Automated Modelling of the Closed Cycle of Armour Plate Production *

Olga Artemenko[1,†], Hanna Hesheva[2,*,†] and Serhii Pasichnyk[3,†]

[1] *PHEI Bukovinian University, Darvina Street 2-A, Chernivtsi, 58000, Ukraine*
[2] *Dmytro Motornyi Tavria State Agrotechnological University, Zhukovskoho street 66, Zaporizhzhia, 69063, Ukraine*
[3] *Lviv Polytechnic National University, Stepana Bandery Street 32-a, Lviv, 79013, Ukraine*

Abstract

The research focuses on developing and implementing a project of an automated closed-loop armourplate manufacturing cycle using modern software tools. The work demonstrates the integration of design, production, and control processes into a single system that ensures high accuracy and efficiency of production. The study emphasises the automation of processes utilising specialised software and developing an API program to optimise the interaction between the system components. This paper presents the development of an algorithm for product creation using modern software tools. A 3D solid model was designed to support further manufacturing steps, with the production process automated within a closed-loop system. Cutting tools and precision measuring equipment were selected to ensure quality and accuracy. To optimize modeling costs of the project and streamline the update process, a Python-based calculation module was integrated into the design system to reduce modeling costs and enable visual tracking of structural changes. The study contributes to the project for automation of design and manufacturing processes in Ukraine's defense industry, offering an efficient, software-driven solution for producing armored components.

Keywords

Computer-aided design of design documentation, technological documentation, numerical control, technological processes, UML diagrams, project management, system analysis.

1. Introduction

Ukraine's defence industry is pivotal in safeguarding national security in the contemporary geopolitical environment. Producing high-quality protective equipment, including armour plates, is a priority area within the industry. The introduction of modern technologies for automation and digitalisation of production processes has the potential to enhance the efficiency and quality of the production of such critical components to a significant degree.

Automating design and production processes represents a pivotal component within the contemporary industrial production paradigm, particularly in fabricating models of complex products such as armour plates. In the context of armour plate manufacturing, this problem is particularly salient, as the quality and reliability of such products directly affect the safety of users.

The employment of contemporary design methodologies, facilitated by CAD/CAM systems such as SolidWorks and Autodesk Inventor, enables the generation of exact three-dimensional models that encompass all the requisite parameters and characteristics of the product.

Nevertheless, the efficacy of this process is contingent not solely on the quality of the design but also on the integration of the various stages of the production cycle and the automation of data

---

* *ITPM-2025: VI International Workshop "IT Project Management", May 22, 2025, Kyiv, Ukraine*
[1*] Corresponding author.
[†] These authors contributed equally.
olga.hapon@gmail.com (O. Artemenko); hanna.hesheva@tsatu.edu.ua (H. Hesheva); sergiypasichnyk@gmail.com (S. Pasichnyk)
0000-0002-4057-1217 (O. Artemenko); 0000-0003-3052-4393 (H. Hesheva); 0009-0006-8011-5618 (S. Pasichnyk)

transfer between them. The present paper focuses on modelling the closed-loop manufacturing cycle, which includes the creation of 3D models, development of process documentation, setting up computer numerical control (CNC) programs, and process automation using software tools such as SolidWorks, Autodesk Inventor, and the Python programming language. This paper not only analyses the theoretical foundations of automation but also offers a practical solution to the problem of creating an armour plate. The developed API program indicates a contemporary approach to software integration, facilitating the simplification and optimisation of production processes. The implemented automation methods aim to improve accuracy, reduce time, and minimise human errors in the production process. The structure of the work includes an analysis of project management methods in engineering, the development of diagrams for process visualisation (in particular, diagrams of use cases, states and transitions), and the practical implementation of an automated armour plate manufacturing cycle using modern software tools.

A systematic approach characterises the present work, as the use of a closed manufacturing cycle allows all stages – from the creation of a 3D model to the checking of the quality of the product – to be combined into a single integrated system. Moreover, the practical nature of the developed Python program is evidenced by its capacity to integrate SolidWorks APIs, thereby automating alterations in model parameters and significantly streamlining production data management processes. Implementing advanced CAD/CAM software (SolidWorks, Autodesk Inventor) facilitates the preparation of production documentation and CNC trajectories with high accuracy. The modelling approach adopted was meticulous, presenting a diagram of use cases, states and transitions, thus ensuring a comprehensive comprehension of the system's operation. The presented work demonstrates the practical application of computer-aided design and technological preparation tools for production. This work is expected to be helpful to process engineers and researchers in production automation.

2. Project management of the armour plate manufacturing project

Integrating visual planning tools, precision manufacturing technologies, and optimised process design is imperative for effective project management in engineering. Modern methodologies use schematic representations, including Gantt and flow charts, to coordinate complex workflows [3]. Concurrently, computer numerical control (CNC) systems facilitate high-precision manufacturing by translating digital designs into machine instructions [4]. Managing engineering projects is the foundation for technical initiatives, ensuring the balanced allocation of resources, the adherence to stipulated timelines, and the fulfilment of quality criteria. In contradistinction to the general project management paradigm, engineering programmes necessitate specialised tools for navigating technical specifications, ensuring regulatory compliance, and fostering interdisciplinary coordination [3]. The Project Management Institute (PMI) framework is adapted to engineering contexts through modified phase processes. The initiation phase involves conducting feasibility studies using computational models to estimate material and energy requirements [6]. The planning phase of an engineering project utilises work breakdown structures (WBS), which are decomposed into machine-readable tasks for CNC operations [4]. The Project Management Institute (PMI) framework is adapted to engineering contexts through modified phase processes.

The implementation of CNC operations necessitates meticulous planning to optimise machine utilisation. The theory of constraints (TOC) identifies the machining centre with the most sluggish performance as the element that can be utilised to accelerate the work, which requires the management of buffers in preceding processes [5].

As illustrated in Figure 1, a work breakdown structure (WBS) has been devised for the armour plate manufacturing process. This is a fundamental aspect of the project, as it enables the team to comprehend the nature of the project.

A work breakdown structure (WBS) is a fundamental project management tool that provides a systematic framework for defining, organising, and controlling project deliverables throughout the life cycle. This comprehensive framework enables project managers to decompose complex projects into manageable components, facilitating more efficient planning, execution, monitoring, and control of project activities. As projects continue to exhibit increased complexity across diverse industry sectors, the strategic implementation of a WBS has become imperative in ensuring the successful realisation of project objectives in terms of scope, time, cost, and quality parameters.

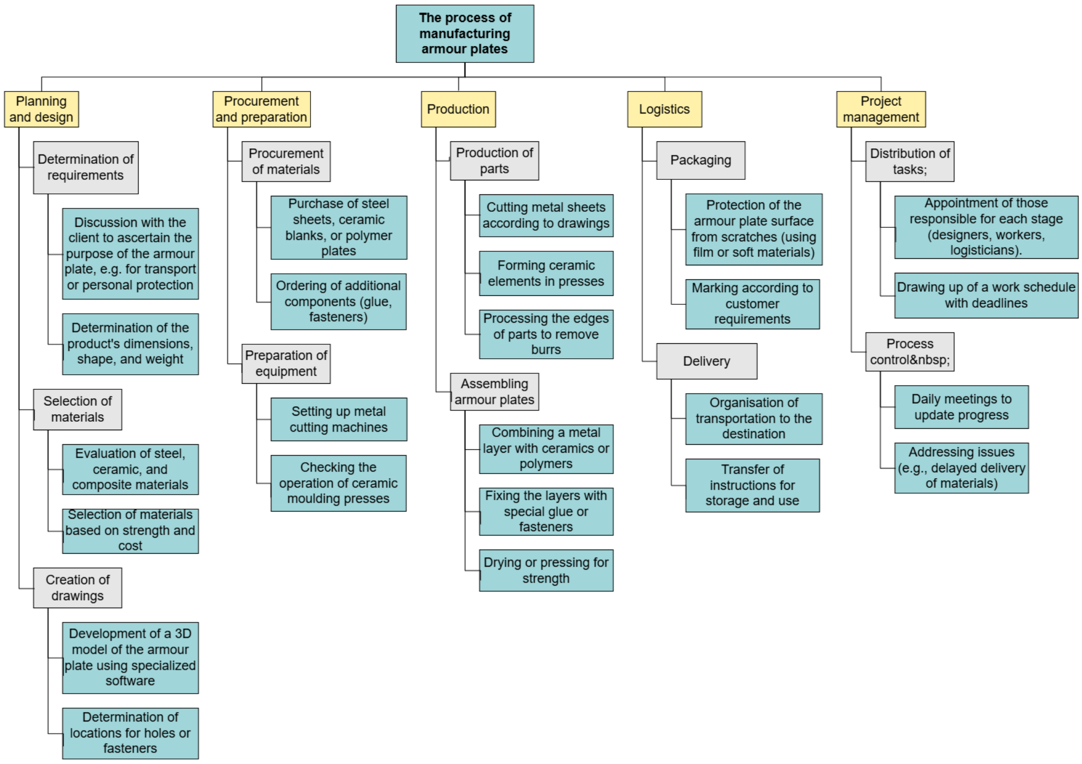

Figure 1: Work breakdown structure (WBS) for the armour plate project.

The task assigned to us involved the development of an armour plate model within the project, with consideration given to all project stages.

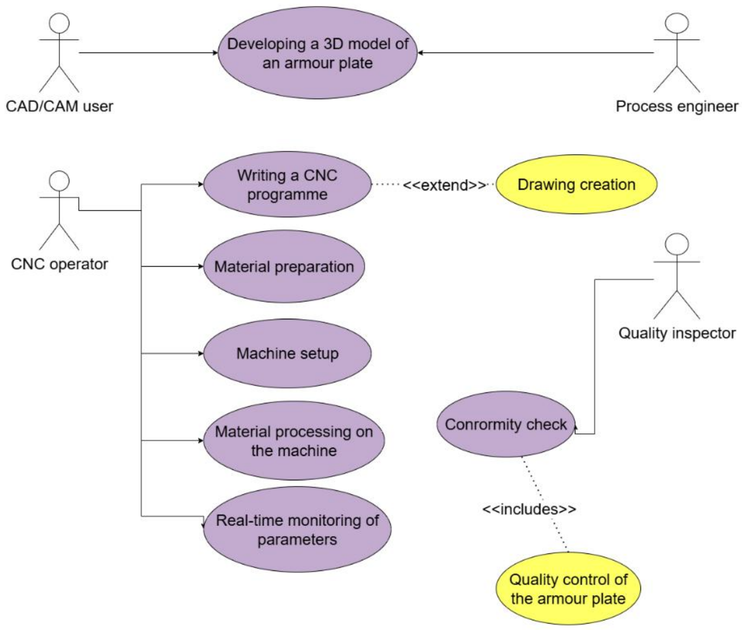

Figure 2: Diagram of options for using the armour plate manufacturing process on a CNC (computer numerical control) machine.

In preparation for the project, developing specific diagrams was necessary to facilitate a comprehensive understanding of the process.

The following discussion will refer to the use cases diagram (Fig. 2).

A thorough examination of the use case diagram reveals the following key components:

1. Actors (participants in the process):

The CNC operator is responsible for the machine's setup, operation and monitoring.

Furthermore, the process engineer's role is to develop drawings and programs for the CNC.

Finally, the role of the CAD/CAM user is to work with the software to create the necessary drawings and trajectories.

Finally, the quality inspector is tasked with examining the finished product.

2. The primary applications are as follows:

Developing a 3D model of an armour plate is a task performed and managed by a process engineer and a CAD/CAM user.

The writing of a CNC programme is based on a drawing.

Furthermore, the CNC operator is responsible for material preparation.

Setting up the machine is also the responsibility of the CNC operator.

Furthermore, the CNC operator is responsible for the real-time monitoring of parameters.

Finally, conformity checking is undertaken, which includes armour plate quality control.

The subsequent diagram was developed to visualise the process, and it is a diagram of states and transitions (Fig. 3).

The following detailed analysis of the diagram is proposed:

The initial state is designated as 'Material ready'. The transition will be: 'Download the application' → Next state: 'Programme downloaded'.

State: 'Programme downloaded'

'Set up machine' → 'Machine set up'.

State: 'Machine set up'

'Start machining process' →

Next state: 'Cutting/machining'.

State: 'Cutting/machining'

'Processing complete' → 'Finishing'.

State: 'Finishing processing'

'Finishing is complete' → 'Inspection' (checking the product for quality and compliance with all requirements).

State: 'Inspection'

Transition 1: 'Successful inspection' →

Status: 'Process completed'.

Transition 2: 'Does not meet standards' →

Status: The final status is 'Process complete'. This indicates that the product is ready to be shipped.

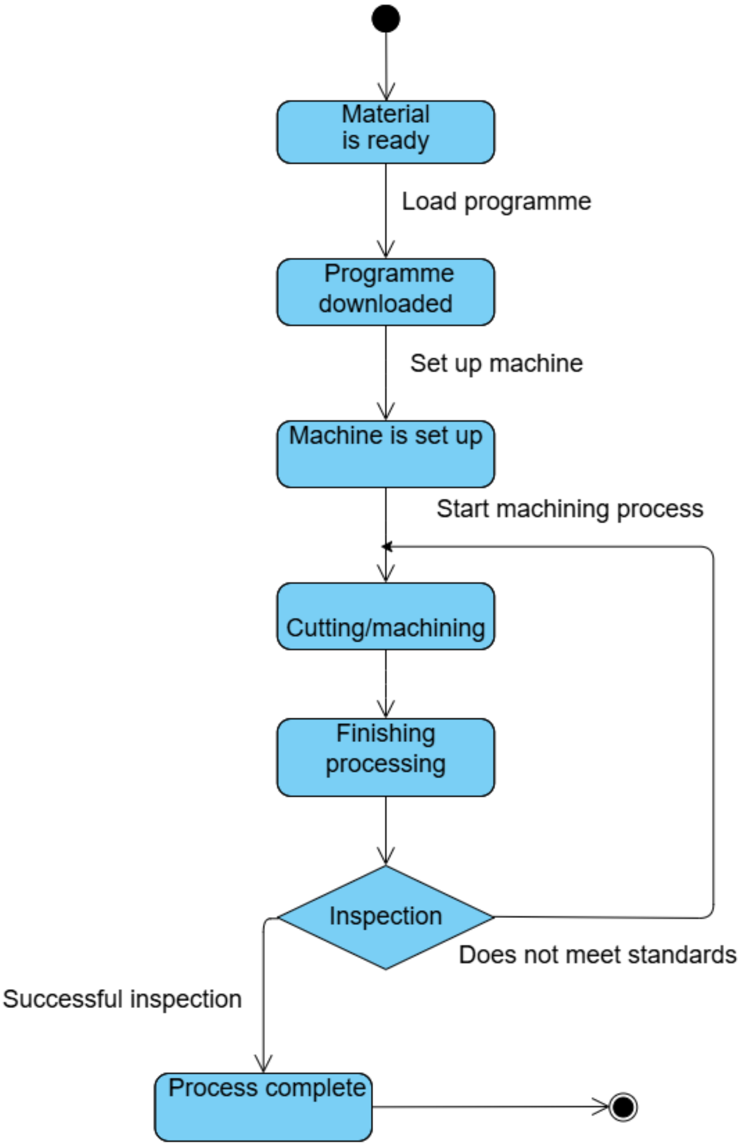

Figure 3: Diagram of states and transitions for the process of manufacturing armour plates using a CNC machine.

3. Development of the application API

During the project, an API program was developed that allows the user to modify any parameter of the Armour Plate model.

The application code was written in the Python programming language. The application is compatible with the parameters of a model created in SolidWorks through the SOLIDWORKS API using the COMtypes library. Alternatively, the code can be modified to operate using the swpy library. The latter is a Python wrapper for the SolidWorks API, thus facilitating the process, albeit not a standard library and necessitating separate installation. The comtypes library was selected for its ability to facilitate direct interaction with the SolidWorks COM API, thereby offering enhanced control. However, it should be noted that this approach necessitates a comprehensive understanding of the particulars of the SolidWorks API.

The code facilitates the manipulation of SolidWorks models by utilising a GUI, enabling the modification of parameters.

A step-by-step examination of the code's workflow is now warranted. The code has been developed to create a graphical user interface (GUI) that works with SolidWorks models. The sequence of operations initiated by the code is as follows: it launches SolidWorks through the API, opens the 3D model (.sldprt or .sldasm), retrieves a list of custom model properties, then obtains the dimensions (geometry parameters) of the model, and then can modify both the custom properties and the dimensions (geometry parameters) (Figure 4, Figure 5).

```python
# Modify a geometric dimension
def set_dimension(sw_model, dim_name, new_value):
    try:
        dimension = sw_model.Parameter(dim_name)  # Example: "D1@Sketch1"
        if dimension:
            dimension.SetSystemValue3(float(new_value), 2)  # 2 - change in mm
            sw_model.EditRebuild3()
            sw_model.Save()
            print(f"Dimension {dim_name} changed to {new_value} mm")

            messagebox.showinfo("Success", f"Dimension {dim_name} changed to {new_value} mm")
        else:
            print(f"Dimension {dim_name} not found")
    except Exception as e:
        print(f"Error modifying dimension: {e}")
```

Figure 4: Modify a geometric dimension.

```python
# Function to modify a dimension
def change_dim():
    if not sw_model:
        messagebox.showerror("Error", "Open a model first.")
        return
    selected = tree_dims.focus()
    if not selected:
        messagebox.showerror("Error", "Select a dimension to modify.")
        return
    dim_name = tree_dims.item(selected, "values")[0]
    new_value = entry_dim.get()
    set_dimension(sw_model, dim_name, new_value)
```

Figure 5: Function to modify a dimension.

Our code utilises the following libraries:
– comtypes.client to work with the SolidWorks API.
Furthermore, tkinter is utilised to facilitate the creation of a GUI.
To ensure the effective operation of the code, it is essential to observe the following dependencies:
1. Python libraries:

- Comtypes
- tkinter (included in the standard Python library)

2. Additionally, the SolidWorks software with COM API support must be installed.

Python was selected for the task at hand due to its numerous advantages, including convenience, speed of development, and several other key benefits. Firstly, Python facilitates effective interaction with COM (Component Object Model), which SolidWorks uses for automation. The comtypes library simplifies the creation of COM objects and the interaction with the SolidWorks API, providing seamless integration.

Compared with other programming languages, such as C++, which requires compilation, Python offers a more streamlined integration with COM. Additionally, while Visual Basic for Applications (VBA) is highly effective for macros, it is not well-suited for comprehensive GUI applications.2. Python's syntax is straightforward and intuitive, making it a breeze to work with. This, in turn, results in a substantial increase in development speed. Additionally, the code is relatively concise, comprising approximately 200 lines, and will result in a ready-made GUI application for working with SolidWorks.

Compared with other programming languages, a similar C++ or C# application would necessitate a substantially larger amount of code.

3. Python contains a built-in Tkinter library that facilitates the straightforward creation of windowed applications without additional dependencies. This is a highly convenient GUI using this library.

Conversely, alternative programming languages such as C++, Qt and C# WPF necessitate greater customisation. To illustrate this point, consider VBA's limited capabilities in creating a GUI. In contrast, Python boasts several advanced libraries, including pandas and openpyxl, which can be instrumental in data management tasks, such as exporting model data to Excel or CSV, generating reports, and automatically checking model parameters. This inherent automation capability is a significant advantage.

In comparison, the process is more convoluted in VBA and necessitates more code in C++. Additionally, if the requirement is to extend the functionality, it is more straightforward to do so in Python. For instance, the following additional operations can be performed: incorporating 3D model processing (utilising numpy or vtk libraries) and creating a web interface (employing the Flask/Django frameworks).

4. Modelling an armour plate

A model of the armour plate was developed using the SolidWorks software. The model is presented in the figures from both the interior and exterior perspectives (see Figures 6 and 7).

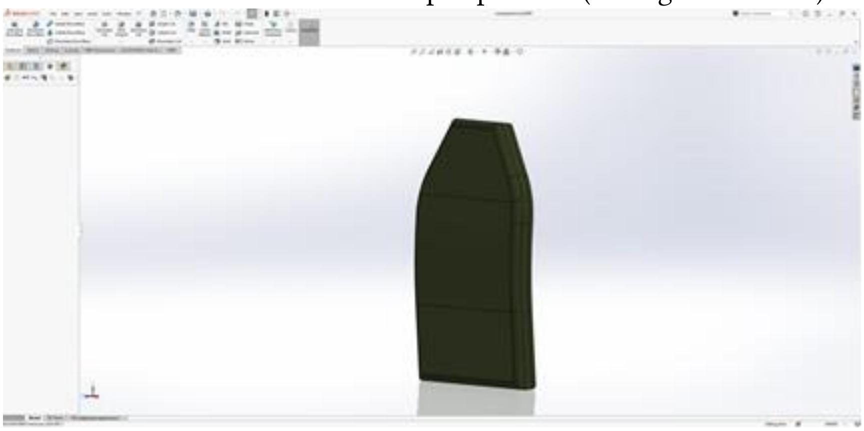

Figure 6: Armour plate model from the outside.

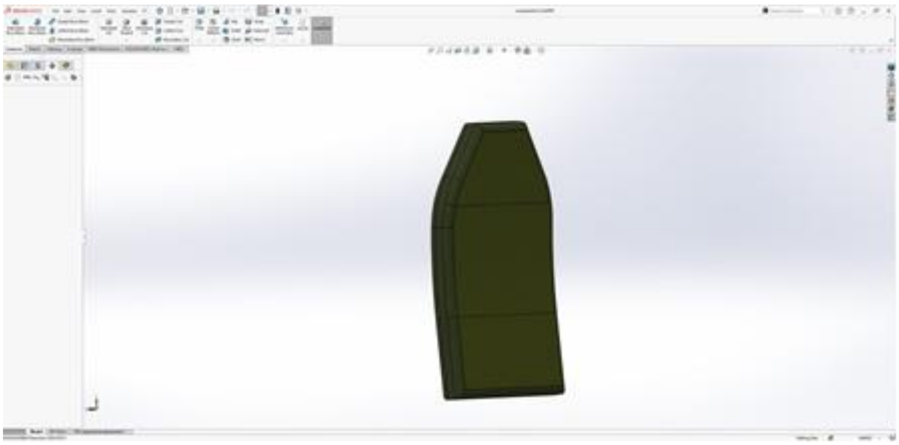

Figure 7: Armour plate model from the inside.

The machining process of the component is initiated with the utilisation of Autodesk Inventor, a software that facilitates the creation of a template. This template serves as a blueprint for the machining processes executed on the CNC machines. The 3D model, which has been previously developed, is imported into Inventor and methodically processed. Additionally, the model can be employed to create 2D drawings, which may be necessary for further production-related activities. Following importation, the model becomes accessible within the Inventor environment as a solid geometry [1].

Furthermore, the software facilitates modification of the model, addition of new elements, and incorporation of threads, among other functions. Before the machining process, a thorough examination of the model is imperative. This is undertaken using analysis tools that assess parameters such as geometry, weight, and centre of mass.

The subsequent processing of the component in Inventor is conducted through the utilisation of CAM (Computer-Aided Manufacturing) modules, wherein trajectories are meticulously prepared for the following operation of a CNC machine (Fig. 8). Additionally, the graphical module of the program facilitates the observation of the machining process's modelling, thereby enabling the comprehension of the tool's performance on the component in a sequential manner (Fig. 9, Fig. 10). Autodesk Inventor has been demonstrated to demonstrate effective integration with a variety of production systems, thus enabling the effective management of imported data. This capacity represents a significant advantage in collaborative environments. Following the requisite processing, the model is exported with configured trajectories for subsequent production utilisation of CNC machinery.

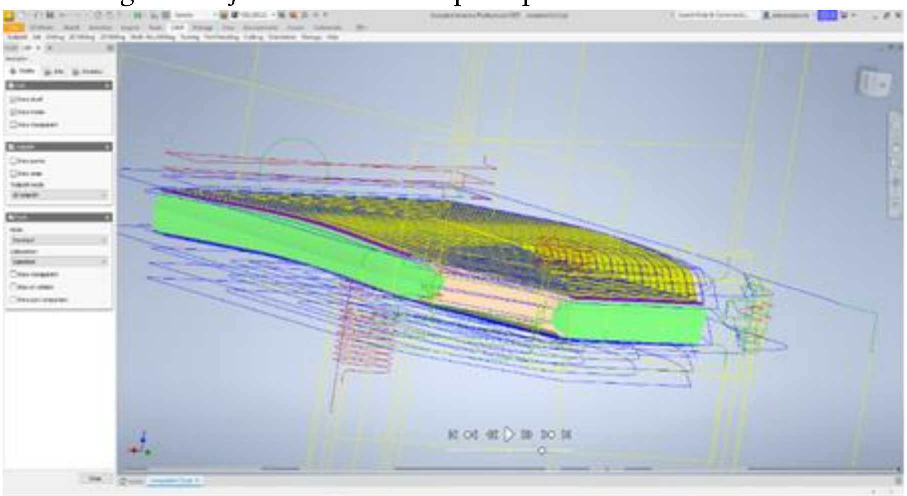

Figure 8: Defining trajectories for model processing.

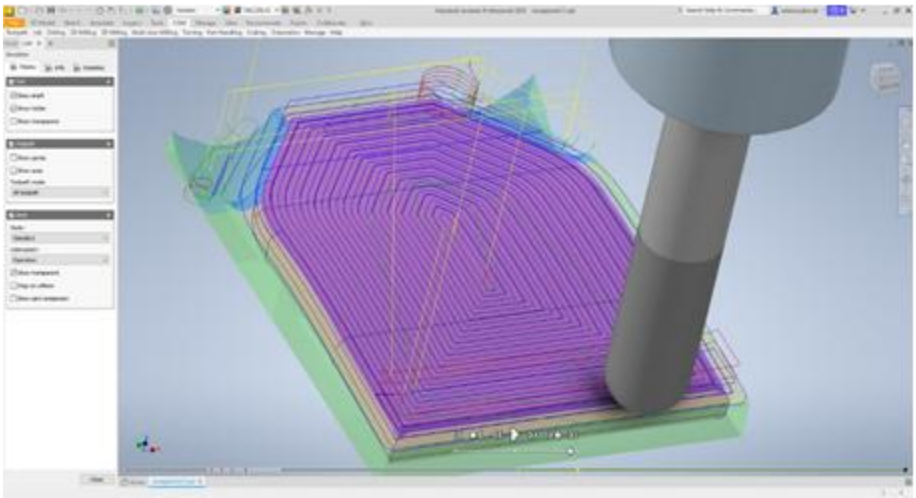

Figure 9: Roughing the model with the selected tool.

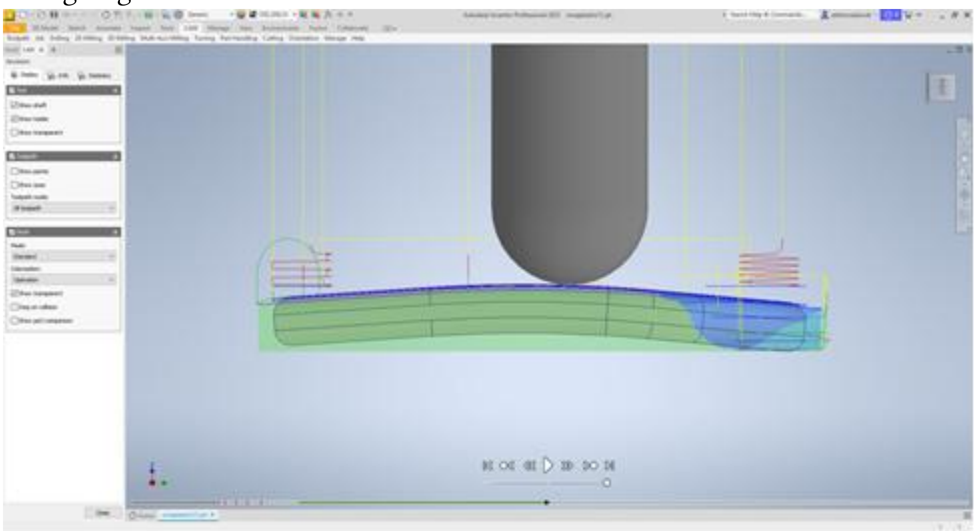

Figure 10: Processing a model using the tool.

5. Strength calculations

COSMOSWorks will be integrated into the SolidWorks software to verify the model's strength. COSMOSWorks is a finite element analysis (FEA) software module integrated into the SolidWorks environment. Its primary function is to assist engineers in performing various types of analysis directly during the design process, including static, dynamic, thermal, and others. This enables them to optimise designs and enhance their reliability at the initial stages. It is a powerful tool that helps to improve design solutions and enhance the quality of the final product. [7]

The utilisation of COSMOSWorks entails the connection of the module and the execution of requisite calculations, thereby facilitating the efficient execution of engineering tasks through its integration with SolidWorks. The module's flexibility enables the selection of essential tools to address specific requirements, while the finite element method employed by COSMOSWorks constitutes a numerical approach to structural analysis. It allows the solution of equations that describe the behaviour of elements, considering their interconnections and assessing the impact of external loads and constraints. The software determines the displacements in the X, Y, and Z axes at each structural node, which is used to calculate the forces acting in different directions. In addition, mathematical expressions and formulas are used to analyse stresses. One can predict its behaviour by determining the stresses based on the part's material, applied loads and constraints. If the stress reaches a critical value, the part will fracture due to the inherent properties of the material from which it is made.

The following methods are employed to limit the mobility of a part:

- The edges and vertices of the element are fixed.
- Restriction is applied in a specific direction.
- Symmetry is utilised to analyse a part of the structure.
- Rigid connections (e.g. bolts, springs, etc.) are employed.
- Different element sizes are used in different areas of the part to improve the accuracy of calculations.

Before the initiation of the task, a 3D model is loaded into COSMOSWorks, and calculations are performed in the form of exercises. In the COSMOSWorks Manager, the Exercise command should be selected from the context menu. Then, the fundamental parameters should be set: analysis type (e.g. static) and mesh type (e.g. solid mesh). The initial data for the strength calculation is then entered. Select the "Apply/Edit Material" command from the context menu of the "Armour Plate" element to specify a material. Materials from SolidWorks libraries are available in the material selection window. It is usual to fix a model using its lower and upper edges. To set the load, in the Manager tree, select the "Pressure" command from the context menu and then select the upper surface of the part to be loaded.

One of the key tasks of the calculation is to determine the safety factor. This coefficient indicates the extent to which the permissible stress is less than the critical stress at which the part may fail, and its value is contingent on the characteristics of the material, how the load is applied, and so forth. The analysis aims to ascertain the force at which the part will not withstand the load; in this case, the safety factor will be close to one. The program automatically calculates this factor after the calculations are completed. The Report section provides detailed information about the strength calculation in the format of an HTML file, which can be obtained by clicking the Report button on the COSMOSWorks results toolbar.

6. Conclusion

The present study demonstrates the effectiveness of employing contemporary automation methods to model the closed cycle of armour plate manufacturing. The developed methodology ensures the integration of all stages of the production process, from design to quality control, which allows for increased accuracy, reduced time costs, and minimised risks of errors. The key achievement of the research is the development of an API program in the Python programming language that provides flexible control of model parameters in SolidWorks. This application exemplifies a contemporary approach to integrating diverse software tools and automating data transfer between them, a particularly salient consideration in intricate production processes. Implementing the proposed methodology and the developed software tools within the production process is anticipated to result in substantial improvements in the efficiency and quality of armour plate manufacturing and the development of direct relevance to the Ukrainian defence industry. A systematic approach to the production organisation, incorporating diagrams of use cases, states and transitions, and a work breakdown structure, provides a clear understanding of the process and effective resource management.

The practical significance of the research results lies in their direct implementation in the production process, not only for manufacturing armour plates but also for other complex products requiring high accuracy and reliability. The developed API programme's adaptability to different types of models and parameters ensures its versatility and vast application possibilities. Prospects for further research in this area include the expansion of the functionality of the developed API program, its integration with other software tools (e.g. manufacturing execution systems (MES) and enterprise resource planning (ERP)), and the development of algorithms for the optimisation of model parameters concerning specific product requirements. It is also advisable to conduct experimental studies to assess the effectiveness of the proposed methodology in real production conditions.

The objective of the task was to develop an algorithm for creating a product utilising computer software packages. A three-dimensional solid-state model was constructed for subsequent tasks, a manufacturing process was formulated, and the design was automated within a closed production cycle using specialised software. The necessary cutting tools for machining the part were selected and analysed, and measuring instruments were identified to ensure the high accuracy and quality of the manufactured structure. A calculation module based on the Python programming language was created to reduce the cost of modelling the part and automate its updating. This module is integrated into the design system, which makes it possible to visually track changes in the structure of the part during its adjustment and make the necessary corrections.

The study significantly contributes to developing the Ukrainian defence industry's design and production automation methods. It offers a comprehensive solution integrating modern software tools and project management methodologies. This integration aims to ensure efficient and high-quality production of armour plates.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT, Grammarly in order to: Grammar and spelling check, Paraphrase and reword. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

[1] Autodesk Inc. Autodesk Knowledge Network (AKN): The official Autodesk knowledge base on importing, processing, and editing models. (2025) [Online]. Available: https://knowledge.autodesk.com

[2] Dassault Systèmes SolidWorks Corporation. SolidWorks Support and Tutorials: Basic guides and articles on exporting models in formats like STEP, IGES, and Parasolid. (2025) [Online]. Available: https://www.solidworks.com

[3] S. Khan, A. Dimache, D. Gorman, and C. Gachon, "Lean in Higher Educational Institutes: A Literature Review," in *Studies in Educational Management*, vol. 15, pp. 1–19, 2024. Available: https://doi.org/10.32038/sem.2024.15.01

[4] R. M. Yojana, E. F. Harahap, W. Septiani, S. Adisuwiryo, and E. Brata, "Jurnal Sistem dan Manajemen Industri," vol. 6, no. 2, pp. 121–134, Dec. 2022. Available: http://dx.doi.org/10.30656/jsmi.v6i2.5088

[5] M. R. Ghani and D. S. Donoriyanto, "Development of Android-Based Management Information System Application for Student Final Project Services," in Jurnal Inovtek Polbeng – Seri Informatika, vol. 10, no. 1, Mar. 2025. ISSN: 2527-9866

[6] Rudin, M.J.: Project Management Plan for the INEL Technology Logic Diagrams. EG&G Idaho, Inc., Idaho Falls, Idaho, Prepared for the US Department of Energy, October 1992.

[7] Rice University, "Introducing COSMOSWorks," [Online]. Available: https://www.clear.rice.edu/mech403/HelpFiles/IntroducingCW.pdf.