示した。Nakamura 他 [30] は、この単純化されたシステムに対する最適配置探索アルゴリズムを提案した。

このように最適配置を効率よく探索するアルゴリズムの開発は行われてきた。しかし、Nakamura 他 [29] のアルゴリズムは $r = m$ の場合、Nakamura 他 [30] のアルゴリズムは $r = m - 1$ かつ $s = n - 1$ の場合のように既存のアルゴリズムは故障条件が限定されており、一般的な $\text{Lin}/(r, s)/(m, n):F$ システムには適用することができなかった。そこで、Nakamura and Yamamoto [31] により、一般的な $\text{Lin}/(r, s)/(m, n):F$ システムに対して、効率的に最適配置を探索するアルゴリズムの開発が行われた。

4 $\text{Lin}/(r, s)/(m, n):F$ システムに対する最適配置探索アルゴリズム

本章では、Nakamura and Yamamoto [31] により提案された $\text{Lin}/(r, s)/(m, n):F$ システムに対する最適配置探索アルゴリズムを紹介する。

4.1 基本となるアイデア

まず、アルゴリズムの基本となるアイデアを述べる。探索木を用いて、すべての配置（解候補）を列挙し、その中から最適配置を見つける場合、多くの計算量を要し、実用的な時間で最適配置を求めることは困難である。特に、 $\text{Lin}/(r, s)/(m, n):F$ システムに対する最適配置問題の場合、目的関数の計算に大きな計算コストが必要となる。これを回避するためには、最適配置となる見込みのない配置を可能な限り列挙しないことが重要である。分枝限定法では、基本的には全列挙と同様に探索木を用いた探索を行うが、その過程で最適配置になりえないことが何らかの理由によって示される場合、それ以上の探索を行わない。これを限定操作、もしくは、枝刈りという。Nakamura and Yamamoto [31] により提案されたアルゴリズムでは、分枝限定法をベースに限定操作により、探索する配置数を抑えることで効率的に最適配置を探索する。具体的には、以下の 3 種類の枝刈り条件により限定操作を行う。

- 必要条件に基づく枝刈り条件
- 対称配置を排除する枝刈り条件
- システム信頼度に基づく枝刈り条件

更に、前処理を行うことで、割り当てるコンポーネン

ト数を減じ、予め問題の規模を小さくする。

4.2 前処理

本節では、分枝限定法を実行する前に実施する前処理を説明する。定理 1 [25] により、最適性を失うことなく、以下の前処理によって割り当てるコンポーネント数を減らすことができる。

系 1. $2r > m$ かつ $2s > n$ ならば、最も信頼度の低いコンポーネントは位置 $(1, 1)$ に、信頼度の高い $(2r - m)(2s - n)$ 個のコンポーネントは以下の集合 C の要素である位置に割り当てる。

$$C = \{(i, j) \mid m - r + 1 \leq i \leq r, n - s + 1 \leq j \leq s\}.$$

$\text{Lin}/(r, s)/(m, n)$:F システムが $2r > m$ かつ $2s > n$ を満たすならば、この前処理を行うことで、割り当てるコンポーネント数を mn 個から $(mn - (2r - m)(2s - n) - 1)$ 個まで減じることができる。

4.3 必要条件に基づく枝刈り条件

本節では、最適配置が満足すべき必要条件 [25] に基づく枝刈り条件を示す。

まず、位置 $(1, 1)$ に割り当てられるコンポーネントを限定するための条件を与える。

枝刈り条件 1. 位置 $(1, 1)$ にコンポーネントが割り当てられたとき、

$$(i) \quad 2r \geq m \text{ かつ } 2s \geq n \text{ ならば, } \pi(1, 1) = 1$$

$$(ii) \quad 2r \geq m \text{ かつ } 2s < n \text{ ならば,}$$

$$\pi(1, 1) \leq mn - 2sm + 1$$

$$(iii) \quad 2r < m \text{ かつ } 2s \geq n \text{ ならば,}$$

$$\pi(1, 1) \leq mn - 2rn + 1$$

$$(iv) \quad 2r < m \text{ かつ } 2s < n \text{ ならば,}$$

$$\pi(1, 1) \leq mn - 4rs + 1$$

を満たさなければ、枝刈りを行う。

例えば、 $2r \geq m$ かつ $2s < n$ ならば、 τ 番目に信頼度の低いコンポーネント $(1 \leq \tau \leq mn - 2sm + 1)$ を位置 $(1, 1)$ に割り当てる。

次に、最適配置において位置 (i, j) に割り当てられるコンポーネントは以下の条件を満たす。

枝刈り条件 2. 位置 (i, j) にコンポーネントが割り当てられたとき、

$$(i) \quad i = 1, 2, \dots, m \text{ に対して, } 2 \leq j \leq s \text{ ならば,}$$

$$\pi(i, j - 1) < \pi(i, j)$$

$$(ii) \quad i = 1, 2, \dots, m \text{ に対して, } \max\{n - s + 2, s + 1\} \leq j \leq n \text{ ならば,}$$

$$\pi(i, j - 1) > \pi(i, j)$$

$$(iii) \quad j = 1, 2, \dots, n \text{ に対して, } 2 \leq i \leq r \text{ ならば,}$$

$$\pi(i - 1, j) < \pi(i, j)$$

$$(iv) \quad j = 1, 2, \dots, n \text{ に対して, } \max\{m - r + 2, r + 1\} \leq i \leq m \text{ ならば,}$$

$$\pi(i - 1, j) > \pi(i, j)$$

を満たさなければ、枝刈りを行う。

4.4 対称配置を排除する枝刈り条件

$\text{Lin}/(r, s)/(m, n)$:F システムにおいて、コンポーネント信頼度が与えられたとき、システム信頼度はコンポーネントの相対的な配置によって決まる。すなわち、回転や反転の操作により、同じシステム信頼度を持つ本質的に等価な複数の配置が現れる。従って、最適配置を求める場合には、本質的に等価な配置のうち 1 つの配置を列挙すれば、他の配置を列挙する必要はない。そのため、提案するアルゴリズムでは、以下の条件を満足する配置のみを考える。

枝刈り条件 3.

$$(i) \quad m \neq n \text{ もしくは } r \neq s \text{ ならば,}$$

$$\pi(1, 1) = \min \left\{ \begin{array}{l} \pi(1, 1), \pi(m, 1), \\ \pi(1, n), \pi(m, n) \end{array} \right\} \quad (4)$$

$$(ii) \quad m = n \text{ かつ } r = s \text{ ならば, 式 (4) と}$$

$$\pi(m, 1) < \pi(1, n) \quad (5)$$

を満たさなければ、枝刈りを行う。

ここで、式 (4) は、位置 $(1, 1), (m, 1), (1, n), (m, n)$ に割り当てられるコンポーネントの中で最も信頼度の低いコンポーネントを位置 $(1, 1)$ に割り当てることを意味する。

4.5 システム信頼度の計算

最適配置探索アルゴリズムでは、位置 (m, j) ($j = 1, 2, \dots, n$) にコンポーネントが割り当てられた時点において、Yamamoto and Miyakawa [6] が導出した再

帰方程式を用いて部分的にシステム信頼度を計算する。その再帰方程式は紙面の都合上、割愛する。

最適配置問題を解く際、効率的な信頼度算出方法を用いても、多くの信頼度計算が求められるため、多くの計算時間を要する。従って、わずかな時間であってもシステム信頼度の計算時間を短縮することは非常に重要となる。そこで、分枝操作では一部のコンポーネントのみが異なる配置が逐次的に列挙されることに着目し、異なる部分に対して差分計算することで信頼度計算の高速を実現する。

例として、 $\text{Lin}/(2, 2)/(3, 5):F$ システムにおいて、以下の配置 Π_A, Π_B を持つ 2 つのシステムを考える。

$$\Pi_A = \left(\begin{array}{ccc|c|c} 1 & 8 & 5 & 10 & \cdot \\ 12 & 15 & 13 & 14 & \cdot \\ 2 & 7 & 6 & 9 & \cdot \end{array} \right) \quad (6)$$

$$\Pi_B = \left(\begin{array}{ccc|c|c} 1 & 8 & 5 & 9 & \cdot \\ 12 & 15 & 13 & 11 & \cdot \\ 2 & 7 & 6 & 4 & \cdot \end{array} \right) \quad (7)$$

ここで、行列内の “ $\cdot$ ” はコンポーネントが割り当てられていないことを意味する。式 (6) と式 (7) に示すように、 $R((2, 2), (3, 5), \mathbf{p}; \Pi_A)$ と $R((2, 2), (3, 5), \mathbf{p}; \Pi_B)$ では 1 列目から 3 列目の配置は共通である。よって、一方の信頼度を求める際に共通となる値を記憶しておくことで、もう一方の信頼度を少ない計算量で求めることができる。このように、多くの配置に対応するシステム信頼度を計算する際には、特定の値を記憶し、利用することで冗長な計算を回避し、効率的な信頼度計算が可能となる。

また、システム信頼度を用いた限定操作を実施する。システム信頼度は列 n に対して非増加関数であるため、部分的にコンポーネントが配置されたシステムの信頼度がこれまでの探索で得られた最良の配置に対応するシステム信頼度を下回った場合は、さらにコンポーネントを割り当てても信頼度は増加しない。つまり、それ以上コンポーネントを割り当てて必要はない。そこで、提案アルゴリズムでは、システム信頼度に基づく枝刈りを実施する。

枝刈り条件 4. $j = s, s+1, \dots, n$ に対して、位置 (m, j) にコンポーネントが割り当てられたとき、

$$\mathbf{Rmax} \leq R((r, s), (m, j), \mathbf{p}; \Pi) \quad (8)$$

を満たさなければ、枝刈りを行う。ここで、 $\mathbf{Rmax}$ はこれまでの探索で得られた最良の配置に対応するシステム信頼度である。

4.6 最適配置探索アルゴリズムの手順

次に、分枝限定法を用いた提案アルゴリズムの手順を簡単に説明する。具体的な手順は、Nakamura and Yamamoto [31] を参照されたい。

まず、 $2r > m$ かつ $2s > n$ を満たすのであれば、系 1 の前処理を行う。その後、コンポーネントを $(2, 1), (3, 1), \dots, (m, 1), (1, 2), \dots, (m, 2), (1, 3), \dots, (m, n)$ の順に割り当てる。このとき、前処理により既にコンポーネントが割り当てられているならば、その位置への割り当てはスキップする。分枝限定法により割り当てられるコンポーネント数を α とすると、

$$\alpha = \begin{cases} mn - (2r - m)(2s - n) - 1 & 2r > m \text{ かつ } 2s > n \text{ の場合} \\ mn & \text{その他} \end{cases} \quad (9)$$

となる。上記の順にコンポーネントを割り当てるとき、 $k = 1, 2, \dots, \alpha$ に対して、 $\text{DFS}(k)$ を k 番目の位置にコンポーネント割り当てのための処理ルーチンとする。処理ルーチン $\text{DFS}(k)$ の流れは以下のとおりである。

STEP 1: 枝刈り条件 1 を用いて限定操作を行う。

STEP 2: 枝刈り条件 2 を用いて限定操作を行う。

STEP 3: 枝刈り条件 3 を用いて限定操作を行う。

STEP 4: m 行目にコンポーネントが割り当てられたとき、再帰方程式 [6] を用いてシステム信頼度を算出する。

STEP 5: 枝刈り条件 4 を用いて限定操作を行う。

STEP 6: mn 個すべてのコンポーネントが割り当てられた配置が $\mathbf{Rmax}$ よりも大きければ、その配置を保存し、 $\mathbf{Rmax}$ を更新する。

上記の操作をすべての配置を列挙するまで繰り返し、すべての配置を列挙し終えた後、システム信頼度が $\mathbf{Rmax}$ である配置が最適配置となる。

4.7 数値実験の結果

本節では、Nakamura and Yamamoto [31] のアルゴリズムと Omura 他 [27] のアルゴリズムを計算時間の観点から比較する。本来、アルゴリズム [27] は、 $r = m - 1$ かつ $2s > n$ の場合にのみ探索可能であるが、比較のため、一般化を行った。表 1 にアルゴリズム [31] のアルゴリズム [27] の計算時間の比較結果を示す。表中の “N/A” は最適配置の探索に 24 時間以上要することを意味する。表 1 に示すように、アルゴリズム [31] は、アルゴリズム [27] に比べて、短い時間で最適配置が求められる。以上のことから、枝刈り条件による限定操作が有効に機能していることがわかる。

表 1 計算時間比較 (秒) [3]

(r, s)	(2, 2)	(3, 2)	(3, 3)	(3, 3)	(3, 4)	(3, 3)	(4, 4)	(4, 5)	(5, 4)	(5, 5)
(m, n)	(4, 4)	(4, 4)	(4, 4)	(4, 5)	(4, 5)	(5, 4)	(5, 5)	(5, 6)	(6, 5)	(6, 6)
(A) アルゴリズム [31]	408.81	52.84	0.08	3686.04	2.30	4548.24	40.84	1144.77	1532.19	19645.68
(B) アルゴリズム [27]	462.77	72.45	7.23	19529.64	6066.08	41857.86	N/A	N/A	N/A	N/A
(A)/(B)	88.34%	72.93%	1.12%	18.87%	0.04%	10.87%	—	—	—	—
システム信頼度	0.328	0.782	0.960	0.943	0.989	0.943	0.998	0.992	0.992	0.997

4.8 ま と め

本章では、 $\text{Lin}/(r, s)/(m, n):F$ システムに対して、分枝限定法をベースとして、必要条件に基づく枝刈り条件、対称配置を排除する枝刈り条件、システム信頼度に基づく枝刈り条件を組み込んだ最適配置探索アルゴリズムを紹介した。更に、前処理によって最適性を失うことなく割り当てるコンポーネント数を減らした。加えて、分枝操作では一部のコンポーネントのみが異なる配置が逐次的に列挙されることに着目し、異なる部分に対して差分計算することで信頼度計算の高速を実現した。Nakamura and Yamamoto [31] で実施された計算機実験では、このアルゴリズムは、既存アルゴリズム [27] と比較して、短時間で最適配置が得られることを確認した。

Nakamura and Yamamoto [31] で得られた成果の 1 つは、最も信頼度が高い (故障に強い) コンポーネント配置が得られることである。先行研究 [27–30] では、適用可能なシステムに制限があったが、提案アルゴリズムでは、(理論的には) 一般的な $\text{Lin}/(r, s)/(m, n):F$ システムの最適配置を得ることができる。最適配置が得られることで、信頼性を保証したシステムを経済的に設計することが可能となり、現実のシステムを設計・運用するための指針が従来よりも的確に提案可能になると考えられる。センシングシステムに限らず、「構成要素の集中異常がシステム全体に影響を与える」という現象は様々なところで見られるため、 $\text{Lin}/(r, s)/(m, n):F$ システムとして表現可能な多様な現実システムの信頼度向上が期待できる。

しかし、現実に現れる問題例は規模が大きいことが多く、その場合、提案アルゴリズムを用いても最適配置を得ることが極めて困難となる。実務上、常に最適解が求められているわけではないため、良質な準最適解を出来るだけ効率良く求めることを目指す発見的解法の構築が求められる。その際、遺伝的アルゴリズムやアントコロニー最適化などの汎用的なメタ戦略を単に適用するのではなく、Nakamura and Yamamoto [31] で導出した枝刈り条件を組み込むことで、解探索空間を狭め、探索効率の向上が期待できる。更に、提案ア

ルゴリズムで得られる解は最適性が保証されているため、発見的解法の評価に理論的裏打ちを与えることができる。このように、発見的解法の構築及び評価においても Nakamura and Yamamoto [31] は貢献可能であると考えている。

5 お わ り に

本稿では、長方形型・円筒型・トラス型システムを紹介し、それらに対してこれまでに行ってきた研究を簡単に紹介した。更に、これまでの研究成果の 1 つとして、信頼性工学分野において、困難かつ重要な問題である最適配置問題に対する効率的な最適配置探索アルゴリズムを説明した。高性能な計算機やソフトウェアが利用できる環境が整ってきているとしても、現実には手に負えない問題は数多く存在する。しかし、問題が持つ数理的な性質を利用し、工夫を凝らした効率的なアルゴリズムを構築することで、様々な現実問題の解決に貢献できると考えている。

近年、第 4 次産業革命ともいわれる、IoT、ビッグデータ、AI をはじめとするデータ活用の技術が急速に進展してきており、これらの技術により、収集されたデータをもとに瞬時に判断することが可能になりつつある。信頼性工学の分野においても、IoT によるデータ収集が行われており、これまで解析が困難であった部材や部品の製造状況や製品の使用状況のデータを容易に収集することが可能となっている [32]。このような変化に伴って新たな課題が生じており、その課題に対する解決策の提供が求められる。AI やデータを軸とする第 4 次産業革命に対応すべく、今後も信頼性工学における諸問題に取り組んでいく。

参 考 文 献

- [1] 福井 泰好,『入門 信頼性工学 (第 2 版) 確率・統計の信頼性への適用』, 森北出版株式会社, 2016.
- [2] T. K. Boehme, A. Kossow, and W. Preuss, “A generalization of consecutive- k -out-of- $n:F$ systems,” *IEEE Transactions on Reliability*, vol. 41, no. 3, pp. 451–457, 1992.
- [3] T. Nakamura, “Study on reliability evaluation and

- optimal design of connected- $\mathbf{X}$ -out-of- (m, n) :F lattice systems,” 首都大学東京大学院システムデザイン研究科博士論文, 2020.
- [4] T. Akiba, T. Nakamura, X. Xiao, and H. Yamamoto, “Evaluation methods for reliability of consecutive- k systems,” in *Systems Engineering: Reliability Analysis Using k-out-of-n Structures*, M. Ram and T. Dohi, Eds. CRC Press, 2019, ch. 1, pp. 1–24.
 - [5] T. Nakamura, H. Yamamoto, and X. Xiao, “Fast calculation methods for reliability of connected- (r, s) -out-of- (m, n) :F lattice system in special cases,” *International Journal of Mathematical, Engineering and Management Sciences*, vol. 3, no. 2, pp. 113–122, 2018.
 - [6] H. Yamamoto and M. Miyakawa, “Reliability of a linear connected- (r, s) -out-of- (m, n) :F lattice system,” *IEEE Transactions on Reliability*, vol. 44, no. 2, pp. 333–336, 1995.
 - [7] Y. Higashiyama, “An algorithm for exact reliability of consecutive 2-out-of- (m, n) :F system with un-equal component probability,” in *Proceedings of the IEEE Region 10 Conference*, vol. 2, 1999, pp. 990–993.
 - [8] H. Yamamoto, T. Akiba, H. Nagatsuka, and Y. Moriyama, “Recursive algorithm for the reliability of a connected- $(1, 2)$ -or- $(2, 1)$ -out-of- (m, n) :F lattice system,” *European Journal of Operational Research*, vol. 188, no. 3, pp. 854–864, 2008.
 - [9] T. Nakamura, H. Yamamoto, T. Shinzato, and T. Akiba, “Upper and lower bounds for reliability of connected- $(1, 2)$ -or- $(2, 1)$ -out-of- (m, n) :F lattice system,” in *Proceedings of the 19th Asia Pacific Industrial Engineering and Management Systems*, 6 pages, 2018.
 - [10] T. Nakamura, H. Yamamoto, X. Xiao, and T. Akiba, “The increasing failure rate property of connected- $(1, 2)$ -or- $(2, 1)$ -out-of- (m, n) :F lattice systems — the case of $m = 2, 3, 4$ —,” in *Proceedings of the 8th Asia-Pacific International Symposium on Advanced Reliability and Maintenance Modeling*, pp. 45–51, 2018.
 - [11] W. Y. Yun and A. J. Endharta, “Preventive maintenance of consecutive multi-unit systems,” in *Advances in Reliability and System Engineering*, M. Ram and J. P. Davim, Eds. Springer, 2017, pp. 27–51.
 - [12] L. Zhou, H. Yamamoto, T. Nakamura, and X. Xiao, “Optimization problems for consecutive-2-out-of- n :G system,” *Communications in Statistics - Theory and Methods*, vol. 49, no. 15, pp. 3792–3807, 2020.
 - [13] L. Zhou, H. Yamamoto, T. Nakamura, and X. Xiao, “Optimization problems for consecutive- k -out-of- n :G systems,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E103-A, no. 5, pp. 741–748, 2020.
 - [14] T. Nakamura, H. Yamamoto, T. Shinzato, X. Xiao, and T. Akiba, “Reliability of a circular connected- $(1, 2)$ -or- $(2, 1)$ -out-of- (m, n) :F lattice system with identical components,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E100-A, no. 4, pp. 1029–1036, 2017.
 - [15] L. Cui, Y. Xu, and X. Zhao, “Developments and applications of the finite Markov chain imbedding approach in reliability,” *IEEE Transactions on Reliability*, vol. 59, no. 4, pp. 685–690, 2010.
 - [16] L. Cui, C. Lin, and H. Yang, “Analysis for a qualification test procedure with FMCIA (finite Markov chain imbedding approach),” in *Mathematics Applied to Engineering*, M. Ram and J. P. Davim, Eds. Academic Press, 2017, ch. 1, pp. 1–20.
 - [17] T. Nakamura, H. Yamamoto, X. Xiao, and T. Akiba, “Reliability of a toroidal connected- $(1, 2)$ -or- $(2, 1)$ -out-of- (m, n) :F lattice system,” in *Proceedings of the 7th Asia-Pacific International Symposium on Advanced Reliability and Maintenance Modeling*, pp. 399–406, 2016.
 - [18] 安島雄一郎, 井上智宏, 平本新哉, 清水俊幸, “スーパーコンピュータ「京」のインターコネクト Tofu,” *Fujitsu*, vol. 63, no. 3, pp. 260–264, 2012.
 - [19] T. Nakamura, H. Yamamoto, T. Shinzato, X. Xiao, and T. Akiba, “Proposal of calculation method for reliability of toroidal connected- $(1, 2)$ -or- $(2, 1)$ -out-of- (m, n) :F lattice system with Markov chain,” in *Reliability Modeling With Computer and Maintenance Applications*, S. Nakamura, C. H. Qian, and T. Nakagawa, Eds. World Scientific, 2017, ch. 7, pp. 139–153.
 - [20] T. Nakamura, H. Yamamoto, and T. Akiba, “Reliability of a toroidal connected- (r, s) -out-of- (m, n) :F lattice system,” *Journal of Risk and Reliability*, 10 pages, 2020 (Early Access).
 - [21] W. Kuo and M. J. Zuo, *Optimal Reliability Modeling: Principles and Applications*. John Wiley and Sons, 2003.
 - [22] H. Yamamoto and T. Akiba, “Survey of reliability studies of multi-dimensional consecutive- k -out-of- n :F systems,” *Reliability Engineering Associ-*

- ation of Japan, vol. 25, no. 8, pp. 783–796, 2003.
- [23] L. Cui and D. Qinglai, “Consecutive k and Related Models—A Survey,” in *International Conference of Celebrating Professor Jinhua Cao’s 80th Birthday*, Q. L. Li, J. Wang, and H.B. Yu, Eds. Springer, 2019, ch. 1, pp. 3–18.
 - [24] M. Zuo, “Reliability & design of 2-dimensional consecutive- k -out-of- n systems,” *IEEE Transactions on Reliability*, vol. 42, no. 3, pp. 488–490, 1993.
 - [25] M. V. Koutras, G. K. Papadopoulos, and S. G. Papastavridis, “Note: Pairwise rearrangements in reliability structures,” *Naval Research Logistics*, vol. 41, no. 5, pp. 683–687, 1994.
 - [26] F. K. Hwang and S. Dinghua, “Redundant consecutive- k systems,” *Operations Research Letters*, vol. 6, no. 6, pp. 293–296, 1987.
 - [27] T. Omura, H. Yamamoto, X. Xiao, and T. Akiba, “Algorithm for obtaining optimal arrangement of a connected- (r, s) -out-of- (m, n) :F system — the case of $m = r + 1$ and $2s > n$ —,” in *Proceedings of the 16th Asia Pacific Industrial Engineering and Management Systems Conference*, 2015, pp. 652–659.
 - [28] T. Omura, T. Akiba, H. Yamamoto, and X. Xiao, “Algorithm for obtaining optimal arrangement of a connected- (r, s) -out-of- (m, n) :F system — the case of $m = r$ and $s = 2$ —,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E98-A, no. 10, pp. 2018–2024, 2015.
 - [29] T. Nakamura, H. Yamamoto, T. Akiba, and K. Shingyochi, “Algorithm for the component assignment problem in redundant consecutive- k -out-of- n :F systems,” in *Proceedings of the 20th Asia Pacific Industrial Engineering and Management Systems Conference*, 2019, pp. 152–157.
 - [30] T. Nakamura, H. Yamamoto, and T. Akiba, “Fast algorithm for optimal arrangement in connected- $(m - 1, n - 1)$ -out-of- (m, n) :F lattice system,” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E101-A, no. 12, pp. 2446–2453, 2018.
 - [31] T. Nakamura and H. Yamamoto, “Algorithm for solving optimal arrangement problem in connected- (r, s) -out-of- (m, n) :F lattice system,” *IEEE Transactions on Reliability*, vol. 69, no. 2, pp. 497–509, 2020.
 - [32] 斎藤 彰, “AI の進化は故障解析に何をもたらすのか ～その期待とリスク～,” *日本信頼性学会誌 信頼性*, vol. 40, no. 2, pp. 64–71, 2018.