

迷路の形と特徴量を一覧できる WEB サイトの構築 ; 「世界共通な固有の迷路名」の提案とともに

今村隆輝¹ 小島俊輔² 木原久美子³

概要 : 迷路は、古くからパズル遊びのツールとして使われ、現在ではゲーム内のダンジョンへ利用されたり、生物の行動学的な実験に用いられたりしている。迷路は限定された空間内に多様な形が構成されているため、ひとことでその形状を表現することが難しい。そこで本研究では、それぞれの迷路に「世界共通な固有の迷路名」をつけることで、迷路の情報を共有できるのではないかと考えた。具体的には、迷路の構造を文字列情報に圧縮する方法として、base64 エンコーディングを適用してユニークな識別子を生成した。さらに、この識別子を文字列として URL に含む迷路ごとの情報を閲覧できる WEB サイトを構築した。ここでは、迷路の固有名と形だけではなく、迷路の特徴量も情報として付加した。また、迷路の正解の経路の形の類似度に基づいて、迷路を段階的に並べ替えて羅列することで、ユーザーにとって使いやすいページを実装した。

キーワード : 迷路、識別子、迷路サイズ、正道、誤道、特徴量、2 次的な構造、パズル、ゲーム

Development of Website for Maze Shapes and Features Perspection

RYUKI IMAMURA¹ SHUNSUKE OSHIMA²
KUMIKO KIHARA³

Abstract: Mazes have long been used as tools for puzzles, and are now used in-game dungeons and for behavioral experiments on biology. However, the mazes were composed of various shapes in a limited space, it is difficult to represent the shape. In this study, we developed a maze website constructed with mazes named by unique identifiers and were aligned to its shapes and features.

1. はじめに

迷路は、古くからパズル遊びのツールとして使われてきた。迷路は単純にスタートからゴールまでの解を探索する方法だけではなく、スタートからゴールまでの解を求めることで絵や文字が浮かび上がる絵画的迷路[1][2][3]や迷路の各マスに漢字が割り振られ、画数の多いマスにのみ進むことができるというような迷路に更に制限を掛けたもの[4][5]など様々なパズル遊びのツールになっており、ベルサイユ宮殿の迷路園や紙の迷路など様々な形で親しまれてきた。

また、現在ではゲーム内のダンジョンの構成にも迷路は使われている[6]。さらにはゲーム内のダンジョンを手続き的に生成するために迷路の自動生成アルゴリズムの開発やアルゴリズムを用いたダンジョン生成が行われている [7][8]。自動生成アルゴリズムを用いることで、ゲームを行うたびに別の形のダンジョンを楽しむことが可能となる。

また、迷路は生物の行動実験にも用いられる。生物の行動学的な実験においては、迷路は実験環境の面積の制御や巣から餌場への移動の困難さを制御するために用いられている[9][10]。

迷路は限定された空間内に多様な形が構成されているため、ひとことでその形状を表現することが難しい。過去

には迷路の自動生成アルゴリズムが生成する迷路の傾向を分析するため、直線の数や分岐点 (T 字路, 十字路) の数などといった基本的な構造などを用いて迷路間の形状の差が解析された[8][11]。しかし、複数の特徴量の値が同じであっても迷路上の出現位置が異なれば迷路の形状も変わる。そのため、現在は迷路が画像のみで共有されており、迷路の情報が共有されない。迷路を用いたほぼすべての研究で迷路が再利用されることはなく、その研究に特異的なものになっている。また迷路を用いたほぼすべての研究でなぜその迷路を選択したのかという理由は書いていない。

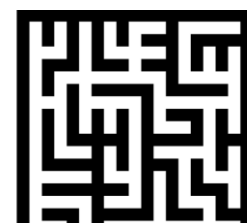


図 1 迷路識別子を生成する迷路の条件を満たしたサイズ 19 の迷路の例

1 熊本高等専門学校専攻科 生産システム専攻
Production Systems Engineering Course, Advanced Courses, National Institute of Technology, Kumamoto College

2 熊本高等専門学校 リベラルアーツ系
Faculty of Liberal Studies, National Institute of Technology, Kumamoto College

3 熊本高等専門学校 拠点化プロジェクト系/生物化学システム工学科
Faculty of Project Centers / Department of Biological and Chemical Systems Engineering, National Institute of Technology, Kumamoto College

そこで本研究では、それぞれの迷路に「世界共通な固有の迷路名」をつけることで、迷路の情報を共有できるのではないかと考えた。本研究では迷路の構造を決定する特定のマスのみに対しエンコードを行うことによって文字列情報に圧縮する迷路のユニークな識別子の生成手法を提案する。そして、迷路の情報共有するために、識別子を URL に含む迷路ごとの情報を閲覧できる WEB サイトを構築する。

2. 迷路配布システム

2.1 要件定義

迷路を実験に用いる実験者が迷路の性質を考慮した上で迷路を選択できることを目指し以下の 3 点を実現する Web サイトの開発を目指した。

- (1) 迷路は構造を決定するユニークな識別子を持つ
- (2) 迷路の性質を示す特徴量を網羅的に確認できる
- (3) 迷路の正解の経路の形の類似度に基づいて、迷路が段階的に並べて一覧される

(1) の要件については、迷路の構造を一意に決定する識別子と迷路データを配布することによって、実験者が以外が迷路を簡単に再現することが可能になることを目指すものである。(2) の要件は、迷路の特徴を数値などの統計値を確認するためである。これまでに迷路の基本的な要素である直線、分岐 (T 字, 十字), 曲がり角, 正解の経路 (正道), 間違いの経路 (誤道) 等の個数や長さを統計的に解析することで迷路間の性質の違いが可視化されてきた[8][11]. 迷路の特徴量を示すことで実験者が自分が用いる迷路がどのような性質を持っているのかを数値として把握することが可能となる。(3) の要件は、迷路を一覧する際に迷路選択において最も重要な特徴である正道の形を基準に段階的に迷路を整列されることで、ユーザがより快適に迷路を一覧, 検索できる仕組みを作ることを目的としたものである。

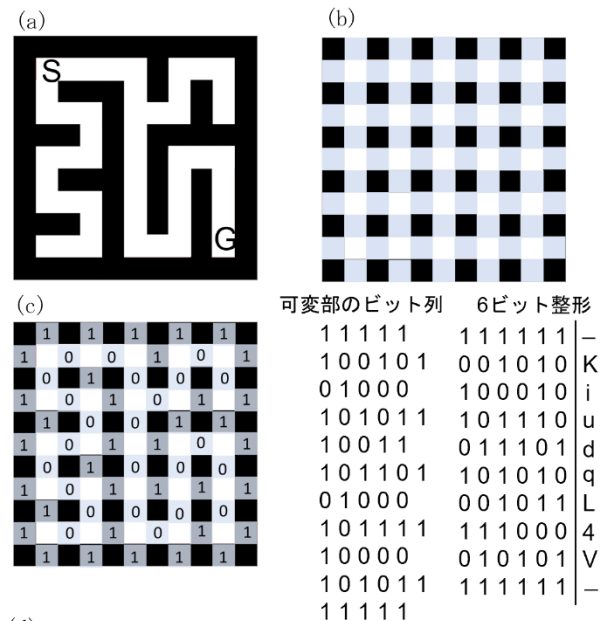
2.2 迷路の識別子の生成

2.1 の要件 (1) を達成するため、本研究では 1,0 のバイナリで表現された迷路の構造を決定するユニークな識別子の生成する。本提案手法では迷路の構造を制御する特定のビット列に着目する。以下の章では、迷路識別子の生成の対象となる迷路の条件と、識別子生成の手順について説明する。

2.2.1 対象とする迷路

本研究で迷路の識別子を生成する迷路の条件を以下に示した。また、図 1 に本研究で取り扱う迷路の条件を満たした迷路の例を示した。

- 迷路を構成する最小単位をマスと呼び、通行できるマスを経路、通行できないマスを壁と呼ぶ。
- 1 マスの形状は正方形である。また、経路と壁の 1 マスの大きさは同じものとする。
- 迷路は外周が 1 マス分壁で覆われているものとする。また、本研究では迷路の左上の壁の座標を (0, 0) として



(d) 生成された迷路識別子 : W11H11S11G99M_KiudqL4V_
図 2 迷路サイズ 11 の迷路の迷路識別子の生成例 : (a) 迷路. (b) 迷路の定常部と可変部. (c) (a) の迷路の識別子生成の例. (d) 生成された識別子. 定常部の壁を黒, 経路を白, 可変部を青色で示した。

扱う。

- 迷路のサイズは外周を覆う壁の縦と横のマス数で定義した。本研究では縦と横のマス数が同じ $n \times n$ マスの迷路を用いた。以降 $n \times n$ の迷路のサイズを迷路サイズ n と表す。また n は必ず奇数となる。
- スタートとゴールを結ぶ解が 1 通りである。これはつまり迷路中にループ (閉路) を含まないことを意味する。
- スタートの位置を迷路の左上の経路, 座標 (1, 1) とし, ゴールの位置を迷路の右下の経路, 座標 ($n-2, n-2$) とする。
- 迷路中には到達できない経路は存在しないものとする。また, 迷路の経路の幅は 1 マスとした。これは, 迷路は左上の座標を (0, 0) とした場合, 行, 列ともに偶数のとき必ず壁となり, 行, 列ともに奇数のとき必ず経路となる。

本研究では、これらの条件を満たすことができる迷路生成アルゴリズムのうち AldousBroder, Division, GrowingTree, Kruskal, Prims, Sidewinder の代表的な 6 つを用いて迷路を生成した。迷路の生成には Python の迷路生成ライブラリである Mazelib を用いた。迷路サイズ 19 の迷路を 6 つのアルゴリズムで 100 個ずつ生成し, 合計 600 個の迷路を本研究では取り扱った。

2.2.2 Base64

Base64 は Multipurpose Internet Mail Extensions (MIME) によって規定されるバイナリデータを扱うための符号化方式である。base64 はバイナリデータを 6 ビットずつ取り出し該

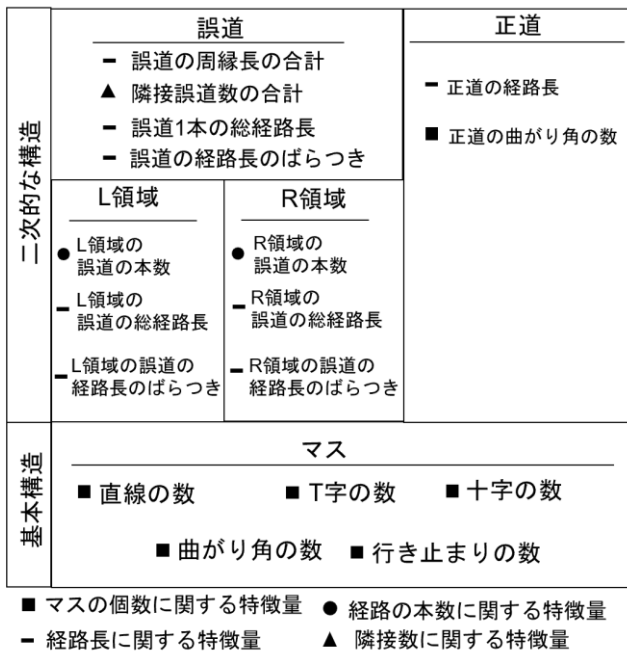


図 3 迷路を構成する特徴量

当するコードへ置き換える符号化方式である。Base64 では $2^6=64$ 通りのパターンをアルファベットの 大文字と小文字の合計 52 文字と数字の 10 文字に加え、+と/を加えた 64 文字に置き換える。しかし、通常の base64 ではエンコード後の文字列に+や/が含まれるため、URL やファイル名として扱づらい。そこで本研究では+を-, /を_に置き換えた base64url encoding を用いた。

2.2.3 迷路識別子の生成手順

迷路の構造からその迷路の構造が一意に定まる迷路識別子を生成する。本研究で対象にする迷路は左上の座標を (0,0) とした場合、偶数行かつ偶数列のとき必ず壁となり、奇数行かつ奇数列のとき必ず経路となる性質をもつ。逆を言えば、この条件以外の、偶数行かつ奇数列のマス値、および奇数行かつ偶数列のマス値によって迷路の形状が決定される。本研究では、マスが壁もしくは経路と定まっている部分を迷路の定常部と定義した。また、迷路の形状を決定するような構造によって変化するマスを可変部と定義した (図 2 (b))。

迷路識別子の生成は迷路の構造を決定する可変部のみに着目して行った。理由として可変部のみに着目することで、識別子の文字列長を抑えることができるためである。また、迷路の定常部と復元した可変部のビット列を組み合わせることで識別子から迷路の構造を復元することも可能である。

識別子の生成は、まず迷路から可変部のビット列を抽出した。抽出した可変部のビット列を先頭の行の左から順に 6 ビットずつに分割し、可変部のビット列を 6 ビットに整形した。整形後の可変部のビット列に対して、base64url encoding を適用し 6 ビットを 1 つの文字に変化し迷路の識

別子を生成した (図 2 (c))。

最後に生成した識別子に迷路の基本情報を付加した。本研究では迷路の基本情報とは迷路サイズ、スタート、ゴールの座標のこととした。迷路の基本情報と対応する文字列を以下の表に示した。スタート、ゴールの座標は S や G の後ろに記した。縦横のマスの桁数がそろっていない場合は先頭を 0 で埋めた。縦横の座標は S や G から続く数字を機械的に半分の長さに分割することで求められる。また迷路識別子の開始を示す文字である M を迷路の基本情報と迷路識別子の間に追加した (図 2 (d))。

表 1 本研究で使用した迷路の特徴量

定義	文字
迷路の縦のマス数	W
迷路の横のマス数	H
スタートの座標	S
ゴールの座標	G

2.3 迷路の特徴量

2.1 の要件 (2) を達成するために提案システムで確認可能な迷路の特徴量を決定した。迷路の性質を評価する特徴量は迷路生成アルゴリズムの分類などに用いられる迷路のマスについての特徴量と、正解の経路の形状などの迷路の構造を記述することができる 2 次的な構造に関する特徴量とがある (図 3) [8][12]。本研究ではその両方を求めることで、多数の尺度での迷路の性質の確認を目指した。

2.3.1 迷路の基本特徴量

迷路中の直線、分岐点 (十字, T 字), 曲がり角, 行き止まりのマスの特徴量として求めた。迷路のマスは接続するマスの数や位置によって直線, 分岐点 (十字, T 字), 曲がり角, 行き止まりの 5 つの基本構造に分類される。迷路のマスの特徴量は迷路の特徴を示す最も基本的なもので、迷路の解析にも頻繁に用いられている [8][11]。

2.3.2 正道に関する特徴量

正道の形状を記述する 2 次的な構造に関する特徴量として、正道の経路の長さとして曲がり角の数を求めた。正道は経路の長さによってある程度形状が制限されるため、正道の経路の長さは正道の形状を示す指標となる。また、曲がり角の数は正道がどれだけ曲がっているかという正道の複雑

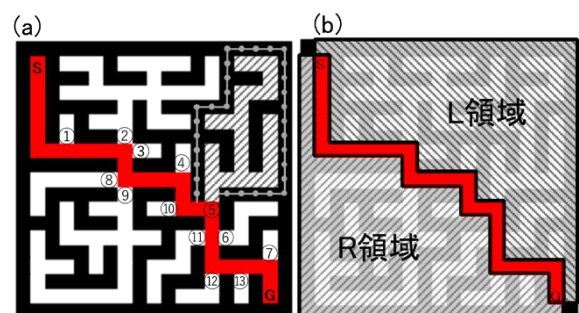


図 4 誤道に関する特徴量のイメージ

さ指標となる[12].

2.3.3 誤道に関する特徴量

誤道の形状を記述する特徴量として、誤道の周縁長の合計と隣接誤道数の合計を求めた。誤道の凹んだ部分を見逃した誤道の外周を周縁とし、外周のマスを周縁長とした。誤道の周縁長の合計は誤道の形状のコンパクトさを記述することができる[12].

また、隣接誤道数は壁を挟んで隣接している誤道の数重複の無いように数えた。隣接誤道数は、経路長が長く、細く迷路全体へ延びるような誤道が存在すると値が高くなる。一方で、経路長が長くても、一部分へ固まるような経路は隣接数が増えないため、隣接誤道数は少なくなる。隣接誤道数の合計はこのような誤道の総経路長や入り組み具合のような構造を記述する指標となる。

誤道 1 本の総経路長を求めた。また各誤道の総経路長から誤道の経路長のばらつきを求めた。本研究ではばらつきとして標準偏差を計算した。

誤道の特徴量のイメージを図 4 (a) に示した。丸数字は各誤道の後述する L 領域から先にスタートからの位置を示したものである。図 4 (a) では誤道の周縁は灰色のドットで示した。この時の周縁長は 31 マスである。また網掛けで示した 5 番目の誤道は 2 番目、4 番目 6 番目の誤道と隣接しており誤道の隣接数は 3 となる。また、5 番目の誤道の総経路長は 26 マスとなる。誤道の経路長のばらつきは 1 番

目から 13 番目までの総経路長の標準偏差を求めたものとなる。

2.3.4 誤道の部分領域に関する特徴量

誤道は正道によって必ず 2 つの領域に二分される。この 2 分された誤道は図 4 (b) のようにスタートからゴールへ向いた時に左側の領域を L 領域、右側の領域を R 領域と定義され、各領域についての特徴量が定義されている[12]. 本研究では L 領域、R 領域の間違った道の本数、総経路長、長さの標準偏差を求めた。

2.4 正道の類似度に基づいた迷路の一覧

2.1 の要件 (3) を達成するために、正道の形の類似度に基づいた段階的な迷路の一覧を行った。本研究では、L 領域と R 領域の誤道の総経路長の比 (式 (1)) を正道の類似度をとした。L 領域と R 領域の誤道の総経路長の比は正道が迷路のどの位置を通っているかを表現することが可能である。例えば、L 領域と R 領域の誤道の総経路長の比が低いほど、L 領域の占める割合は大きくなり、正道は迷路の左下側を通り、比の値が高いほど R 領域の占める割合は大きくなり、正道は迷路の右上側を通る。図 5 では、迷路をランダムで並べたものと、L 領域と R 領域の誤道の総経路長の比で並べたものを比較している。図 5 では L 領域と R 領域の誤道の総経路長の比によって正道の形の類似度に基づいて段階的に迷路が整列されている。

$$LR_size_ratio = \frac{R_size}{R_size + L_size} - 0.5 \quad (1)$$

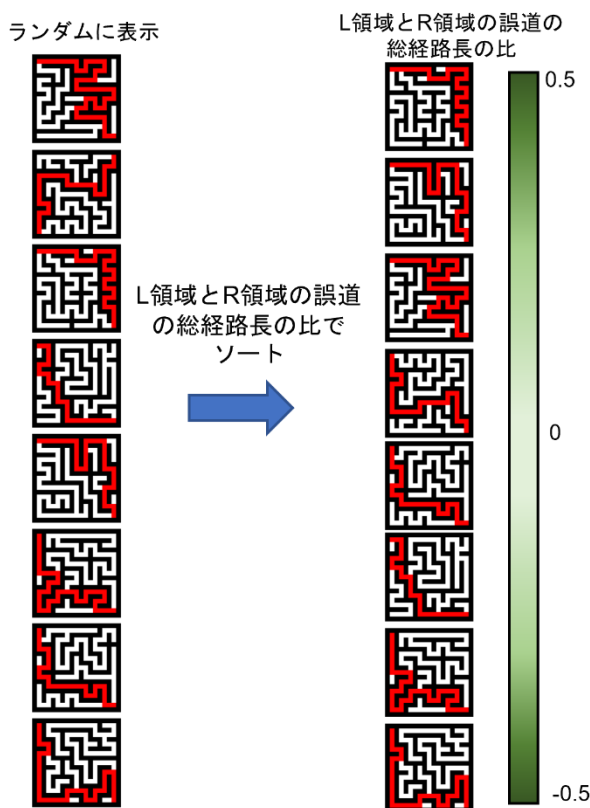


図 5 L 領域の総経路長と R 領域の総経路長の比に基づいて並べた迷路

2.5 システム構成

提案システムの構成を以下の図 6 に示した。本システムは迷路を配布する Web システムのフロントエンド部とフロントエンド部を公開するためのバックエンド部からなる。フロントエンド部は迷路の閲覧、迷路データのダウンロードの 2 つのみが行える簡便なものを目指した。機能が迷路の閲覧とデータのダウンロードのみであるため、自前で実装するサーバーサイドのバックエンド処理を必要とせず、HTML, CSS, JavaScript からなるフロントエンドのみで機能

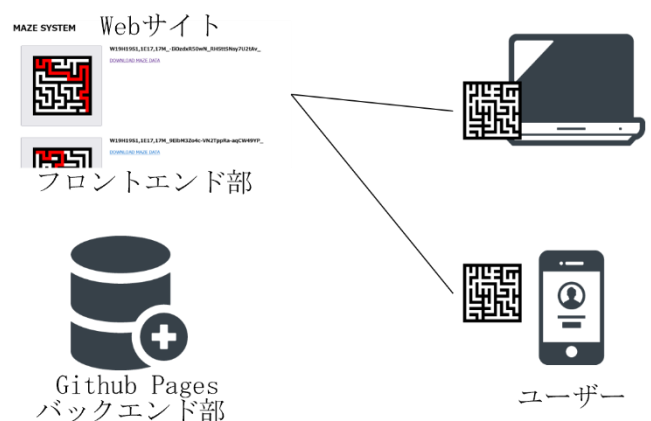


図 6 提案システムの構成

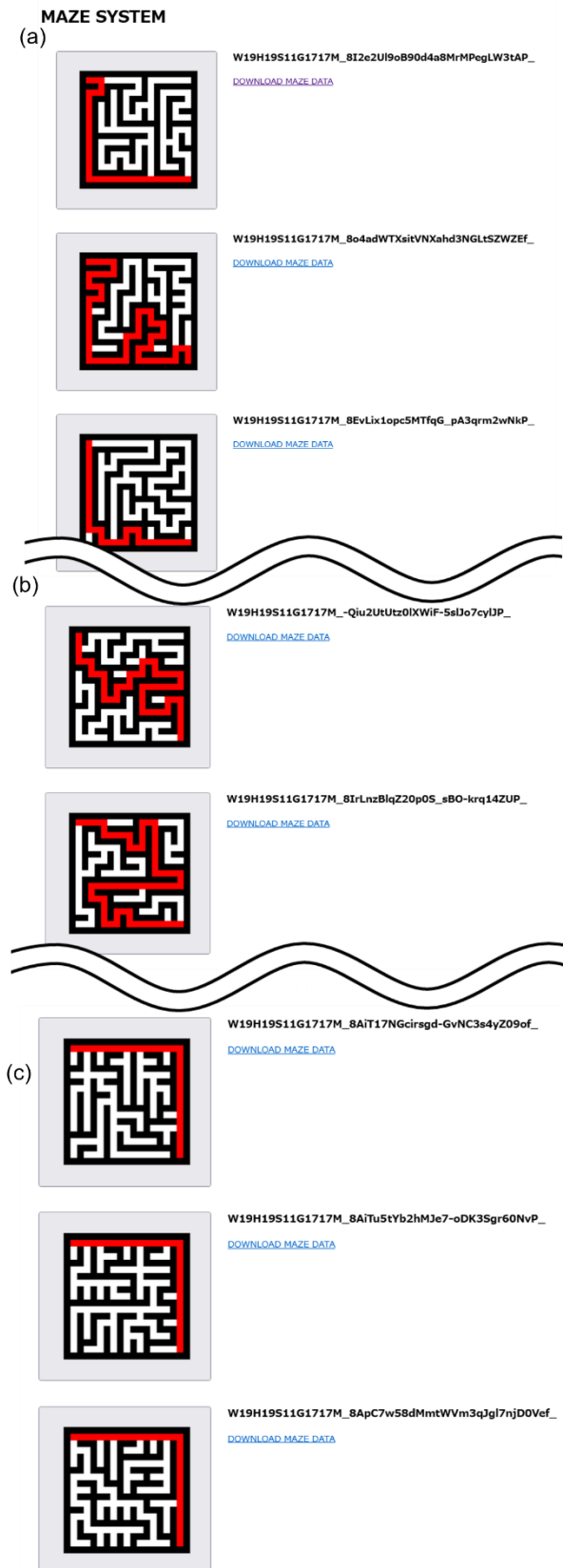


図 7 迷路一覧画面の例:段階的に並んだ迷路を示すため、(a) から (c) まで任意の箇所を抜き出し表示した。



図 8 ダウンロードした迷路データの例

を実装した。本提案システムは Web システムであるため、ブラウザのみがあればどのデバイスであろうと特別な環境を構築することなく、迷路の閲覧、データのダウンロードが行える利点がある。

バックエンド部ではシステムの公開を行った。システムの公開については公開用のサーバーを用意するなどいくつかの方法が考えられたが、最終的に Github Pages を利用してシステムの公開を行った。Github Pages は静的な Web サイトのホスティングサービスの一つであり、HTML, CSS, JavaScript が動作するサイトを簡単に公開できる。また、Github を用いることで迷路画像などのデータを保存するストレージとしての機能も兼ねている。また、Git を用いたバージョン管理やホスティングサービスを使用することによる保守作業の軽減が Github Pages を使用した理由である。

3. 提案システムの開発と実装

本章では実装した提案システムの機能について説明する。提案システムは正道の形の類似度に基づいて整列された 600 個の迷路データを一覧できる迷路一覧画面と、選択した迷路の特徴量や特徴を強調した様々な迷路の外見などの詳細情報を閲覧できる迷路詳細画面の 2 ページからなる。

3.1 迷路一覧画面の実装

迷路一覧画面では、迷路の一覧の表示機能、迷路識別子の表示機能、迷路データのダウンロード機能の 3 つの機能を実装した (図 7)。迷路一覧の表示機能では迷路を正道の形が似ている順に並べて表示した。図 7 (a) は先頭から 3 番目までの迷路、(b) は 279 番目と 280 番目の迷路、(c) は 597 番目から 600 番目の迷路を示した。図 7 (a) (b) (c) を見ると正道が迷路の左下側を通る迷路から迷路の右上側を通る迷路へ順に並んでいることがわかる。また正解の経路の形状は迷路を用いた実験において最も重要かつ最初に確認する項目であると考えられるため、正解の経路に色付けした迷路の画像を表示した。また迷路はその迷路の詳細画面へのリンクとなっており、迷路をクリックすることで後述する迷路詳細画面へと移動することができる。各迷路の右側には、対応する迷路識別子と迷路データのダウンロードリンクを表示した。迷路データのダウンロード機能で

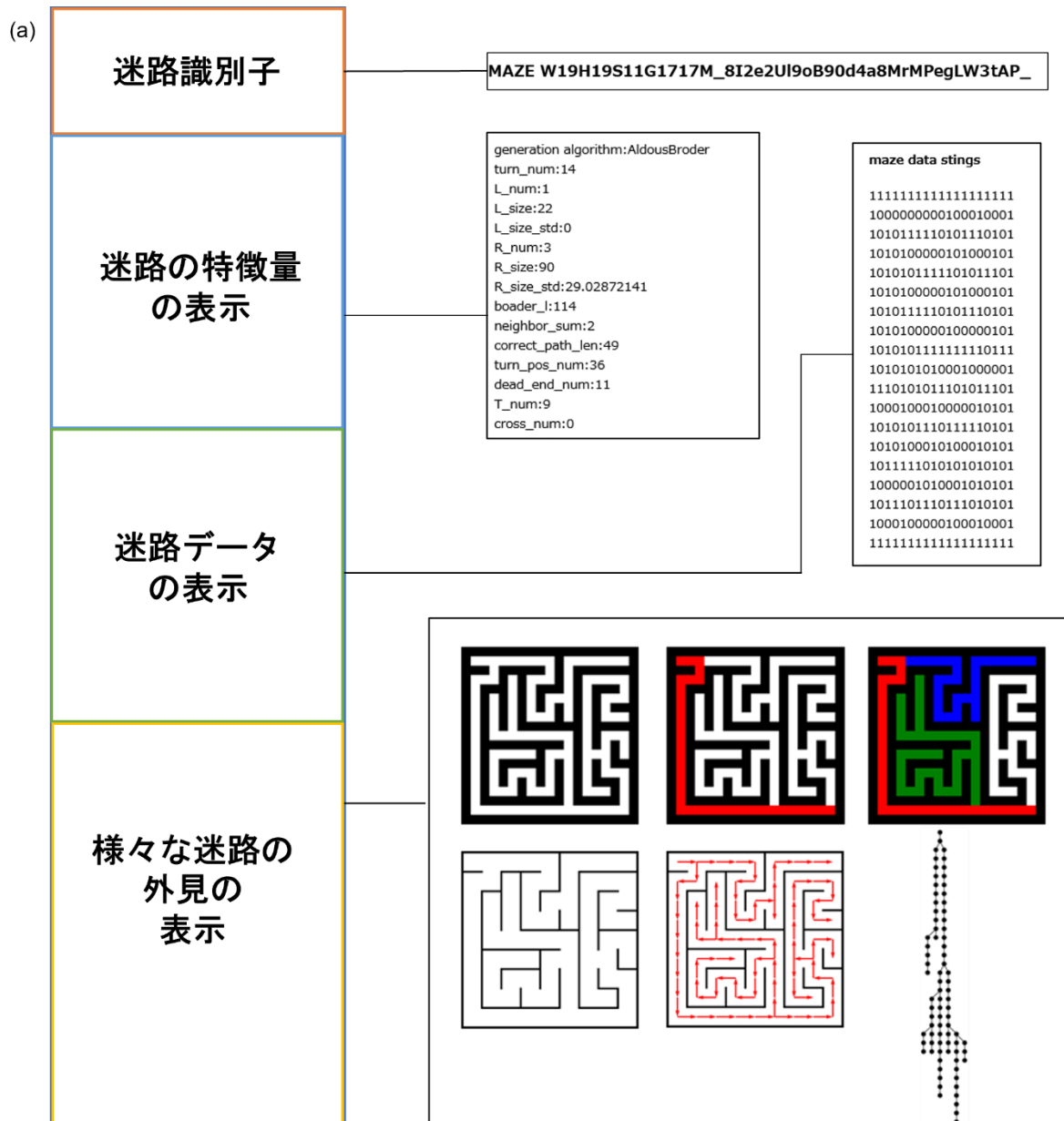


図 9 迷路詳細画面の例：(a) 迷路詳細画面の内容を表した模式図．迷路識別子，迷路の特徴量の表示，迷路データの表示，様々な迷路の外見画像の部分は実際の迷路詳細画面の一部をトリミングし右に拡大し表示した．

は，ダウンロードリンクをクリックすることによって，迷路のバイナリデータをテキスト形式でダウンロードすることが可能である (図 8)．

3.2 迷路詳細画面の実装

迷路詳細画面では，迷路の特徴量の表示，迷路データの表示，様々な迷路の外見の表示の 3 つを実装した (図 9)．迷路詳細画面ではまず先頭に詳細を表示している迷路の識別子を表示した (図 9 (b))．次に迷路の特徴量は前述した迷路の特徴量の値を表示した (図 9 (c))．迷路の特徴量の数値を確認することで，詳細を表示している迷路の性質を確認することができる．次に，ダウンロードできる迷路データを表示した (図 9 (d))．最後に通常の迷路と特徴量の

数値のみでは迷路の選択が困難であると考え，様々な迷路の外見を表示した (図 9 (e))．図 9 (e) では，通常の迷路，正道を赤色で示した迷路，正道を赤色，誤道を各色で色付けた迷路，壁が薄い (経路：壁=9:2) の迷路，壁が薄い迷路のスタートから末端までの方向を示した迷路，壁が薄い迷路を元に作成した木の画像を示した．

迷路を選択するユーザーによっては正道に着目をする場合と誤道に着目する場合や，迷路自体も壁と経路が 1:1 の迷路を想定しているユーザーと壁が薄い迷路を想定しているユーザーの両方がいるであろう．また迷路を抽象化した木としての外見で迷路を選択するユーザーもいると考えられる．提案システムのように多くの迷路の外見を表示す

ることで幅広くユーザーのニーズに対応できる。

4. 結論

本研究では迷路の構造を決定する特定のマスのみに対しエンコードを行うことによって文字列情報に圧縮することで、迷路の構造を決定するユニークな識別子の生成手法を提案した。そして、迷路の情報共有するために、識別子を URL に含む、迷路ごとの情報を閲覧できる WEB サイトを構築した。

参考文献

- [1] 池田 心, 橋本 隼一. 確率的最適化を用いた絵画的迷路生成. 情報処理学会論文誌. 2012, vol. 53, no.6, p.1625-1634.
- [2] 岡田吉央, 上原隆平. 絵画的迷路の作り方. 数理解析研究所講究録. 2009, vol. 1649, p. 58-65.
- [3] 中井亮平, 岡田吉央. 絵画的迷路作成アルゴリズムの改善. 数理解析研究所講究録. 2010, vol. 1691, p. 162-166.
- [4] 嘉来進之助. 左脳を鍛える大人の迷路. 主婦の友社, 2004,p. 6-11.
- [5] 米山公啓. 右脳と左脳を鍛える！大人の迷路パズル傑作選. 廣済堂出版, 2006, p30-31.
- [6] Nintendo. 2017. The Legend of Zelda: Breadth of the Wild.
- [7] 藤平 啓汰, シュエ ジュウシュエン, 池田 心. 人間らしさを考慮したテストプレイヤを用いる迷路の自動生成と難易度評価. The 26th Game Programming Workshop 2021, 2021, p.192-199.
- [8] Kim et al. Design-Centric Maze Generation. FDG'19, 2019, p.26-30.
- [9] Lloyd D Stringer et al, Mazes to Study the Effects of Spatial Complexity, Predation and Population Density on Mate Finding, insects, 2020, vol.256, p.11-4.
- [10] Darar Bega et al. The effect of maze complexity on maze-solving time in a desert ant. Behavioural Processes. 2019, vol.166, 103893.
- [11] Jamis Buck. Mazes for Programmers: Code Your Own Twisty Little Passages. 1, Pragmatic Bookshelf, 2015,p249.
- [12] 今村隆輝,小島俊輔,木原久美子. 迷路の構造を記述するための新たな特徴量の提案と特徴量同士の関係性の考察. 火の国情報シンポジウム 2022.