

Systematic Enumeration and Identification of Unique Spatial Topologies of 3D Systems Using Spatial Graph Representations

Satya R. T. Peddada^{1a}, Nathan M. Dunfield^{2a}, Lawrence E. Zeidner^b, Kai A. James^{3a},
James T. Allison^{1a}

¹Department of Industrial and Enterprise Systems Engineering

²Department of Mathematics

³Department of Aerospace Engineering

^aUniversity of Illinois at Urbana-Champaign, Urbana, IL 61801

^bRaytheon Technologies Research Center, East Hartford, CT 06108

Email: {[speddad2](mailto:speddad2@illinois.edu), [nmdkaijames](mailto:nmdkaijames@illinois.edu), [jtalliso](mailto:jtalliso@illinois.edu)}@illinois.edu;
lawrence.zeidner@rtx.com

Abstract

Systematic enumeration and identification of unique 3D spatial topologies of complex engineering systems such as automotive cooling layouts, hybrid-electric power trains, and aero-engines are essential to search their exhaustive design spaces to identify spatial topologies that can satisfy challenging system requirements. However, efficient navigation through discrete 3D spatial topology options is a very challenging problem due to its combinatorial nature and can quickly exceed human cognitive abilities at even moderate complexity levels. Here we present a new, efficient, and generic design framework that utilizes mathematical spatial graph theory to represent, enumerate, and identify distinctive 3D topological classes for an abstract engineering system, given its system architecture (SA) – its components and interconnections. Spatial graph diagrams (SGDs) are generated for a given SA from zero to a specified maximum crossing number. Corresponding Yamada polynomials for all the planar SGDs are then generated. SGDs are categorized into topological classes, each of which shares a unique Yamada polynomial. Finally, for each topological class, one 3D geometric model is generated for an SGD with the fewest interconnect crossings. Several case studies are shown to illustrate the different features of our proposed framework. Design guidelines are also provided for practicing engineers to aid the utilization of this framework for application to different types of real-world problems.

1 INTRODUCTION

Systems engineering [1, 2] often involves choosing the most suitable candidate among many alternative design solutions to meet specific performance criteria using techniques such as comparative design analysis and optimization [3, 4]. In most cases, the component technologies and the component-to-component connectivity (system architecture) remains fixed between different feasible designs to preserve the functionality of the system. *System architecture (SA)* specifies what components comprise a system, and how ports on components are connected to specific ports on other components which represent specific technology options to perform each function, and the flows of material, energy, and/or information from one component to another. System architecture enumeration problems were previously studied for engineering design examples such as hybrid electric power trains [5–7], automotive vehicle suspension systems [8], and optimal cooling system layouts [9]. However, for each system architecture, many *spatial topologies (STs)* may exist, each with its own optimal geometry. The geometric optimization problem can be simplified by decomposing the problem into identification of unique spatial topologies, and then geometrically optimizing each one, subject to all relevant physical interactions. For example,

To appear in the proceedings of the ASME 2021 International Design Engineering Technical Conferences. An expanded journal version, with more details about the case studies, is in preparation.

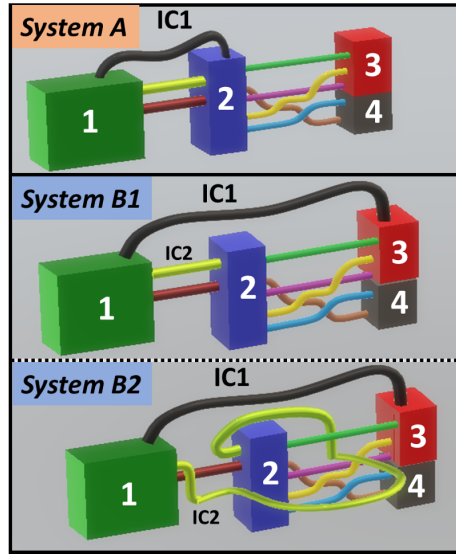


FIGURE 1: A simple illustration of system architecture and spatial topology concepts. 3D systems A and B1 have different system architectures. Systems B1 and B2 have the same system architecture but are two distinct spatial topologies.

if an interconnect links ports $P1$ and $P2$, many options exist for how this interconnect passes around various other interconnects and components in the system. Two spatial topologies are *equivalent* when there is a continuous deformation of component locations and interconnect trajectories that takes one topology to the other. Figure 1 shows 3D systems A and B1 having two different system architectures because interconnect IC1 is connected between components $\{1, 2\}$ in A but between $\{1, 3\}$ in B1. In other words, component-to-component connectivity is different in A and B1 respectively. Systems B1 and B2 have the same system architecture as all the component-to-component interconnections are the same. However, B1 and B2 have different spatial topologies because the interconnect IC2 between the component 1 and 2 is topologically different (please see the crossing patterns in Fig. 1). As both the ends of the interconnect IC2 are fixed, it cannot be continuously morphed between B1 and B2. Hence, B1 and B2 are topologically different systems. The scope of this paper deals with enumerating such unique spatial topologies for each system architecture within a design problem. The example shown in Fig. 1 is kept simple for illustration purposes, but the framework can be used to generate STs for more complex architectures and larger interconnected systems with multiple crossings.

Any systematic design process [10, 11] involves four key tasks: *representation*, *generation*, *evaluation*, and *design guidance*. Representation refers to the task of describing a system using a generic model that captures the functionality of the various system elements. Depending on the design analysis tools and application requirements, design representations can be mathematical, graphical, physics-based, or conceptual [12, 13]. The generation phase involves creating feasible design alternatives using the representation based on design rules. Evaluation is the process of measuring the design quality in terms of the performance criteria. Finally, design guidance is providing feedback for the generation task based on the evaluation output to find better alternatives in the design space. The generation, evaluation, and design guidance tasks are usually performed in an automated loop that converges finally to a design solution. However, what actually enables high design accuracy, comprehensive search space navigation, and computational efficiency of the output from the last three tasks is the design representation selected in the first task. Hence, a design representation for 3D interconnected systems that captures the relevant problem attributes and aligns well with a generation computation is critical to navigate through the

discrete 3D spatial topology options efficiently.

The design elements of a three-dimensional interconnected system are its components, their 3D spatial locations and orientations, their port valencies, interconnections, and the crossings of their interconnections. Unlike 2D system layout enumeration performed in the development of very large scale integrated circuits (VLSI), the 3D spatial layout enumeration problem is fundamentally different and even more challenging. More specifically, even for a given system architecture, options exist for how the interconnects are routed relative to one another and to the components (e.g., say, should duct A go over cable B or under, or should a pipe be routed through the hole in the casing, or around the edge of the casing, etc.). These are topologically discrete design differences. To cater to that need, in this paper we have identified a mathematical design representation to describe 3D interconnected spatial layouts as *spatial graph diagrams (SGDs)*.

1.1 Objectives and Contributions

Complex engineering systems such as autonomous aerial vehicles [14], electric power trains [5, 15], aero-jet engines [16], or vehicle thermal management and cooling systems [9, 17] have different kinds of components connected together either through wires, ducts, or pipes entangles with one another in a tightly packed three-dimensional volume. Design alternatives with distinct spatial topologies may exist, with different values of metrics such as efficiency, spatial packaging density, maintenance costs, and design complexity. Current practice for exploring different system spatial topologies relies largely upon human expertise, design rules, modification of existing designs, and manual adjustments. This approach precludes use of spatial topologies for practical application to typical complex systems, and suggests that automated methods are needed to apply spatial topologies for typical systems design. In addition, the realization of optimal functionality or performance is not guaranteed with current practice due to incomplete design space coverage. Therefore, the main objective of this paper is to develop an automated and systematic enumeration framework to both represent 3D engineering systems using spatial graph diagrams (SGDs) and efficiently generate distinct spatial topologies (STs) for a given system architecture using a rigorous mathematical approach. Advantages of using a spatial graph representation are: 1) simplicity, while capturing necessary system elements and features, 2) easy to visualize, 3) flexibility to add additional geometric features, 4) distinct topologies can be detected using polynomial invariants, 5) scalable or even decomposable into a set of smaller graphs, 6) supports automated 3D model generation, and 7) features such as node locations, edge diameters, edge trajectory shape functions, port locations, crossing information, etc., can be parameterized for performing numerical optimization. For example, items 6 and 7 can be very useful for using different 3D models as initial start points for multi-physics packing and routing optimization.

The major contributions of the proposed design framework include:

1. A new way to represent 3D engineering systems using spatial graph theory. This representation supports description of components as nodes, interconnects as edges, multiple crossings, and variable component valency.
2. Combinatorial enumeration of all SGDs for a given system architecture up a maximum number of crossings.
3. Efficient and systematic identification of unique SGDs from the exhaustively enumerated SGD set using Yamada polynomial invariants. This serves as a foundation to remove redundant topologies and explore the 3D spatial topology design space more thoroughly.

4. Topological equivalence between spatial graphs is tested here using Yamada polynomials rather than comparing diagrams directly.
5. Several case studies that illustrate the efficacy of this design automation framework.
6. Practical guidelines to help system design engineers apply this proposed framework to different kinds of problems.

The remainder of this paper is organized as follows. The terminology and notation of the proposed spatial graph representation are discussed in detail in Sec. 2. Section 3 describes the characteristics of Yamada polynomials and how they are evaluated for an individual SGD. Section 4 demonstrates the proposed six-step design framework that utilizes spatial graphs to represent, enumerate, and identify distinctive 3D topological classes for an engineering system, given its system architecture (SA). SGDs are generated for a given SA from zero to a specified maximum crossing number. Corresponding Yamada polynomials for all the enumerated SGDs are then generated. SGDs are categorized into topological classes, each of which shares a unique Yamada polynomial. Finally, for each topological class, 3D geometric models are generated. Section 5 presents several practical case studies based on the proposed framework. The results are discussed in Sec. 6. Finally, the conclusion and future work items are presented in Sec. 7.

2 SPATIAL GRAPHS

The study of graphs in 3-space has been mathematically formalized using *spatial graphs* [18–20], which we now describe. Suppose G is a graph, that is, a set of vertices and a set of edges, where an edge is just a pair of vertices. (Edges are undirected and multiple edges between the same pair of vertices are allowed.) A *spatial embedding* of a graph G is a set of points (nodes) in $\mathbf{R}^3$ corresponding to the vertices of G , and a set of smooth arcs (links) corresponding to the edges of G that join appropriate pairs of vertices; here, each arc meets the vertices only at its two endpoints, and it intersects other arcs only at these vertices. Collectively, these points and arcs form a *spatial graph* with underlying (abstract) graph G . More formally, the spatial embedding is a function $f : G \rightarrow \mathbf{R}^3$, whose image $\tilde{G} := f(G)$ is the spatial graph. See Figure 2(a) for a sample spatial graph. The natural topological notion of equivalence for spatial graphs is *isotopy*, when two spatial graphs $\tilde{G}_1$ and $\tilde{G}_2$ can be continuously deformed from one to the other without any arc passing through another arc or itself.

Spatial graphs are a natural extension of knot theory, which is the study of circles embedded in $\mathbf{R}^3$, since we can put vertices on a knot to make it into a spatial graph. While the study of knot theory has its origin in the physics of the late 19th century [21], spatial graph theory has its roots in chemistry [22, 23] and is different from graph theory because graph theory studies abstract graphs while spatial graph theory studies embeddings of graphs in $\mathbf{R}^3$ or even in other 3-manifolds [24–26]. This theory was used in polymer stereochemistry [22, 27] and molecular biology (e.g., protein folding) to distinguish different topological isomers. A folded protein can be thought of as a spatial graph where residues are the nodes and edges connect the residues in close proximity.

If a spatial graph is projected onto a plane, then some arcs (edges) may appear to cross in the projection. If information about which arc is on top at the apparent crossings is omitted, the projection is called a *shadow* of the spatial graph, as shown in Fig. 2(b). If we keep track of which arc is on top at each apparent crossing, the projection or planar representation is called a *diagram* of the spatial graph, as shown in Fig. 2(c). In other words, diagrams are the images of embedded graphs under a projection $\mathbf{R}^3 \rightarrow \mathbf{R}^2$ whose singularities are a finite number of crossings of edges equipped with over-under crossing information. Hence, many different spatial diagrams of a spatial graph G may have the same shadow.

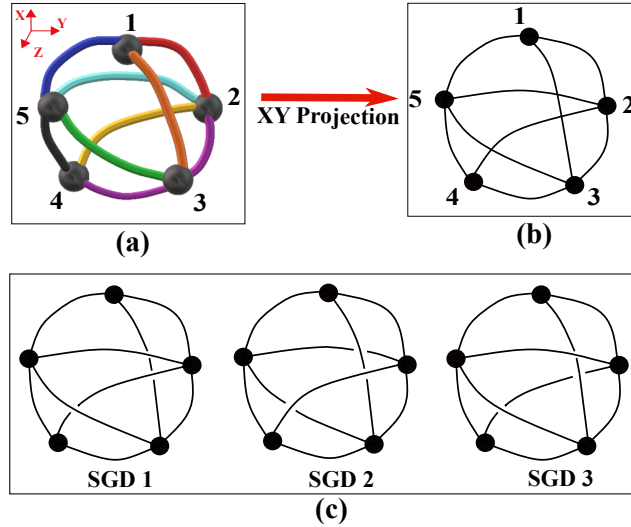


FIGURE 2: Figure (a) represents a five component interconnected 3D system; (b) is the shadow of the 3D diagram; and (c) shows three distinct spatial graph diagrams (SGD1, SGD2, and SGD3)

We can produce this family of spatial graph diagrams (SGDs) by assigning all possible permutations of overstrand or understrand information.

2.1 Reidemeister moves

Given an abstract graph G , we can use the spatial graph diagrams above to begin enumerating spatial embeddings of G . The challenge is then to determine which of these SGDs actually describe isotopic spatial embeddings (i.e., are topologically equivalent), so that later steps in the design process consider each topological possibility only once. Fortunately, it has been shown that two diagrams represent isotopic embeddings if and only if they are related by a finite sequence of fundamental *Reidemeister moves* ($R0$ to $R6$) [28–30] as shown in Fig. 3. Figure 4 shows a simple illustration of three diagrams where SGDs A and B are *topologically equivalent* under the first Reidemeister move $R1$ whereas C is not equivalent to either A or B as its edges cannot be continuously deformed using the Reidemeister moves to attain A or B, so they represent *topologically distinct* spatial graphs.

2.2 Flat vertex graphs and ribbon graphs

The topological formulation of spatial graphs is quite idealized in that each vertex has no local structure and the edges are infinitely thin. We can impose additional, but still purely topological, structure by considering *flat vertex graphs* and *ribbon graphs*, which may be more suitable for certain design applications. A flat vertex graph is a spatial graph where the vertices correspond to flat disks in $\mathbf{R}^3$ as shown in Fig. 5. In particular, this gives the edges coming in to each disk a cyclic order. A flat vertex graph can also be encoded by a SGD, with the convention that each disk is rotated parallel to the projection plane before projecting. Two SGDs represent isotopic flat vertex graphs if and only if they differ by a series of Reidemeister moves $R0$ to $R5$; here $R6$ is no longer allowed since it would change the order of the edges coming into the vertex disk.

A ribbon graph is a spatial graph whose vertices have become flat disks and whose edges have become thin bands, depicted in Fig. 6. These too can be encoded as SGDs by using the blackboard framing

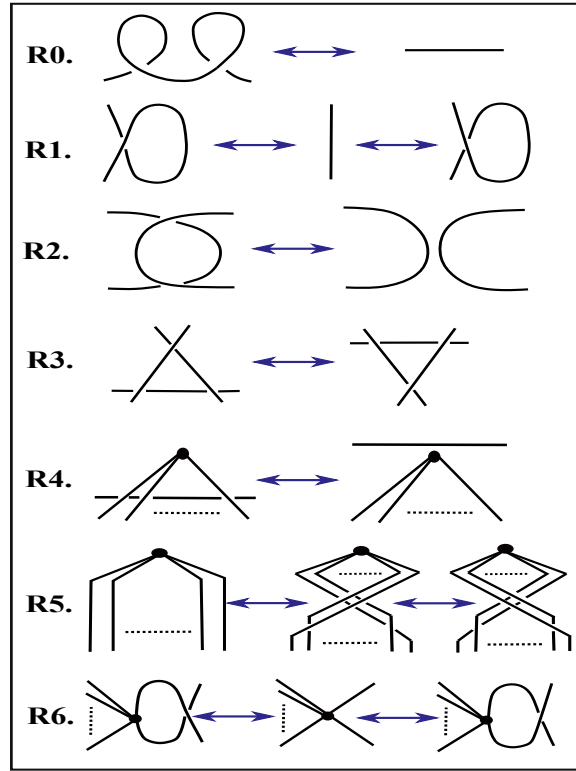


FIGURE 3: Fundamental Reidemeister moves for spatial graphs. R5 describes the move of taking the node and rotating along its vertical axis, dragging the strands. In R6 move, only the two strands to the right of the node are being moved.

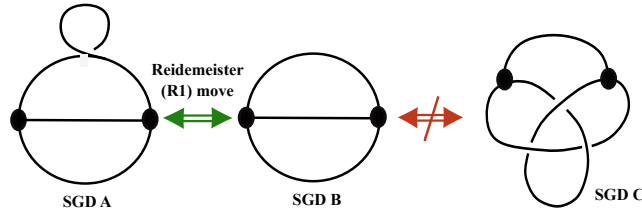


FIGURE 4: SGD A and B are topologically equivalent θ_1 graphs under Reidemeister-I (RI) move. C is a θ_2 graph and is topologically distinct from A and B under any fundamental R moves.

convention (a way to view a knot diagram on a plane) illustrated in Fig. 6. This framing is obtained by converting each component to a ribbon lying flat on the plane. Two SGD s represent isotopic ribbon graphs if and only if they differ by a sequence of Reidemeister moves $R0$ and $R2$ to $R5$. The basic notation of a spatial graph introduced in Sec. 2 is sometimes referred to as a *pliable* spatial graph to contrast the notion with flat vertex and ribbon graphs. Here, we will focus on pliable and flat vertex graphs, but note that ribbon graphs would be useful for measuring twisting along interconnects in the final 3D system.

3 YAMADA POLYNOMIAL INVARIANTS

Reidemeister moves are valuable for identifying when two embeddings are isotopic (that is, topologically equivalent); however, finding the specific sequence of moves between two equivalent spatial diagrams can be extremely challenging, especially when the spatial graphs have many nodes and edges. Even for knots, which are the simplest class of spatial graphs, it is unknown whether there exists a polynomial-time

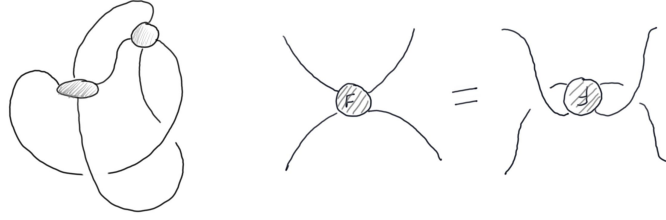


FIGURE 5: At left is a typical flat vertex graph. The Reidemeister $R5$ move on this kind of spatial graph is shown at right.

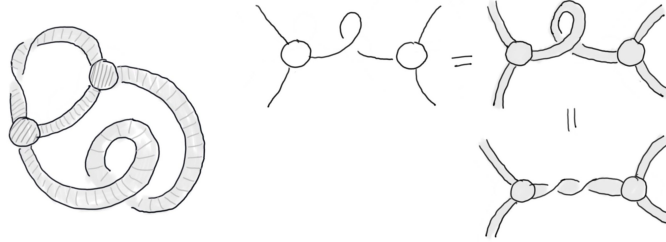


FIGURE 6: At left is a typical ribbon graph; notice the twists in the topmost band. The upper-right shows how a SGD encodes a ribbon graph via the blackboard framing, with the vertical isotopy showing how loops in the SGD can describe twists in the ribbons.

algorithm for determining when two knots are isotopic. (It is not impossible that such an algorithm exists: the question of whether a knot is equivalent to a round circle is in $\mathbf{NP} \cap \mathbf{coNP}$ [31, 32].) To show that two embeddings are not *isotopic* requires an *invariant*: a function of the embeddings whose output is not changed by isotopies, and which takes different values on the two embeddings [33–36]. Mathematicians often use such invariants that are computable and yet powerful enough to detect some delicate differences of embeddings of the same graph. Over the last century, many *polynomial invariants* [34, 37–39] were discovered by knot theorists, such as the Alexander-Conway [40], Jones [41], Kauffmann [42], and Yoshinaga [43] polynomials. Some of these have been extended to spatial graph theory [44–46] using similar constructions. These invariants satisfy nice *skein relations* which are mathematical tools that give linear relationships between the polynomials of closely related diagrams. Relevant skein relations are sufficient to calculate the polynomials recursively and are relatively convenient to use for this purpose. The proof of invariance then relies on using the skein relation to show the value of the invariant is unchanged by Reidemeister moves.

3.1 Yamada polynomial properties

The specific polynomial invariant used here is the Yamada polynomial, which associates to each SGD a polynomial in an indeterminate A , which is an arbitrary independent variable. For example, it turns out that the Yamada polynomial for the SGD C in Fig. 4 is $-A^{-6} - A^{-5} - A^{-4} - A^{-3} - A^{-2} - 1 - A^2 + A^6$. The Yamada polynomial is defined in terms of a polynomial invariant H of ordinary (non-spatial) graphs. Continuing the example, the abstract theta graph that underlies all the SGDs in Fig. 4 has

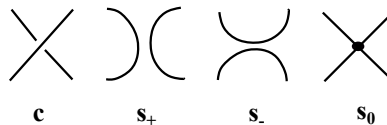


FIGURE 7: Class of spins for a crossing.

$H(A) = -A^{-2} - A^{-1} - 2 - A - A^2$. We first discuss the polynomial invariant H and its properties P1–5, which can be used to recursively compute H for any graph. We then turn to the Yamada polynomial and its fundamental properties S1–4 which similarly allow for recursive computation of it for any SGD. After giving some helpful additional properties Y1–8 of the Yamada polynomial, it is computed for four different example SGDs in Sec. 3.2.

Let $G = (V, E)$ be an abstract graph, where V is the vertex set and E is the edge set of G . For two graphs G_1 and G_2 , $G_1 \sqcup G_2$ denotes the disjoint union of G_1 and G_2 , and $G_1 \vee G_2$ denotes a wedge at a vertex of G_1 and G_2 , that is $G_1 \vee G_2 = G_1 \cup G_2$ and $G_1 \cap G_2 = \{\text{vertex}\}$. In addition, a graph G has a cut-edge e (also known as bridge or isthmus) if $G - e$ has more connected components than G . First, following Ref. [47], a polynomial invariant $H(G)(A)$ of an abstract graph G is described, where A is an indeterminate (arbitrary independent variable); precisely, our $H(G)(A)$ is Yamada's $h(x, y)$ with $x = -1$ and $y = -A - 2 - A^{-1}$. The polynomial $h(G)(A)$ is characterized by the following properties:

- P1. $H(\text{empty graph}) = 1$ and $H(\text{simple loop}) = A + 1 + A^{-1}$.
- P2. $H(G_1 \sqcup G_2) = H(G_1)H(G_2)$
- P3. $H(G_1 \vee G_2) = -H(G_1)H(G_2)$
- P4. If G has a cut edge, then $H(G) = 0$.
- P5. Let e be a non-loop edge of a graph G . Then $H(G) = H(G/e) + H(G - e)$, where G/e is the graph obtained from G by contracting e to a point and $G - e$ is G with e deleted.

Now we can define a powerful and much-studied invariant of spatial graphs, the *Yamada polynomial* [46–50]. Let g be a spatial graph diagram. For a crossing c of g , three *graph reductions* are defined as: s_+ , s_- , and s_0 (denotes a vertex), with the class of spin $+1$, -1 , and 0 , respectively, as shown in Fig. 7. These graph reductions are used to replace crossings in a spatial graph for Yamada polynomial calculation. Let S be the planar graph obtained from g by replacing each crossing with a spin. S is called a *state* on g and $U(g)$ denotes the set of states on g obtained by applying all possible reductions in its crossings. Set $\{g|S\} = A^{n_1 - n_2}$, where n_1 and n_2 are the numbers of crossings with spin of $+1$ and spin of -1 , respectively, and A is an indeterminate. The Yamada polynomial $R\{g\}(A) \in \mathbf{Z}[A, A^{-1}]$ is defined as:

$$R(g) = R(g)(A) = \sum_{S \in U(g)} \{g|S\} H(S),$$

In particular, if the diagram of g does not have crossings, then $R(g) = H(g)$. This Yamada polynomial for a spatial graph can be computed recursively using the following skein relations and the properties of H :

- S1. $R(\text{crossing}) = AR(\text{cup}) + A^{-1}R(\text{cap}) + R(\text{crossing})$,
- S2. $R(\text{edge with crossing}) = R(\text{edge with cup}) + R(\text{edge with cap})$, where e is a non-loop edge.
- S3. $R(g_1 \sqcup g_2) = R(g_1)R(g_2)$,
- S4. $R(g_1 \vee g_2) = -R(g_1)R(g_2)$.

So far, the Yamada polynomial is a function of the given diagram g and we need an invariant of the spatial graph $\tilde{G}$ it describes. Yamada showed:

- I1. Any two diagrams g and g' whose flat vertex graphs $\tilde{G}$ and $\tilde{G}'$ are isotopic have $R(g') = (-A)^n R(g)$ for some integer n .
- I2. If every vertex has valence at most three, then two diagrams g and g' whose spatial graphs $\tilde{G}$ and $\tilde{G}'$ are isotopic have $R(g') = (-A)^n R(g)$ for some integer n .
- I3. Any two diagrams g and g' whose associated ribbon graphs $\tilde{G}$ and $\tilde{G}'$ are isotopic have $R(g') = R(g)$.

The next set of relations for the Yamada polynomial can be derived from the previous ones and are very useful aides for its calculation. Detailed proofs for these relations (Y1-Y8) are provided in Ref. [47]. They are as follows:

$$Y1. R(\bigcirc) = B, \quad \text{where } B = A + 1 + A^{-1},$$

$$Y2. R(\bigcirc \bigcirc) = -BR(\bigcirc \bigcirc),$$

$$Y3. R(\bigcirc \bigcirc \bigcirc) = -AR(\bigcirc \bigcirc \bigcirc) - (A^2 + A)R(\bigcirc \bigcirc \bigcirc),$$

$$Y4. R(\bigcirc \bigcirc \bigcirc) = -A^{-1}R(\bigcirc \bigcirc \bigcirc) - (A^{-2} + A^{-1})R(\bigcirc \bigcirc \bigcirc),$$

$$Y5. R(\bigcirc \bigcirc \bigcirc) = -AR(\bigcirc \bigcirc \bigcirc), \quad R(\bigcirc \bigcirc \bigcirc) = -A^{-1}R(\bigcirc \bigcirc \bigcirc),$$

$$Y6. R(\bigcirc \bigcirc) = A^2 R(\bigcirc \bigcirc), \quad R(\bigcirc \bigcirc) = A^{-2} R(\bigcirc \bigcirc),$$

Y7. Edge subdivision does not change the polynomial:

$$R(\overset{1}{\bullet} \text{---} \overset{3}{\bullet} \text{---} \overset{2}{\bullet}) = R(\overset{1}{\bullet} \text{---} \overset{2}{\bullet})$$

Y8. Petals to concentric self-loops:

$$R(\bigcirc \bigcirc) = R(\bigcirc \bigcirc) = -B^2 = -(A + 1 + A^{-1})^2.$$

3.2 Illustrative Examples

Yamada polynomials for a few spatial graphs are calculated by reducing the spatial graph diagram into a linear combination of smaller elements based on the skein relations stated above.

Example 1 - Theta (θ_1) graph: The Yamada polynomial for a standard theta graph is calculated as follows:

$$\begin{aligned}
 R(\bigcirc \bigcirc) &= \underbrace{H(\bigcirc \bigcirc)}_{\substack{\text{(Apply S2 on} \\ \text{the center edge)}}} = \underbrace{H(\bigcirc \bigcirc)}_{\downarrow \text{(Apply Y7)}} + \underbrace{H(\bigcirc \bigcirc)}_{\downarrow \text{(Apply Y8)}}, \\
 &\Rightarrow R(\bigcirc \bigcirc) = H(\bigcirc \bigcirc) + H(\bigcirc \bigcirc), \\
 &\Rightarrow R(\theta_1) = B - B^2, \quad (\text{where } B = A + 1 + A^{-1}) \\
 &\Rightarrow R(\theta_1) = -(2 + A + A^{-1} + A^2 + A^{-2})
 \end{aligned}$$

Example 2: A spatial graph  which is isotopic (by the R6 move) to the standard theta graph. Its Yamada polynomial is calculated as follows, though one could instead use property Y5 as a shortcut.

$$\begin{aligned}
R\left(\text{figure-eight}\right) &= AR\left(\text{two loops}\right) + A^{-1}R\left(\text{figure-eight}\right) + R\left(\text{figure-eight}\right), \\
R\left(\text{two loops}\right) &= 0, & \text{(because of isthmus based on property P3)} \\
R\left(\text{theta graph}\right) &= H\left(\text{theta graph}\right) = H\left(\text{two circles}\right) + H\left(\text{figure-eight}\right), \\
&= B - B^2, & \text{(where } B = A + 1 + A^{-1}\text{)} \\
R\left(\text{figure-eight}\right) &= H\left(\text{figure-eight}\right) = H\left(\text{theta graph}\right) + H\left(\text{figure-eight}\right), \\
&= H\left(\text{three loops}\right) + H\left(\text{theta graph}\right) + H\left(\text{figure-eight}\right), \\
&= B^3 + (-B^2 + B) + -B^2 = B^3 - 2B^2 + B, \\
\Rightarrow R\left(\text{figure-eight}\right) &= A(0) + A^{-1}(B - B^2) + (B^3 - 2B^2 + B), \\
\Rightarrow R\left(\text{figure-eight}\right) &= A^3 + A^2 + 2A + 1 + A^{-1}, \\
&= -A(R(\theta_1)). \quad \text{Note the } -A \text{ factor permitted in I2.}
\end{aligned}$$

Example 3: The spatial graph is . This example involves extensive use of Yamada polynomial skein relations (Y1-Y8). Its Yamada polynomial is calculated as follows:

$$\begin{aligned}
&= \underbrace{AR\left(\text{figure-eight}\right)}_{\downarrow \text{(Apply Y5)}} + \underbrace{A^{-1}R\left(\text{figure-eight}\right)}_{\downarrow \text{(Apply Y6)}} + R\left(\text{figure-eight}\right), \\
&\text{Drop the Rs for simplicity,} \\
&= A(-A\left(\text{theta graph}\right)) + A^{-1}(A^{-2}\left(\text{figure-eight}\right)) + \underbrace{\left(\text{figure-eight}\right)}_{\downarrow \text{(Apply Y4)}}, \\
&= (-A^2 + A^{-3})\left(\text{theta graph}\right) + -A^{-1}\left(\text{figure-eight}\right) - (A^{-2} + A^{-1})\left(\text{figure-eight}\right) \\
&= (-A^2 - A^{-2} - A^{-1} + A^{-3})\left(\text{theta graph}\right) - A^{-1}\left(\text{figure-eight}\right), \\
&= A^4 + A^3 + A^2 + A - A^{-2} - A^{-3} - A^{-4} - A^{-5} \\
&\text{Since taking } B = A + 1 + A^{-1} \text{ we have:}
\end{aligned}$$

$$\begin{aligned}
\text{Diagram} &= \text{Diagram 1} + \text{Diagram 2} = \text{Diagram 3} + \text{Diagram 4} + \text{Diagram 5}, \\
&= +B^3 + (-B^2 + B) + -B^2 = B^3 - 2B^2 + B.
\end{aligned}$$

Example 4: The spatial graph is . This example involves extensive use of skein relations (Y1-Y8). Its Yamada polynomial is calculated as follows:

$$\begin{aligned}
&= AR\left(\text{Diagram 1}\right) + A^{-1}R\left(\text{Diagram 2}\right) + R\left(\text{Diagram 3}\right), \\
&\text{Drop the Rs for simplicity,} \\
&= A^3 \underbrace{\text{Diagram 1}}_{\text{Example 2}} + A^{-1} \underbrace{\text{Diagram 2}}_{\text{Example 3}} - A \text{Diagram 3} - (A^2 + A) \text{Diagram 4}, \\
&= -A^4 \text{Diagram 5} + -A^{-1} \text{Diagram 6} - A(-A^{-1}) \text{Diagram 7} \\
&\quad - (A^{-2} + A^{-1}) \text{Diagram 8} - (A^2 + A)(-A) \text{Diagram 9} \\
&= (-A^4 + A^{-1} + 1 + A^3 + A^2) \text{Diagram 5} + A^{-1} \text{Diagram 6} + \text{Diagram 7} \\
&= A^6 - A^2 - 1 - A^{-2} - A^{-3} - A^{-4} - A^{-5} - A^{-6}.
\end{aligned}$$

4 PROPOSED SGD ENUMERATION FRAMEWORK

Figure 9 shows the steps of the proposed design framework to represent, enumerate and categorize unique spatial topologies of a 3D system (assuming inter-component connectivity is fixed). The detailed steps of the enumeration design framework are:

- **1. Define system architecture:** Provide the specific 3D system architecture (SA) for which spatial topologies must be enumerated. From the SA, extract the number of nodes (components), their valencies, system interconnectivity, and the corresponding edges (interconnects in the system).
- **2. Enumerate spatial graph diagrams:** Combinatorially enumerate all possible spatial graph diagrams (SGDs) for the SA from zero crossings to the maximum crossing number ($k = 0, 1, \dots, k_m$) using their corresponding shadows.
- **3. Check graph planarity:** Planar diagrams (PDs) of spatial graphs are used for the calculation of the Yamada polynomials. However, before calculation, each enumerated graph must be checked to determine whether it is planar, for which there are linear-time algorithms [51]. The graphs start with a circular order of the edges at each vertex, making the planarity check even easier. The algorithm shown in Fig. 8 recursively contracts the edges of a graph until the diagram is a bouquet of circles, and then use the fact that if the diagram is planar, there must exist at least one loop edge whose endpoints come consecutively in the cyclic ordering around the vertex, i.e., a self-loop. This self-loop can be removed without altering planarity. The recursive steps for the planarity check (PC) algorithm are enumerated below:

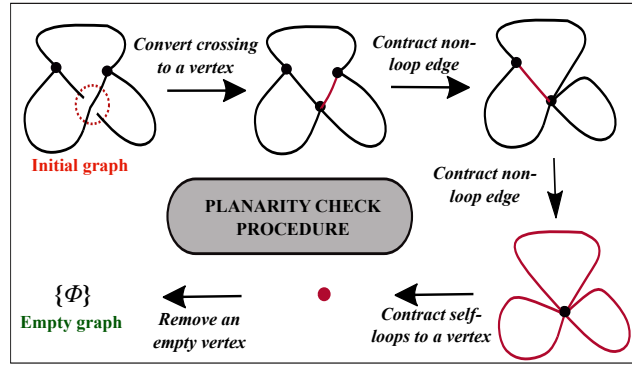


FIGURE 8: Identifying connected planar spatial diagrams from the combinatorially enumerated set is performed using this procedure.

- PC1. Convert all vertices to crossings, as it does not affect planarity.
- PC2. Contract all non-loop edges (edges shared between two vertices) to a vertex, as it does not affect planarity.
- PC3. Remove all planar self-loops at a vertex.
- PC4. Empty vertex does not affect planarity, so remove it. By doing all these steps recursively, if the result is an empty diagram, then the original diagram is planar. If not, the diagram is non-planar.

- **4. Evaluate Yamada polynomials:** The Yamada polynomials for all the valid planar SGD's are evaluated using the Yamada polynomial properties detailed in Sec. 3.
- **5. Categorize different spatial topologies:** Cluster SGD's into classes so that the SGD's in each class have the same Yamada polynomial and no two classes have the same polynomial. Any pair of SGD's that represent isotopic spatial graphs will be in the same Yamada class, so we then consider only one SGD per Yamada class. In most engineering applications, multiple crossings between a pair of interconnects is undesirable. For example, there is no advantage to two wires intertwining several times unless they intentionally function as a twisted pair. Similarly, a pair of pipes that intertwine are more costly to fabricate, and more complex to install and remove. Hence, from each class an SGD is selected having the fewest crossings, and is used to generate a 3D geometric model.
- **6. 3D model generation:** Simpler SGD's from step 5 are utilized as underlying skeleton structures for generating various 3D system geometric models.

4.1 Comparison to Other Methods

Steps 2–5 above follow the standard strategy in low-dimensional topology for enumerating knots, which are a special class of spatial graphs, see [21] for an overview. Using additional techniques from hyperbolic geometry, it is possible to exactly enumerate all knot topologies with less than 20 crossings, of which there are more than 350 million [52].

Compared to such massive computations, prior work on spatial graphs where the underlying graph is more complicated than just a loop is limited: mostly tabulations of less than 100 topologies [53–58]. For example, the authors in Ref. [56] generated two vertex bouquet spatial graphs with a maximum

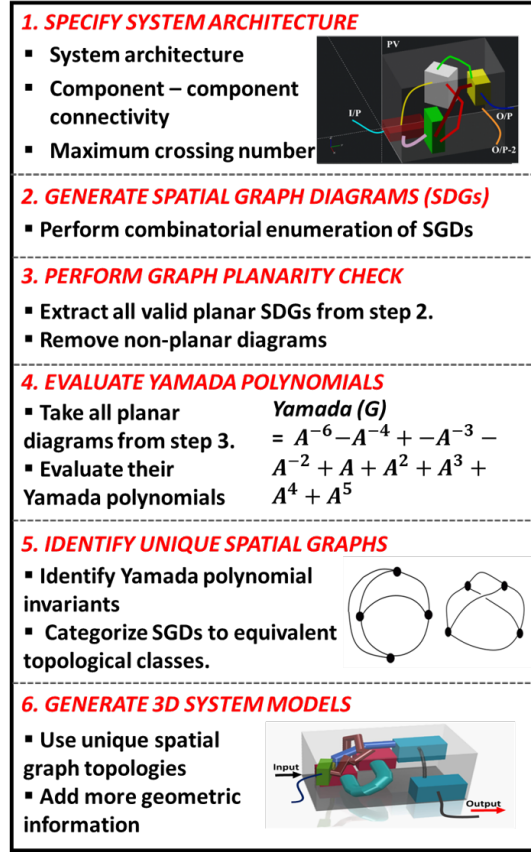


FIGURE 9: The six sequential steps of the proposed spatial graph-based topology enumeration framework.

of seven crossings. As our cases studies demonstrate, the strategy above allows for much larger-scale enumerations, with arbitrary specified system architectures. This is very suitable for representing large-scaled complex engineering systems easily, and for enumerating their spatial topologies efficiently.

5 CASE STUDIES

In this section, a number of case studies are provided to demonstrate the proposed enumeration framework discussed in Sec. 4. All computations (Yamada polynomial calculation, planarity check, etc.) in the case studies were performed using WOLFRAM MATHEMATICA 11.3 software with an Intel Xeon E5-2660 CPU @ 2.00 GHz, 64 GB DDR4-2400 RAM, WINDOWS 10 64-bit workstation.

5.1 Case Study 1: Components with equal valencies

In this case study, we consider a 3D system with architecture as shown in Fig. 10. This system contains four identical trivalent components (nodes), and six interconnects (edges). We find the unique 3D spatial topologies (STs) of the system for crossing numbers varying from zero to three. The notation used to indicate each SGD is given by SGD_k where k is the crossing number of that diagram, and the letter refers to the specific SGD. SGD_0 is the original system architecture without any crossings. Using the proposed framework, we combinatorially enumerate all the SDGs and pass them through a planarity check procedure. Yamada polynomials were then calculated for 3, 31, 118, and 231 valid planar SDGs having 0, 1, 2 and 3 crossing numbers respectively. We group the SDGs having the same Yamada polynomial or differing by a factor $(-A)^n$ under the same topological class based on property

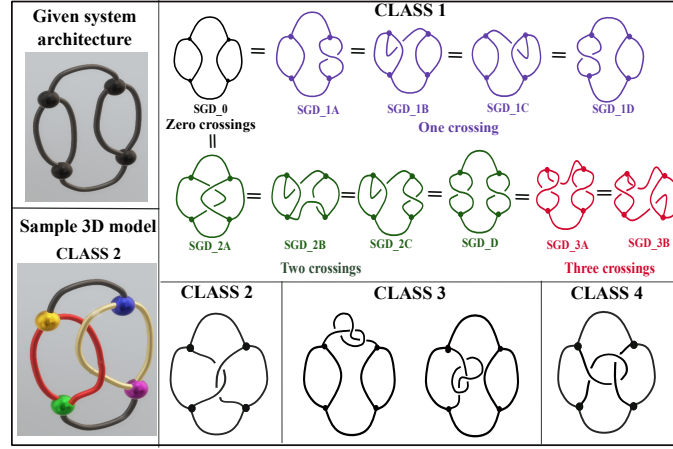


FIGURE 10: Results from case study 5.1 for a given system architecture for maximum crossing numbers from 0 through 3.

12. Through this we attain a total of four unique Yamada classes as shown in Fig. 10. The Yamada polynomials for these different classes of topologies are shown in Table 1. For class 1, it is observed that three distinct Yamada polynomials exist for different crossing numbers, but they all differ by a factor $(-A)^n$. This strongly suggests that the SGDs shown under class 1 are isotopic to SGD_0, and indeed this can be verified using the R6 move. In contrast, the two SGDs in class 3 can be shown to be nonisotopic using other tools. A sample 3D model of class 2 spatial topology candidate design solution is also shown in Fig. 10. The computational time for this entire case study is 78.3 seconds (s).

TABLE 1: Yamada polynomials of diagrams shown in Fig. 10 (case study 5.1).

Classes	Yamada polynomials
Class 1 SGD_0:	$A^{-3} + A^{-2} + 3A^{-1} + 2 + 3A + A^2 + A^3$
Class 1 SGD_1:	$(-A)^1(A^{-3} + A^{-2} + 3A^{-1} + 2 + 3A + A^2 + A^3)$
Class 1 SGD_2:	$(-A)^2(A^{-3} + A^{-2} + 3A^{-1} + 2 + 3A + A^2 + A^3)$
Class 1 SGD_3:	$(-A)^3(A^{-3} + A^{-2} + 3A^{-1} + 2 + 3A + A^2 + A^3)$
Class 2	$A^{-4} + A^{-3} + A^{-2} + 2A^{-2} + 2A + 2A^3 + A^4 + A^5 + A^6$
Class 3	$A^{-7} + A^{-6} + 3A^{-5} + 3A^{-4} + 4A^{-3} + 3A^{-2} + 3A^{-1} - 3A^2 - 2A^3 - 2A^4 - A^5 + A^6 + A^8$
Class 4	$A^{-7} + A^{-6} + A^{-5} + 2A^{-4} + 2A^{-3} + A^{-2} + 3A^{-1} + 2A - A^2 - A^5 + A^6$

5.2 Case Study 2: Scaling the system

The goal of this case study is to provide insight about how the number of unique spatial topologies varies with increasing number of components and the maximum number of crossings in a 3D system. For consistency, here all the components are assumed to be trivalent. Table 2 shows the number of unique Yamada classes obtained for each of the component and crossing number combinations, where here SGDs having the same Yamada polynomial or differing by a factor $(-A)^n$ are categorized together. The computational time increased drastically as one moves from a small-scaled (2 components and 2 crossings) to a large-scaled (10 components and 10 crossings) system case study from 12.3 sec to 24,256.6 sec (6.73 hrs), respectively.

TABLE 2: Case study 2: Unique spatial topology classes with increasing number of components and crossings.

Components	Crossings				
	2	4	6	8	10
2	1	2	3	4	6
4	1	4	6	8	9
6	2	4	8	15	21
8	4	6	9	20	28
10	5	12	16	27	36

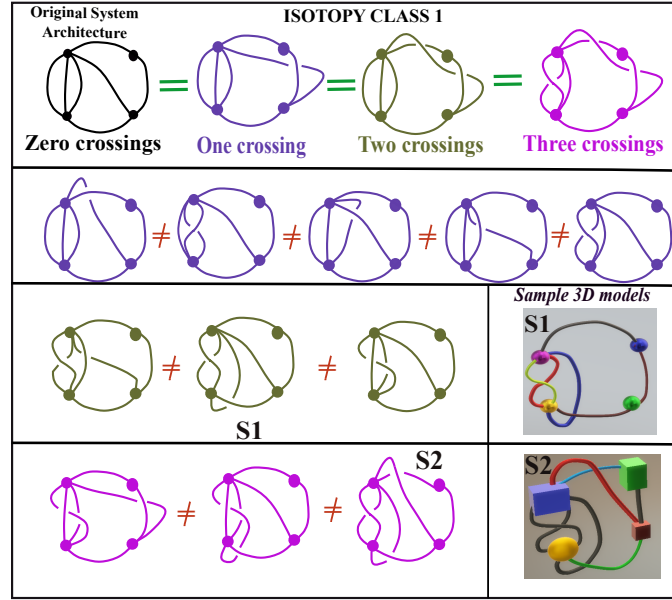


FIGURE 11: Results from case study 5.3 for a given system architecture for maximum crossing numbers from 0 through 3 with components having different valencies.

5.3 Case Study 3: Components with different valencies

Unlike case study 5.1, a system with four components but each with different valencies (number of ports) is considered. In addition, flat vertex graphs (FVGs) as described in Sec. 2.2 are used. As FVGs have local structures at nodes, the edge connectivity order around the nodes is preserved, and thus FVG representations are highly suitable for design applications where nodes have a specific cyclic ordering of ports. Here R0 to R5 moves are valid but not R6. Figure 11 shows some of the results obtained in this study. After computing the Yamada polynomials of hundreds of planar SGDs, a total of 27 unique Yamada classes are obtained. For illustration purposes, we show some isotopes of SGD₀ (original system architecture) as class 1 isotopes. Furthermore, unique SGDs belonging to some unique Yamada classes are shown for crossing numbers one, two, and three respectively. Two final 3D system geometric models (referred as S1 and S2) are also shown in Fig. 11. The total computational time taken for study B is 211.4 sec. It can be observed from this study that with components with different valencies, we get more unique Yamada classes than those with identical components. Thus, manually generating such designs is very challenging and the automated enumeration framework we proposed here is very valuable.

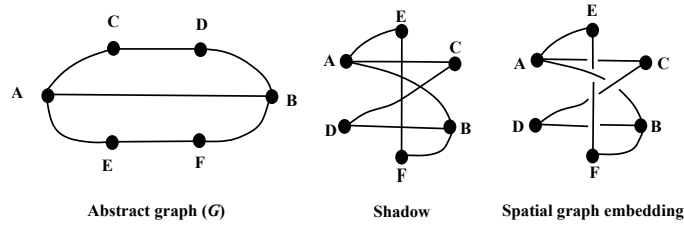


FIGURE 12: Implementation of the circular spatial graph representation technique in Sec. 5.4 to avoid unnecessary or extra twists between any two edges of a diagram.

5.4 Case Study 4: Circular graph representation

While filtering out isotopic spatial graph diagrams in the previous case studies, we observed that in a few occasions where system topologies have many crossings, two edges in that diagram twist around each other multiple times. Although a higher crossing number is satisfied, intertwining between edges can often be reduced by Reidemeister moves to a smaller crossing number, so essentially, no unique spatial topology is attained. Such intertwining is practically not observed or desirable between pipes or ducts in most complex systems (e.g., aero-engine externals, hydraulic systems). Some syntactic constraints need to be imposed to prevent more than a simple crossing between any two edges. This requires a representation that implicitly forbids twisting of two edges multiple times around each other. One way to get different spatial embeddings of an input abstract graph G as shown in Fig. 12 is to: 1) Pick an ordering of the nodes and use that to arrange them along a circle on the plane, 2) Connect the nodes by straight lines corresponding to the edges of G . This gives the “shadow”, and 3) Resolve the intersection lines of the shadow into over or under crossings. Figure 12 shows the shadow of graph G based on a particular cyclic order of nodes and one spatial graph embedding. As there are five crossings, a total of $2^5 = 32$ spatial embeddings are possible. The unique ones can be identified using the proposed design framework.

5.5 Case Study 5: Large-scaled system - spatial graph decomposition approach

From the observations made in the previous case studies, it is evident that enumerating spatial topologies for most real-world systems containing many components and approximately hundreds of crossings is intractable with manual processes and can become computationally expensive with automated methods such as the one presented in this paper. In contrast, enumerating spatial topologies of each subsystem of components can be a simple and efficient process. A complex spatial graph can thus be converted to a set of sub-graphs, and the unique spatial topologies of these sub-graphs can then be enumerated separately. The sub-graphs can be decoupled and can be considered as super-nodes. This decouples the task into two subtasks: 1) Enumerate STs of the system graph with only the subsystems as super-nodes, and 2) enumerate unique STs within each sub-graph. This presents fewer design candidates about which to make decisions, which greatly reduces the overall computational expense. Figure 13 shows a random complex spatial graph with 14 nodes, 20 edges and allowing at most 10 edge crossings. Approximately 1.134×10^4 SGD are attained for this entire system that fall under 434 unique Yamada polynomial categories. As this is a very large set, decomposition of the graph into sub-graphs (as super-nodes) is appropriate. First, a unique spatial topology of the super-nodes graph is found. Case study 1 in Sec. 5.1 is utilized as a sub-graph for demonstration purposes. Note that while enumerating STs for the spatial sub-graph, the rest of the system is condensed as an extra node in the sub-graph to preserve spatial connectivity information. Finally, using the proposed design framework, unique STs of the sub-graph can be plugged into the original system to attain system configurations. The scope of this paper only

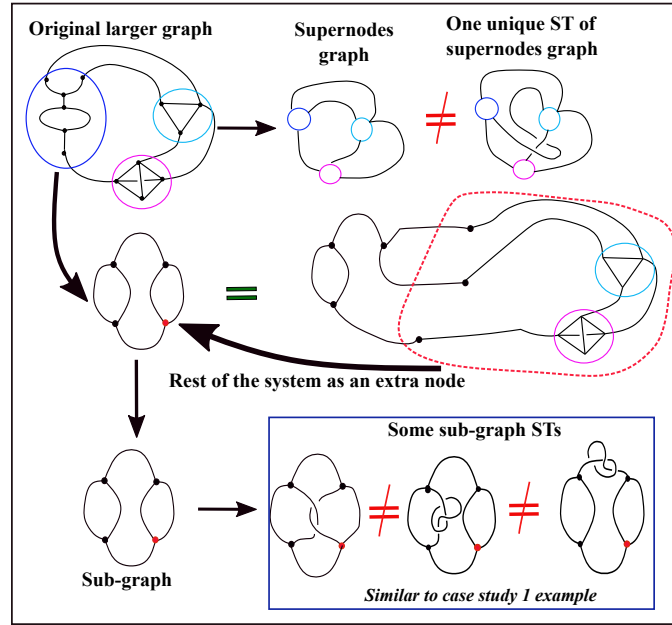


FIGURE 13: Demonstration of spatial graph diagram decomposition approach discussed in case study 5.5.

deals with enumerating unique STs, so we plan to show how each of these unique topologies affect overall system performance in future work.

To explain this decomposition concept using a concrete engineering design example, suppose that the complex network represents the spatial topology of a hybrid-electric vehicle powertrain; one possible subsystem could be a fluid-thermal cooling circuit. Each distinct circuit topology can be geometrically optimized for fair comparison, revealing how the topological features contribute to the overall system efficiency, fuel economy, thermal loss management, and other figures of merit due to physical interactions between components, interconnections, and the environment. The best candidate ST can then be chosen according to the desired performance requirements as in Refs [8, 9] where same procedure was followed but for ranking different system architectures (SAs).

6 DISCUSSION

This section summarizes a list of important observations from the five case studies as follows:

1. From the above case studies, especially 5.1, and 5.2, it can be observed that the number of unique spatial topologies finally attained for a given interconnect crossing complexity are much smaller than the combinatorially enumerated set of spatial graph diagrams as most of them are isotopic to each other under the Reidemeister moves.
2. Case study 5.3 presents another contribution of this work as the proposed method enumerates and identifies newer spatial topologies for components with different valencies in contrast to existing work that is mostly limited to two or three equivalent vertices. [53–57].
3. The circular graph representation method, presented in case study 5.4, is a simple way to enumerate and realize SGDs and avoid edge intertwining, although Yamada polynomials should still be used for identifying unique STs. Furthermore, specific syntactic constraints can be added to significantly reduce the initial set of SGDs obtained for planarity checking and Yamada polynomial evaluation. For example, by adding constraints on total crossings allowed between two edges of a system,

there is greater control on the type of spatial topologies finally obtained. This will be studied more in future work.

4. As seen in case study 5.5, for large-scaled systems, the best way to achieve different STs and search effectively is by graph decomposition. The spatial graph of the subsystem, which plays a critical role in performance impact, can be extracted to find its unique topologies. This avoids the need to enumerate thousands of diagrams of a complex network, compute their polynomials and compare them. Moreover, sub-graph designs can be optimized for performance independently and then combined with the remaining system.
5. Another very impactful aspect of this framework is that for one system architecture, there can be a range of spatial topologies from zero to many crossings. The spatial embeddings with fewer interconnect crossings are generally more useful for practical engineering purposes. Therefore, for existing, complex real-world system designs with 10s of crossings, using the proposed design framework, a simpler spatial topology can be found for that network with a much lower crossing number, but still keeping the same system connectivity.

In this work, it is more important to illustrate different topological designs than to analyze the difference in the corresponding Yamada polynomials, hence, the polynomials for all the diagrams shown in these case studies are not listed. However, the software code for generating these diagrams shown in the case studies and the corresponding Yamada polynomials data will be released for the benefit of the community in the near future.

7 CONCLUSION

The design representation presented in this paper greatly enhances the study of 3D engineering system spatial topologies in a systematic manner and is supported by rigorous mathematical foundations in spatial graph theory. Topologies of complex engineering systems, designed for particular applications, are conventionally created manually. But for more effective performance and efficiency, systematic identification, enumeration, and classification of possible system topologies is necessary. A framework for representing three-dimensional interconnected engineering systems using spatial graph embeddings is presented. Initially, all the combinatorial spatial graph descriptions up to some fixed topological complexity are enumerated for an input system architecture. A polynomial invariant, the Yamada polynomial is then calculated for the set of all the spatial graphs attained from the combinatorial permutations. The Yamada polynomial helps identify the duplicate spatial graph topologies from the exhaustive set and a smaller set of unique spatial embeddings (equivalent topological classes) is obtained. This smaller set of spatial graphs can be used for generating three dimensional geometric system models. Five case studies have been demonstrated using the proposed enumeration strategy. The results show that this method is efficient, scalable, applicable to all general 3D interconnected system networks, allows comprehensive exploration of the design space, and greatly aids in the design and development of unprecedented system topologies.

Future work includes adding more geometric features to these spatial graph embeddings, such as representing nodes with geometric shapes and ports. Investigation of braid-based representations of interconnect networks is also anticipated. As the system becomes larger, evaluating Yamada polynomials for many SGD is very time-consuming. This can be overcome by implementing a mix of Reidemeister moves to eliminate isotopic diagrams quickly to produce a smaller set of diagrams that require Yamada calculations. Other application aspects include utilizing the unique spatial topologies obtained here as starting points for physics-based packing and routing optimization of 3D systems. Furthermore, research

areas that can benefit from SGD representations are 3D pipe routing, topological 3D path planning for robotic operations, aerial drone navigation, generation of new automotive cooling system configurations, 3D integrated circuit interconnect technology, and many others. In this paper, the general concept of 3D spatial topology enumeration using spatial graphs is discussed. In the future, we plan to demonstrate our framework using a real world engineering design example. We hope that this initial work serves as a foundation to bridging the gap between engineering design and mathematical low-dimensional topology. There are many interesting aspects which are yet to be explored and can have a great impact when applied to classical engineering design problems.

ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation Engineering Research Center (NSF ERC) for Power Optimization of Electro-Thermal Systems (POETS) with cooperative agreement EEC-1449548. Dunfield was partially supported by NSF grant DMS-1811156 and the Simons Foundation. The authors also thank Kyle Miller from the University of California, Berkeley for his helpful discussions on topics related to spatial graphs and Yamada polynomials.

References

- [1] Goode, H. H., and Machol, R. E., 1957. *System engineering: an introduction to the design of large-scale systems*. McGraw-Hill New York.
- [2] Sydenham, P. H., 2003. *Systems approach to engineering design*. Artech House Boston, MA.
- [3] Field, B. W., 2006. *Introduction to engineering design*. University of Melbourne, Dept. of Mechanical Engineering, Clayton, Vic.
- [4] Ashrafioun, H., and Mani, N. K., 1990. "Analysis and optimal design of spatial mechanical systems". *J. Mech. Des.*, **112**(2), June, pp. 200–207.
- [5] Bayrak, A. E., Ren, Y., and Papalambros, P. Y., 2016. "Topology generation for hybrid electric vehicle architecture design". *J. Mech. Des.*, **138**(8), June.
- [6] Ai, X., and Anderson, S., 2005. An electro-mechanical infinitely variable transmission for hybrid electric vehicles. doi: [10.4271/2005-01-0281](https://doi.org/10.4271/2005-01-0281)
- [7] Ramdan, M. I., and Stelson, K. A., 2016. "Optimal design of a power-split hybrid hydraulic bus". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, **230**(12), Jan., pp. 1699–1718.
- [8] Herber, D. R., and Allison, J. T., 2019. "A problem class with combined architecture, plant, and control design applied to vehicle suspensions". *J. Mech. Des.*, **141**(10), May.
- [9] Peddada, S. R. T., Herber, D. R., Pangborn, H. C., Alleyne, A. G., and Allison, J. T., 2019. "Optimal Flow Control and Single Split Architecture Exploration for Fluid-Based Thermal Management". *Journal of Mechanical Design*, **141**(8), 04. doi: [10.1115/1.4043203](https://doi.org/10.1115/1.4043203)
- [10] Field, B. W., 1997. *Introduction to design processes*, 2nd ed. Department of Mechanical Engineering, Monash University Clayton, Vic.
- [11] Challenger, S., 2000. "Systems Thinking, Systems Practice. By Peter B. Checkland. Published by John Wiley, Chichester, UK, 1981, 330 pp., (republished 1999 in paperback, with a 30-year retrospective). From a practitioner perspective". *Systems Research and Behavioral Science*, **17**(S1), pp. S78–S80.
- [12] Wyatt, D. F., Wynn, D. C., and Clarkson, P. J., 2013. "A scheme for numerical representation of graph structures in engineering design". *J. Mech. Des.*, **136**(1), Nov.
- [13] Schmidt, L. C., Shetty, H., and Chase, S. C., 1999. "A graph grammar approach for structure synthesis of mechanisms". *J. Mech. Des.*, **122**(4), July, pp. 371–376.
- [14] Moguel, E., Conejero, J. M., Sánchez-Figueroa, F., Hernández, J., Preciado, J. C., and Rodríguez-Echeverría, R., 2017. "Towards the use of unmanned aerial systems for providing sustainable services in smart cities". *Sensors (Basel, Switzerland)*, **18**(29280984), Dec., p. 64.
- [15] Yu, J., Cao, Z., Cheng, M., and Pan, R., 2018. "Hydro-mechanical power split transmissions: Progress evolution and future trends". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, **233**(3), Jan., pp. 727–739.

- [16] Kyprianidis, K., 2017. “An approach to multi-disciplinary aero engine conceptual design”. In International Symposium on Air Breathing Engines, ISABE 2017, Manchester, United Kingdom.
- [17] Park, S., and Jung, D., 2010. “Design of vehicle cooling system architecture for a heavy duty series-hybrid electric vehicle using numerical system simulations”. *J. Eng. Gas Turbines Power*, **132**(9), June.
- [18] Flapan, E., Mattman, T. W., Mellor, B., Naimi, R., and Nikkuni, R., 2017. “Recent developments in spatial graph theory”. In *Knots, links, spatial graphs, and algebraic invariants*, Vol. 689 of *Contemp. Math.* Amer. Math. Soc., Providence, RI, pp. 81–102. [arXiv:1602.08122](https://arxiv.org/abs/1602.08122).
- [19] Taylor, S. A., 2019. Abstractly planar spatial graphs. [arXiv:1902.01719](https://arxiv.org/abs/1902.01719).
- [20] Flapan, E., Mellor, B., and Naimi, R., 2012. “Spatial graphs with local knots”. *Revista Matemática Complutense*, **25**, pp. 493–510.
- [21] Hoste, J., Thistlethwaite, M., and Weeks, J., 1998. “The first 1,701,936 knots”. *Math. Intelligencer*, **20**(4), pp. 33–48.
- [22] Liang, C., and Mislou, K., 1994. “Classification of topologically chiral molecules”. *J. Math. Chemistry*, **15**(1), pp. 245–260.
- [23] Flapan, E., and Fletcher, W., 2013. “Intrinsic chirality of multipartite graphs”. *J. Math. Chemistry*, **51**(7), pp. 1853–1863.
- [24] Flapan, E., 1995. “Rigidity of graph symmetries in the 3-sphere”. *J. Knot Theory Ramifications*, **4**(3), pp. 373–388.
- [25] Mellor, B., 2018. Invariants of spatial graphs. [arXiv:1812.08885](https://arxiv.org/abs/1812.08885).
- [26] Fleming, T., and Mellor, B., 2007. “An introduction to virtual spatial graph theory”. In Proceedings of the International Workshop on Knot Theory for Scientific Objects, A. Kawachi, ed., Vol. 1 of *OCAMI Studies*, Osaka Municipal Universities Press.
- [27] Rapenne, G., Crassous, J., Echegoyen, L. E., Echegoyen, L., Flapan, E., and Diederich, F., 2000. “Regioselective one-step synthesis and topological chirality of trans-3, trans-3,trans-3 and e,e,e [60]fullerene-cyclotrimeratrylene tris-adducts: Discussion on a topological meso-form”. *HCA*, **83**(6), June, pp. 1209–1223.
- [28] Trace, B., 1983. “On the Reidemeister moves of a classical knot”. *Proc. Amer. Math. Soc.*, **89**(4), pp. 722–724.
- [29] Hass, J., and Lagarias, J., 1998. “The number of Reidemeister moves needed for unknotting”. *J. Amer. Math. Soc.*, **14**, pp. 399–428.
- [30] Hayashi, C., 2005. “The number of Reidemeister moves for splitting a link”. *Mathematische Annalen*, **332**(2), pp. 239–252.
- [31] Hass, J., Lagarias, J. C., and Pippenger, N., 1999. “The computational complexity of knot and link problems”. *J. ACM*, **46**(2), pp. 185–211.
- [32] Lackenby, M., to appear. “The efficient certification of knottedness and Thurston norm”. *Adv. Math.* [arXiv:1604.00290](https://arxiv.org/abs/1604.00290).
- [33] Ishii, A., 2011. “On normalizations of a regular isotopy invariant for spatial graphs”. *International J. Math.*, **22**, pp. 1545–1559.
- [34] Negami, S., 1987. “Polynomial invariants of graphs”. *Trans. Amer. Math. Soc.*, **299**, pp. 601–622.
- [35] Cho, S., and Koda, Y., 2011. “Topological symmetry groups and mapping class groups for spatial graphs”. *Michigan Math. J.*, **62**, pp. 131–142.
- [36] Flapan, E., Henrich, A., Kaestner, A., and Nelson, S., eds., 2017. *Knots, links, spatial graphs, and algebraic invariants*, Vol. 689 of *Contemporary Mathematics*, American Mathematical Society, Providence, RI.
- [37] Bar-Natan, D., 1995. “On the Vassiliev knot invariants”. *Topology*, **34**, pp. 423–472.
- [38] Kauffman, L., 1988. “New invariants in the theory of knots”. *Amer. Math. Monthly*, **95**, pp. 195–242.
- [39] Thompson, A., 1992. “A polynomial invariant of graphs in 3-manifolds”. *Topology*, **31**, pp. 657–665.
- [40] Alexander, J. W., 1928. “Topological invariants of knots and links”. *Trans. Amer. Math. Soc.*, **30**(2), pp. 275–306.
- [41] Murasugi, K., 1987. “Jones polynomials and classical conjectures in knot theory”. *Topology*, **26**, pp. 187–194.
- [42] Kauffman, L., 1989. “Invariants of graphs in three-space”. *Trans. Amer. Math. Soc.*, **311**, pp. 697–710.
- [43] Dobrynin, A., and Vesnin, A., 2003. “On the yoshinaga polynomial of spatial graphs”. *Kobe J. Math.*, **20**, pp. 31–37.
- [44] Yokota, Y., 1996. “Topological invariants of graphs in 3-space”. *Topology*, **35**, pp. 77–87.
- [45] Mellor, B., Kong, T., Lewald, A., and Pigrish, V., 2016. “Colorings, determinants and Alexander polynomials for spatial graphs”. *J. Knot Theory Ramifications*, **25**(4), pp. 1650019, 19. [arXiv:1506.06083](https://arxiv.org/abs/1506.06083).
- [46] Murakami, J., 1993. “The Yamada polynomial of spacial graphs and knit algebras”. *Comm. Math. Physics*, **155**, pp. 511–522.
- [47] Yamada, S., 1989. “An invariant of spatial graphs”. *J. Graph Theory*, **13**(5), Nov, pp. 537–551.
- [48] Vesnin, A., and Dobrynin, A., 1996. “The Yamada polynomial for graphs, embedded knot-wise into three-dimensional space”. *Vychislitel'nye Sistemy*, **155**, Jan.
- [49] Li, M., Lei, F., Li, F., and Vesnin, A., 2018. “The Yamada polynomial of spatial graphs obtained by edge replacements”. *J. Knot Theory Ramifications*, **27**(9), pp. 1842004, 19. [arXiv:1801.09075](https://arxiv.org/abs/1801.09075).
- [50] Deng, Q., Jin, X., and Kauffman, L. H., 2019. “The generalized Yamada polynomials of virtual spatial graphs”. *Topology Appl.*, **256**, pp. 136–158. [arXiv:1806.06462](https://arxiv.org/abs/1806.06462).
- [51] Hopcroft, J., and Tarjan, R., 1974. “Efficient planarity testing”. *J. Assoc. Comput. Mach.*, **21**, pp. 549–568.
- [52] Burton, B. A., 2020. “The Next 350 Million Knots”. In 36th International Symposium on Computational Geometry

- (SoCG 2020), S. Cabello and D. Z. Chen, eds., Vol. 164 of *Leibniz International Proceedings in Informatics (LIPIcs)*, Schloss Dagstuhl–Leibniz-Zentrum für Informatik, pp. 25:1–25:17.
- [53] Oyamaguchi, N., 2015. “Enumeration of spatial 2-bouquet graphs up to flat vertex isotopy”. *Topology and its Applications*, **196**, pp. 805–814.
 - [54] Kanenobu, T., and Sugita, K., 2012. “Finite type invariants of order 3 for a spatial handcuff graph”. *Topology and its Applications*, **159**, pp. 966–979.
 - [55] Moriuchi, H., 2008. “Enumeration of algebraic tangles with applications to theta-curves and handcuff graphs”. *Kyungpook Math. J.*, **48**, pp. 337–357.
 - [56] Moriuchi, H., 2009. “A table of θ -curves and handcuff graphs with up to seven crossings”. In *Noncommutativity and singularities*, Vol. 55 of *Adv. Stud. Pure Math.* Math. Soc. Japan, Tokyo, pp. 281–290.
 - [57] Soma, T., 1996. “Spatial-graph isotopy for trivalent graphs and minimally knotted embeddings”. *Topology and its Applications*, **73**, pp. 23–41.
 - [58] Fominykh, E., Garoufalidis, S., Goerner, M., Tarkaev, V., and Vesnin, A., 2016. “A census of tetrahedral hyperbolic manifolds”. *Exp. Math.*, **25**(4), pp. 466–481.