

Received: 16 February 2026 / Accepted: 21 April 2026 / Published online: 04 May 2026

*CAD/CAPP/CAM/CNC integration,
manufacturing analytics,
non-productive motions,
manufacturing feedback*

Adam Piotr ZALEWSKI^{*1},
Sergiusz SOBIESKI²

NON-PRODUCTIVE MOTIONS IN MILLING: A CAM-CONNECT DATA-DRIVEN IDENTIFICATION METHOD

Manufacturing data typically flows one way (CAD→CAPP→CAM→CNC), expanding as it approaches the shop floor through technologists' know-how and parameter choices. Using the CAM-Connect module integrated with Mastercam, we automatically extract these CAM settings and export a structured process description (STEP/STEP-NC XML) for scalable analysis. This paper introduces a practical CAM-driven method to identify and quantify non-productive motions in 3-axis milling. We focus on vertical auxiliary motions: Z-axis approaches, withdrawals, retracts and safe-height traverses. Auxiliary distances and times are reconstructed by interpreting CAM option logic, including clearance modes and retract/feed-plane definitions. The method is applied to 242 industrial milling operations comprising 2912 toolpaths. After excluding special drilling cycles and extremely short operations, auxiliary motion averages 7.96% of total operation time. Moreover, 14.5% of operations exceed a 15% auxiliary-time share. While rapid and feed auxiliary distances are similar, feed-executed auxiliary segments generate about 96% of auxiliary-motion time. The findings provide actionable diagnostics tied to CAM decisions and support systematic auditing and improvement of machining processes.

1. INTRODUCTION

In CNC milling, productive cutting is accompanied by auxiliary (non-cutting) motions—any tool movements without material removal, such as approaches and withdrawals, rapid repositioning, and positioning between cutting segments. These motions do not create value (no chip is produced), yet they increase cycle time and energy consumption [1]. In modern machine tools, the share of non-cutting time can be unexpectedly high: research and review reports indicate that effective cutting may occupy only a limited part of machining time, and energy balances can reveal situations where non-productive states consume most of the total energy, while cutting accounts for only a small fraction of the

¹ Faculty of Mechanical and Industrial Engineering, Warsaw University of Technology, Poland

² Technology company, TIZ-IMPLEMENTS Sp. z o.o., Poland

* E-mail: adam.zalewski@pw.edu.pl

<https://doi.org/10.36897/jme/220902>

overall demand [2]. Industrial estimates also exist in which roughly 20% of the energy is attributed to cutting, implying that approximately 80% is spent in “non-cut” states (positioning, idle periods, and auxiliary motions of machine subsystems) [3].

Consequently, methods for modelling and optimising non-productive trajectories have been developed for years, aiming to shorten idle times and reduce energy demand [4]. Particular attention has been given to advanced approaches: path optimisation and multi-objective methods (simultaneous minimisation of time and energy, e.g., using NSGA-II) [5], the use of artificial intelligence and digital twins for monitoring and real-time adaptation [6], and heuristics inspired by combinatorial optimisation (e.g., variants of TSP/PC-TSP for feature sequencing). In parallel, feedrate scheduling methods (without geometric path deformation) have been developed for linear–circular segments described in NC files, enabling time reduction without changing the toolpath geometry [7]. An important research stream also addresses air-cutting elimination at the geometric level by modifying the trajectory while considering tool–workpiece relations; reported results show cycle-time reductions obtained by removing non-productive parts of the trajectory [8].

Auxiliary motions in milling include all axis movements during a cycle when cutting does not occur, for example:

- Tool approach and withdrawal – entry into the workpiece before cutting and exit after cutting (e.g., Z-axis motion to/from the part surface).

- Transfers between cutting paths – so-called air-cutting when the tool travels to the next machining region without removing stock [9].

- Rapid positioning (G0) – rapid traverses to reference points, between features, or between operations.

- Auxiliary actions during the cycle – e.g., motion to a tool-change position, spindle acceleration/deceleration, coolant waiting, etc. These actions are typically sequential and do not directly contribute value to the part.

Auxiliary motions are necessary (they ensure safety and enable reliable execution), but they create losses of time and energy and therefore represent a key domain for optimisation. The CNC cycle time is the sum of all activities, both cutting and non-cutting. According to application guides, removing unnecessary transfers and optimising sequences can shorten the cycle by tens of percent. For example, Gunawan et al. [9] proposed a dynamic trajectory optimisation method that detects and shortens idle segments in real time, obtaining up to 42% reduction in total machining time by limiting air-cutting. Similarly, Nishida (2020) reported that automated toolpath modification aimed at eliminating empty passes can significantly shorten 3-axis milling [10].

A common recommendation is to audit the process time by decomposing the cycle and measuring how many seconds or minutes are spent in each activity [11]. Such analysis often reveals that idle motions dominate rather than cutting. Based on this, suggests prioritising improvements where the largest “gaps” occur, such as excessive moves between workpiece regions or unnecessary motions to a tool magazine. Typical issues include overly conservative clearance heights and suboptimal feature sequencing, causing the tool to travel unnecessary distances. Even simple corrections—such as lowering the retract height when workpiece geometry permits—reduce the approach time before cutting. Another improvement is overlapping auxiliary actions: for instance, starting the spindle or coolant while approaching

the part instead of triggering actions sequentially. Modern controllers offer M-codes supporting parallel execution, i.e., performing multiple actions during one tool move.

It is also important that the machining strategy (operation order) affects auxiliary time. When multiple workpieces are machined in one fixture, programming should minimise transfers between them. Practical recommendations include starting and ending near the tool magazine to shorten tool entry/exit paths. If a programmer completes all features on one fixture side and then repeats the same sequence on another side, the machine repeatedly travels from the headstock to the farthest position—an avoidable loss. Intelligent sequencing (e.g., alternating the nearest regions across sides) can eliminate redundant transfers and noticeably reduce idle time [11].

CAM software plays a key role in generating efficient toolpaths. Classic strategies (zigzag, contour, radial) focus primarily on geometry and do not guarantee minimal non-cutting time. Therefore, recent development has intensified around automatic toolpath optimisation targeted at shortening non-productive motions and smoothing connections between cutting segments. At the problem-definition level, the goal is often framed as minimising the length of non-cutting motion and eliminating unnecessary retracts. Industry reviews and application materials indicate that purposeful toolpath optimisation reduces cycle times and machine energy demand, while improving motion stability and surface quality [12].

In practice, the implementation of these methods follows two main routes. First, CAM systems themselves are extended to generate improved trajectories (or provide built-in optimisation). Second, optimisation algorithms are integrated with CNC control by analysing and adjusting NC code. Pajaziti et al. (2025) presented an approach for direct G-code optimisation with respect to time: the method analyses command sequences, identifies inefficient motions, and modifies trajectories and parameters to remove them [13]. Using simulation (ICAM3D v3.1.0 and ArtCAM/Aspire) to verify correctness and time impact, selected projects were shortened from 15 min 23 s to 13 min 33 s without affecting geometric accuracy—approximately a 12% improvement.

Beyond offline optimisation (CAM programming or NC editing), there is an increasing interest in online optimisation, where CNC control may adapt sequencing or approach parameters in response to the current machine state. Such methods require very fast decision algorithms and often rely on simpler heuristics. In industrial practice, predefined controller functions for parallel execution are therefore more common—for example, returning to a safe position while a rotary table reorients or while a tool change is prepared.

Simulation and virtual analysis are also increasingly used at the CAM planning stage to assess toolpath efficiency. Machining simulators (e.g., Vericut, NX CAM, Mastercam Simulator [14]) estimate the time of individual motion segments and detect collisions or unnecessary moves. In research, simulation is a key validation tool; for instance, Karuppanan (2019) compared algorithmic optimisation results with trajectories generated and simulated in commercial CAM systems [15], showing that even advanced packages can leave significant improvement potential.

The above examples primarily address toolpath preparation or NC program execution. However, important practical questions remain: Can we trace the history of executed jobs and assess their efficiency at scale? Can we provide technologists with targeted guidance on where improvements will likely yield the largest benefit in future processes? Can we capture the

practical production knowledge that reflects the specifics of a particular company? The method proposed in this paper addresses these questions by drawing conclusions from CAM settings and decisions rather than only from measured outcomes. These settings (technologist choices) are decisive for future solutions and therefore valuable for monitoring and diagnostics.

2. A KNOWLEDGE BASE BUILT BY CAM TECHNOLOGISTS

2.1. DATA FLOW AND THE CAM-CONNECT CONCEPT

Preparing NC programs for CNC machine tools using a CAM system begins with assumptions and constraints defined during process planning. With the support of CAPP (Computer-Aided Process Planning) systems—of varying maturity depending on company size—a process plan is prepared and passed to CAM. It typically includes geometric models of the part and stock, lists of machines, fixtures, tools, setup definitions, and a decomposition into setups, operations and toolpaths, together with basic technological parameters. This information can be considered general (“A” in Fig. 1). Based on it, the technologist builds the detailed CAM process by adding information “B” and “C” (Fig. 1), ultimately generating an NC program.

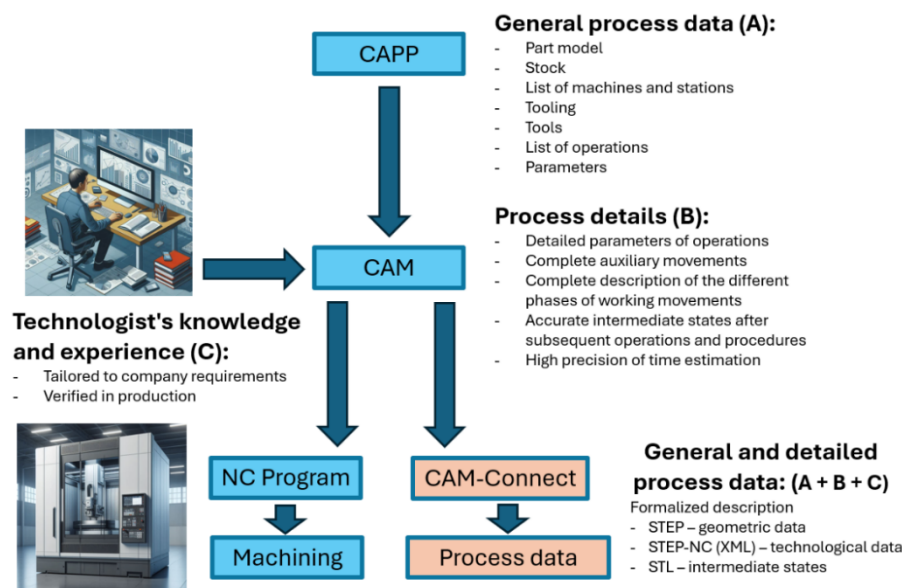


Fig. 1. Manufacturing data flow and the CAM-Connect feedback concept (described in Section 2.2) [16]

Technologist work in CAM combines theoretical knowledge with experience interpreted through real conditions. Depending on the completeness of CAPP inputs, this work may include selecting motion strategies, defining approach/withdrawal logic, setting auxiliary motions and safety planes, splitting axial and radial depths of cut, and adapting local cutting conditions. At this stage, differences in process efficiency can reach tens of percent [17].

These detailed data (“B” and “C” in Fig. 1) are directly tied to successful machining because they are adapted to system stiffness, vibration tendencies, required productivity, accuracy and surface quality. Therefore, validated CAM processes implicitly encode information about the dynamic capabilities of the machine–fixture–tool–workpiece system and about a company’s manufacturing know-how.

Such technological knowledge accumulates naturally as subsequent jobs are executed; however, the practical use of hundreds or thousands of CAM projects as a company knowledge base is challenging. For this reason, we developed CAM-Connect to automatically harvest this knowledge and structure it for analysis and decision support.

2.2. DATA EXTRACTION AND REPRESENTATION

Each CAM system stores process data in its own internal format. Although the process structure (and even the content of dialog windows) is often similar, data storage relies on intermediate files subsequently translated by a postprocessor into NC code (Fig. 1). These intermediate representations only apparently describe the complete process: they contain the information required to generate an NC program for a specific part, typically as explicit coordinates and motion parameters (e.g., move to $Z = 11$ mm at rapid speed). They do not explicitly encode the *rules* used by the technologist (e.g., “rapid to 1 mm above the surface, then approach at reduced feed”), which limits their value as a knowledge repository. Attempts have been made to extract knowledge directly from NC programs (e.g., by analysing ISO 6983 code) [18], but this approach makes it difficult to relate execution to the original CAM decisions.

CAM-Connect operates inside Mastercam by automatically opening subsequent CAM projects and identifying their operations and toolpaths. For each toolpath, CAM-Connect locates the associated geometric inputs used by the technologist and exports them into a universal format (STEP, STL), accessible outside Mastercam. In parallel, CAM-Connect generates a formal, highly detailed description of the machining process aligned with the CAM data structure. In the analysed configuration, each toolpath is represented by up to 5665 technological parameters (depending on toolpath type), together with identifiers linking parameters to exported geometry.

Geometry export uses the STEP format (Standard for the Exchange of Product Data [ISO 10303-11 1994]), which enables data exchange across CAD systems. For process data, a STEP-NC-based representation is applied [ISO 14649-1 2003] [19]. To support CAM integration, the STEP-NC representation used in CAM-Connect was extended and stored in an XML-based structure. A distinctive feature is the export of intermediate stock states after each toolpath in STL format.

By connecting process settings (what the technologist decided) with the detailed structure of a job, CAM-Connect provides a missing feedback link between shop-floor execution and upstream engineering (CAD/CAPP/CAM). In addition to supporting ramp-up and new process design, the resulting database offers labelled training material for AI, as demonstrated in project Cybertech 4. In this paper, we use this infrastructure to diagnose non-productive motions in historical milling processes.

3. CASE STUDY: AUXILIARY-MOTION ANALYSIS

3.1. DATA SET AND SCOPE

This paper focuses on auxiliary motions. The study analysed 242 Mastercam project files from one company, created by the same technologist between 2012 and 2023. The parts were machined on a 3-axis milling machine. CAM-Connect generated, among other artefacts, 242 XML files, each containing data for one operation. Here, an operation is defined as the machining of a part in one workpiece position (one setup), consisting of multiple toolpaths (CAM cycles).

The analysis is restricted to auxiliary motions in the vertical Z axis occurring before or after cutting. Although CAM-Connect also enables auxiliary-motion analysis in the XY plane, the vertical component is particularly relevant because it is directly affected by safety and clearance conventions. In future work, we plan to use indicators normalised by tool diameter. In the analysed dataset, the XY approach distance was approximately proportional to tool diameter (about 75% of diameter), while tool diameter had no significant impact on Z -axis auxiliary distance.

Across the 242 operations, the total number of toolpaths was 2912, i.e., about 12 toolpaths per operation on average. The maximum number in one operation was 177; a small number of operations contained only a single toolpath.

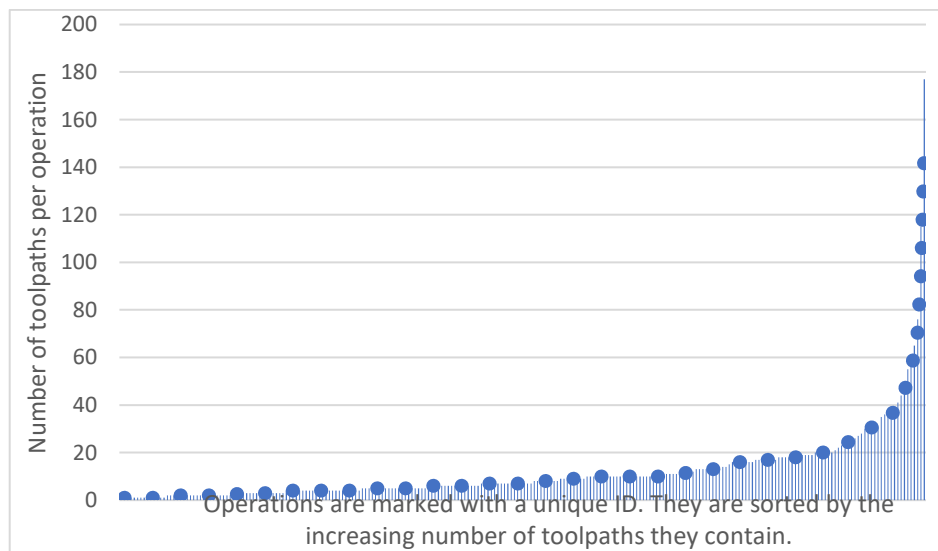


Fig. 2. Number of toolpaths per operation (operations sorted by increasing toolpath count)

Figure 2 shows the number of toolpaths per operation, revealing substantial variability in process structure across the analysed dataset. This is relevant to the present study because a larger number of toolpaths generally implies more auxiliary events and a higher potential for accumulated non-productive motion. A high number of manually prepared toolpaths indicates high programming effort. CAM programming time can be a critical production bottleneck and is performed by a skilled person under non-trivial risk (tool breakage,

collision). A greater number of toolpaths also implies a higher frequency of tool changes, each typically lasting 1–2 seconds. This represents a systematic factor that is readily quantifiable. However, it is not further discussed in the present paper.

The available processes used approximately 25 distinct toolpath types (CAM cycles). For reporting clarity, 15 of the most frequent types were presented individually and the remainder were grouped (Fig. 3).

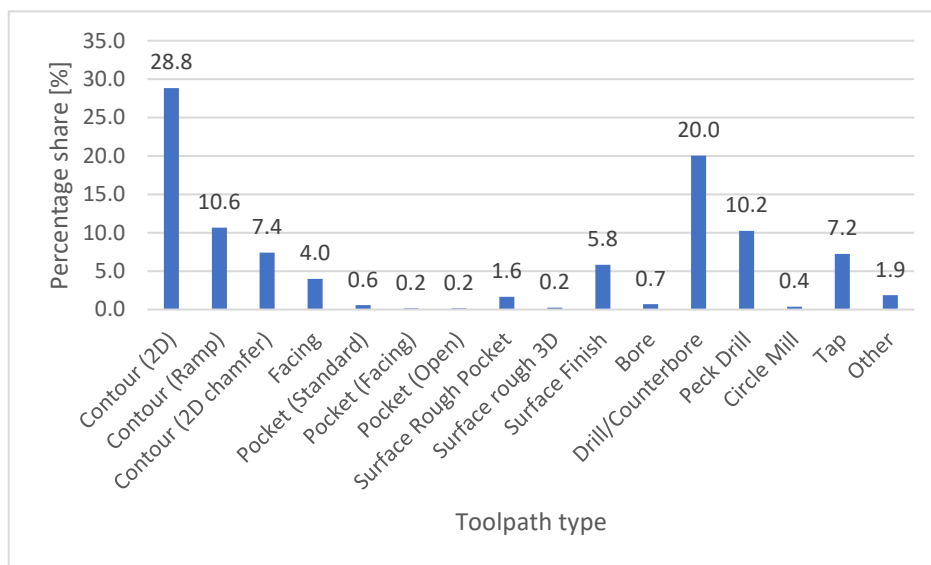


Fig. 3. Distribution of toolpath types across 242 operations

The distribution indicates that approximately 46.8% were contour-based operations (toolpath shape derived from boundary contours). Approximately 38.5% were hole-making toolpaths. The share of 3D freeform toolpaths (Surface...) was only 7.6%. This profile is consistent with machining of high-precision machine parts of moderate geometric complexity. It also suggests a relatively “traditional” technology approach, with limited use of modern cycles targeting constant tool load (e.g., constant engagement angle).

3.2. METHODOLOGY: RECONSTRUCTING EFFECTIVE AUXILIARY Z MOTION FROM CAM SETTINGS

The study focused on parameters related to 3-axis milling and vertical auxiliary motions. Depending on the toolpath strategy, these parameters are defined by technologists in different CAM dialog windows (highlighted fields in Fig. 4). A basic scheme is shown in Fig. 4 (left): it covers rapid vertical segments (yellow) and approach segments executed at a reduced feed (blue). Values entered by the technologist in the highlighted fields are extracted by CAM-Connect and stored in STEP-NC (XML). More advanced dialog windows exist to further refine auxiliary motions, especially for 3D toolpaths (Fig. 4, right).

A key point is that the interpretation of approach parameters depends on the selected option logic (e.g., Absolute, Incremental, Associative, Use clearance only). Therefore, parameter values must be analysed in the context of these selections.

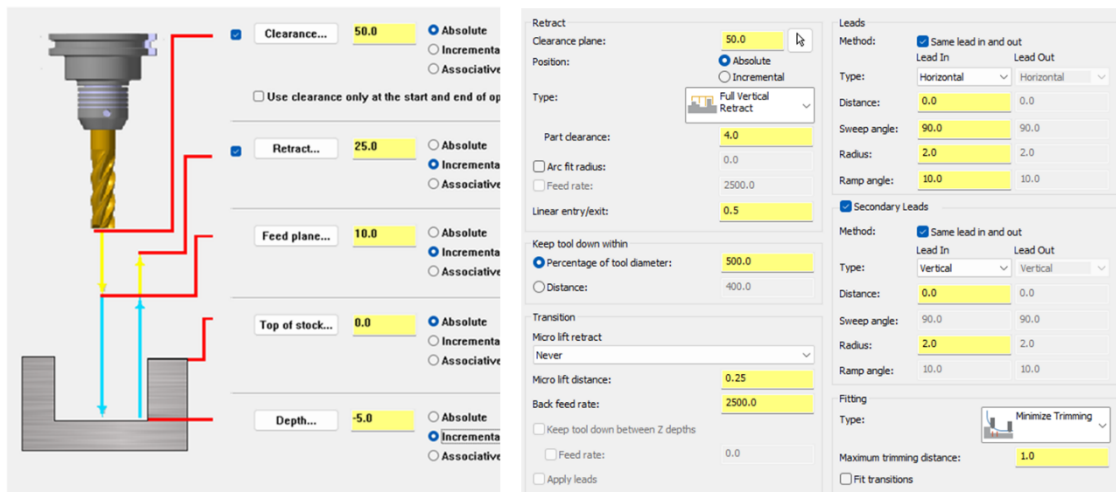


Fig. 4. Mastercam dialog windows used to define approach/withdrawal distances and speeds (left) and advanced auxiliary-motion settings for 3D toolpaths (right)

Moreover, toolpaths are often executed with stepped axial depths (a_p) and radial widths (a_e), including roughing and finishing passes. Many of these patterns can introduce additional Z-axis moves whose number and magnitude depend on CAM parameters.

For this reason, XML parameters must be further processed to compute the *effective Z* displacement (distance travelled in Z) and to separate it into phases executed at rapid and at feed. Accordingly, the XML parameters were processed in two complementary stages. First, the XML_READER module written in Python was used to analyse the 242 XML files and export the selected parameters to a consolidated CSV dataset. Second, Excel workbooks were used to interpret the parameters, calculate auxiliary distances and times, and generate statistical summaries and charts.

3.3. VERIFICATION AGAINST CAM AND NC SIMULATION

Verification of the computational rules was based on simple controlled examples and randomly selected CAM projects from the dataset. We compared computed rapid and feed auxiliary times with values reported by the built-in toolpath and material-removal simulations. In addition, generated NC programs were simulated using Cimco Edit.

During verification, discrepancies were observed in the reporting of auxiliary motion time and distance. We developed a dedicated NC-code analysis script and found that the differences originate from a systematic limitation of common simulations: Mastercam and Cimco Edit typically count only rapid moves (G0) as “auxiliary” when reporting non-productive-motion statistics, while feed moves (G1) are treated as “productive” by default. However, in practice, a substantial portion of non-productive motion can be executed with G1, e.g., feed approaches to a feed plane, controlled retracts, or clearance moves executed at a programmed feed for safety.

The CAM-Connect-based analysis is therefore more informative for diagnostics, because it explicitly identifies auxiliary moves executed with both G0 and G1 and can relate them back to the CAM settings that caused them.

3.4. RESULTS

We quantified the share of auxiliary Z-axis motion time relative to the total time of each toolpath type (Fig. 5). Figure 5 presents the percentage share of auxiliary-motion time within individual toolpath types. The highest shares observed for hole-making and selected contour-based operations indicate that these categories are particularly sensitive to CAM-defined approach, retract, and clearance settings, and therefore offer the greatest potential for targeted improvement. The results identify the largest improvement potential in hole machining and 2D contour milling. These categories should be examined with regard to parameters such as Retract, Feed plane, and Top of stock, as well as additional Z-motion parameters defining local retract/approach behaviour. The choice of these parameters depends on technologist experience, company conventions, machine capabilities, and the “human factor” related to operator preferences. Conservative long approaches and retracts are comfortable and perceived as safe during “calm” machining, but they can accumulate into significant time losses.

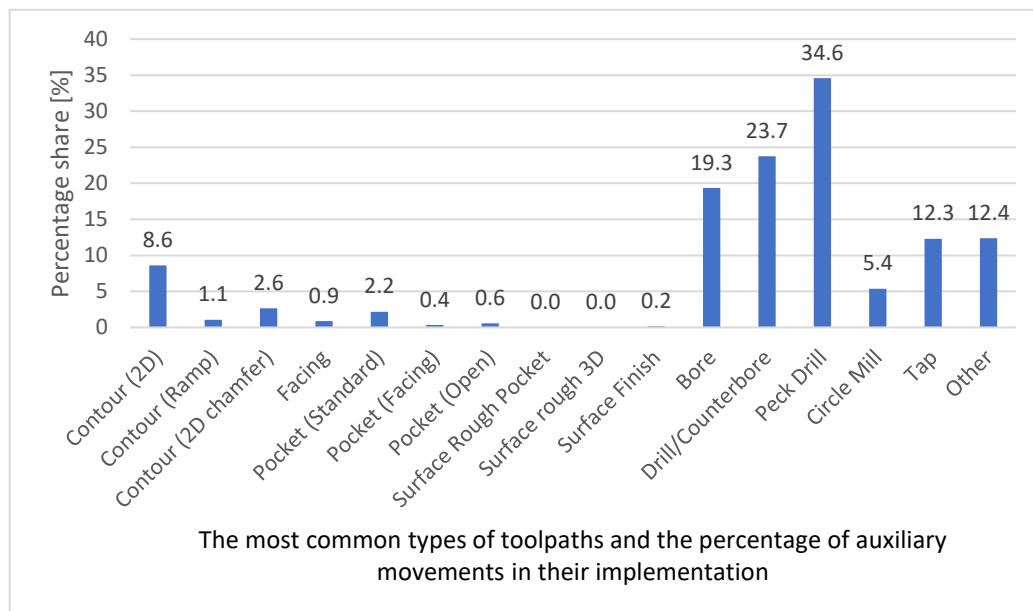


Fig. 5. Percentage share of auxiliary-motion time within a toolpath type (relative to total toolpath time) [%]

Next, we analysed the effect of auxiliary motions at the level of whole operations (Fig. 6). Figure 6 illustrates the share of auxiliary-motion time at the operation level, showing that the burden of non-productive motion is unevenly distributed across the analysed processes. This distribution is important because it identifies a subset of operations in which CAM-related parameter choices have a disproportionately strong impact on total machining time. At this stage, we applied a practical correction motivated by interpretability. Very short operations (e.g., 1–3 toolpaths) often exhibited a large auxiliary-motion share and long auxiliary distances in absolute terms, but when the entire operation lasted only a few seconds, the percentage could become misleading and not representative for general improvement decisions.

Additionally, some drilling cycles (e.g., specialised variants) inherently include significant Z-axis retracts required for chip evacuation. Such motions should not be interpreted as avoidable “losses” because they are necessary for process reliability. A small number of such cases (approximately a dozen) were therefore excluded from the operation-level statistics.

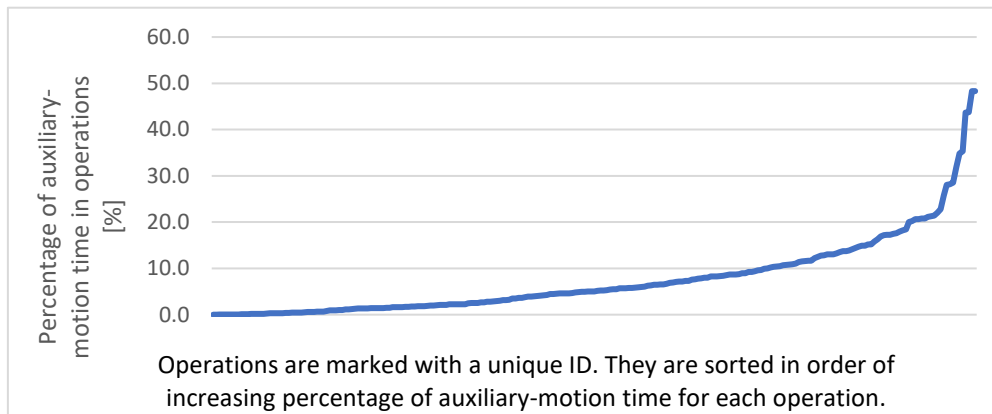


Fig. 6. (operation level). Percentage share of auxiliary-motion time across operations

After filtering, the arithmetic mean auxiliary-motion time across operations was 7.96% of total operation time. The maximum value reached 48.31%, but it occurred in a short operation (44 s) dominated by the excluded-like drilling behaviour. Overall, 14.5% of operations had auxiliary-motion time above 15%.

These values compare favourably with the literature indications summarised in Section 1, which often refer to broader categories of non-productive states (including idle, waiting, and system auxiliaries). The difference may be explained by the fact that this study uses real industrial jobs, an experienced technologist, and a focused definition limited to Z-axis auxiliary motions.

We further analysed the Z-axis auxiliary travel distance, distinguishing between rapid and feed components (Fig. 7). Rapid speeds were determined from machine settings and were assumed to be 20 m/min based on an interview with the technologist (average across machines). Feed speeds for auxiliary segments were taken from the programmed feed rates associated with the relevant toolpaths.

Figure 7 compares auxiliary Z-axis travel executed at rapid and at feed across operations. Although both distance components are of similar magnitude, the much lower speed of feed-executed segments makes them the dominant contributor to auxiliary-motion time, highlighting the limitations of analyses based solely on rapid traverses. Although the *distances* travelled at rapid and feed are comparable (Table 1), the speed difference implies that feed auxiliary segments dominate time. In the analysed dataset, auxiliary motions executed at feed accounted for approximately 96% of total auxiliary-motion time.

This result highlights a critical simplification in auxiliary-motion analysis based solely on NC-code categories (G0 = auxiliary, G1 = productive). A large portion of non-productive time may be embedded in G1 segments, and its identification is possible either at the CAM-project stage (the approach proposed here) or via shop-floor monitoring (e.g., force/vibration

thresholds) and material-removal simulation. However, monitoring and simulation typically do not provide a direct mapping from *CAM settings* to observed motion segments, which is essential for actionable guidance to technologists.

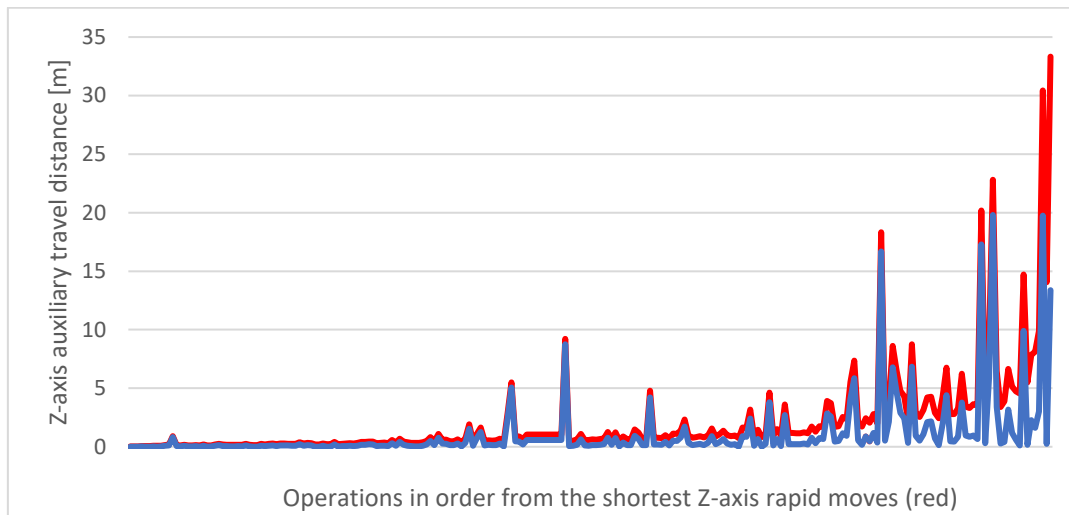


Fig. 7. Z-axis auxiliary travel distance [m]. Operations sorted by increasing rapid auxiliary distance (red). For the same operations, total auxiliary travel executed at feed is shown in blue

Table 1. Z-axis auxiliary travel distances

Motion type	Rapid [m]	Feed [m]
Maximum	33.3	19.8
Mean	2.2	2.2

4. SUMMARY

This paper demonstrated a CAM-based approach to analysing historical manufacturing jobs with a focus on non-productive motions. Using CAM-Connect to extract technologist-entered settings and a STEP-NC (XML) representation of process structure, we presented a practical method for identifying auxiliary motions—specifically vertical Z-axis approaches and withdrawals—including both rapid and feed-executed segments.

Key findings and contributions are:

CAM settings enable actionable diagnostics. By interpreting the same parameters edited by technologists (e.g., retract height, feed plane, clearance logic), the method links observed non-productive motion directly to decisions that can be improved.

Feed auxiliary segments can dominate time. While common CAM and NC simulations often count only G0 as “auxiliary”, the analysed dataset shows that auxiliary moves executed at feed can account for ~96% of auxiliary-motion time, despite similar travel distances to rapid moves.

Industrial baseline indicators were quantified. Across 242 historical 3-axis milling operations (2912 toolpaths) from one industrial environment, mean auxiliary-motion time was 7.96% of operation time after filtering special drilling cycles and extremely short operations; 14.5% of operations exceeded 15% auxiliary time.

CAM-Connect supports feedback loops and AI-ready data. The approach fills part of the missing feedback link between production and upstream planning/design by generating structured process records, exported geometry, and intermediate stock states. This can support ramp-up, training of new engineers, detection of inefficiencies, and AI-based process planning tailored to company specifics, as in Cybertech 4.

The present study is limited to one company, one technologist, a defined time span, and a focused subset of motion (*Z*-axis auxiliary). Future work will extend the analysis to *XY* auxiliary motions, normalised indicators (e.g., by tool diameter and feature size), and multi-company datasets. Another important direction is correlating CAM-setting-driven auxiliary-motion indicators with machine monitoring data (power, vibration, spindle utilisation) to strengthen diagnostic conclusions.

ACKNOWLEDGEMENTS

This research work was carried out based on data from "CYBERTECH 4 – intelligent design system of the technological process for the 4.0 platform", no. POIR.01.01.01-00-0547/20.

REFERENCES

- [1] MAKINO, *Cycle Time Reduction Secrets Revealed: Optimizing Non Cut Time*, <https://www.makino.com/en-us/resources/content-library/articles/cycle-time-reduction-secrets-revealed>.
- [2] PAWANR S., GUPTA K., 2024, *A Review on Recent Advances in the Energy Efficiency of Machining Processes for Sustainability*, *Energies* 2024, 17, 3659, <https://doi.org/10.3390/en17153659>.
- [3] SECOTOOLS, *Smart and Sustainable Machining Production*, https://www.secotools.com/article/smart_and_sustainable_machining_production.
- [4] STRÖBEL R., PROBST Y., HUTT L., FLEISCHER J., 2023, *Software-Defined Workpiece Positioning for Resource-Optimized Machine Tool Utilization*, *J. Mach. Eng.*, 23/1, 71–84, <https://doi.org/10.36897/jme/161660>.
- [5] DEMIR G.; ACAR VURAL R., 2024, *Non-Cutting Moving Toolpath Optimization with Elitist Non-Dominated Sorting Genetic Algorithm-II*, 14, 4471, <https://doi.org/10.3390/app14114471>.
- [6] HEO E.; YOO N., 2021, *Numerical Control Machine Optimization Technologies Through Analysis of Machining History Data Using Digital Twin*, *Appl. Sci.* 2021, 11, 3259, <https://doi.org/10.3390/app11073259>.
- [7] KELEKCI E., KIZIR S., 2023, *A Novel Tool Path Planning and Feedrate Scheduling Algorithm for Point to Point Linear and Circular Motions of CNC-Milling Machines*, *Journal of Manufacturing Processes*, Elsevier, 9, <https://www.sciencedirect.com/science/article/abs/pii/S1526612523003316>.
- [8] NISHIDA I., SHIRASE K., 2020, *Machining Time Reduction by Tool Path Modification to Eliminate Air Cutting Motion for End Milling Operation*, *International Journal of Automation Technology*, 14/3, 459–466, <https://da.lib.kobe-u.ac.jp/da/kernel/0100481863/0100481863.pdf>.
- [9] GUNAWAN H., SUGIONO D., TU R.Q., JONG W.-R., MUFARRIH A., 2025, *Enhanced Computer Numeric Controller Milling Efficiency Via Air-Cutting Minimization Using Logic-Based Benders Decomposition Method*, *Electronics*, 14, 2613, <https://doi.org/10.3390/electronics14132613>.
- [10] NISHIDA I., SHIRASE K., 2020, *Machining Time Reduction by Tool Path Modification to Eliminate Air Cutting Motion for End Milling Operation*, *Int. J. Autom. Technol.*, 14, 459–466, <https://www.fujipress.jp/ijat/au/ijate001400030459/>.
- [11] WINTECH, 2023, *How to Reduce Cycle Time without Reducing Tool Life in CNC Machining*, <https://ko.wintecht.com/resources/how-to-reduce-cycle-time-without-reducing-tool-life.html#:~:text=process%20tool%20by%20tool%20to,cut%20part%20of>.
- [12] JAROSZ K., LÖSCHNER P., NIEŚŁONY P., KROLCZYK G., 2017, *Optimization of CNC Face Milling Process of Al-6061-T6 Aluminum Alloy*, *J. Mach. Eng.*, 17/1, 69–77.

- [13] PAJAZITI A., TAFILAJ O., GJELAJ A., BERISHA B., 2025, *Optimization of Toolpath Planning and CNC Machine Performance in Time-Efficient Machining*, Machines, 13, 65, <https://doi.org/10.3390/machines13010065>.
- [14] VERICUT (<https://vericut.com/>), NX CAM (<https://www.siemens.com/en-us/products/nx-manufacturing/cam-software/>), Mastercam Simulator (<https://mastercam.com>).
- [15] KARUPPANAN B.R.C., SARAVANAN M. J., 2019, *Optimized Sequencing of CNC Milling Toolpath Segments Using Metaheuristic Algorithms*, Mech. Sci. Technol. 33, 791–800, <https://doi.org/10.1007/s12206-019-0134-3>.
- [16] ZALEWSKI A., SOBIESKI S., 2024, *Integration and Automation of Technological Knowledge in CNC Machining with CAM Systems*, Presentation at the KSN24 conference, <https://ksn2024.mech.pg.gda.pl/en/25-27.09.2024>.
- [17] ZALEWSKI A., SOBIESKI S., 2015, *Selection Strategy for Milling Parts on CNC Machines*, Advanced Technologies in Mechanics, 2/3, 9–23, <http://atim.media.pl>.
- [18] ZHANG X., NASSEHI A., NEWMAN.S, 2014, *Feature Recognition from CNC Part Programs for Milling Operations*, Int. J. Adv. Manuf. Technol., 70, 397–412.
- [19] RAUCH M., LAGUIONIE R., HASCOET J.-Y., XU X., 2009, *Enhancing CNC Manufacturing Interoperability with STEP-NC*, J. Mach. Eng., 9/4, 26–38.