Role of IoT and Cloud Computing in Automated Assembly Modeling Systems

Student Name: Karunya HariKrishnan
Register Number: 21011101059

Summary:

Assembly model is the digital representation of an assembled product of multiple parts and components that helps simulate and analyze the behaviour and functionality of the product. Assembly model systems are computer-aided design (CAD) systems that are specifically designed for creating, editing, and analyzing assembly models. They typically provide a wide range of tools and features for creating, editing, and analyzing assembly models. Examples include: SolidWorks, CATIA, and more.

IoT has the potential to greatly impact the manufacturing sector by providing a way to connect and collect data from various devices and machines within a factory. This data can then be used to improve efficiency, reduce downtime, and optimize production processes, wherein the present scenario is presented with the challenge of increased use of automation and advanced technologies has led to more complex production processes and a greater need for real-time data and decision-making.

The objective of this paper was to integrate IoT and cloud computing systems into a conventional assembly modeling system that helps it evolve into an advanced system, which is capable of dealing with complexity, and changes automatically.

The Proposed Solution and its methodology:

The proposed solution for an automated assembly modeling system includes utilizing a modularized architecture for robustness, reliability, flexibility, and expandability, incorporating object-oriented templates for easy interface and reuse of system components, and implementing automated algorithms for efficient assembly planning through the retrieval of relational assembly matrices.

0.1 Methodology:

0.1.1 Object Oriented Model Templates:

It is beneficial to have a modularized product structure, where components in assembly are not tightly connected, allowing participants to modify and maintain their own models without affecting the overall structure and is achieved by exploiting IoT's potential to link multiple resources. Using an object-oriented model template also simplifies product development by reducing the number of interactions required. This template includes basic elements, relations, constraints and assembly sequence for a product family. The assembly template is based on the physical connections between parts and defined separately for every assembly model.

0.1.2 Algorithms Proposed To Automate Assembly Modeling:

For assembly modeling, the key tasks involve defining the assembly relations from existing CAD models automatically. To achieve this, algorithms are proposed to automatically retrieve the matrices for these relations and thus automating the most crucial part of the said solution.

1) Matrix for Assembly Relations:

The most important information in a model template for the product assembly is the connection relations of parts. To retrieve it from the model template, a matrix for assembly relations is defined in Figure 1.

2)Interference Analysis in Generating Matrix M: To better automate a model it is critical to analyze the interference in determining an assembly plan. By using matrices to represent the components in the assembly, the authors use mathematical operations to analyze and compare the different components and identify potential areas of interference , and thus generate derived matrices from the original Matrix used to define Assembly Relations. These derived matrices are then utilized in sequence planning to define assembly or disassembly paths or other crucial parts of assembly planning.

$$m_{ij} = \begin{cases} 0 & \text{if } p_i \text{ and } p_i \text{ are separated} \\ 1 & \text{if } p_i \text{ is a function part, } p_j \text{ is a function part,} \\ & \quad \text{and } p_\mathrm{i} \text{ and } p_j \text{ are contacted} \\ 2 & \text{if } p_i \text{ is a function part, } p_j \text{ is a function part,} \\ & \quad \text{and } p_i \text{ and } p_j \text{ are connected} \\ 3 & \text{if } p_i \text{ is a connection part (screw or bolts)} \\ 4 & \text{if } p_i \text{ is a connection part (nuts)} \\ 5 & \text{if } p_i \text{ is a connection part (keys)} \\ 6 \geq 6 & \text{if } p_i \text{ is a connection part (others).} \end{cases}$$

Figure 1: Matrix M to retrieve template

3) Derived Matrix for Assembly Paths: Currently, there is no known algorithm that can automatically identify assembly or disassembly paths without interference between parts. The proposed solution is When creating a derived matrix, each part is positioned based on its local coordinate system (LCS) which is typically located on the base feature of the part. It is preferable for the direction of assembly or disassembly to align with an axis of the LCS or the global coordinate system (GCS).

0.2 My Views:

Validation for IoT's influence in the manufacturing sector has yet been fully explored. On the other hand, existing computer-aided software tools are experiencing a bottleneck in dealing with complexity, dynamics, and uncertainties in their applications of modern enterprises.

The paper promises to offer a solution to the above mentioned problem by automating the process and the regularization and it seems to have worked well for their particular case study of Assemblies of Aircraft Engines.

But I personally think that the paper has a few foibles , the absence for optimising Lack of control and Limited flexibility [Common problems faced when automating assembly models]. Also when generating a Derived Matrix dynamically , there's no account for how they plan to deal with the increasing number of computation over time and scale.