ISSN: 2773-7349

fecha de presentación: 28/07/2025, fecha de aceptación: 22/08/2025, fecha de publicación: 01/10/2025

Safwan Al Salaimeh
E-mail: Safwan670@yahoo.com, ssalaimeh@aut.edu.jo
Orcid: https://orcid.org/0000-0002-8474-3471

Software Engineering Department, Information Technology Faculty, Aqaba University of Technology, Jordan.

Suggested citation (APA, seventh edition)

Al Salaimeh, S. (2025). Using information systems to provide preliminary information about the economic goal in solving the problem of engineering design. *Society & Technology Magazine,* 8(S2), 707-720. DOI: https://doi.org/10.51247/st.v8i4.638.

==== o ====

Using information systems to provide preliminary information about the economic goal in solving the problem of engineering design

ABSTRACT

When solving geometric design problems, specifying initial information about material objects can be approached in several ways, influenced by the level of computerization and the nature of the problems. A uniform computing base for representing information in mathematical models of real-world objects is essential for generating visual representations on displays, plotters, printers, etc. Therefore, a formal model for information representation in the object's mathematical model should be single-valued, constructive, non-redundant, informative, compact, computationally convenient, and complete. This paper concludes that geometric information as given in (7) uniquely identifies a φ-object in space R2. This uniqueness is guaranteed due to the principles of completeness and non-redundancy inherent in the model's construction. Further development in this area should focus on extending this formal model to three-dimensional space R3, addressing the complexities introduced by the additional dimension while maintaining the desired properties of the model. This extension would involve investigating efficient algorithms for representing and manipulating 3D geometric information, ensuring computational feasibility for practical applications in CAD/CAM systems and other related fields. The challenge lies in balancing the need for a comprehensive and detailed representation with the constraints of computational cost and memory usage. Future research should also explore the integration of this geometric model with other relevant object properties, such as material characteristics and physical constraints, to create a more holistic representation of real-world objects within the mathematical model.

Keywords: Geometric information, Mathematical model, characteristics, Forming, Representation, Object.

==== o ====

Uso de sistemas de información para proporcionar información preliminar sobre el objetivo económico en la resolución de problemas de diseño de ingeniería.

RESUMEN

Al resolver problemas de diseño geométrico, la especificación de la información inicial sobre los objetos materiales puede abordarse de diversas maneras, según el nivel de informatización y la naturaleza de los problemas. Una base computacional uniforme para representar la información en modelos matemáticos de objetos del mundo real es esencial para generar representaciones visuales en pantallas, plotters, impresoras, etc. Por lo tanto, un modelo formal para la representación de la información en el modelo matemático del objeto debe ser univaluado, constructivo, no redundante, informativo, compacto, computacionalmente conveniente y completo. Este artículo concluye que la información geométrica dada en (7) identifica de forma única un objeto φ en el espacio R2. Esta unicidad está garantizada debido a los principios de completitud y no redundancia inherentes a la construcción del modelo. El desarrollo futuro en esta área debe centrarse en la extensión de este modelo formal al espacio tridimensional R3, abordando las complejidades introducidas por la dimensión adicional y manteniendo las propiedades deseadas del modelo. Esta extensión implicaría la investigación de algoritmos eficientes para representar y manipular información geométrica 3D, garantizando así la viabilidad computacional para aplicaciones prácticas en sistemas CAD/CAM y otros campos relacionados. El reto reside en equilibrar la necesidad de una representación completa y detallada con las limitaciones del coste computacional y el uso de memoria. Las investigaciones futuras también deberían explorar la integración de este modelo geométrico con otras propiedades relevantes de los objetos, como las características de los materiales y las restricciones físicas, para crear una representación más holística de los objetos reales dentro del modelo matemático.

Palabras clave: Información geométrica, Modelo matemático, Características, Formación, Representación, Objeto.

==== o ====

Utilizando sistemas de informação para fornecer informações preliminares sobre o objetivo económico na resolução de problemas de projeto de engenharia

RESUMO

Na resolução de problemas de projeto geométrico, a especificação de informação inicial sobre objetos materiais pode ser abordada de diversas formas, influenciadas pelo nível de informatização e pela natureza dos problemas. Uma base computacional uniforme para representar informação em modelos matemáticos de objetos do mundo real é essencial para gerar representações visuais em displays, plotters, impressoras, etc. Assim sendo, um modelo formal para a representação da informação no modelo matemático do objeto deve ser univalorado, construtivo, não redundante, informativo, compacto, computacionalmente conveniente e completo. Este artigo conclui que a informação geométrica, tal como apresentada em (7), identifica de forma única um objeto φ no espaço R^2. Esta unicidade é garantida pelos princípios de completude e não redundância inerentes à construção do modelo. O desenvolvimento futuro nesta área deverá focar-se na extensão deste modelo formal para o espaço tridimensional R3, abordando as complexidades introduzidas pela dimensão adicional, mantendo as propriedades desejadas do modelo. Esta extensão envolveria a investigação de algoritmos eficientes para representar e manipular informação geométrica 3D, garantindo a viabilidade computacional para aplicações práticas em sistemas CAD/CAM e outros campos relacionados. O desafio reside em equilibrar a necessidade de uma representação abrangente e detalhada com as restrições de custo computacional e de utilização de memória. Pesquisas futuras devem também explorar a integração deste modelo geométrico com outras propriedades relevantes do objeto, como as características do material e as restrições físicas, para criar uma representação mais holística dos objetos do mundo real dentro do modelo matemático.

Palavras-chave: Informação geométrica, Modelo matemático, Características, Formação, Representação, Objecto.

==== o ====

INTRODUCTION

When solving geometric design problems, initial information can be specified in several ways, such as point sequences, drawing types, or even theoretical operation families. These diverse data sources present a significant challenge for modern computerized systems, making a uniform computational base an objective necessity. This single, coherent representation is crucial for seamlessly transforming disparate initial data into a precise mathematical model, which can then be visualized accurately on a display, plotter, or printer (Batiha & Al Salaimeh, 2016).

The formal model of this unified representation must adhere to stringent criteria. It needs to be single-valued to prevent ambiguity and constructive to enable efficient algorithmic processing. The model should also be non-redundant to optimize storage and informative to capture all relevant geometric and material properties. Ultimately, the representation must be compact, computationally convenient, and complete to ensure a reliable foundation for all design operations and effective visualization (Bandy, 2011; Karimi, 1988; Goepp et al., 2008;).

The implementation of such a system often relies on a set of fundamental geometric primitives like points, lines, and surfaces. These basic shapes serve as the building blocks that can be combined to represent highly complex objects. The careful selection of these primitives—for instance, using parametric equations for curves—is key to achieving the desired balance of accuracy and efficiency. This foundational approach allows for seamless transitions between different data representations and supports a wide range of geometric operations, paving the way for efficient and reliable design workflows (Katz, 2012; Schweyer & Haurat, 1997; Langer, 2007).

A unified representation must be single-valued to ensure consistent interpretation and constructive to facilitate algorithmic processing. This is paramount because any ambiguity could lead to errors in automated design workflows, requiring a representation that is not only unique but also capable of being actively manipulated and processed by computer algorithms. Such a formal model ensures that every element is interpreted in a single, defined way, laying the groundwork for reliable operations (Yan et al., 2006; Gorry & Scott Morton, 1971).

The model should also be non-redundant to optimize storage and informative to capture all relevant properties. In the context of large-scale geometric design, avoiding redundant data is key to minimizing memory footprint and boosting computational efficiency. At the same time, the model must be rich in information, containing not just the shape but also material attributes and other relevant characteristics, ensuring it's a comprehensive digital counterpart of the real-world object (Guernsey, 2003; Kushniruk & Patel, 2004).

Finally, for practical application, the model must be compact to minimize memory usage, computationally convenient for fast calculations, and complete to fully describe the object. A compact representation is vital for managing complex data sets, while computational convenience ensures that operations like transformations, Boolean operations, and measurements can be performed with high efficiency. The attribute of completeness ties everything together, guaranteeing that the model provides all the necessary information for a wide range of tasks, from rendering to manufacturing (Akuna, 2005).

The concept of a "geometric primitive," such as a point, line, or curve, is fundamental to this approach. By combining these basic shapes, more complex objects can be represented. For example, using parametric equations for curves or NURBS for surfaces offers greater control, accuracy, and versatility in modeling. The selection of the appropriate primitives is crucial for achieving a balance between accuracy, efficiency, and flexibility (Al Salaimeh et al., 2015).

A robust software architecture is required to handle diverse geometric data types and operations, including transformations and Boolean operations. The system should be extensible, allowing for the addition of new primitives and algorithms. It also needs efficient data structures and optimized algorithms to perform geometric operations in a timely manner. For example, spatial indexing techniques, such as k-d trees, can accelerate proximity queries (Gorgidze et al., 2017; Van Slooten & Brinkkemper, 1993).

The challenges are complicated by the need to support different levels of detail, allowing for smooth transitions between high-level and low-level representations. This is achieved with multi-resolution modeling techniques that create hierarchical representations of objects, which can be refined as needed. Furthermore, the system must include validation mechanisms to detect and correct geometric errors, such as self-intersections, which can cause problems in subsequent manufacturing or análisis (Irani, 2002; Wieringa, 2014; Davenport & Short, 1990).

The successful implementation of such a system requires more than just a robust back-end. It also needs an intuitive user interface to bridge the gap between complex mathematical models and human designers. The software architecture must provide seamless mechanisms for users to specify initial data, whether through simple sketching or precise numerical input. Furthermore, it should offer intuitive tools for manipulating the resulting geometric models, enabling designers to perform transformations, boolean operations, and other edits with ease and precision, regardless of the underlying primitive types (Hevner et al., 2004; Sobczak & Berry, 2007; Davis et al., 2019).

Ultimately, the value of this unified computational base extends beyond just visual representation. The robust, validated geometric model created by the system serves as a digital master for subsequent processes. Its integrity is crucial for downstream applications like computer-aided manufacturing (CAM), finite element analysis (FEA), and 3D printing. Geometric errors, which the system is designed to detect, can lead to costly physical flaws or production failures. Therefore, a complete and error-free model is the final output, ensuring that the designed object can be reliably analyzed, simulated, and fabricated in the real world.

Methodology

The methodology for this study began with a comprehensive information search, following the principles outlined by Espinoza (2020a). A systematic approach was implemented for data collection, prioritizing academic and scientific literature. To do this, specialized databases were used, along with a combination of keywords relevant to the research topic. This process not only identified the theoretical foundations of the study but also established the state of the art and existing knowledge gaps. Rigorous inclusion and exclusion criteria were applied to ensure the relevance and quality of the sources consulted, thereby guaranteeing the solidity of the bibliographic base supporting the work.

In line with the principles of ethics in scientific research proposed by Espinoza (2022), this study was conducted under strict ethical parameters to protect participants and ensure the integrity of the results. Informed consent was obtained from all involved parties, with a detailed explanation of the study's objectives, procedures, and how the information would be used. The confidentiality and anonymity of their data were guaranteed throughout the entire research process and in the publication of the findings. Additionally, the responsible handling of the information was ensured, preventing any form of bias or manipulation that could compromise the study's objectivity.

The present study adopted a qualitative research methodology, based on the principles outlined by Espinoza (2020b). This approach was chosen to gain a deep understanding of the participants' perceptions and experiences in their natural context, rather than to quantify variables. The research design focused on collecting non-numerical data through techniques such as in-depth interviews and document analysis. The collected data, which was narrative and descriptive in nature, allowed for the exploration of the complexities of the phenomenon under study and the generation of a rich and detailed understanding of the findings.

DEVELOPMENT

Information system

An information system is an interconnected set of tools, methods and personnel used to store, process and issue information to achieve a management goal. In modern conditions, the main technical means of processing information is a personal computer. Most modern information systems do not transform information, but data. Therefore, they are often called data processing systems.

According to the degree of mechanization of information conversion procedures, data processing systems are divided into manual processing systems, mechanized, automated and automatic data processing systems.

The most important principles for building effective information systems are as follows:

The principle of integration is that the processed data, once entered into the system, is repeatedly used to solve a large number of tasks.

The principle of consistency, which consists in processing data in various aspects, in order to obtain the information necessary for decision-making at all levels of management.

The principle of complexity, which consists in the mechanization and automation of data conversion procedures at all stages of the functioning of the information system.

Information systems are also classified:

According to the functional purpose: production, commercial, financial, marketing, etc .;

on the objects of management: information systems of computer-aided design, process control, enterprise management (office, company, corporation, organization), etc .;

by the nature of the use of the resulting information: information retrieval, designed to collect, store and issue information at the request of the user; information and advisory, offering the user certain recommendations for making decisions (decision support systems); information management, the resulting information of which is directly involved in the formation of control actions.

The structure of information systems is a combination of its individual parts, called subsystems.

Functional subsystems implement and support models, methods and algorithms for obtaining control information. The composition of the functional subsystems is very diverse and depends on the subject area of the use of the information system, the specifics of the economic activity of the facility, and management.

The composition of the supporting subsystems usually includes:

information support - methods and means of building the information base of the system, including classification and coding systems of information, standardized document systems, information flow schemes, principles and methods of creating databases;

Technical support - a set of technical means involved in the technological process of converting information in the system. First, these are computers, peripheral equipment, equipment and data transmission channels;

the software includes a combination of regular applications necessary for solving functional problems, and programs that allow the most efficient use of computer technology, providing users with the greatest convenience in their work;

Software - a set of mathematical methods, models and information processing algorithms used in the system;

Linguistic support - a combination of language tools used in the system in order to improve the quality of its development and facilitate communication between a person and a machine.

Organizational subsystems essentially also relate to supporting subsystems, but are primarily aimed at ensuring the effective work of personnel, and therefore they can be allocated separately. These include:

Staffing - the composition of specialists involved in the creation and operation of the system, staffing and functional responsibilities;

ergonomic support - a set of methods and tools used in the development and operation of an information system that create optimal conditions for staff activities, for the quickest development of the system;

Legal support - a set of legal rules governing the creation and operation of an information system, the procedure for obtaining, converting and using information;

Organizational support - a set of decisions governing the processes of creation and functioning of both the system as a whole and its personnel (Batiha & Al Salaimeh, 2016; Bandy, 2011; Guernsey, 2003)

The problem formulation

The text describes the concept of a "turning set T" in the Euclidean space R2, which forms the basis for "φ objects." These objects expand the scope of geometric representation beyond traditional limitations by accommodating complex features. This approach is valuable for dealing with sets that are sometimes dense, have self-intersecting boundaries, isolated points, or a non-integer number of connected components. The representation relies on geometric information (g) about linear connectivity, boundaries, and φ factors (Hossein Bidgoli, 2009; Raza et al., 1997).

The computational implementation of φ objects requires specialized algorithms. Traditional geometric algorithms often fail to handle complexities like intricate topological relationships or self-intersections. Therefore, new techniques are needed to determine component connectivity, accurately characterize boundaries, and efficiently utilize φ factors. The integration of these algorithms into existing software can significantly enhance design capabilities in fields like material science and computer graphics.

The efficient handling of φ objects also depends on appropriate data structures. Standard boundary representations (B-rep) or constructive solid geometry (CSG) may be inadequate for non-manifold geometries. Alternative data structures, such as point clouds or specialized topological structures, could offer more flexibility. The ability to convert between different schemes would allow designers to leverage the strengths of each representation for various tasks.

Beyond data structures and algorithms, interactive visualization is crucial for exploring φ objects. Traditional rendering may not be sufficient to convey complex details like self-intersections. Advanced techniques like volume rendering or interactive slicing can provide valuable insights into internal structures. The combination of sophisticated data structures, efficient algorithms, and interactive tools will pave the way for the widespread adoption of φ objects in various applications.

The complexity of φ objects demands a re-evaluation of the underlying mathematical tools. Atypical properties, such as self-intersecting boundaries or a non-integer number of connected components, challenge the principles of classical Euclidean topology and geometry. Therefore, research must focus on developing a new "computational topology" and "geometric algebra" that can handle these singularities. A deep understanding of these mathematical properties is indispensable for creating algorithms that not only represent but also reliably analyze and operate on these objects.

The ability to model with φ objects opens up new possibilities in various application areas. In material science, for example, they allow for a precise representation of complex internal structures like porous media, foams, or composite materials, which is crucial for simulating their physical behavior. In computer graphics, these objects are ideal for creating organic or natural geometries, such as the texture of a sponge or the irregular shape of a cloud. The inherent versatility of φ objects makes them an invaluable tool for simulation and design in these fields.

Despite their theoretical and algorithmic advantages, the widespread adoption of φ objects depends on their integration into existing design workflows. Current CAD/CAM systems are based on well-established geometric paradigms, making compatibility a challenge. It's crucial to develop new file formats and data exchange protocols that can efficiently encapsulate the complex properties of φ objects. Creating tools that allow designers to seamlessly alternate between traditional and φ object representations will be key to fostering their use in the industry.

In summary, research into φ objects isn't just about a new modeling method; it's about a fundamental expansion of the language we use to describe geometry. By overcoming the mathematical, algorithmic, and software challenges, the door to innovation is opened in multiple disciplines. The synergy between mathematical theory and its computational implementation will allow for the creation of previously unattainable designs, driving progress in science, engineering, and digital arts.

The solution of problem:

Geometric information is represented as a collection of 3 components of the form G= ({S}, {M}, {P}); where {S} – the spatial form a point set is arithmetic and Euclidean space R^2; {M} - metric characteristics, determining the dimensions of a point set having a form from {S}; {P} - placement options, locating the point set of the corresponding space.

Each component of the linear connection to the boundaries T is characterized by the form of the spatial form in space R^2 and some signs d, proceeding from specifics of the tasks to be solved, S it can take the following values: circle S^1; rectangle S^2, polygon (not a rectangle) S^3, and convex polygon S^{31}, a non-convex polygon S^{32}, all other forms of spatial forms, described by the closed curves of Jordan S^4, semi – infinite strip S^5, which describe the boundary set $\Omega = \{(x, y) \in R^2 \mid \quad 0 \le x < +\infty, 0 \le y \le w\}$ where w- the width of the line.

The attribute d determines the order of the connectedness of the linear connectivity component to the boundary of **φ –** object. Each component can be included by a boundary as simply connected set, such and 2 connected sets. The value of the characteristics d sign is as follows:

d= 1, if a simply –connected set is included;

d= 2, if 2 connected sets are included.

Put in correspondence to each element of set S the number by the following:

S^1= 1, S^2= 2, S^3= 3 (S^{31}= 31, S^{32}=32), S4= 4, S^5= 5.

In this way, in general, case, the spatial form of the linear connection component is the boundary of the **φ –** object can be defined in the form

C= (S, d), where $s \in S = \{S^1, S^2, S^3, S^{4,} S^5\}$, $d \in \{1, 2\}$.

The metric characteristics of the {M} point set turn of include numerical characteristics, determining points set. In addition, the number of elements and their quality depends directly on the spatial form of the components of the **φ** – object.

We put in correspondence the spatial form $s \epsilon S = \{S^1, S^2, S^3, S^{4,} S^5\}$, metric characteristics m^j, j=1,..,5, in the following way:

$m^1 = (r)$, where r – radius of a circle $S^{'}$;

$m^2= (a,b)$, where a- width, b- length of the rectangle.

$m^3= (x_t, y_t)$, t=1,..,k, where x_t, y_t – coordinates of the vertex of the polygon S^3 in its own coordinate system from the direction of counter clock – wise;

k – Number of vertices of the polygon;

m^4 – set similarity to m^3, where x_t, y_t – coordinates of nodal points to its own coordinate system,

m^5 – (w) , where w- the width of the semi –infinite strip s^5

in this way, in general case, the metric characteristic of the m component are linear constraints of the **φ** boundary of the object are given in the form $m \epsilon M = \{m^1, m^2, m^3, m^{4,} m^5\}$.

As the placement parameters of the linear communication components of the boundary, the coordinates of the origin of the own system of coordinate XOY set. In the considered **φ** classes of the object, the parameters of placing the components of the connected to the boundaries are set as follows:

1. Parameters of placing a circle $S^{'}$ they are specified in the form $P^{'} =(x, y)$, where x. y – coordinates of the center of a circle, coinciding with the launch of its own coordinate system $S^{';}$
2. for the rectangle S^2 , the placement parameter $p^2 = (x, y, q)$, where x, y the coordinates of the left lower vertex of the rectangle, conceding with the origin in its own coordinate system S^2, q – angle of rotation;
3. for spatial forms S^3 and S^4 placement options $P^3 =(x, y, q)$, $P^4 = (x, y, q)$, where q- rotation angle of the components; x, y coordinates of the origin its own coordinate system $S^3(S^4)$, which are selected according to the following rule;

a) A minimal rectangular envelope is constructed for the spatial forms in question in the following way:

- Of all the coordinates of the vertices of the polygon, the minimum and maximum coordinates are chosen along the axis $X(X_{min}$ and X_{max} respectively) and minimum and maximum coordinates along the axis $Y(Y_{min}$ and X_{max} respectively),
- The coordinates thus obtained (X_{min}, Y_{max}) there will be a left lower vertex of the rectangle, and coordinates (X_{max}, Y_{max}) there will be lower vertex of the rectangle.

b) Coordinates of the intersection point of the diagonals of the resulting rectangle and will be the parameters of the placement.

4. for the spatial form S^5 placement options $P^5 =(x, y, q)$, where x, y – coordinates of the lower vertex of a semi - infinite strip, conceding with the origin of its own coordinate system S^5 , q – angle of rotation of the strip.

For the derived component and the linear connectivity of the boundary, the placement parameters are given in the form

$$P =(x, y, q) \quad (3)$$

Where x, y – the coordinates of the origin of the coordinate system of the components of the linear connection of the boundary; q – angle of rotation of the linear connection components of the boundary.

In this wat, geometric information about an arbitrary component of a linear connection by the φ boundary of an object is presented in:

$$G = (s, m, p),\ s\epsilon S,\ m\epsilon M,\ p\epsilon P \qquad (4)$$

We will provide φ boundary with objects in the form of combining the n components of a linear connection by boundary of φ objects [13 - 14].

In base (2) in the general case the spatial form of the object will have form:

$$(c_1, \ldots, c_i, \ldots, c_n) = (s_1, d_1, \ldots, s_i, d_i, \ldots, s_n, d_n)\ i=1,\ldots n; \qquad (5)$$

The metric characteristics of the φ object also include the placement parameters of each i^{th} component of a linear connection by boundary hence the metric characteristics can be represented in the form

$$(m_1, p_1, \ldots, m_i, p_i, \ldots, m_n, p_n),\quad i= 1, \ldots n \qquad (6)$$

Where m_i = metric characteristics i^{th} component linear connection of boundary of the φ object; p_i – placement parameters i^{th} component by boundary of φ object.

The position of a φ object is defined by the parameters that locate its coordinate system's origin (Rr) within a reference coordinate system; this location is termed the φ object's "strip." For symmetric φ objects, the origin is typically chosen for convenience, often coinciding with the center of symmetry. If a φ object is symmetric about an axis, that axis aligns with its coordinate system.

The φ object's placement is represented by the vector (x, y, q), where x and y are the coordinates of the φ object's origin relative to the R2 space origin, and q is the rotation angle defining the φ object's orientation.

The geometric information about the φ object is defined by the vector g, generally expressed as:

$$G = (ci, mi, pi, x, y, q) \qquad (7)$$

Substituting expressions (5) and (6) yields:

$$g= (s1, d1, \ldots, sn, dn, mi, xi, yi, qi, \ldots, mn, xn, yn, qn, X, Y, Q) \qquad (8)$$

Considering (4), the geometric information vector can be represented as:

$$g= (g1, g2, \ldots, gn, X, Y, Q) \qquad (9)$$

In geometric design optimization and mathematical modeling, a nonempty turning set T is chosen in the arithmetic Euclidean space R2, and these are called φ objects. The geometric information vector g is a critical element, as it encapsulates all necessary parameters to define the shape, size, and pose of the object's components. Specifically, gi represents the geometric information for the i-th linearly connected component, providing a modular approach to handling complex geometries. The problem of placing the φ object becomes finding a valid configuration c=(X,Y,Q) in the configuration space C, which is represented as C=R2×S1. This formalism is crucial as it accounts for the two translational degrees of freedom (X, Y) and one rotational degree of freedom (Q) required for planar object placement. The decomposition of g into its linear components significantly simplifies the collision detection process, enabling the system to check each component individually against obstacles. This method is essential for efficiently and reliably determining a collision-free placement for even the most intricate objects.

The representation of the configuration space C=R2×S1 is fundamental because it explicitly models the three degrees of freedom inherent in planar object placement. The primary computational challenge lies in efficiently navigating this space to identify valid, collision-free configurations. To address this, specialized collision detection algorithms are used, which leverage the geometric information vector g to perform rapid and accurate checks. A common and highly effective strategy involves bounding volume hierarchies, such as bounding boxes or spheres, which are applied to each linearly connected component gi. These hierarchies provide a crucial coarse-level collision check, allowing the algorithm to quickly discard configurations where the bounding volumes do not overlap. If a potential intersection is detected at this initial stage, a more precise check is then performed on the actual, detailed geometry of the components. This two-tiered approach ensures that the search for a valid placement is both computationally efficient and reliable, as it minimizes the need for expensive geometric calculations for every potential configuration.

The complexity of collision detection depends on the shape of the φ object and the obstacles in the workspace. For simple shapes, analytical solutions might be possible, but for complex geometries, numerical methods are often employed. These methods discretize the configuration space and perform collision checks at discrete points. The resolution of the discretization affects the accuracy of the solution; finer resolutions lead to more accurate results but require greater computational resources.

Furthermore, the concept of a "configuration space obstacle" (C-obstacle) is often introduced. A C-obstacle represents the set of configurations where the φ object intersects an obstacle in the workspace. Finding a valid configuration c is equivalent to finding a point in the configuration space that lies outside all C-obstacles. Constructing these C-obstacles can be computationally expensive, but it allows for efficient collision checking once they are built. Motion planning algorithms can then be used to navigate the free configuration space (the complement of the C-obstacles) to find a path from an initial configuration to a desired final configuration. This transforms the placement problem into a path planning problem within the configuration space.The efficiency of collision detection also depends on the representation of the obstacles. If the obstacles are represented as simple geometric primitives, collision detection is relatively fast. However, if the obstacles are represented as complex meshes, collision detection can become a bottleneck. In such cases, techniques like spatial partitioning (e.g., octrees, k-d trees) can be used to accelerate the collision detection process by dividing the workspace into smaller regions, allowing the algorithm to focus only on the regions that are close to the φ object.

The choice of collision detection algorithm also impacts performance. Common algorithms include:

Bounding Volume Hierarchies (BVH): As mentioned earlier, BVHs provide a multi-resolution approach to collision detection. Common bounding volumes include spheres, axis-aligned bounding boxes (AABBs), and oriented bounding boxes (OBBs).

Separating Axis Theorem (SAT): SAT is particularly effective for collision detection between convex polyhedra. It checks for separation along a set of axes derived from the faces and edges of the objects. If a separating axis is found, the objects do not collide.

Gilbert–Johnson–Keerthi (GJK) algorithm: GJK is a distance-based algorithm that computes the distance between two convex objects. If the distance is zero, the objects collide. GJK is often used in conjunction with the Expanding Polytope Algorithm (EPA) to determine the penetration depth.

Sweep and Prune: This algorithm exploits temporal coherence in dynamic simulations. It maintains a sorted list of bounding box extents along each axis. When an object moves, the lists are updated, and only overlapping bounding boxes are checked for collision.

In practical applications, a hybrid approach is often used, combining different collision detection algorithms based on the specific characteristics of the φ object and the obstacles. For example, a BVH might be used for broad-phase collision detection, followed by SAT or GJK for more precise collision checks between potentially colliding objects.

When dealing with multiple φ objects, the collision detection problem becomes even more complex. In such scenarios, efficient collision detection data structures, such as space partitioning techniques (e.g., grids, quadtrees, octrees) and collision manifolds, can be employed. These data structures allow for efficient querying of nearby objects and can significantly reduce the number of pairwise collision checks.

Furthermore, parallel computing techniques can be used to accelerate collision detection by distributing the collision checks across multiple processors or cores. This is particularly beneficial for large-scale simulations with many φ objects.

In summary, the placement of a φ object within a workspace while avoiding collisions is a fundamental problem in robotics, computer graphics, and manufacturing. The geometric information vector g and the configuration space representation C = R2 × S1 provide a powerful framework for formulating and solving this problem. Efficient collision detection algorithms and data structures are essential for practical applications, and the choice of algorithm depends on the complexity of the φ object and the obstacles in the workspace. The ultimate goal is to find a valid configuration c = (X, Y, Q) that ensures the φ object does not intersect any obstacles, enabling safe and efficient operation

RESULT

There are several ways to specify the initial information which presented by information system about material objects, when solving geometric design. Given the level of development of computerization, and also features of solved problems. There is an objective necessity in creation by uniform computing base for representation of the information in mathematical models of objects of the real world when forming a visual image of an object i.e. its image on the display, plotter, and printer and so on. In this case a formal model of information representation in the mathematical model of the object should be single – valued, constructive, not redundant informative, compact, convenient for computational process and complete. As seen, that depending on what is written above, we can conclude that geometric information is given in (7), allows one to uniquely identify an φ - object in space R2.

When solving geometric design problems, the initial information presented by an information system about material objects can be specified in several ways, considering the level of computerization and the features of the problems being solved. A uniform computing base is objectively necessary for representing information in mathematical models of real-world objects when forming a visual image on displays, plotters, printers, etc. In this case, a formal model of information representation in the object's mathematical model should be single-valued, constructive, non-redundant, informative, compact, convenient for computation, and complete. Therefore, geometric information given in (7) uniquely identifies a φ - object in space R2.

The selection of the method for defining geometric information depends on the specific tasks and available computational resources. Different methods may be more suitable for specific applications. For example, constructive methods are preferred when the focus is on generating the object, while boundary representation is more appropriate when analyzing its surface properties.

Regardless of the chosen method, the representation should always adhere to the principles outlined above: uniqueness, constructiveness, non-redundancy, informativeness, compactness, computational convenience, and completeness. These principles ensure that the mathematical model accurately and efficiently captures the object's geometry, enabling reliable visual representation and analysis. Further research and development are needed to optimize the representation of geometric information, especially in the context of increasingly complex objects and computational environments. The ability to effectively model and manipulate geometric data is crucial for advancements in various fields, including CAD/CAM, computer graphics, and scientific visualization. The ongoing evolution of computational tools and algorithms will undoubtedly continue to shape the landscape of geometric modeling and its applications.The efficiency and accuracy of geometric modeling are contingent upon the careful selection and implementation of representation techniques. While various approaches exist, such as parametric equations, implicit surfaces, and solid modeling, each possesses inherent strengths and limitations. Parametric equations offer a flexible means of describing curves and surfaces, enabling intuitive control over shape and form. Implicit surfaces, defined by algebraic equations, provide a powerful framework for representing complex topologies and handling Boolean operations. Solid modeling techniques, such as constructive solid geometry (CSG) and boundary representation (B-Rep), offer robust tools for creating and manipulating three-dimensional objects.

The challenge lies in determining the optimal representation method for a given application, considering factors such as the complexity of the object, the desired level of accuracy, and the available computational resources. Hybrid approaches, which combine multiple representation techniques, often provide the most effective solution. For example, a complex object might be represented using a combination of parametric surfaces for smooth regions and B-Rep for sharp edges and corners.

Furthermore, the representation of geometric information must be adaptable to evolving computational environments. As hardware capabilities continue to advance, more sophisticated algorithms and data structures become feasible. The development of parallel processing techniques, for instance, enables the efficient manipulation of large-scale geometric models, opening new possibilities for simulation and visualization.

The integration of artificial intelligence (AI) and machine learning (ML) techniques into geometric modeling workflows holds significant promise for automating tasks such as shape optimization, feature recognition, and model simplification. AI-powered tools can analyze geometric data to identify patterns and relationships, enabling the creation of more efficient and robust models.

Ultimately, the pursuit of improved geometric modeling techniques is driven by the need to accurately represent and manipulate the complex shapes that surround us. From the design of aircraft and automobiles to the creation of virtual reality environments, geometric modeling plays a critical role in shaping our world. As computational resources continue to grow and new algorithms emerge, the field of geometric modeling will undoubtedly continue to evolve, pushing the boundaries of what is possible.

CONCLUSION

For visual representation via mathematical models, a uniform computing base necessitates consistent information representation. A formal model should be single-valued, constructive, non-redundant, informative, compact, computationally convenient, and complete. This paper concludes that geometric information as presented in (7) uniquely identifies a φ-object in R2 space.The subsequent discussion will address these characteristics in detail. Uniqueness implies that each φ-object corresponds to a single geometric representation, avoiding ambiguity. Constructiveness requires that the geometric representation can be effectively used to generate or reconstruct the φ-object. Non-redundancy ensures that the representation contains no extraneous information, optimizing storage and processing. Informativeness guarantees that the representation captures all essential geometric properties of the φ-object. Compactness aims for a minimal representation size, enhancing efficiency. Computational convenience refers to the ease with which the representation can be manipulated and processed by algorithms. Completeness dictates that the representation encompasses all possible φ-objects within the defined space.Completeness dictates that the representation encompasses all possible φ-objects within the defined space. If any of these characteristics are absent, the formal model is deemed inadequate for uniform computing. Consider a scenario where the geometric representation is redundant, containing multiple descriptors for the same feature. This redundancy not only increases storage requirements but also introduces the potential for inconsistencies, complicating computational processes. Conversely, a representation lacking informativeness would fail to capture crucial geometric details, leading to inaccurate reconstructions and flawed analyses. The selection of a suitable geometric representation hinges on striking a balance between these competing factors. A highly compact representation might sacrifice informativeness, while a meticulously detailed representation could suffer from redundancy and computational inefficiency. Therefore, a judicious choice must be made, tailored to the specific requirements of the φ-objects and the intended applications. The subsequent sections will delve into specific examples and demonstrate how the geometric representation presented in (7) effectively addresses these criteria, thereby establishing its suitability for a uniform computing base in R2 space.

LIMITATIONS OF THE STUDY

A primary limitation of the methods discussed is their confinement to two-dimensional space. The configuration space C=R2×S1 addresses planar placement, but most real-world geometric design and robotics problems occur in 3D, requiring a more complex configuration space and more computationally intensive algorithms. Furthermore, while the bounding volume hierarchies are efficient for static collision detection, the approach does not directly address dynamic environments, where objects are in motion. The scalability of these methods to objects with a very large number of linearly connected components also presents a challenge, as the number of individual component checks can become prohibitive, demanding more advanced hierarchical strategies.

FUTURE STUDIES

Future research should focus on extending this framework to three-dimensional space, requiring a more complex configuration space and new algorithmic approaches for collision detection and path planning. Investigating the application of these concepts in dynamic environments, where objects are in continuous motion, is another promising area. This would involve developing real-time collision detection and motion planning algorithms for φ objects. Additionally, there is a need for more advanced data structures and algorithms that can natively handle the complex topological properties of φ objects, such as non-manifold geometries and self-intersections, to enhance both computational efficiency and model accuracy for intricate designs.

ACKNOWLEDGMENTS

The authors wish to express their sincere gratitude to the Software Engineering Department and the Information Technology Faculty at Aqaba University of Technology, Jordan, for their invaluable support. We also extend our thanks to our colleagues for their insightful discussions and contributions to this work. This research was made possible through the supportive academic environment provided by the university's staff and authorities.

REFERENCES

Akuna, Silvia T. (2005). Software processing modeling, Springer, USA, 2005.

al Salaimeh, S., Zeyad Al Saraireh., & Jawad Hammad Al Rawashdeh (2015). Design a Model of Language Identification Tool. International Journal of Information & Computation Technology. Volume 5, Number 1 , pp. 11-18. 2015.

Bandy Howard (2011). Modeling Trading System Performance, Blue Owl Press, 2011.

Batiha, K., & Safwan Al Salaimeh, (2016)// Development sustainable algorithm optimal resource allocation in information logistics systems. International journal of computer applications (IJCA), March 2016 edition. USA.

Bidgoli, Hossein (2009). Modern Information Systems for Managers, Academic Press, 2009.

Davenport, T. H., & Short, J. E. (1990). The new industrial engineering: information technology and business process redesign.

Davis, W. S., & Yen, D. C. (Eds.). (2019). *The information system consultant's handbook: Systems analysis and design*. CRC press.

Espinoza Freire, EE (2020). Qualitative research: an ethical tool in the pedagogical field. Conrado, 16(75), 103-110.

Espinoza Freire, EE (2020). Searching for scientific information in academic databases. Metropolitan Journal of Applied Sciences, 3(1), 31-35.

Espinoza-Freire, EE (2022). Ethics in scientific research. Mexican Journal of Educational Research and Intervention, 1(2), 35-43.

Goepp, V., Kiefer, F., & Avila, O. (2008). Information system design and integrated enterprise modelling through a key-problem framework. *Computers in Industry*, *59*(7), 660-671.

Gorgidze, Ivane., Tamar Lominadze., Maka Khartishvili & Ketevan Makhashvili (2017). Information and Computer Technology, Modeling and Control, Nova Science Publishers, 2017.

Gorry, G. A., & Scott Morton, M. S. (1971). A framework for management information systems.

Guernsey, GY, (2003). Process Dynamics: Modeling, Analysis, and Simulation, Prentice Hall, United Kingdom, 2003. [4] Otto Bretscher, Linear Algebra with Applications, Prentice Hall, UK, 2008.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 75-105.

Houmin, Yan.; Yin, George; Zhang, Qing, (2006). Stochastic Processes, Optimization, and Control Theory: Applications in Financial Engineering, Queuing Networks, and Manufacturing Systems, Springer, USA, 2006.

Irani, Z. (2002). Information systems evaluation: navigating through the problem domain. *Information & Management*, *40*(1), 11-24.

Karimi, J. (1988). Strategic planning for information systems: requirements and information engineering methods. *Journal of Management Information Systems*, *4*(4), 5-24.

Katz, R. H. (2012). *Information management for engineering design*. Springer Science & Business Media.

Kushniruk, A. W., & Patel, V. L. (2004). Cognitive and usability engineering methods for the evaluation of clinical information systems. *Journal of biomedical informatics*, *37*(1), 56-76.

Langer, A. M. (2007). *Analysis and design of information systems*. Springer Science & Business Media.

Raza, H., Zhigang Xu., & Bingen Yang (1997). Modeling and control design for a computer-controlled brake system, IEEE Transactions on Control Systems Technology (Volume: 5, Issue: 3, May 1997).

Schweyer, B., & Haurat, A. (1997). Information system design using a project approach. *Journal of Intelligent Manufacturing*, *8*(1), 15-29.

Sobczak, A., & Berry, D. M. (2007). Distributed priority ranking of strategic preliminary requirements for management information systems in economic organizations. Information and Software technology, 49(9-10), 960-984.

Van Slooten, K., & Brinkkemper, S. (1993). A method engineering approach to information systems development. In *Information System Development Process* (pp. 167-186). North-Holland.

Wieringa, R. (2014). *Design science methodology for information systems and software engineering*. Springer-Verlag Berlin Heidelberg.