Chemical Engineering Science 250 (2022) 117416

Contents lists available at ScienceDirect

Chemical Engineering Science

journal homepage: www.elsevier.com/locate/ces

Transport in mazes; simple geometric representations to guide the design of engineered systems

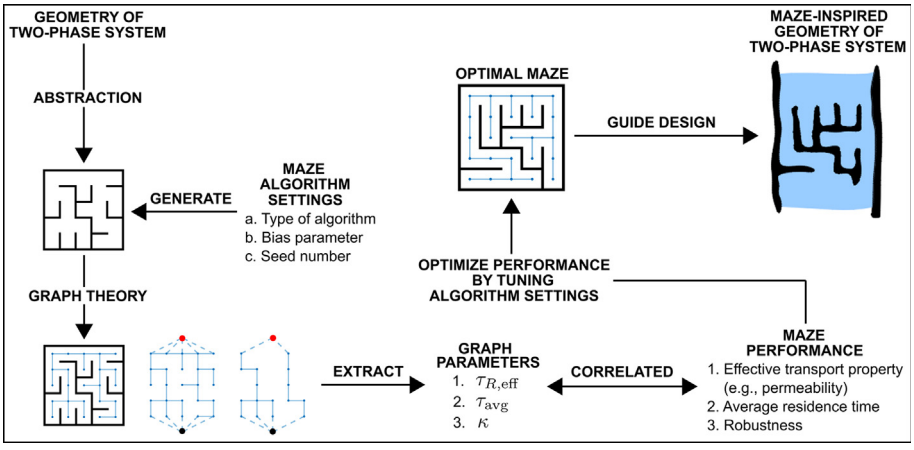
Alex Guo, William C. Marshall, Corey C. Woodcock, Joel L. Plawsky *

Howard P. Isermann Department of Chemical and Biological Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA

HIGHLIGHTS

- Abstraction using mazes increases understanding of geometric effects on transport.
- Graph theory can be used to determine a maze's transport-related performance.
- Maze generation algorithms can be modified to tune a maze's performance.
- An optimized maze's geometry can guide the rational design of engineered systems.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 11 September 2021
Received in revised form 4 December 2021
Accepted 30 December 2021
Available online 5 January 2022

Keywords:
Mazes
Graph theory
Transport processes
Tortuosity
Porous media
Permeability

ABSTRACT

Although engineers can control the internal geometry of materials down to the micro-scale, it is unclear what configuration is ideal for a given transport process. We explore the use of mazes as abstract representations of two-phase systems. Mazes can be easily generated using many different algorithms and then represented as graphs for analysis. The three, dimensionless graph parameters of effective tortuous resistance, average tortuosity, and minimum-cut-size were derived and then correlated to the maze's effective transport property (e.g., permeability), average residence time, and robustness, respectively. It was shown that by tuning the settings of the maze algorithm, one can obtain desired maze performance. Finally, a composite maze was constructed and shown to mimic the geometry and permeability of a real commercial membrane. In principle, a surrogate maze geometry can be optimized/tuned for a given transport process and then used to guide the rational design of the engineered system it represents.

1. Introduction

Labyrinths and mazes have fascinated human beings for generations — from the Minotaur's labyrinth to the tombs of ancient Egypt. Today, mazes are viewed primarily as puzzles found in novelty books, Victorian gardens, spent corn fields, and the heart of most adventure video games. We are interested in whether mazes could have any useful application for transport processes. Modern mazes are usually not crafted by hand but are generated by a type of goal-seeking algorithm (Buck, 2015). A square maze with 25 cells generated by one such algorithm is shown in Fig. 1.

* Corresponding author.
E-mail address: plawsky@rpi.edu (J.L. Plawsky).

https://doi.org/10.1016/j.ces.2021.117416
0009-2509/

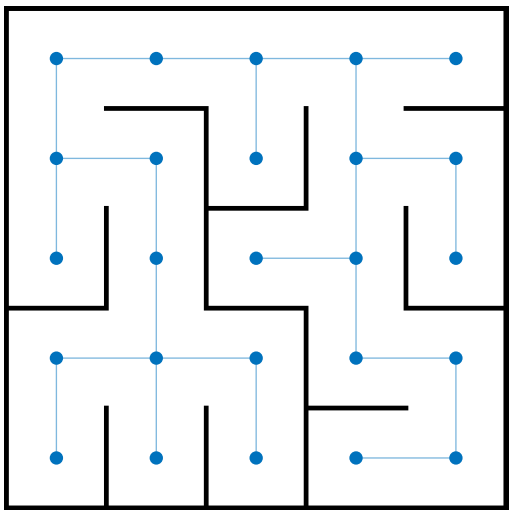


Fig. 1. (single column; online color only): A 5 × 5 maze, generated with the Sidewinder algorithm, with its graph overlayed on top of it.

The cellular nature of mazes ensures that they are readily represented and analyzed by the tools of graph theory, which is a versatile theory that can be used to model pair-wise relationships in complex systems (Biggs et al., 1986; Chartrand and Zhang, 2012). We convert a maze into its graphical representation following Fig. 1. The graph *nodes* represent the center of maze cells, and the graph *edges* represent the *channel segments* of the maze. Moreover, each edge contains an *edge weight* that is used to store information about the edge, such as the geometric distance between the two nodes that it connects. An edge weight can also be used to store additional information if required (e.g., channel segment width).

Fundamentally, we operate under the assumption that many transport processes in natural or engineered systems can be viewed as negotiating some type of maze. However, we restrict our scope here to systems that contain two phases, and we call these systems *transport media*. Examples of transport media include fluid flow within a porous medium, the flow of heat in a composite material, the flow and distribution of electrical signals within an integrated circuit, and the flow of optical signals through a photonic crystal. There has been extensive work using periodic media, stochastically generated media, and fractal media to abstractly represent transport media and to develop the insights and the mathematics needed to describe how the geometry of a material affects its transport properties (Matyka et al., 2008; Koponen et al., 1997; Andrade et al., 1997; Wang et al., 2007; Hyman et al., 2012; Martys and Garboczi, 1992; Siena et al., 2014; Zhou et al., 2016; Araújo et al., 2006; Tartakovsky, 2010; Ye et al., 2017; Adler, 1992; Sahimi, 2003; Sahimi, 2003). However, in each class of the artificially-generated media, the produced geometries are either too deterministic or too random or not readily fabricated. Moreover, the algorithms used almost always focus on generating the solid phase instead of generating the "active" phase (i.e., channels) that will host the flow of fluid, heat, charge, etc. On the other hand, mazes have their channels directly generated and in a pseudo-random manner — there is a balance between randomness and control over how the flow paths will be connected to each other. Thus, mazes are the ideal, abstract representations of reality needed to better understand how geometry affects transport and to build new useful geometries for transport operations.

To the best of the authors' knowledge, mazes have not been used as an abstraction to study the effects of geometry on transport; however, mazes have been used as an abstraction to find the shortest solution within multi-solution systems. These multi-solution systems can include transport media (Kesavan and Chandrashekar, 1972; Marle, 1981; Bermond et al., 1986), traffic networks (Kalamaras et al., 2000), logistics operations (Bramel and Simchi-Levi, 1998), and many other systems that now also come under the umbrella of network theory (Barnes and Harary, 1983). A *solution* to a maze is defined as a unique sequence of channel segments that connects a specified inlet cell to a specified outlet cell. A *path* is a sequence of edges that connects any two nodes in a graph. A *multi-solution* maze is a maze with more than one possible solution. The maze in Fig. 1 is a single-solution maze; however, if one wall of any cell is removed, then it becomes a multi-solution maze. Once a multi-solution system is converted into a graph, the shortest path between two specified nodes can be found using the Breadth-First Search (BFS) algorithm if the graph's edges all have equal weight. In general, a system's graph will have edge weights that aren't equal; in this case, Dijkstra's algorithm can be used (Cormen and (Ed.), 2009). In the literature, mazes are used as an abstraction for multi-solution systems and the focus has been to develop novel methods that can find the shortest path of a multi-solution maze more efficiently than the BFS or Dijkstra's algorithm. These novel maze-solving methods have employed the use of fluid flow (Fuerstman et al., 2003; Lovass et al., 2015; Lagzi et al., 2010), slime molds (Adamatzky, 2012), and memristors (Sarmiento-Reyes and Rodriguez-Velasquez, 2018). Although finding the shortest path is important, for a generic transport process or operation, the three transport properties of throughput, average residence time (i.e. average travel time), and system robustness (Najjar and Gaudiot, 1990) are more important to the design. These properties of transport through mazes are what we will focus on.

There has been research where the complex geometries of natural or engineered systems (e.g., porous media) have been converted directly into graphs. These graphs have been analyzed as fast, convenient ways to understand some aspects of the transport process occurring in the media without having to use expensive simulations (Cocco et al., 2017; Xu et al., 2014). However, graph theory is a mathematical tool and cannot provide insight into exactly how the internal geometry (i.e., configuration of channels) of a material affects its transport properties. To address this challenge, we believe that artificial mazes can be used as acceptable abstractions or surrogates for representing transport media. Once a suitable fitness metric for a transport process is specified, the deterministic and stochastic elements of maze generation can be leveraged, and a form of guided natural selection can be applied to find the optimal or "fittest" configuration of its channels. With the internal geometry of the optimal maze in hand, one can use it to guide the rational design of transport media or even use additive manufacturing to fabricate and directly use the maze as a transport medium. In Section 2, we compare various maze generation algorithms and discuss how the different maze algorithm biases deterministically lead to mazes with vastly different geometrical biases. In Section 3, we use graph theory to derive three graph parameters that can be correlated to the three transport properties of throughput, average residence time, and robustness of the maze. In Section 4, we show how the act of simply choosing a maze algorithm bias can act as a coarse tuning knob and the modification of an algorithm parameter can act as fine tuning knob for obtaining desired graph parameters and thus desired transport properties. Finally, in Section 5, we will show how to construct a composite maze that can mimic the unique geometric features and throughput of a two-dimensional image of an actual membrane material.

2. Common maze algorithms and characteristics

The most common maze generation algorithms, or maze algorithms for short, first require the generation of a 2D tiling of regular polygons. Then, certain walls of each polygonal cell are removed/-carved out to form passages. Here, we focus only on the simplest 2-D tiling composed of square cells. Maze generation is pseudo-random in that it uses a seeded, random number generator whenever a random decision (such as which wall to remove upon visiting a cell) is required; knowing the seed number allows one to reproduce a maze geometry identically. Maze algorithms incorporate randomness in different forms and to different degrees, leading to vastly different generated mazes (Buck, 2015). 3-D and higher dimension mazes are fundamentally based on analogous algorithms.

Four prominent and distinct maze algorithms are presented here: Sidewinder, Prim's, Recursive Backtracker, and Aldous-Broder algorithms (Buck, 2015). All four algorithms will generate *perfect mazes*, that is mazes that have no inaccessible areas nor looping channels. Perfect mazes are always solvable, and they are single-solution if exactly one inlet cell and exactly one outlet cell are specified. The maze in Fig. 1 is a perfect maze. Three-dimensional mazes with regions of inaccessibility and loops may have many advantages, or even be essential, in membrane and wicking systems for example, but their detailed properties are outside the scope of this paper. The primary difference between each maze algorithm lies in the maze bias. *Maze bias,* or equivalently *channel bias,* is defined as the characteristic property that the channels in a generated maze have. Maze bias can be visualized by performing a breadth-first search from the center cell/node of the maze and then coloring all the other cells based on their distance to the center cell. Examples of mazes generated by the four algorithms and their biases are presented in Fig. 2.

Sidewinder: The Sidewinder algorithm generates mazes with a vertical, or top-down, bias. The starting cell is the bottom-left corner cell, and the algorithm traverses the bottom row of cells while randomly carving out a north, east, or west wall upon visiting each cell. Then, the algorithm works on the row of cells above and repeats the process until the top row of the grid is reached. Sidewinder mazes will never have vertical walls in the top row of cells because the algorithm must carve a north or vertical wall as it visits a cell, of which the former type cannot be removed if we're at the top row of the grid. A 25 $\times$ 25 maze generated using the Sidewinder algorithm and its bias are shown in Fig. 2a.

Prim's algorithm: Prim's algorithm generates mazes with a radial bias. (Note that there is a separate algorithm of the same name used for finding minimum spanning trees of weighted, cyclic graphs (Kershenbaum and Van Slyke, 1972); the two algorithms are similar but are for completely different applications.) Since Prim's algorithm does not traverse the grid in a linear fashion like Sidewinder, it requires the selection of a starting cell in which to start the algorithm; the radial bias originates from the chosen starting cell. Additionally, the usual maze-generating Prim's algorithm assigns a random weight to each cell before randomly traversing cells and carving out walls; however, we use a simplified version of Prim's algorithm that doesn't perform this weight-assigning step in order for the resulting radial bias around the chosen cell to be more prominent (Buck, 2015). Fig. 2b shows a Prim's maze generated with the cell at the center of the maze chosen as the starting cell. Prim's maze bias looks like the channels diffuse outward from the starting cell.

Recursive Backtracker: The Recursive Backtracker algorithm generates mazes with a long-channel bias and requires the selection of a starting cell. The algorithm performs a random walk but with a memory of its path for the purpose of backtracking once a dead-end is reached; so, there is a long-channel bias, but no directional bias like with Sidewinder and Prim's. Moreover, there is no dependence of the maze bias on which starting cell is chosen, unlike with Prim's mazes. In Fig. 2c, the chosen starting cell is at the center of the maze.

Aldous-Broder: The Aldous-Broder algorithm is designed to generate mazes with no bias at all. It is similar to the Recursive Backtracker algorithm and uses a random walk traversal method. The Aldous-Broder algorithm differs from Recursive Backtracker in that it retains no memory of past steps, so there is no passage-length bias. The chosen starting cell in Fig. 2d is also at the center of the maze.

Since Sidewinder mazes have a vertical direction for their bias, they can be used to represent transport media that has flow occurring primarily from top to bottom (e.g., the top-down flow through a membrane that we consider in Section 5). Similarly, the radially-outward bias of Prim's mazes could be appropriate for electrical interconnect networks or clock distribution signals. The Sidewinder and Prim's algorithms generate mazes efficiently because each cell in the grid is visited exactly once. The Recursive Backtracker algorithm is also efficient because it visits each cell exactly twice, once when doing a forward-search and again when backtracking. The Aldous-Broder algorithm has no memory of the random walk it has taken, so there is no limit to how many times each cell is visited before the maze can be finally generated. There are alternative algorithms for generating mazes with no directional or channel-length bias, however they also suffer from the same form of inefficiency exhibited by the Aldous-Broder algorithm (Buck, 2015). Maze algorithms simply prescribe rules for how the channels should be generated, and less complicated rules yield more randomness (less maze bias) in the final geometry. When using a maze to represent how a transport medium should be constructed, the most suitable form of channel bias will change depending on

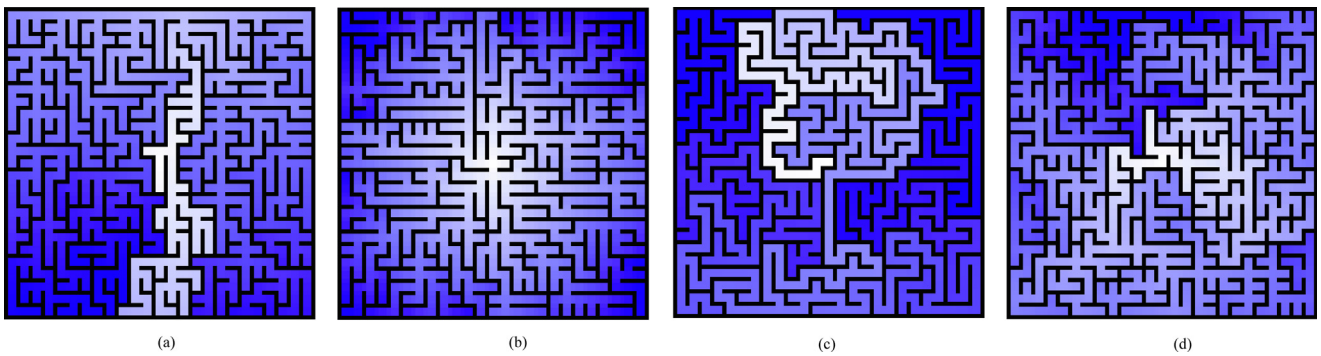


Fig. 2. (double column; online color only): Channel bias visualized for 25 $\times$ 25 mazes generated with the (a) Sidewinder, (b) Prim's, (c) Recursive Backtracker, and (d) Aldous Broder algorithm. Darker colors correspond to longer distances to the center cell.

the engineering application in question; however, some amount of bias will generally be beneficial.

3. Mazes, graphs, and metrics

Many transport processes can be seen as a flow of elements through a resistive medium. These elements can be fluid/solid particles, phonons, electrons, molecules, etc., while the resistance is represented using transport and material properties. In this section, we convert mazes to graphs and use aspects of graph theory to derive three parameters that will be related to the transport properties of the maze. These parameters are: 1) the normalized overall graph resistance, which is inversely related to the overall throughput of the maze; 2) the normalized average distance through the graph, which relates to the average residence time through the maze; and 3) the "cost" required to completely restrict flow through the graph, which is related to the robustness of the maze. All three parameters are highly dependent on how the inlet(s) and outlet(s) of the maze/graph are chosen.

3.1. Preliminary considerations

We first clarify the analogies between mazes and graphs by using the 5 × 5 maze in Fig. 1 as an example. This maze was generated using the Sidewinder algorithm and is thus a perfect maze without any channels that form loops; consequently, its corresponding graph does not contain *cycles*, which are defined as a looping sequence of edges. A *channel segment* in the maze corresponds to a single edge in its graph. A channel in the maze corresponds to a path in the graph. A *channel* is a sequence of channel segments in a maze, while a *path* is a sequence of edges that connects two nodes in the graph without passing through any given edge or node twice. A solution of the maze corresponds to a path that connects any inlet with any outlet of the graph. The number of neighboring cells that a cell is connected to in the maze corresponds to the degree of the node in the graph, where the *degree* of a node is the number of neighboring nodes it is connected to.

Many instances of transport involve flow from multiple inlet cells (source nodes) to multiple outlet cells (sink nodes). For these cases, a *supernode* can allow multiple nodes to be effectively represented as a single node. The graph from Fig. 1 is shown with a supersource and supersink node in Fig. 3a. The maze now allows elements to flow from the entire top boundary (represented by supersource node S) to the entire bottom boundary (represented by supersource node T). Note that the addition of the supernodes (i.e. allowing multiple inlet and outlet cells) makes the maze have multiple solutions and introduces cycles into its graph. Unless stated otherwise, "graph" will henceforth refer to the entire graph containing the two supernodes. Also, note that the graphs of perfect mazes will almost always have cycles but will never have *interior cycles*, which we define as cycles that don't intersect either of the supernodes. The decision of which nodes to connect to the supernodes will affect the direction of flow and may be fine-tuned for a given physical model. For brevity, we define any set of nodes connected to the supersource as N_s and any set connected to the supersink as N_T.

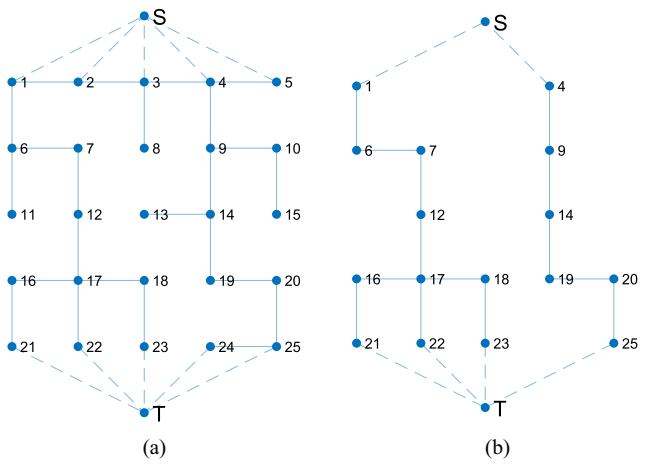


Fig. 3. (single column; online color only): (a) Graph from Fig. 1 but with the supersource and supersink nodes added in (dashed-line edges have half the weight of solid-line edges for reasons described in the text). (b) Reduced graph with inactive edges removed.

To facilitate the calculation of the three transport properties of the graph, *edge weights* provide useful information and give the edges a magnitude. Throughout the majority of this paper, edge weights will be used to simply represent the distance between nodes. Since we place each node at the center of a maze cell, the distance between two adjacent nodes is equal to the cell side length. Referring to Fig. 1, one can see that the distance between the supersource node (top boundary of the maze) and any of the inlet nodes is ½ of the cell side length; so, the dashed edges connected to the supersource node S of the graph in Fig. 3a carry ½ the edge weight of solid-line edges. The same is true for the supersink node T. Other variables affecting the overall conductance or resistance of the medium being modeled, like material properties, channel dimensions, etc., can be included in the edge weights as needed (this will be done in Section 3.5).

Finally, depending on the nature of the transport process being modeled, one may reduce the computational costs of analyzing the maze geometry by removing dead-end edges/channel segments. Such instances are common in many steady-state processes where the movement of elements to and from dead-end channel segments (possibly due to diffusion) is negligible compared to the flow in channel segments with advection. One can perform this reduction in the graph by recursively removing all nodes of degree one until one is left with a *reduced graph* or proper subgraph of the original. If one were to follow this proposed reduction process, one can see that the edge connecting S to node 2 and the edge connecting node 2 to node 1 in Fig. 3a will remain, despite the fact that these edges may contribute almost nothing to the overall flow of elements from S to T (as a result of a shorter path involving the edge that directly connects S to node 1). Thus, the proposed reduction procedure is insufficient, and some nodes need to be removed in the node sets N_S and N_T, even if those nodes have a degree greater than one. The actual procedure we use for reducing a graph is to first remove all edges between the nodes in N_S, do the same for nodes in N_T, and then recursively remove all nodes of degree one. The reduced graph corresponding to Fig. 3a is shown in Fig. 3b, where the four paths from S to T, which represent the four solutions to the maze, retain only the active edges of the original graph. This is the final version of the graph from which we extract the three graph parameters in Sections 3.2-3.4. In Section 3.5, we relate the graph parameters to their corresponding transport properties.

3.2. Effective graph resistance

According to constructal theory, "For a finite-size system to persist in time (to live), it must evolve in such a way that it provides easier access to the imposed (global) currents that flow through it" (Bejan, 1997). There is an optimization problem known as the "max-flow problem," which tries to optimize the overall throughput/flow through a graph while satisfying constraints on the maximum flow allowed through each edge (Schrijver, 2002). Though related to the analysis here, we consider a slightly different

problem where there are no *a priori* fixed limits on the maximum, allowable flow through each edge. The overall throughput of a graph, or its ability to permit flow through it, is instead related to the reciprocal of its effective resistance. To follow constructal theory, we assume that the effective resistance of the reduced graph that contains only the active flow paths is essentially equivalent to that of its original graph. The effective resistance of a reduced graph depends on the resistance to the flow of elements through the edges connecting the supersource to the supersink. For generality, we don't account for parameters such as channel segment widths and viscosity until Section 3.5 and assume here that the resistance to flow in the i th edge of the reduced graph, R_i, is equivalent to just its length or the physical distance, l_i, between the two nodes that it connects. After setting all edge weights of the reduced graph to $R_i = l_i$, we can calculate the effective resistance of a reduced graph using classical ideas from the theory of resistor networks, where series and parallel resistor networks within the graph are simplified using Equations (1) and (2), respectively.

$$R_{series} = \sum_i R_i \tag{1}$$

$$R_{parallel} = \left(\sum_i \frac{1}{R_i}\right)^{-1} \tag{2}$$

We calculate the effective length resistance, $l_{R,\mathrm{eff}}$, between S and T by distilling (i.e., simplifying) the entire resistor network down until we obtain a single resistor, which is shown on the right in Fig. 4. Although $l_{R,\mathrm{eff}}$ has units of length, we use the subscript "R" to denote that it fundamentally represents a kind of effective resistance of the graph, and it can be greater or less than the nominal length or size of the graph, L. The effective length resistance $l_{R,\mathrm{eff}}$ calculated here is usually termed the *effective resistance distance* in graph theory (Klein and Randić, 1993; Ellens et al., 2011).

The effective length resistance of a graph is a length variable akin to a physical separation and represents a fundamental form of resistance in most transport processes. Tortuosity is a parameter used in many different academic fields (Clennell, 1997), and it is generally defined as the ratio between the length of a curved path between two points and the straight-line distance between them. Classically, tortuosity can never be less than 1. Here, we define a tortuosity variant, $\tau_{R,\mathrm{eff}}$, as:

$$\tau_{R,\mathrm{eff}} = \frac{l_{R,\mathrm{eff}}}{L}, \tag{3}$$

where $\tau_{R,\mathrm{eff}}$ is the effective tortuous resistance and L is the straight-line separation distance between S and T. Since $l_{R,\mathrm{eff}}$ can be less than L when many parallel paths linking S and T exist, it is possible and likely for $\tau_{R,\mathrm{eff}}$ to be less than 1. For the reduced graph in Fig. 3b, $\tau_{R,\mathrm{eff}} = 0.56$.

The *effective length resistance* $l_{R,\mathrm{eff}}$ describes the overall length resistance to the flow of elements through a graph. We made this measure dimensionless by defining an *effective tortuous resistance* $\tau_{R,\mathrm{eff}}$, which is our first dimensionless graph parameter. It provides a ratio of the effective length resistance to the minimum separation resistance distance between the supersource and supersink nodes. In the context of mazes, the effective tortuous resistance accounts for a reduction of flow resistance with an increasing number of parallel channels between the inlet and outlet (boundaries) of the maze.

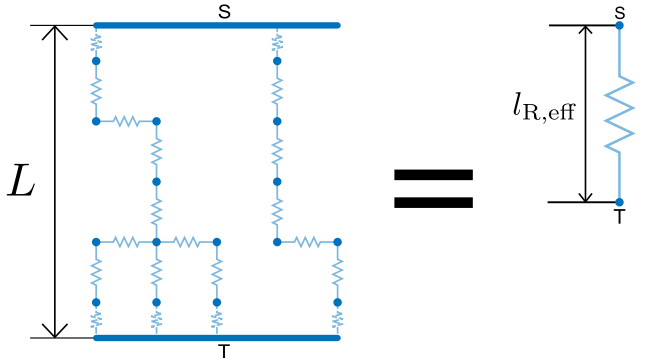


Fig. 4. (single column; online color only): A resistor network corresponding to the reduced graph from Fig. 3b but with supernodes represented as lines (left) and its single, distilled, equivalent resistor (right). The length of each resistor is proportional to its length resistance value.

3.3. Normalized average path length

To obtain the second dimensionless graph parameter, we now extract the second length variable from the reduced graph in Fig. 3b, which is the *average path length*, l_{avg}. Our definition of the average path length is the average length of all paths that specifically connect the supernodes S and T, and it should not be confused with the standard definition in graph theory which instead pertains to the average length of the shortest path between all possible pairs of nodes. The procedure for obtaining l_{avg} is to first assign each edge a weight based solely on physical distance such as in Section 3.2, find all paths between S and T, and average the length of those paths. Note that the second step of finding all possible paths between two nodes in a graph is NP-hard, meaning it is among the class of problems with the highest computational complexity (e.g. the traveling salesman problem) (Matyka et al., 2008); so, finding the average path length for very large graphs with many interior cycles (which we don't deal with in this paper) will be computationally infeasible. Here, we use a modified breadth-first search algorithm to find and record all possible paths between the two supernodes. We normalize the length variable by the nominal maze length to obtain the average tortuosity τ_{avg} as:

$$\tau_{\mathrm{avg}} = \frac{l_{\mathrm{avg}}}{L} \tag{4}$$

The average tortuosity, which is our second dimensionless graph parameter, behaves in a familiar way to the classical tortuosity definition: as the average path length approaches the minimum separation distance between S and T, the average tortuosity approaches a lower limit of 1, and as the average path length increases, so does the average tortuosity. In terms of mazes, l_{avg} physically represents the average length of all possible solutions to the maze. For the reduced graph in Fig. 3b, $\tau_{\mathrm{avg}} = 1.27$.

3.4. Maze/graph robustness

The third, and final, dimensionless graph parameter we extract from the reduced graph in Fig. 3b is the *min-cut-size*, κ, which is a measure of the resilience or robustness of a graph with supersource node S and supersink node T. To measure the resilience of an arbitrary graph to flow disruption, we use ideas from the $S-T$ *minimum cut problem* (Nagamochi and Ibaraki, 1992). We let the nodes of the arbitrary graph be partitioned into two disjoint sets of nodes, N_A and N_B, where N_A contains the source S and N_B contains the sink T (note that these definitions are vastly different from the definition of the inlet nodes N_S and the outlet nodes N_T). Then, the S–T cut, C, is defined as the set of edges in the graph that connect one node in N_A and the other node in N_B. The cut-size of C, sometimes called the capacity, is the sum of the edge weights of all edges in C. The $S-T$ minimum cut problem aims to find a partitioning of nodes that yields an optimal set of edges C_{opt} with the smallest cut-size. The smallest cut-size of the graph, or its *min-cut-size* κ, is

$$\kappa = \sum_{e_i \in C_{opt}} e_i, \tag{5}$$

where e_i is the edge weight of the i th edge. If the graph is unweighted (has equal edge weights), then κ is just the minimum number of edges that must be removed to completely disconnect flow from S to T (i.e., no paths exist between them).

The min-cut algorithm finds the "weakest links" in the graph, so an edge that is more vulnerable to being cut-out should have its edge weight set to a smaller magnitude than the other edges. Although the lengths between nodes differ in the reduced graphs of mazes (due to the presence of supernodes), we consider all edges to be equally vulnerable and so we set all of the edge weights to 1 before calculating the min-cut-size. Following the min-cut algorithm, we find that the reduced graph in Fig. 3b has a min-cut-size of $\kappa = 2$. As the number of cycles increases (the number of solutions in the maze increases), κ and the resilience of the graph both increase because it is harder to completely cut off the flow between S and T. Unlike the previous two tortuosity parameters, κ can be calculated from either the original or reduced graph and give the same answer.

3.5. Relating graph parameters to transport properties

We now relate the three graph parameters we derived in the three previous subsections to the transport properties of the 2D, square-cell maze from which the graph was born (Fig. 1). We aim to describe a generalized steady flow of elements from the source to the sink of the maze in the framework of laminar, pressure-driven flow to provide a familiar context. We will exactly relate the effective tortuous resistance to the *permeability* of the maze. Then, we will use the average tortuosity to estimate the *residence time* in the geometry. Finally, we will use the min-cut size to describe the *maze robustness* to channel blockage. The main reason that these graph parameters are extensible to describing transport processes in 2D channel segments (not just 1D edges) is because geometrical and transport aspects of flow through channel segments can be easily incorporated into the edge weights of the graph.

First, we show how the effective tortuous resistance, $\tau_{R,\text{eff}}$, can be used to calculate the permeability of a maze. Similar to Hagen-Poiseuille's equation for volumetric flow rate through a channel with a circular cross-section, the volumetric flow rate (per unit depth) through the maze's i th channel segment with a rectangular cross-section can be expressed as (Cornish, 1928):

$$\frac{Q_i}{z} = \frac{w^3}{12\mu}\frac{\Delta p_i}{l_i}, \tag{6}$$

where Q_i is the volumetric flow rate in m^3/s, w is the width of the channel segment, $z \gg w$ is the depth into the channel segment, l_i is the length of the channel segment (note that $l_i = w$ for the solid line edges of the graph in Fig. 3b), μ is the dynamic viscosity of the fluid, and Δp_i is the pressure drop across the length of the channel segment. We rearrange Equation (6) into a form of the driving force (the pressure drop) over a resistance, and isolate the local resistance, R_i:

$$\frac{Q_i}{z} = \frac{Driving\ Force}{Resistance} = \frac{\Delta p_i}{\left(\frac{12\mu}{w^3}\right) l_i} \tag{7}$$

$$R_i = \frac{12\mu}{w^3} l_i \tag{8}$$

Since the constants multiplying l_i in Equation (8) can be factored out when using Equations (1) and (2), we calculate the effective length resistance $l_{R,\text{eff}}$ as described in Section 3.2 and express the effective hydraulic resistance in this case as:

$$R_{\text{eff}} = \frac{12\mu}{w^3} l_{R,\text{eff}} \tag{9}$$

The factoring and isolation of the length variable in the local resistance term is not done for the calculation of the effective resistance of systems such as microfluidic networks (Oh et al., 2012; Ajdari, 2004), likely due to the fact that channel segment widths are non-homogeneous in almost all real systems. See Appendix A1 for how R_{eff} can be expressed as a function of the effective length resistance even for a maze with non-homogeneous channel segment widths (e.g., for the composite maze discussed in Section 5). For the rest of this derivation, we maintain the assumption that the channel segment widths are homogeneous. Upon obtaining the effective resistance R_{eff} from Equation (9), the total flow rate, Q_{total}, through the maze can be expressed as:

$$\frac{Q_{\text{total}}}{z} = \frac{\Delta P}{R_{\text{eff}}} = \frac{\Delta P}{\left(\frac{12\mu}{w^3}\right) l_{R,\text{eff}}}, \tag{10}$$

where ΔP is the overall pressure drop across the maze.

Equation (10) allows us to use the effective length resistance derived from the graph to create a constitutive relationship between a pressure driving force, ΔP, and total flow rate of fluid elements through a maze. The cell width in the original (square) grid is w, and the viscosity, μ, is of the fluid we choose to model. To relate the total flow rate to the classical permeability, Darcy's law is used. For a porous domain with a linear pressure drop, Darcy's law is:

$$v_s = \frac{k}{\mu}\frac{\Delta P}{L}, \tag{11}$$

where v_s is the superficial velocity (i.e., volume flux) through the outlet, k is the permeability, and L is the overall length of the porous domain — the distance between S and T. Let W be the superficial width of the porous domain (i.e., overall width of the entire maze) so that the superficial cross-sectional area is $A_s = Wz$. Equation (10) can now be rewritten as:

$$v_s = \frac{Q_{\text{total}}}{A_s} = \frac{w^3}{12\mu W}\frac{\Delta P}{l_{R,\text{eff}}}. \tag{12}$$

By equating Equations (11) and (12) and then using the definition for $\tau_{R,\text{eff}}$ in Equation (3), we finally get an exact relationship between the first graph parameter and the first transport property of permeability:

$$k = \frac{w^3}{12 W \tau_{R,\text{eff}}}. \tag{13}$$

Now, we use the second graph parameter of average tortuosity τ_{avg}, derived in Equation (4), to obtain an approximate expression for the transport property of average residence time $t_{\text{res,avg}}$ for a physical element (e.g., a particle on a streamline) traveling through a maze. Unlike the previous graph parameter $\tau_{R,\text{eff}}$, we can use l_{avg} and τ_{avg} directly here without first needing to find an expression like Equation (8) and factoring out constants. We derive the expression for $t_{\text{res,avg}}$ for an idealized maze with a reduced graph that is composed only of paths between S and T (i.e., solutions) that do not overlap at any node or edge. The steady-state, cross-sectionally-averaged velocity through the j th solution of length $l_{j,\text{sol}}$ and cross-sectional area $A = wz$ can be written as $v_{j,\text{sol}} = \frac{Q_j}{A} = \frac{w^3 \Delta P}{12\mu l_{j,\text{sol}}}/w = \frac{w^2 \Delta P}{12\mu l_{j,\text{sol}}}$. Then, the residence time within this solution is $t_{j,\text{res}} = \frac{l_{j,\text{sol}}}{v_{j,\text{sol}}} = \frac{12\mu\left(l_{j,\text{sol}}\right)^2}{w^2 \Delta P}$. We assume here that the cells of this idealized maze are square and uniform such that w is the same for all solutions of the maze. If we make a simplification that

assumes that the average of $l_{j,\mathrm{sol}}^2$ over all j is the same as the square of the average value of $l_{j,\mathrm{sol}}$ over all j (i.e., it's the same as the square of $L\tau_{\mathrm{avg}}$), then we can write the average residence time as

$$t_{\mathrm{res,avg}} = \frac{12\mu L^2 \tau_{\mathrm{avg}}^2}{w^2 \Delta P}. \tag{14}$$

For any non-idealized maze that contains paths that branch into or out of each other, the solutions of the maze will overlap. In this case, or in the case of non-homogeneity in channel segment widths, the equality in Equation (14) will not hold; however, to a first-order approximation, the average residence time will still be proportional to the square of the average tortuosity parameter. See Appendix A2 for more discussion on the effects of branching and non-homogeneous channels segment widths on the relationship between average residence time and average tortuosity.

For the last graph parameter, the min-cut-size κ, we make an analogy to the robustness of the maze. The min-cut-size quantifies the number of edges that can be removed in a graph before the supersource and supersink nodes become disconnected and no paths exist between them. Analogously, it quantifies the number of channel segments that can be blocked in a maze before it becomes unsolvable. We qualify a robust maze as one that is more resistant to blockage and possesses a high min-cut-size value. Quantifying the robustness of a maze, however, heavily depends on the transport process in question; so, we can only say that robustness is proportional to the min-cut-size κ. See Appendix A3 for a discussion of the effects of non-homogeneous channel segment widths on the relationship between κ and the robustness of a maze.

The example of fluid flow through porous media demonstrated in this subsection can easily be extended to other forms of flow through transport media. For any transport process that can be expressed as driving force over resistance like in Equation (7), an appropriate flux-gradient transport law can be used to obtain an expression for the corresponding effective transport/material property that describes throughput (Table 1). The analysis involving graph theory and the effective tortuous resistance here provides a balance between accurate but costly numerical simulations which completely discretize the geometry, and simple but low-fidelity equations like Kozeny-Carman's equation (Kozeny, 1927; Carman, 1956) which do not discretize the geometry at all and cannot directly account for inhomogeneity in the geometry. For the second and third graph parameters, they have only been approximately related to their corresponding transport properties in the fluid flow case here but are believed to be easily generalizable to approximating analogous transport/system properties of other types of transport processes.

4. Properties of maze algorithms

This section explores two "knobs" that can be turned to achieve desired values of the three graph parameters $\tau_{R,\mathrm{eff}}$, τ_{avg}, and κ, and hence the transport properties of a maze geometry. In Section 4.1, we show that a coarse tuning knob could simply be one's choice of which maze algorithm to use, since maze algorithms create mazes in vastly different parts of the graph parameter space. In Section 4.2, we show how we can modify one algorithm, specifically the Sidewinder algorithm, to create a fine tuning knob for the maze's graph parameters.

Table 1
Effective transport equations for three analogous types of flow.

	Flow in Single Channel Segment i	Flux-Gradient Transport Law (for a linear domain)	Effective Transport Property Describing Throughput
Laminar Flow	$Q_i = \frac{w^3 z}{12\mu}\frac{\Delta p_i}{l_i}$	$v_s = \frac{k}{\mu}\frac{\Delta P}{L}$	$k = \frac{w^3}{12W\tau_{R,\mathrm{eff}}}$
Mass Flow	$\dot{n}_i = (wz)D_{ab}\frac{\Delta c_i}{l_i}$	$N_a = D_{\mathrm{eff}}\frac{\Delta c}{L}$	$D_{\mathrm{eff}} = \frac{D_{ab}w}{W\tau_{R,\mathrm{eff}}}$
Heat Flow	$\dot{Q}_i = (wz)k\frac{\Delta T_i}{l_i}$	$q'' = k_{\mathrm{eff}}\frac{\Delta T}{L}$	$k_{\mathrm{eff}} = \frac{kw}{W\tau_{R,\mathrm{eff}}}$

4.1. Choosing a maze algorithm

To eventually use maze geometries for transport operations, we first explore how the graph parameters $\tau_{R,\mathrm{eff}}$, τ_{avg}, and κ change when: 1) different maze algorithms are used to define the geometry, and 2) the flow direction is changed by varying which nodes in the graph are connected to the supersource (S) and supersink (T) nodes. The four maze algorithms used are Sidewinder, Prim's with the starting cell as the center cell, Recursive Backtracker with the starting cell as the center cell, and Aldous-Broder with the starting cell randomly chosen. We consider three flow directions from $S \rightarrow T$: top-down, left–right, and radially-outward. The diversity in flow directions allows the analyses of mazes here to be applicable to a wide range of engineered materials or systems, while the diversity in maze algorithms is what will provide the coarse tunability of graph parameters. For top-down and left–right flow, the distance from S to T is the overall length L and the overall width $W = L$ for the square mazes considered here, respectively. For radially-outward flow, S is linked to only the graph's middle node, while T is linked to all nodes on the graph's outer perimeter. The distance between the supernodes, L, needed for the calculation of $\tau_{R,\mathrm{eff}}$ is not trivial to obtain. For radially-outward flow, we chose to let L be the average of the shortest and longest distances between S and T; specifically, we used $L = (r + \sqrt{2}r)/2$, where r is half the overall width or length of the maze. For each of the twelve cases of maze algorithm and flow direction combinations, 500 25×25 mazes were generated to obtain average values of the three graph parameters.

Before comparing the graph parameters of the mazes across the 12 cases, we first consider just the case of Aldous-Broder mazes (mazes with channels that don't have any directional or length bias) under top-down flow. A significant spread in the two tortuosity parameters for this case is observed in Fig. 5a, which shows the $\tau_{R,\mathrm{eff}}$ vs. τ_{avg} *tortuosity parameter space* of the 500 mazes. In the four extreme corners of the tortuosity parameter space, the channels of mazes take on distinct characteristics (Fig. 5b); these channel characteristics can be easily seen in the graphs in Fig. 5c-5f. In Fig. B.1 in Appendix B, the tortuosity parameter space is shown for the other 11 cases. The simplicity of mazes and the graphs associated with them means we can run many thousands of seed-numbers in a short amount of time and quickly determine the part of the tortuosity parameter space that is associated with a certain maze algorithm and flow direction. In most engineering systems the flow direction is predetermined, so one can see that the choice of maze algorithm entirely dictates what graphs are possible and thus can be used as a coarse tuning knob for obtaining desired values of the first two graph parameters and transport properties. The reason that the min-cut-size, κ, is excluded from the parameter spaces shown is because all graphs considered in this subsection contain no interior cycles and thus will have a low integer value of κ in the range of 1 through 4 (note that for almost all 12 cases considered here, this upper limit is not related to the fact that the maximum degree of a node is equal to 4). Automating the introduction of interior cycles to increase κ and understanding how they affect the other two graph parameters is the subject of future work.

The average and standard deviation of the three graph parameters for each of the twelve cases is shown in Table 2. A representative graph for each case is shown in Fig. 6. The variation in the average values of the parameters shows that the choice of maze algorithm and flow direction changes the average value of the graph parameters significantly. For Prim's mazes in particular,

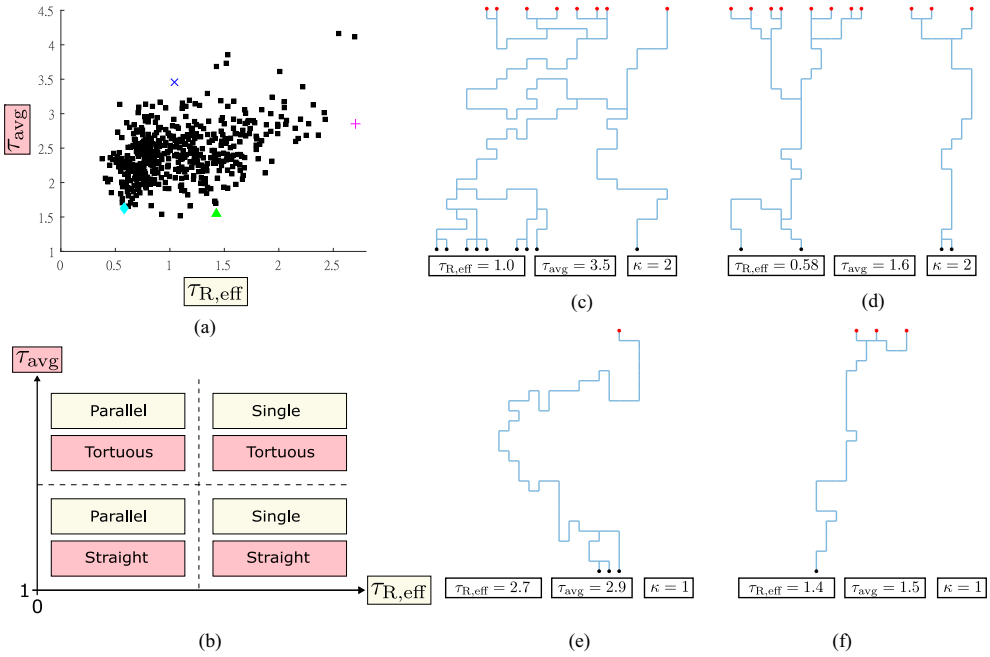


Fig. 5. (1.5 column; online color only): (a) Tortuosity parameter space for 500 Aldous-Broder mazes with top-down flow. (b) Characteristics of the channels of mazes situated in the extreme corners of the tortuosity parameter space. (c) Reduced graph of the maze corresponding with the blue cross in Fig. 5a. (d) Reduced graph corresponding with the cyan diamond. (e) Reduced graph corresponding with the magenta plus sign. (f) Reduced graph corresponding with the green triangle.

Table 2
The average value (and standard deviation in parentheses) of $\tau_{R,eff}$, τ_{avg}, and κ obtained from 500 different mazes/graphs for each of the 12 cases. The three graph parameters describe a graph's normalized effective length resistance, normalized average path length, and robustness to blockage, respectively.

	Sidewinder			**Prim's**			**Recursive Backtracker**			**Aldous-Broder**		
	$\tau_{R,eff}$	τ_{avg}	κ	$\tau_{R,eff}$	τ_{avg}	κ	$\tau_{R,eff}$	τ_{avg}	κ	$\tau_{R,eff}$	τ_{avg}	κ
Top-Down	0.53 (0.14)	1.71 (0.1)	2.5 (0.6)	0.47 (0.10)	1.6 (0.069)	1.6 (0.6)	2.2 (1.4)	3.6 (1.1)	1.6 (0.6)	1.1 (0.4)	2.4 (0.4)	1.5 (0.6)
Left-Right	0.85 (0.09)	1.71 (0.4)	1.04 (0.2)	0.46 (0.10)	1.6 (0.065)	1.6 (0.6)	2.2 (1.3)	3.5 (1.1)	1.6 (0.6)	1.0 (0.4)	2.4 (0.4)	1.6 (0.6)
Radially-Outward	0.93 (0.37)	2.0 (0.6)	1.2 (0.4)	0.16 (0.03)	1.3 (0.054)	2.5 (0.8)	3.9 (2.0)	3.9 (2.0)	1 (0)	0.8 (0.4)	2.5 (0.6)	1.2 (0.4)

when the direction of the maze bias (equivalently channel bias) matches the direction of the flow (i.e., both radially-outward), mazes are more robust, resulting in a larger min-cut-size κ, and they have straighter channels and lower resistances, resulting in smaller values in both tortuosities. Similar, though less prominent behavior is seen for the Sidewinder algorithm. In Prim's algorithm, more active-flow channel segments are generated for the radially-outward case, evident by there being a greater number of edges being retained after the reduction of the graph (see Fig. 6). When more channel segments are retained, parallel maze channels are more likely to be present in the reduced graph, thus lowering $\tau_{R,eff}$. Since these retained channels also tend to point in the same direction as the flow direction, τ_{avg} is also lowered. The standard deviation in the tortuosity graph parameters is also smaller when channel bias and flow direction are aligned (both radially-outward), meaning that these graphs with small values in their tortuosities are being consistently generated. As for the third graph parameter κ, its average value is also the largest during this alignment of channel bias and flow direction for the same reason of there being more active channel segments in the reduced graph. It is worth noting that the standard deviation of κ is not the smallest during alignment due to the fact that κ can only take on a small range of values. Also, for the case of radially-outward Prim's in particular, the min-cut-size is capped at a value of 4 because there is only one inlet node.

The Sidewinder algorithm has a top-down channel bias. We observed for Prim's mazes that alignment of the directional channel bias with the flow direction leads to the two tortuosity parameters being the smallest in their average values and having the least standard deviation when compared with the other flow direction cases. Two discrepancies occur for Sidewinder mazes that conflict with this observation about alignment: 1) the average value of τ_{avg} is the lowest for left–right instead of top-down flow by a small amount, and 2) the standard deviation in $\tau_{R,eff}$ is the lowest for left–right instead of top-down flow. The reason behind both discrepancies is that a straight, fully connected, top row of nodes will always be present in the unreduced graph of Sidewinder mazes (see Section 2 for why); when the flow direction is left–right, the top row of nodes will be retained after the graph reduction and so we observe this artifact for all of the 500 seed numbers. Thus, the first discrepancy occurs because the extremely short path between S and T skews the calculation and makes τ_{avg} much smaller than it should be for left–right flow, while the second discrepancy occurs because of the guaranteed appearance of the connected top row of nodes in the reduced graph no matter the seed number. Moreover, this artifact augments the value of $\tau_{R,eff}$ to be much smaller than it would be — without this artifact, the flow direction being perpendicular to the channel bias can lead to very few and possibly no edges being retained after the graph reduction. Rectifying this artifact by modifying the Sidewinder algorithm, or choosing a different algorithm, is the subject of future work. Similar to Prim's mazes, the average value for κ is highest for Sidewinder in the top-down flow direction because mazes with multiple paths in parallel with each other tend to be generated more easily due to the alignment of flow and channel biases. For Sidewinder

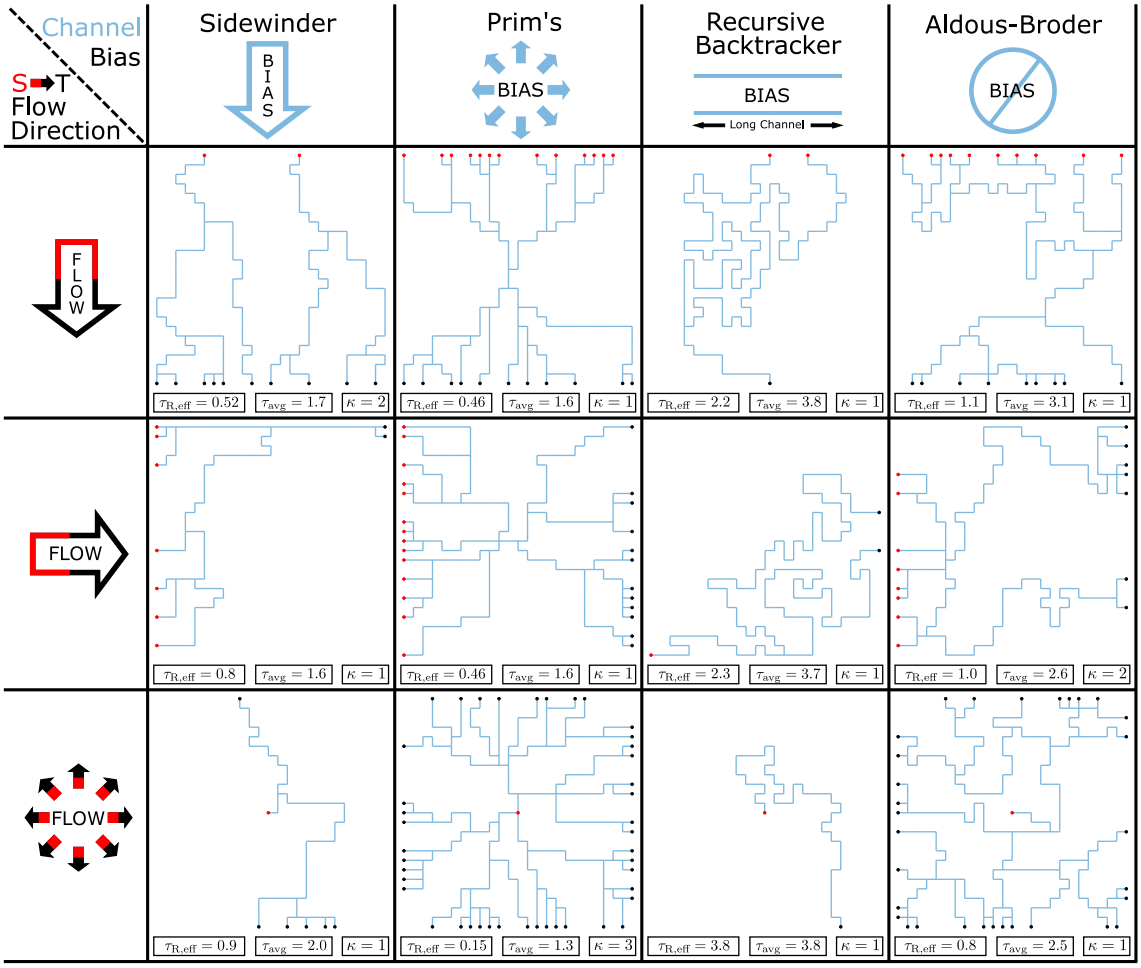


Fig. 6. (1.5 column; online color only): For each of the 12 cases in Table 2, one representative graph was chosen such that its values of $\tau_{R,\text{eff}}$ and τ_{avg} were both within ± 10% of their respective averages, and its value of κ was within a value of ±1 to its average. The set of source nodes N_S connected to S are colored red, and the set of sink nodes N_T connected to T are colored black; however, S and T (and the dashed-line edges connected to them) are omitted from the graphs for clarity.

top-down mazes in particular, κ is tied to the number of inlet nodes; although none of the mazes considered in this subsection had a value of κ larger than 4, just generating a larger maze with more cells (or by modifying the Sidewinder algorithm, which is what is done in the next subsection) can lead to a value of κ larger than 4.

In contrast to Prim's and Sidewinder, the Recursive Backtracker algorithm generates mazes with a non-directional, long-channel bias. These mazes have very tortuous channels, but few are in parallel with each other and so there are limited opportunities to increase the throughput between S and T. Due to this preference for long, tortuous paths, the average values of $\tau_{R,\text{eff}}$ and τ_{avg} are the largest out of all four algorithms (see Table 2). Although the Recursive Backtracker has no directional bias, $\tau_{R,\text{eff}}$ is larger by nearly a factor of 2 in the case of radially-outward flow. The reason is that having only one node connected to S restricts the number of paths connecting S and T to be identically one. For the radially-outward case, having a single path between S and T makes the two tortuosity values equal, while the random-walk and memory aspects of the algorithm make the standard deviations of the tortuosity values very large relative to the averages. The min-cut-size κ is low for all flow-directions. For the radially-outward case, the average value of κ is exactly one and the standard deviation is exactly zero because single-solution mazes are generated for every single seed number.

The Aldous-Broder algorithm generates mazes with no directional or long-channel bias. Thus, this algorithm has the highest level of randomness. From Table 2, one can see that the magnitude of the values of $\tau_{R,\text{eff}}$ and τ_{avg} both fall in-between those for the other maze algorithms and the relative standard deviations in the tortuosities are generally larger compared to the Sidewinder or Prim's mazes due to the random walk nature of the Aldous-Broder algorithm. Although Aldous-Broder has no directional bias, the values of $\tau_{R,\text{eff}}$ and τ_{avg} are slightly lower in the case of radially-outward flow; the reason is most likely because having four boundaries instead of one boundary as the outlet in the maze allows for additional channels to be retained after the reduction of the graph. The min-cut-size is low for all three flow-directions; however, it is slightly lower for radially-outward flow for the same reason the two tortuosity values are slightly smaller.

Equation (13) explicitly relates $\tau_{R,\text{eff}}$ to permeability, while Equation (14) approximately relates τ_{avg} to the average residence time of an element flowing through the maze. There are many situations where the path length and the flow resistance are pitted against one another and where a high throughput (low $\tau_{R,\text{eff}}$) and long average residence time (large τ_{avg}) are desired. Membrane separation, chromatographic separation, and leaching are examples where the system properties of particle retention and flow throughput are both increased if one uses a geometry with channel properties described by the upper-left quadrant of Fig. 5b (many parallel, tortuous channels) rather than the lower-right quadrant (a single, straight channel) due to more interaction of particles with the channel walls. Picking a geometry that optimizes both system properties is required and using mazes may provide some insight on how to accomplish that. Here, we examine the ratio of

τ_{avg} to $\tau_{R,eff}$ of the mazes, where a larger ratio corresponds to a maze with more parallel and tortuous channels; note that the two tortuosity values are not entirely uncorrelated, since $\tau_{R,eff}$ contains information about the length of the paths in addition to the number of parallel paths. Table 3 shows a comparison of the ratio of τ_{avg} to $\tau_{R,eff}$ for each of the 12 graphs in Fig. 6.

In Table 3, we see that the Prim's mazes provide the largest spatial sampling (largest τ_{avg}) for the least resistance (smallest $\tau_{R,eff}$), followed by Sidewinder, Aldous-Broder, and Recursive Backtracker. Note that the data in Table 3 is calculated from the graph parameters of the representative graphs in Fig. 6, so the seed number can also have a large effect on the ratio of τ_{avg} to $\tau_{R,eff}$. Not surprisingly, Recursive Backtracker has the lowest tortuosity ratio because it is biased toward long channels and very few paths (exactly one if it is the radially-outward case) that span from *S* to *T*. Clearly both tortuosities are related to one another, but the exact relationship depends on the maze algorithm, flow direction, and seed number. Still, the variation between algorithms provides a coarse but viable way to determine which type of geometry might be most suitable for a given application.

4.2. Fine tuning a maze algorithm

Since mazes involve a component of randomness in choosing which walls to remove, it is possible to bias the random number generator to fine-tune the details of the maze generated. For example, the classic Sidewinder algorithm uses equal weighting when deciding to carve out walls in the north direction vs. the east/west direction. We altered this carving-bias parameter to be between 0 and 1, such that a bias closer to zero will prefer to carve north, while a bias closer to 1 will prefer to carve east or west. Fig. 7a and 7b show the results of these changes for $\tau_{R,eff}$ and τ_{avg}. $\tau_{R,eff}$ approaches a value near 0 as the bias approaches zero, whereas τ_{avg} approaches 1 under the same conditions. Fig. 7c shows how we use the carving-bias to manipulate the ratio of $\tau_{R,eff}$ to τ_{avg} to vary from 1 to 25. As shown in Fig. 7d, at low values of the carving bias a Gaussian relationship exists between $\tau_{R,eff}$ and τ_{avg}, while at high bias the relationship is linear, like what would occur in Recursive Backtracker mazes. This observation is further confirmed by Fig. 7e, which shows that κ can take on a value of 1 or 2 for a carving bias larger than 0.75, or it can take on a value larger than 2 for a carving bias of less than 0.75. Fig. 7f shows representative maze solutions at the two extremes of carving bias. Using this carving-bias parameter, a Sidewinder maze can be fine-tuned to have a specific average tortuosity and effective tortuous resistance. The exact maze will vary with the seed number, which may also be used as a fine-tuning knob to produce the desired values of tortuosities. The modification of other maze algorithms is also viable but not explored in this paper.

5. Composite mazes as membrane surrogates

Throughout this paper, we have asserted that mazes can abstractly represent any form of a two-phase transport medium and can be used as a design tool for guidance in engineered systems. Before we can use mazes for this purpose, we first show they can be crudely and easily designed to approximate system-specific features and the transport property of a real material. To demonstrate this, we chose a system we have investigated in the past, which is a commercial, microporous, polyethersulfone (PES) membrane (Sorci et al., 2020). The structure of this membrane is shown in Fig. 8a, and it is a two-dimensional slice through the membrane obtained using a focused ion beam within a scanning electron microscope (SEM), then converted into a CAD image by creating a trace of said SEM image. The geometry was then imported into finite element software to compute the flow through the slice. The structure of the real membrane indicates that there are two skin layers, regions of smaller pores and lower porosity, at the top and bottom. Thus, a single maze composed of homogeneous-width channels would not be able to represent this system-specific feature of the actual structure. Producing a maze with inhomogeneous channel segment widths required the construction of a composite maze formed by vertically stacking three mazes with varying cell size. Fig. 8b shows the composite maze structure we generated. Fig. 8c shows the reduced graph of this composite maze.

When generating the three component mazes of the composite maze, the primary parameters considered were: 1) the maze algorithm, 2) the number of rows and columns, and 3) the cell size, which determines the width of the channel segments. For each component maze, a Sidewinder algorithm was selected to match the maze bias to the vertical flow bias for dead-end filtration. The composite maze's overall dimensions were adjusted to match the real membrane's domain width of 100 μm and thickness of 130 μm. The number of rows and columns as well as the cell size for each component maze were also adjusted to fit into the overall width and height desired in each component maze. The overall heights of the top and bottom component mazes were set to roughly match the thickness of the two skin layers of the real membrane; information about this system-specific feature was obtained qualitatively from the real membrane domain image in Fig. 8a and quantified using the simulated pressure drop data shown in Fig. 9. A change in the slope is observable in Fig. 9 at a height of ≈15 and ≈115 μm, creating an S-shaped curve. Additionally, the component maze cell sizes were chosen to approximate the total volumetric flowrate through the membrane at the given pressure drop — this is equivalent to matching the permeability — and to match the nominal, reported pore size of the commercial membrane. Within the framework we have developed, the permeability of a candidate composite maze was calculated using the effective tortuous resistance parameter. The methodology for calculating the effective tortuous resistance, as well as the other two graph parameters, are discussed in Appendix C1. A total of three trials for getting qualitative and quantitative match-up were performed before obtaining the composite maze in Fig. 8b. Its corresponding graph and its three graph parameters are shown in Fig. 8c. The number of rows and columns, cell sizes used, and more details about each of the three component mazes can be found in Appendix C2.

Fig. 10a shows the flow field through the real membrane structure with a pressure of 13,750 kPa applied at the top and a pressure of 0 kPa applied at the bottom. What is interesting is the channeling (i.e., funneling of the flow through a single pore) that appears near the bottom, which can limit the performance of the membrane. This geometry only represents one two-dimensional slice though the membrane. The real membrane has a three-dimensional pore structure, so channeling, like that observed in

Table 3
Ratio of τ_{avg} to $\tau_{R,eff}$ for the graphs in Fig. 6.

Flow Direction	Sidewinder	Prim's	Recursive Backtracker	Aldous Broder
Top-Down	3.2	4.0	1.6	2.4
Left-Right	1.8	4.1	1.6	2.4
Radially-Outward	1.9	7.8	1.0	2.3

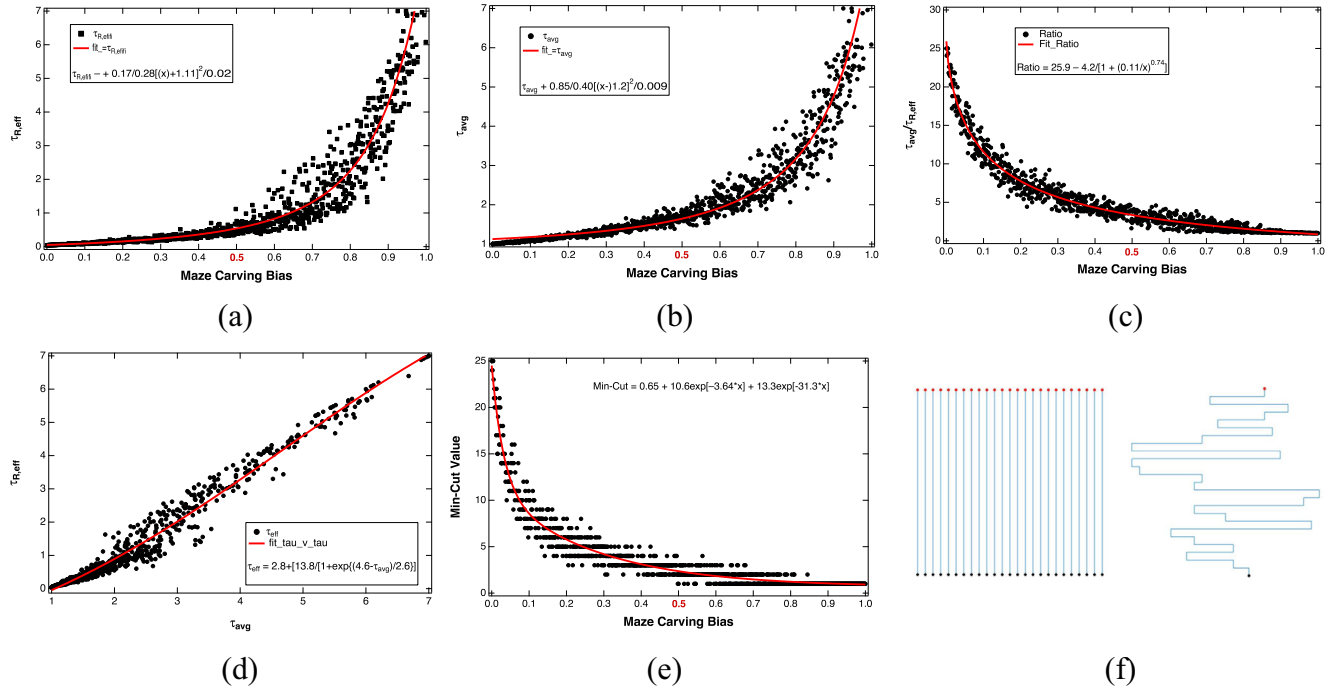


Fig. 7. (1.5 column; online color only): (a) Effect of carving bias on effective tortuous resistance. (b). Effect of carving bias on normalized average path length. (c) Ratio of path length to length resistance as a function of bias. (d) Relationship between $\tau_{R,\text{eff}}$ and τ_{avg} in two regimes of the carving bias. Circles behave as a traditional Sidewinder maze would, while the squares behave more like a Recursive Backtracker maze. (e) Robustness of the transport through the maze measured by the min-cut-size. (f) Example mazes at the extremes of bias (0 left; 1 right).

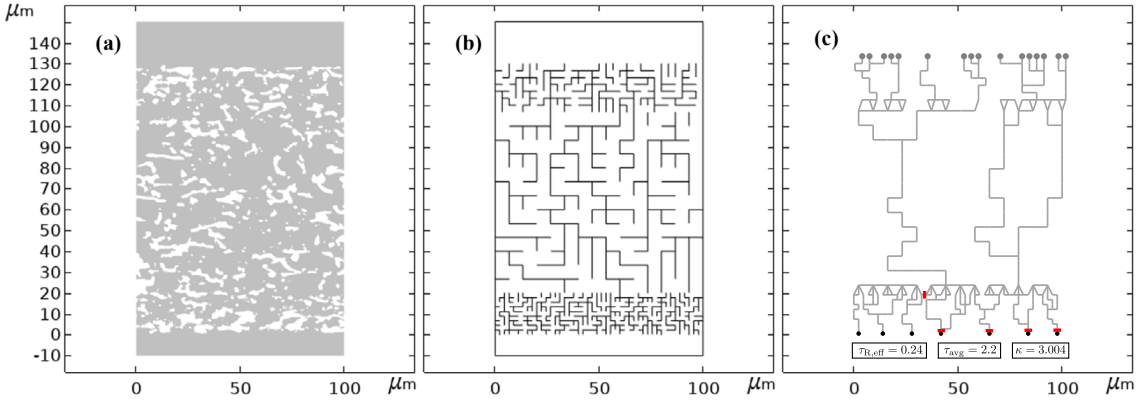


Fig. 8. (double column; online color only): (a) Commercial PES membrane, with grey as the fluid phase. (b) Simulation domain of composite maze, with white as the fluid phase. (c) Reduced graph of composite maze, with grey edges as the fluid phase and red lines crossing the edges that belong in C_{opt}, which is the minimal $S-T$ cut that completely restricts flow in the graph; see Appendix C1 for why C_{opt} doesn't instead just contain two of the edges in the middle component maze.

Fig. 10a, would likely not be as severe. Fig. 10b shows the results for the composite maze. While not reproducing the exact features of the real, 3-D membrane, the overall features appear similar and include the presence of a finite number of primary channels and a moderate channeling effect. Fig. 10c shows the calculated pressure distribution within the real membrane slice, and Fig. 10d shows the pressure distribution within the composite maze structure. Slices of pressure are averaged every two microns throughout the length of the domain and plotted against the pressure drop for the PES membrane and composite maze structure in Fig. 11. Clearly 50% of the overall pressure drop occurs in the final 20% of the membrane. Here too, about 50% of the pressure drop occurs near or within the bottom component of the composite maze. The calculated permeability of the real membrane is 2.41×10^{-13} m^2. The permeability of the composite maze, calculated using the same finite element software, is 2.05×10^{-13} m^2. The permeability of the composite maze, calculated using Equation (13) (and the modification to the equation as discussed in Appendix C1), is 1.78×10^{-13} m^2. While not as precise as the finite element calculation, the graph approach is faster and can screen many more geometries to find common or hidden patterns.

Despite the minimal amount of optimization effort (three trials) used in generating the composite maze structure, the flow field and permeability of the composite maze are remarkably similar to the membrane. It is important to note that the goal of this section is not to perfectly mimic a transport medium using a composite maze — it is to show that mazes are flexible and can be used, creatively, to capture the unique geometrical and transport-related features of a system (here, these are the skin layers and non-linear pressure drop).

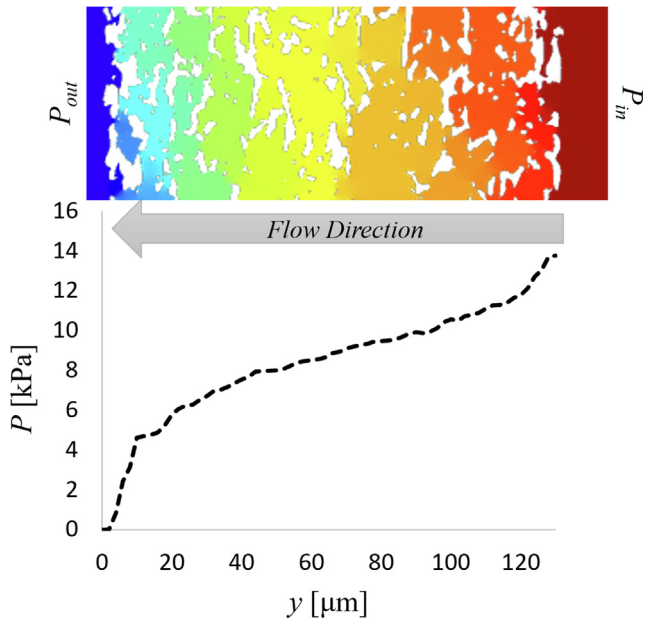


Fig. 9. (single column; online color only): Nonlinear pressure drop from the PES membrane simulation. The S-shaped pressure curve is due to the pore distribution forming a "skin layer" on the membrane where larger pressure drops are observed. The pressure data is averaged across the width of the domain.

In a membrane, selectivity may be as important or more important than permeability. There is no guarantee that the maze structure that mimics the permeability can also mimic the selectivity. Outside of pure size exclusion, an exact selectivity metric is difficult to derive. However, as discussed at the end of Section 4.1, selectivity/particle-retention should scale with average residence time (i.e., the average time a particle spends interacting with the channel walls); in Section 3.5, the average residence time was found to approximately scale with the square of the average tortuosity τ_{avg}. So, selectivity should approximately scale with the square of the average tortuosity. The ratio of τ_{avg} to $\tau_{R,eff}$, introduced at the end of Section 4.1, can provide a starting point to deriving a metric that accounts for both permeability and selectivity. Moreover, the min-cut-size κ can provide a metric for the robustness of the composite maze to fouling. The values of $\tau_{R,eff}$, τ_{avg}, and κ for the composite maze are 0.24, 2.23, and 3.004, respectively; see Appendix C1 for the calculation of these graph parameters (and why the min-cut-size is not an integer). The ratio of the tortuosities for the composite maze is 9.4, which is larger than that of the graphs of the single-component mazes in Fig. 6 (see Table 3). Finally, membranes are three dimensional structures and so cycles are likely the rule rather than the exception. Since the graph of the composite maze still has a small value of κ, more cycles need to be introduced to increase its robustness. Although incorporating additional cycles in mazes is not difficult to do, especially for 3-D mazes, the extraction of the three graph parameters of the maze can become very difficult. Appendix C1 discusses how

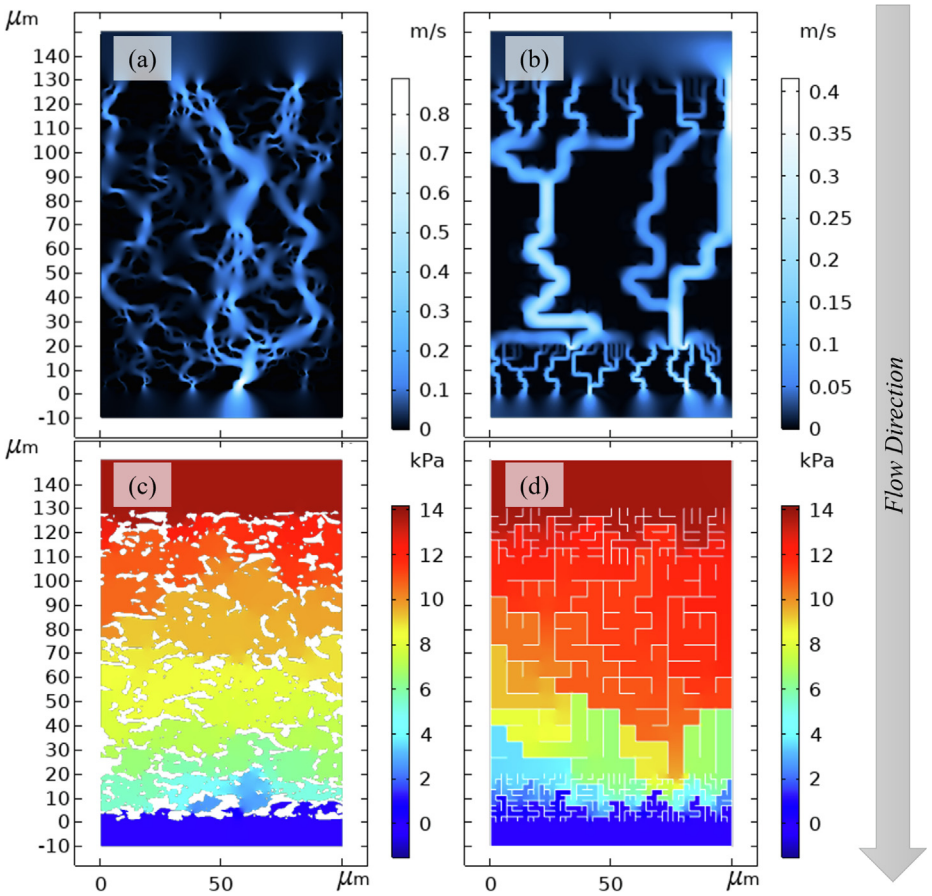


Fig. 10. (1.5 column; online color only): Pressure-driven Stokes flow modeled with a 13750 Pa transmembrane pressure. Velocity field (a,b) and pressure gradient (c,d). Data is presented for a commercial PES membrane (left) and a composite maze (right).

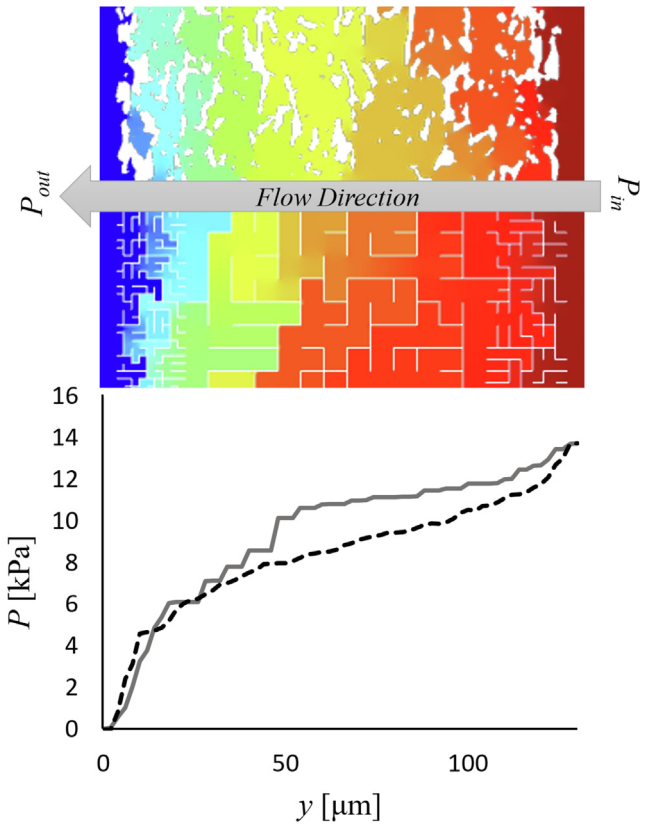


Fig. 11. (single column; online color only): Average pressure taken every y = 2 μm. The dashed line is data from the PES membrane simulation, while the solid line is from simulation data on the composite maze. The composite maze captures the nonlinear pressure drop through the membrane.

we have already generalized our analyses in Section 3 to apply to mazes with interior cycles, which is what the composite maze studied in this section was. For graphs of mazes that have even more (interior) cycles and/or nodes with a degree larger than 4, it is expected that $\tau_{R,\text{eff}}$ and κ will remain viable to compute while τ_{avg} will not due to the algorithmic complexity of finding all paths between two nodes in a highly cyclic graph.

6. Conclusions

Mazes can be more than just puzzles to be solved by children, animals, molds, or robots: they can be thought of as simple abstractions that resemble the internal structure of complex systems found in nature and in various engineering applications. Thus, mazes can be used as surrogate materials and studied to fundamentally understand how geometry affects transport processes and systems. With the new understanding, maze-inspired transport geometries/materials can be rationally-designed and optimized for a given engineering application and then realized with additive manufacturing. The structure of a maze can be easily translated and represented as a graph, where the tools of graph theory can be applied to quantify the features of the maze. We presented methods to extract three dimensionless graph parameters from the graph structure and related each of them to a transport property of the maze, providing a platform for guiding the design of maze-based materials from first principles. The first graph parameter is the effective tortuous resistance $\tau_{R,\text{eff}}$ of the graph, which can be used to explicitly relate the structure of the graph to the effective transport resistance of the maze and used to calculate effective transport properties such as permeability, effective thermal conductivity, or effective diffusivity. The second graph parameter is the average tortuosity τ_{avg}, and it is a normalized average path length that can be used to obtain an estimate of the average residence time within the maze geometry. The third graph parameter is the min-cut-size κ, which gives a measure to the resilience of the graph to edge removal and a measure to the robustness of the maze geometry to channel blockage.

There is a plethora of maze generation algorithms, each with their own inherent biases, and we examined only four common types. Each algorithm incorporates some degree of randomness, so the same algorithm can generate multiple mazes that have a spread in their graph parameters. However, it was found that different algorithms can generate mazes in very different regions of the graph parameter space. Consequently, the choice of a maze algorithm is in itself a way to coarsely tune the graph parameters of a maze. We also showed that the maze generation bias can be manually adjusted within one of the algorithms to fine-tune the values of the graph parameters and especially the ratio of the average tortuosity to the effective tortuous resistance. Finally, we showed how a surrogate, composite maze geometry — formed by stacking together three individual mazes — can accurately mimic the permeability and even the system-specific nonlinear pressure drop and flow patterns (caused by the skin layers) observed in a real membrane. The eventual goal is to leverage the flexibility and tunability of mazes to construct and optimize a surrogate maze geometry that can be used as a guide to design an engineered geometry supporting a specific transport process.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors would like to acknowledge the support of Rensselaer Polytechnic Institute, the National Science under grant CBET-1603318, and the National Aeronautics and Space Administration under grant 80NSSC18K0332 for their assistance in funding this work.

Appendix A: Relating graph parameters to transport properties for mazes with non-homogeneous channel segment widths (such as the composite maze studied in Section 5)

Appendix A1: Relating effective tortuous resistance to permeability

Let $\bar{w}$ be the average width of the channel segments in the reduced graph. Then, Equation (8) for the hydraulic resistance of edge i can be rewritten as $R_i = \frac{12\mu}{\bar{w}^3}\left(l_i \frac{\bar{w}^3}{w_i^3}\right)$. Factoring out the constants, the edge weights of the reduced graph can be set to $l_i \frac{\bar{w}^3}{w_i^3}$ (instead of l_i for the case of homogeneous channel segment widths) before proceeding to calculating the effective length resistance $l_{R,\text{eff}}$ as described in Section 3.2. Note that the edge weights still have a dimension of length, so the calculated $l_{R,\text{eff}}$ still does as well. Following the rest of the derivation in Section 3.5, the final permeability equation is $k = \frac{\bar{w}^3}{12W\tau_{R,\text{eff}}}$, and it is very similar to Equation (13).

Appendix A2: Relating average tortuosity to average residence time

In momentum transport, the flow rate in a channel segment changes if the pressure difference across the channel segment

changes; moreover, at a set flow rate, the cross-sectionally-averaged velocity (or simply velocity) in a channel segment changes if the channel segment width changes. For a parent channel bifurcating into two child channels, different pressure differences across the child channels as well as different channel widths can cause the velocities through them to be very different. So, complications arise when trying to obtain the average residence time of a fluid parcel (or inertia-less particle) traveling in a maze with branching channels even if channel segment widths are homogeneous. The average tortuosity τ_{avg} simply cannot provide enough information to be used in an explicit expression for the average residence time in a graph. However, there is still a procedure one can use to obtain the average residence time for a steady-state transport process in a maze with branching channels and/or inhomogenous channel segment widths:

1. Solve for the potential (i.e., pressure) at every node in the reduced graph using Kirchhoff's law and Ohm's law by setting up a system of flow-balance equations at each node. Calculate the pressure drop Δp_i across all the edges.
2. Use Equation (6) to get the velocity v_i through all the edges.
3. Get the residence time of all the edges using $t_{\text{res},i} = l_i / v_i$.
4. Use the procedure for getting l_{avg}, which is discussed in Section 3.3, but with edge weights being $t_{\text{res},i}$ instead of l_i in order to obtain the average residence time $t_{\text{res,avg}}$.

Depending on the transport process, even the above procedure might be insufficient. For example, particles could become trapped in the medium and never make their way out; in this case, the probability of a particle becoming stuck on the walls of channel segments could be incorporated into the edge weights and possibly lead to a modified way of calculating $t_{\text{res,avg}}$. This is the subject of future work.

Appendix A3: Relating min-cut-size to robustness

Consider the specific transport process of fluid flow in a membrane (which Section 5 considers) with particles. The membrane can foul if the solid particles clog up too many channel segments. Recall that the min-cut algorithm finds the "weakest links" in the graph, and so an edge that is more vulnerable to being cut-out should have a smaller edge weight than the other edges so that it will be more easily picked out by the algorithm; one can enforce this policy by setting the edge weights equal to $\frac{w_i}{w}$. One can see that wider channels are more resilient to being blocked than narrower channels by a particle and will thus have a larger contribution to the min-cut-size κ calculated in Equation (5). There is no quantitative basis for this edge-weighting-scheme (for example we could have set the edge weights to be $\left(\frac{w_i}{w}\right)^3$ or $\left(\frac{w_i}{w}\right)^2$ instead), but it is the one that we use to calculate the min-cut-size κ for the composite maze in Section 5. Note that κ, the sum of edges weights of the edges in the discovered optimal cut-set C_{opt}, will be dimensionless like the other two tortuosity graph parameters. The subject of future work is to use a physics-informed approach to modify the edge weights to calculate the min-cut-size more accurately and relate it to the robustness of a maze more quantitatively.

Appendix B: Additional information for Section 4

Fig. B.1 shows the tortuosity parameter spaces for the 12 cases considered in Section 4. A graph with a data point lying on the dashed red line of $\tau_{\text{avg}} = \tau_{R,\text{eff}}$ will contain just a single path between S and T and will have a value of $\kappa = 1$. By definition of the two tortuosities, no graphs can exist below this line.

Appendix C: Details related to the composite maze studied in Section 5

Appendix C1: Calculation of the three graph parameters of the composite maze

Here we provide details on how the three graph parameters $\tau_{R,\text{eff}}$, τ_{avg}, and κ were calculated for the composite maze. We start with $\tau_{R,\text{eff}}$. The first complication related to the composite maze is that the channel segment widths are non-homogeneous; however, as discussed in Appendix A1, one can just set the edge weights of the reduced graph to $l_i\left(\frac{\bar{w}}{w_i}\right)^3$. The average channel segment width was found to be $\bar{w} = 3.70$. The second complication of the compos-

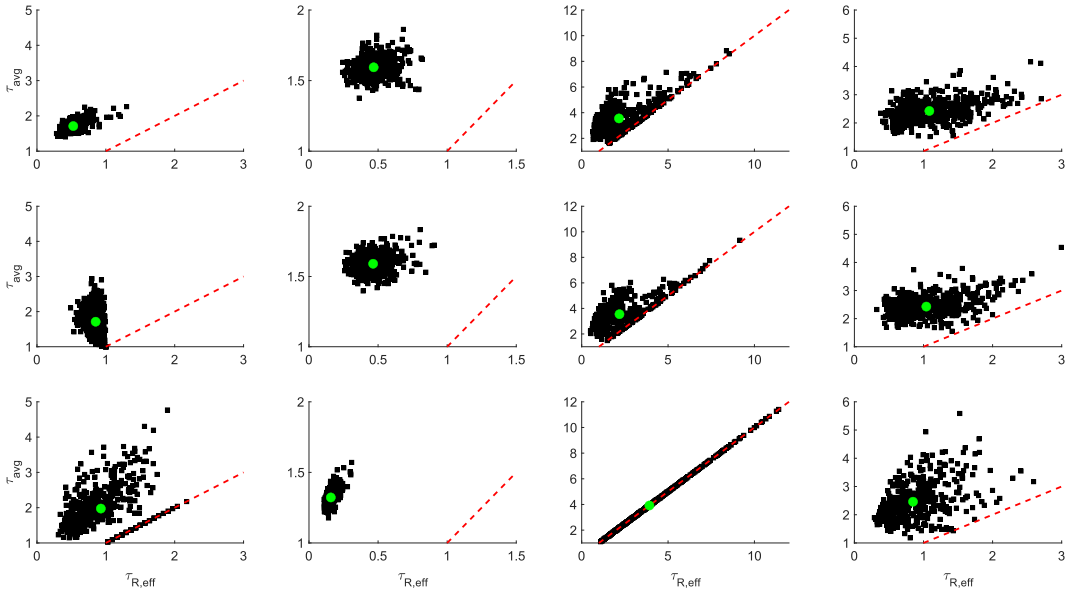


Fig. B1. (double column; online color only): Tortuosity parameter space for the 12 cases. Positions of subplots match those in Fig. 6, the dashed red line represents the $\tau_{\text{avg}} = \tau_{R,\text{eff}}$ curve, and the green circle represents the average of the data points.

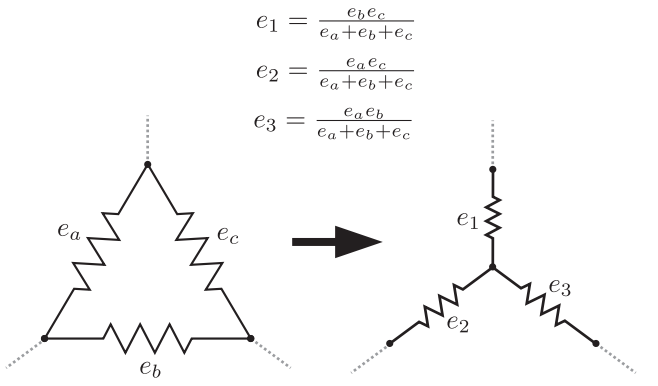


Fig. C1. (single column; online color only): Transformation of resistor network where the edge weights e_i represent length resistance.

ite maze is that the calculation of $l_{R,\text{eff}}$ (which uses the series and parallel rules for simplifying a reduced graph / resistor network into a single edge and two supernodes) can fail if the reduced graph has interior cycles, which we defined in Section 3.1 as cycles that don't intersect either of the supernodes (see Fig. 8c). Similar to the replacement of multiple edges in parallel or in series with an effective edge, the $Y-\Delta$ transform can be used to replace a "Y"-shaped resistor network with a "Δ"-shaped resistor network, or vice-versa (Kennelly, 1899). Fig. C.1 shows the transformation of a resistor network from a shape of "Δ" to "Y. " Note that if a reduced graph has interior cycles and contains many nodes with a degree of 4 or larger, it is likely that even the $Y-\Delta$ transform won't be sufficient for simplifying the resistor network; in this case, the pseudo-inverse of the Laplacian matrix of the graph (reduced or unreduced) can be used to calculate the effective length resistance $l_{R,\text{eff}}$ (Klein and Randić, 1993). (We don't use this method because all graphs we work with have nodes with degree less than or equal to 4.) The algorithm in Section 3.2 was modified to incorporate the use of the $Y-\Delta$ transform to break interior cycles during the simplification of the reduced graph's resistor network and thus enable the calculation of $l_{R,\text{eff}}$. The reduced graph in Fig. 8c was calculated to have a value of $\tau_{R,\text{eff}} = 0.24$.

Unlike the method used for the calculation of $l_{R,\text{eff}}$, the method discussed in Section 3.3 for calculating the average path length l_{avg} (involves finding all paths between S and T) is not complicated by the inhomogeneity in channel segment widths. Furthermore, the method is robust enough to work for any graph no matter how many interior or normal cycles it has or how large the degrees of the nodes are; however, since the composite graph contains interior cycles, the number of paths (and thus computation time for calculating the average path length) will be orders of magnitude higher than that compared to the graphs in Section 4 that don't have any interior cycles. To greatly reduce the number of allowable paths between S and T, the composite graph was first converted into a *directed graph*, which is a graph where the edges have directions; this was done by using the first two steps detailed in the procedure in Appendix A2 to find the velocity (and therefore flow direction) in each edge. The decision to use a directed graph is not only practical here, but sensible due to the context wherein solid/fluid particles can only follow streamlines in pressure-driven flow. The number of paths in the directed version of the composite graph in Fig. 8c was found to be around 11,000, and a value of $\tau_{\text{avg}} = 2.2$ was calculated. (As a side note, one can follow the other steps detailed in the procedure in Appendix A2 to find the average residence time instead of the averge path length for a general graph containing interior cycles, but this calculation was not done for the graph considered in this paper.)

The calculation of κ requires one to first set the edge weights in the reduced graph to $e_i = \frac{w_i}{\bar{w}}$ (as discussed in Appendix A3). For the composite maze, $\bar{w} = 3.70$. The min-cut solution is shown in Fig. 8c, where the 5 edges marked by red dashes through them are the edges that compose the optimal S–T cut, C_{opt}. Although the middle component maze has 2 edges that could also compose an S–T cut, the cut would not be optimal since these 2 edges each have a width that is three times larger than the width of any edge in the bottom component maze (see Table C.1 in Appendix C2). So, choosing 5 edges from the bottom component maze would yield a smaller min-cut-size than choosing 2 edges from the middle component maze. From Equation (5), the capacity of C_{opt} (i.e the min-cut-size κ) is 3.004. Together, the three graph parameters of the composite maze are more ideal than those of any of the single-component mazes in Fig. 6 of Section 4.

Appendix C2: Construction of the composite maze

Here we provide further details on how the composite maze was constructed. Table C.1 summarizes the information for each of the three mazes. For the edges connecting two nodes at the interface between two different component mazes, the channel segment width w_i of these edges were approximated by setting them equal to the average of the channel segment widths of the two adjacent mazes. Recall that the Sidewinder algorithm generates mazes with the top row of cells containing no vertical walls. After using the Sidewinder algorithm to generate each component maze, a modification was done. This modification involved adding a west wall to each cell along the top row of cells at a 50% probability, and the reason was to minimize the amount of horizontal flow in the row of cells (i.e. interface) between the mazes; in Fig. 8b, these added, vertical walls actually appear on the bottom row of cells in each component maze, since each component maze was further modified by flipping it vertically (this flipping was done to better capture the downwards funneling aspect of the flow field in the real membrane). Finally, the component mazes were stacked on top of each other and all the horizontal walls between

Table C1
Geometrical features of component mazes.

Structure	Rows	Columns	Cell size/channel segment width (μm)	Overall thickness (μm)
Top Maze	7	30	3.33	23.33
Middle Maze	13	15	6.67	86.67
Bottom Maze	9	45	2.22	20.0

them were removed to "stitch" them together and create the composite maze.

References

Buck, J., 2015. Mazes for Programmers: Code Your Own Twisty Little Passages. Pragmatic Bookshelf.

Biggs, N., Lloyd, E.K., Wilson, R.J., 1986. Graph theory, 1736–1936. Oxford University Press.

Chartrand, G., Zhang, P., 2012. A First Course in Graph Theory. Dover Publications Inc..

Matyka, M., Khalili, A., Koza, Z., 2008. Tortuosity-porosity relation in porous media flow. Phys. Rev. E 78, (2). https://doi.org/10.1103/PhysRevE.78.026306 026306.

Koponen, A., Kataja, M., Timonen, J., 1997. Permeability and effective porosity of porous media. Phys. Rev. E 56 (3), 3319–3325.

Andrade, J.S., Almeida, M.P., Mendes Filho, J., Havlin, S., Suki, B., Stanley, H.E., 1997. Fluid flow through porous media: the role of stagnant zones. Phys. Rev. Lett. 79 (20), 3901–3904. https://doi.org/10.1103/PhysRevLett.79.3901.

Wang, M., Wang, J., Pan, N., Chen, S., 2007. Mesoscopic predictions of the effective thermal conductivity for microscale random porous media. Phys. Rev. E 75, (3). https://doi.org/10.1103/PhysRevE.75.036702 036702.

Hyman, J.D., Smolarkiewicz, P.K., Winter, C.L., 2012. Heterogeneities of flow in stochastically generated porous media. Phys. Rev. E 86, (5). https://doi.org/10.1103/PhysRevE.86.056701 056701.

Martys, N., Garboczi, E.J., 1992. Length scales relating the fluid permeability and electrical conductivity in random two-dimensional model porous media. Phys. Rev. B 46 (10), 6080–6090. https://doi.org/10.1103/PhysRevB.46.6080.

Siena, M., Riva, M., Hyman, J.D., Winter, C.L., Guadagnini, A., 2014. Relationship between pore size and velocity probability distributions in stochastically generated porous media. Phys. Rev. E 89, (1). https://doi.org/10.1103/PhysRevE.89.013018 013018.

Zhou, L., Qu, Z.G., Ding, T., Miao, J.Y., 2016. Lattice Boltzmann simulation of the gas-solid adsorption process in reconstructed random porous media. Phys. Rev. E 93, (4). https://doi.org/10.1103/PhysRevE.93.043101 043101.

Araújo, A.D., Bastos, W.B., Andrade, J.S., Herrmann, H.J., 2006. Distribution of local fluxes in diluted porous media. Phys. Rev. E 74, (1). https://doi.org/10.1103/PhysRevE.74.010401 010401.

Tartakovsky, A.M., 2010. Langevin model for reactive transport in porous media. Phys. Rev. E 82, (2) 026302.

Ye, G., Sun, Y., Zhou, X., Zhu, K., Zhou, J., Coppens, M.-O., 2017. Method for generating pore networks in porous particles of arbitrary shape, and its application to catalytic hydrogenation of benzene. Chem. Eng. J. 329, 56–65. https://doi.org/10.1016/j.cej.2017.02.036.

Adler, P.M., 1992. Porous Media: Geometry and Transports, Butterworth/Heinemann.

Sahimi, M. Heterogeneous Materials, Volume I: Linear Transport and Optical Properties, Springer, New York, 2003.

Sahimi, M. Heterogeneous Materials, Volume II: Nonlinear and Breakdown Properties and Atomistic Modeling, Springer, New York, 2003.

Kesavan, H.K., Chandrashekar, M., 1972. Graph-theoretical models for pipe network analysis. J. Hydraulics Div. 98 (2), 345–364.

Marle, C., 1981. Multiphase flow in porous media. Éditions technip.

Bermond, J.-C., Delorme, C., Quisquater, J.-J., 1986. Strategies for interconnection networks: Some methods from graph theory. J. Parallel Distrib. Comput. 3 (4), 433–449. https://doi.org/10.1016/0743-7315(86)90008-0.

Kalamaras, G.S., Brino, L., Carrieri, G., Pline, C., Grasso, P., 2000. Application of multicriteria analysis to select the best highway alignment. Tunn. Undergr. Space Technol. 15 (4), 415–420. https://doi.org/10.1016/S0886-7798(01)00010-4.

Bramel, J., Simchi-Levi, D., 1998. The logic of logistics: theory, algorithms and applications for logistics management. J. Operat. Res. Soc. 49 (9), 1016–1017.

Barnes, J.A., Harary, F., 1983. Graph theory in network analysis. Social Networks 5 (2), 235–244.

Cormen, T. H. (Ed.)., 2009. Introduction to algorithms (3rd ed). MIT Press.

Fuerstman, M.J., Deschatelets, P., Kane, R., Schwartz, A., Kenis, P.J.A., Deutch, J.M., Whitesides, G.M., 2003. Solving mazes using microfluidic networks. Langmuir 19 (11), 4714–4722. https://doi.org/10.1021/la030054x.

Lovass, P., Branicki, M., Tóth, R., Braun, A., Suzuno, K., Ueyama, D., Lagzi, I., 2015. Maze solving using temperature-induced Marangoni flow. RSC Adv. 5 (60), 48563–48568. https://doi.org/10.1039/C5RA08207B.

Lagzi, I., Soh, S., Wesson, P.J., Browne, K.P., Grzybowski, B.A., 2010. Maze solving by chemotactic droplets. J. Am. Chem. Soc. 132 (4), 1198–1199. https://doi.org/10.1021/ja9076793.

Adamatzky, A., 2012. Slime mold solves maze in one pass, assisted by gradient of chemo-attractants. IEEE Trans. Nanobiosci. 11 (2), 131–134. https://doi.org/10.1109/TNB.2011.2181978.

Sarmiento-Reyes, A., Rodriguez-Velasquez, Y., 2018. Maze-solving with a memristive grid of charge-controlled memristors. In: 2018 IEEE 9th Latin American Symposium on Circuits & Systems (LASCAS), pp. 1–4. https://doi.org/10.1109/LASCAS.2018.8399973.

Najjar, W., Gaudiot, J.-L., 1990. Network resilience: a measure of network fault tolerance. IEEE Trans. Comput. 39 (2), 174–181.

Cocco, A.P., Nakajo, A., Chiu, W.K., 2017. Analytical transport network theory to guide the design of 3-D microstructural networks in energy materials: Part 1. Flow without reactions. J. Power Sources 372, 297–311.

Xu, Z.H., Ma, G.W., Li, S.C., 2014. A graph-theoretic pipe network method for water flow simulation in a porous medium: GPNM. Int. J. Heat Fluid Flow 45, 81–97. https://doi.org/10.1016/j.ijheatfluidflow.2013.11.003.

Kershenbaum, A., Van Slyke, R., 1972. Computing minimum spanning trees efficiently. Proc. ACM Ann. Conf. 1, 518–527.

Bejan, A., 1997. Constructal-theory network of conducting paths for cooling a heat generating volume. Int. J. Heat Mass Transf. 40 (4), 799–816. https://doi.org/10.1016/0017-9310(96)00175-5.

Schrijver, A., 2002. On the history of the transportation and maximum flow problems. Math. Program. 91 (3), 437–445. https://doi.org/10.1007/s101070100259.

Klein, D.J., Randić, M., 1993. Resistance distance. J. Math. Chem. 12 (1), 81–95. https://doi.org/10.1007/BF01164627.

Ellens, W., Spieksma, F.M., Van Mieghem, P., Jamakovic, A., Kooij, R.E., 2011. Effective graph resistance. Linear Algebra Appl. 435 (10), 2491–2506. https://doi.org/10.1016/j.laa.2011.02.024.

Clennell, M.B., 1997. Tortuosity: A guide through the maze. Geol. Soc. Lond. 122 (1), 299–344. https://doi.org/10.1144/GSL.SP.1997.122.01.18.

Nagamochi, H., Ibaraki, T., 1992. Computing edge-connectivity in multigraphs and capacitated graphs. SIAM J. Disc. Math. 5 (1), 54–66.

Cornish, R.J., 1928. Flow in a pipe of rectangular cross-section. Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, 120(786), 691-700. https://doi.org/10.1098/rspa.1928.0175.

Oh, K.W., Lee, K., Ahn, B., Furlani, E.P., 2012. Design of pressure-driven microfluidic networks using electric circuit analogy. Lab Chip 12 (3), 515–545. https://doi.org/10.1039/c2lc20799k.

Ajdari, A., 2004. Steady flows in networks of microfluidic channels: Building on the analogy with electrical circuits. C.R. Phys. 5 (5), 539–546. https://doi.org/10.1016/j.crhy.2004.02.012.

Kozeny, J., 1927. Uber kapillare leitung der wasser in boden. Royal Academy of Science, Vienna, Proc. Class I, 136, 271-306.

Carman, P.C., 1956. Flow of Gases Through Porous Media. Academic Press.

Sorci, M., Woodcock, C.C., Andersen, D.J., Behzad, A.R., Nunes, S., Plawsky, J., Belfort, G., 2020. Linking microstructure of membranes and performance. J. Membr. Sci. 594, 117419. https://doi.org/10.1016/j.memsci.2019.117419.

Kennelly, A.E., 1899. The equivalence of triangles and three-pointed stars in conducting networks. Electr. World Eng. 34, 413–414.