Digital Twin Based Photogrammetry Field-of-View Evaluation and 3D Layout Optimisation for Reconfigurable Manufacturing Systems

Zi Wang[a], Likun Wang[a,b], Giovanna Martínez-Arellano[a], Joseph Griffin[a], David Sanderson[a], Svetan Ratchev[a]

[a]*Omnifactory Research Centre, Institute for Advanced Manufacturing, University of Nottingham, Advanced Manufacturing Building, Jubilee Campus, Nottingham, NG7 2GX, Nottingham, United Kingdom*
[b]*National Engineering Laboratory for Robot Visual Perception and Control , Hunan University, Changsha, Hunan, China*

Abstract

Photogrammetry is extensively used in manufacturing processes due to its non-contact nature and rapid data acquisition. Positioning photogrammetry cameras requires knowledge of the manufacturing process and time in manual field-of-view (FoV) adjustment. Such a lengthy and labour-intensive process is not suitable for modern manufacturing systems, where automation, robotics and dynamic reconfigurable layout are used to shorten production time and adapt to demand changes. Hence, there exists the need for a fast layout planning approach ensuring manufacturing process feasibility and maximising camera FoV effectiveness. This paper introduces a digital twin based FoV evaluation method and a computationally efficient 3D layout optimisation framework for reconfigurable manufacturing systems. The framework computes optimal layout for photogrammetry cameras and the object of interest (OOI). The automated nature of the proposed framework can speed up planning processes and shorten dynamic system commissioning time. At a technical level, the framework takes advantage of a 3D digital twin, and uses point clouds to represent the camera FoV. Iterative Closest Point (ICP) registration and K-Dimensional Tree (KDTree) intersection techniques are applied to calculate OOI visibility and target coverage ratio. Experimental validation attested to a digital-physical similarity exceeding 93%, indicating a high level of fidelity and the feasibility of station-level 3D layout design in digital twin environments. Feeding into the 3D layout planning, the optimisation framework considers robot reachability, FoV effectiveness, and estimated uncertainty. Given characteristics of the objective function, genetic algorithm, simulated annealing, and Bayesian optimisation were evaluated within a computational budget (100 function calls). The optimised results are compared against a baseline best obtained through brute force grid search. All tested algorithms achieved results within 98% of the grid search's best solution within 5 minutes. Genetic algorithm and simulated annealing outperformed the baseline best by 0.16% and 0.25% respectively for OOI visibility, and Bayesian optimisation exceeded the baseline best by 0.12% for target coverage. These findings emphasise the feasibility of the proposed approach and the efficiency of the overall framework, highlighting its applicability across various development stages from design to execution in a dynamic manufacturing environment.

Keywords: Digital Twin, Photogrammetry, Field-of-View, Measurement-Assisted Manufacturing, Layout Optimisation, Reconfigurable Manufacturing Systems, Heuristic Methods

1. Introduction

The manufacturing industry is facing global challenges of demand fluctuation, supply chain instability and labour shortage. As a result, today's production systems are evolving towards reconfigurable manufacturing systems (RMS). Introduced by Koren et al. (2018), RMS aims to maximise utilisation of robots, machines and automation tools for responsiveness under varying demands. Aligned with the same goals, Industry 4.0 is bringing in new opportunities through the utilisation of data to improve efficiency and quality. This evolution also triggered the increasing implementation of metrology. By providing measurement data, metrology integration enabled a new era of zero-defect manufacturing(Azamfirei et al., 2024). In the perspective of RMS, measurement-assisted assembly (MAA), introduced by Maropoulos et al. (2014), is emerging as an integral technology in the effort to achieve both flexibility and quality. Within this regime, optimisation techniques are often used to maximise system performance and effectiveness. For RMS, process flow (Azab and ElMaraghy, 2007; Chaube et al., 2012), scheduling (Bensmaine et al., 2014) and asset combinations (Torayev et al., 2023a,b) are optimised for cost, time, energy and capacity. Similarly, positions of metrology cameras are optimised for feature visibility and coverage (Zhang et al., 2021; Wang et al., 2023b,a).

Photogrammetry systems are extensively used for MAA applications. Their non-contact nature and rapid data acquisition allow for high-accuracy feedback which plays a crucial role in quality control and inspection (Li et al., 2018). However, measurement effectiveness is highly dependent on cameras' field-of-view (FoV) with respect to visibility of the object of interest (OOI), and coverage of measurement features. Both are directly dependent on layout design, although layout design prioritises manufacturing process over measurement process when integrated. As an example, industrial robots should be prioritised and positioned in the centre of a cell/workstation, maximising

their functional motion envelopes. This often leads to view obstruction and compromises measurement performance. Finding optimal camera positions and orientations is a tedious manual task, requiring high levels of expertise and experience (Wang et al., 2023b, 2022d). In RMS, where layout is constantly changing, the current process is not only labour-intensive, but also poses significant safety risks in robotic environments. Furthermore, manual evaluation without considering overall measurement uncertainty and process constraints cannot provide a conclusive answer. Nonetheless, existing optimisation developments for RMS and metrology are rather isolated, due to the different levels of abstraction at system-level and machine-level. To the best of authors' knowledge, there is very limited research in the automated and computationally efficient evaluation of camera FoV and 3D layout optimisation in the context of MAA.

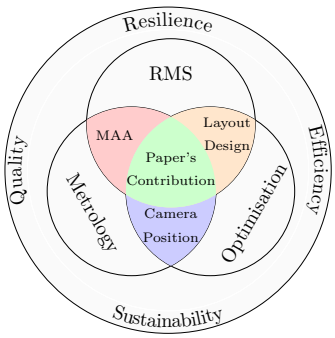


Figure 1: Research gap and paper's contribution with respect to RMS, metrology and optimisation

This paper proposes a novel approach to simulate and evaluate photogrammetry camera FoV in a digital twin environment, and provide an efficient and integrated framework for layout planning and optimisation in these scenarios, considering both camera-level and system-level constraints. As illustrated in Figure 1, the paper is bridging a gap between the research areas of RMS, metrology and optimisation. The proposed methodology for FoV evaluation and optimisation is novel in the following perspectives:

1. It allows FoV simulation of photogrammetry cameras in measurement-assisted manufacturing scenarios within a digital twin environment.
2. It is the first layout optimisation approach for metrology-assisted manufacturing systems at workstation/cell-level in 3D, enabling 6 degrees of freedom (DOF) analysis.
3. It is able to quantify visibility/coverage for the OOI against nominal Computer-Aided-Design (CAD) data and actual retro-reflective feature (also known as targets) characterisation, allowing high level of geometry complexity and realism.
4. The integrated optimisation framework is computationally efficient, making it suitable for both online and offline decision-making across various stages in RMS lifecycle.

The rest of the paper is structured as follow: A review of relevant investigations, emphasising the research gap, is presented in Section 2. With camera FoV evaluation processes and optimisation methodology outlined in Section 3, a use case study is presented in Section 4 with experimental validation in Section 5. Finally, the optimisation results are discussed in Section 6.

2. Related Work

2.1. MAA in Flexible Manufacturing

As Industry 4.0 introduced a boom in metrology integration in manufacturing processes, applications and challenges of optical metrology in digital manufacturing are summarised in a recent review by Catalucci et al. (2022).

MAA is usually implemented in large-scale, high-complexity and low-rate scenarios, such as aerospace manufacturing. Typical MAA processes include measurement-assisted determinate assembly (MADA), work-piece positioning, and robot tool centre point (TCP) control. MADA uses measurements from interfacing features, such as hole patterns or topography, to manufacture the corresponding interface. The reduced deviation between interfacing features minimises in-built stress and reduces scrap rate, which is critical in load-bearing aerostructures Silk and Andrews (2010). Work-piece positioning is necessary in flexible systems, where the use of bespoke and monolithic jigs are minimised. Position measurements around complex geometry and assembly environments bring challenges to feature visibility. To overcome the visibility issue, it is common to combine different metrology systems. Wang et al. (2022b) combined laser tracker, 3D scanner and photogrammetry cameras to measure gaps in wing-fuselage assembly, and to optimise assembly poses, where operating and positioning of these metrology devices are manual. Similar manual operation of metrology systems was applied by Wang et al. (2022d) in an automated environment, where the combination of photogrammetry cameras and laser tracker to achieve robotic reconfiguration repeatability. Both studies still heavily rely on the manual setup to position respective metrology systems. Lastly, robot TCP can be controlled with metrology feedback to improve accuracy, a move-measure-correct approach is often adopted (Drouot et al., 2017; Sanderson et al., 2019; Azamfirei et al., 2021). It is evident that the integration of metrology systems can elevate modern manufacturing to another level of accuracy and precision. However, these studies are based on static environments for specific processes. FoV effectiveness in these integrated systems is rarely studied, largely because these investigations focused on feasibility of the process itself rather than its practicality within a constantly changing environment. Addressing requirement changes, a Unified Modeling Language (UML) based framework to select metrology systems on measurability was proposed by Muelaner et al. (2010). The paper outlined the importance of a 3D assessment of the measurement environment, but did not provide any metrics or algorithm for this purpose. Yet Barazzetti (2017) noted that "the growing number of users of photogrammetric/computer vision automated software has also led to an increment of crude digital reconstructions without metric integrity". Therefore, it is evident that there exists the need and a research gap for parameterised

FoV evaluation and the corresponding layout optimisation for a dynamic manufacturing environment.

2.2. Layout Design in RMS

Layout optimisation, aiming to find the best spatial arrangement of assets, is another established area of research. For RMS, existing research focuses on system-level time/cost reduction for the next production configuration or for the reconfiguration process itself. Yamada (2006) used particle swarm optimisation (PSO) to minimise transitioning distancing between configurations. Guan et al. (2012) investigated material handling cost through autonomous guided vehicle (AGV) movement and minimising distance between different stations with genetic algorithms (GA). In a similar setting, Besbes et al. (2021) included considerations of human safety bubbles. With added complexity of object-oriented definition of workstations/cells, discrete event simulations (DES) were adopted in this context. Zheng et al. (2013) used DES with GA to minimise reconfiguration cost. Petroodi et al. (2019) minimised total production time with simulated annealing. In some cases, layout is treated as a combinational problem, where only feasible combinations are generated (Benderbal et al., 2018; Santos et al., 2021; Leng et al., 2020; Touckia et al., 2022). Yet all aforementioned work is based on system-level simplification that a functional workstation/cell is contained as a unit and a 2D approximated shape is assigned to it as its volume. The parameters of investigation focused on the distancing between them, and the combination of numerical attributes. Consequently, functional consideration at the station level and the process level is seldom accounted for.

With the advancement of computer graphics and Industry 4.0 technology, digital twin (Grieves, 2014; Grieves and Vickers, 2016), an accurate virtual representation of a physical system, can be used for station/process level layout planning and optimisation. However this is rarely used in 3D layout planning. The only related research that considers 3D digital twin are by Wang et al. (2022a) and Arnarson et al. (2023). Wang et al. (2022a) used a multi-agent approach with PSO obtaining feasible layouts with minimised reconfiguration cost and overall lead time. Arnarson et al. (2023) used an multi-objective Non-dominated Sorting Genetic Algorithm (NSGA-II) and solved a similar set of constraints. However, robot pick and place was the main subject in both studies, yet no camera was involved, limiting the research to low-accuracy applications. In summary, previous studies on RMS layout design and optimisation focused on 2D planning with system-level objectives, such as cost and time. There exists a gap in 3D planning and process-level objectives. When adopting metrology-assisted processes in RMS, layout design requires 3D planning, motivating the proposed research that existing literature fails to address.

2.3. Camera Position Optimisation

When it comes to maximising FoV coverage, 2D assumptions are often made to simplify large-scale problems. Konda and Conci (2013) studied maximum coverage for complex indoor environments with a PSO approach. Strubel et al. (2017) used a hybrid approach combining PSO and GA to optimise positions in a camera network mounted on an unmanned aerial vehicle (UAV) for ground coverage. Rangel et al. (2019) adopted a multi-objective optimisation, combining a lexicographic algorithm + NSGA-II, to maximise adaptive redundancy in a large camera network to improve network performance in civil applications. Hocine and Benaissa (2021) proposed a new binary PSO to maximise visual coverage for a surveillance camera network in 2D. Zhou et al. (2024) considered optimising both reconstruction error and coverage ratio using a hybrid heuristic combining PSO and GA. In industrial inspection and dimensional measurement, objects and environments need to be in 3D for station-level process-level planning.

Camera modelling for 3D objects has a higher level of complexity and the optimisation can be time-consuming (Zhang et al., 2019). Some attempts have been made using knowledge-based systems, (Mason and Grün, 1995), simulation (Olague, 2001), and point cloud representation (Zhang et al., 2021). However, they focused on inspection-only scenarios and did not discuss their application when integrated with other agents in manufacturing processes. When combined with MAA, it requires additional consideration of process constraints, such as reachability, collision and uncertainty requirement. Wang et al. (2023a,b) investigated camera positions considering robot motion constraints in a 3D digital twin. A deep reinforcement learning approach was used in both, producing reusable layout knowledge. But extended training time plus a rigid reward function definition make it inefficient to implement under constantly changing environment for RMS, hence have limited value for execution.

2.4. Knowledge Gap and Motivation

Even though MAA allows an extended production flexibility by removing the need for monolithic fixtures, its development focused on precision and accuracy of the enabled process, rather than how to design and optimise the process for reconfiguration. This paper guides for rapid design and optimisation of such integrated systems.

Traditional layout research is focused on system-level objectives, such as material flow and other logistic factors, therefore uses 2D simplification for workstation/cells. However, manufacturers are moving towards more modular and flexible systems. For example, BAE Systems' Factory of the Future (BAESystems) and Advanced Manufacturing Research Centre's Factory 2050 (AMRC) as discussed by Sanderson et al. (2024). These systems contain modular workstations with closely integrated sensors/measurement data/robotic processes. Reconfiguration of these multi-functional workstations is rarely investigated, and cannot be addressed by logistic-focused layout research. Therefore, this paper introduces a unique layout optimisation approach for process-level objectives related to workstation layout.

Similarly, camera positioning at large scale also assumes 2D coverage. 3D object visibility analysis is computational demanding, and 3D camera positioning in manufacturing context is under studied. With emerging simulation capability and digital twin technology, 3D cell layout design and optimisation is becoming feasible, yet an efficient approach should be investigated for rapid reconfiguration.

The paper builds upon our previous works on the 3D digital twin layout optimisation (Wang et al., 2022a) and camera positioning framework (Wang et al., 2023a,b). It extends layout optimisation to MAA applications, allowing cameras and the OOI to be considered together. Meanwhile, new camera FoV evaluation approach and optimisation strategies are introduced to account for more realistic and complex OOI geometry in all 6 DoFs in a computationally efficient manner.

3. Methodology

The proposed framework follows a structure depicted in Figure 2. A digital twin environment outputs process information, such as reachability status, point cloud and estimated uncertainty. Point cloud is then analysed through the FoV evaluation process to compute OOI visibility and target coverage, before formulating the objective function describing layout suitability. The optimiser relates the layout state to a suitability score, and iteratively searches for the optimal state within a defined space.

The evaluation scheme is based on processing 3D point cloud. With the advancement of machine vision and image analysis techniques, 2D image-based analysis is also possible for the same purpose, when quantifying and maximising the OOI visibility for each camera view frame in pixel space. However, there lacks a holistic perspective when merging collective output from a multi-camera system. Especially with photogrammetry, multiple viewpoints are essential for triangulation, leading to the choice of 3D point cloud based evaluation for this study.

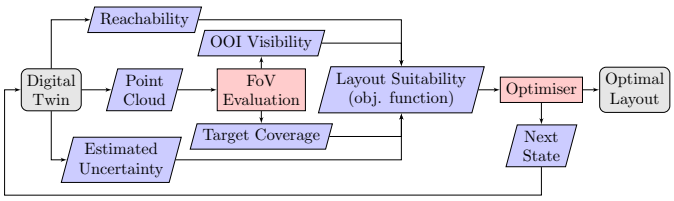


Figure 2: Digital-Twin Based FoV Evaluation and 3D Layout Optimisation Framework Structure, where red cells indicate analysis processes, and purple cells illustrate data, gray cell showing input and output of the framework.

This section will explain the FoV evaluation workflow for both OOI visibility and target coverage in Section 3.1, then, in more details, the point-cloud analysis techniques used in the evaluation process in Section 3.2, and the optimisation strategy and algorithms in Section 3.3. Lastly, the software architecture to enable these elements is summarised in Section 3.5.

3.1. FoV Evaluation

Photogrammetry triangulates multiple 2D images to calculate the 3D coordinate of a point in space, it is contact-free and able to measure a large volume in a very short duration, hence being a preferred option for automation. As stated by photogrammetry practitioners (Liu et al., 2012; Geodetic Systems), a measurement is only effective with

1. well-distributed common feature/target points across the camera's FoV;
2. as many as possible common feature/overlapping target points from different camera viewpoints.

When applied in MAA, measurements are taken for OOI positions. Therefore, one needs to quantify common area on the OOI to maximize the count of and distancing between retro-reflective targets. In a later stage, when retro-reflective targets are added to the OOI, one needs to maximise the target coverage within FoV to take quality measurement. However, robotic processes would often prioritize robot reachability, resulting in view obstruction in camera FoV. In non-contact metrology, such as photogrammetry, visibility (line of sight) of the targets is critical for measurement accuracy. Manual positioning requires a high level of expertise, yet poses safety risks in automated environments.

In this section, a novel two-step FoV evaluation approach is proposed, simulating camera measurements for 1) quantifying visibility of the OOI, and 2) examining retro-reflective target coverage ratio. Maximising OOI visibility is the prerequisite for a well-distributed arrangement of retro-reflective targets, and maximising target coverage ratio is critical in reducing measurement uncertainty. Therefore, OOI visibility evaluation should be used in the early planning stage, and target coverage analysis, based on real-world target arrangement, should be implemented to refine the solution in the commissioning stage.

3.1.1. OOI visibility

Figure 3 shows the evaluation scheme for OOI visibility based on point cloud analysis with a dual-camera system. When a photogrammetry measurement is fully automated, it requires multiple viewpoints to perform triangulation. This implies that a minimum of two cameras are required for in-process near-real-time measurement. The FoV evaluation starts with performing Red-Blue-Green-Depth (RGBD) snapshot from the 3D digital twin environment, capturing point clouds from two viewpoints. A two-step registration process is performed to reconstruct the 3D scene. Based on spatial information from the digital twin, the OOI can be cropped from the scene before analysing point clouds overlap. K-Dimensional Tree (KDTree) intersection is then carried out to extract the common area that is being seen by both camera viewpoints. From here, a CAD point cloud, sampled from a stereolithography (STL) file, is imported and compared with the extracted common area, returning 1) the OOI visibility p_{vis}, as a ratio between visible area against the entire object, and 2) common area mapping, as a visual guidance to setup retro-reflective targets.

3.1.2. Target coverage

In commercial photogrammetry applications, retro-reflective features are used to increase measurement efficiency. These retro-reflective features are referred to as targets. Instead of measuring the entire OOI, the 3D position of these targets are measured, and the OOI is characterised as sparse target points labeled and linked to a datum coordinate system. Knowing the OOI visibility and the overlapping area should help engineers to best layout well-distributed targets. This also means that the aforementioned parameter OOI visibility (p_{vis}) can no longer

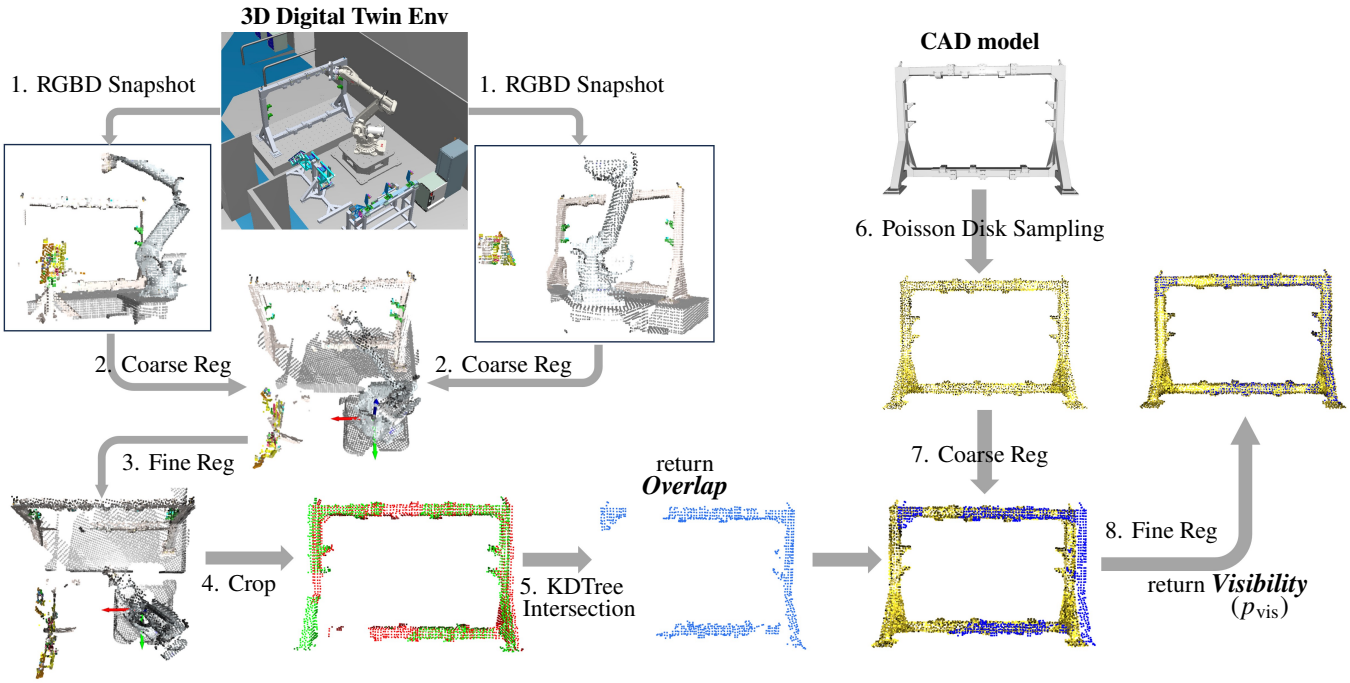


Figure 3: FoV evaluation scheme for OOI visibility: 1) RGBD Snapshot: Capture point cloud from 3D digital twin environment from camera view frames; 2) Coarse Registration: align the point clouds with known camera view frame transformation; 3) Fine Registration: iterative closest point registration to align and reconstruct the scene; 4) Crop: extract OOI from the scene, Red: from right camera; Green: from left camera; 5) KDTree Intersection: Obtain overlapping area between two camera viewpoints, return visualization of common area; 6) Poisson Disk Sampling: Translate CAD model into a uniform point cloud; 7) Coarse Registration: align CAD point cloud and scene point cloud with known datum transformation; 8) Fine Registration: ICP registration to calculate fitness (aka object visibility).

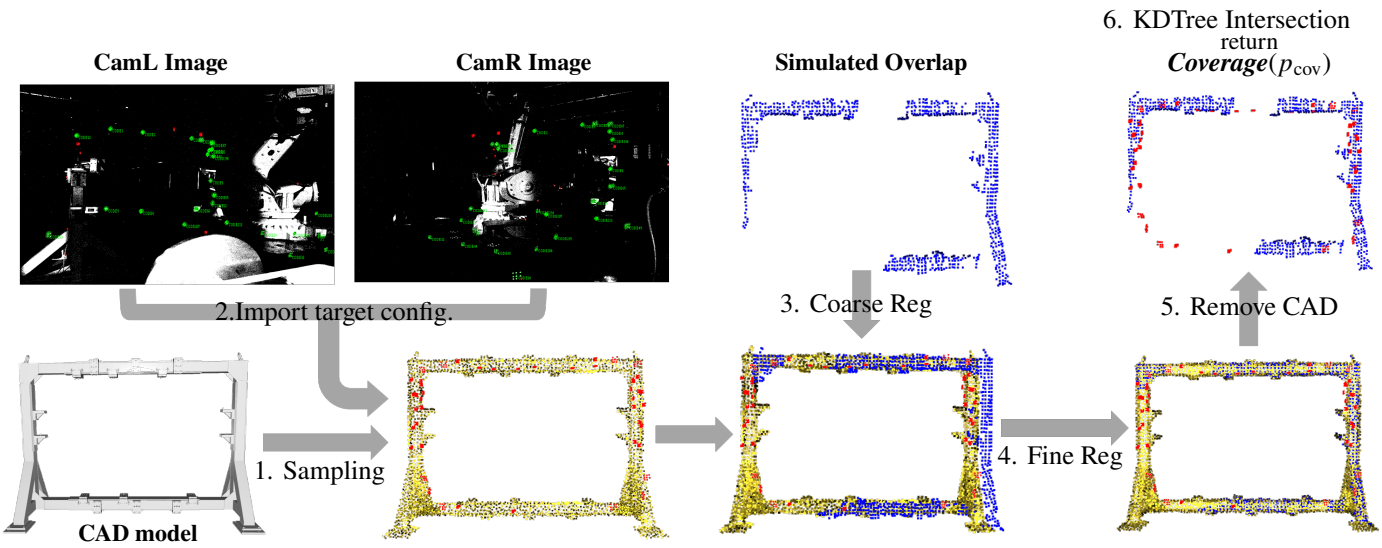


Figure 4: FoV evaluation scheme for target coverage: 1) Sampling: Poisson down-sampling of CAD model; 2) Triangulation: Photogrammetry system measures retro-reflective target locations relative in object datum coordinate; 3) Coarse Registration: import simulated overlap to CAD in object datum coordinate; 4) Fine Registration: ICP registration to align simulated overlap (blue) to CAD (yellow) point clouds; 5) Remove CAD: Remove CAD point cloud; 6) KDTree Intersection: Find overlap between retro-reflective targets (red) and simulated overlap (blue) and return target coverage ratio.

indicate FoV effectiveness. Instead, this should be represented by how many targets are observed in the camera FoV. Therefore, we introduce target coverage p_{cov} for when the OOI is configured with targets, often in system commissioning stage.

A similar process is proposed and illustrated in Figure 4. The process starts with Poisson sampling a point cloud from the CAD model. Target configuration, captured from actual camera measurements, is then imported to the CAD datum and formed the combined baseline. Simulated overlap, as output from Figure 3 - Step 5, is compared to the combined baseline through coarse-to-fine registration, aligning the scene observation to the CAD datum. Then, CAD point cloud is removed and the target coverage ratio p_{cov} is obtained by KDTree intersection between the scene observation and the target configuration.

3.2. Point-cloud Analysis

The FoV evaluation processes used point cloud representation of the scene to quantify OOI visibility and target coverage, and relied on the generation and analysis of point cloud data. This section will introduce in more detail the steps and techniques used for the FoV evaluation process.

3.2.1. RGBD Snapshot

RGBD snapshot is a process that simulates a depth camera taking an RGB picture and computes the depth of each pixel. Although photogrammetry and depth cameras have different working principles in calculating depth, they share commonalities in optical imaging and projection, where a FoV is defined. To simulate the FoV of a photogrammetry camera, a camera object in the digital twin environment is defined with three parameters, $\{w, h, \alpha_{\text{FoV}}\}$, where (w, h) are the width and height for the image plane, α_{FoV} is the diagonal FoV angle. This is based on the assumption that RGB and depth sensors share the same parameters, which is true when simulating photogrammetry cameras. In the cases of an RGBD camera having different RGB and depth settings, this is possible to adjust these parameters too.

When a snapshot is triggered, the camera object returns a RGB bitmap and a binary file containing the depth of each pixel. In addition, spatial information of a defined list of assets, including camera extrinsic parameters, is also captured, each as a 4x4 transformation matrix. These constitute necessary output files to create the point cloud representation of the scene for the evaluation process.

A dual-camera system is used in the demonstration of the proposed approach, however, the FoV evaluation method does not restrict the number of cameras, and the processing time required will proportionally increase with the number of cameras.

Computational efficiency is an important attribute of any online/offline layout planning tool. In the context of RMS technologies, this is the prerequisite, as factory downtime is directly linked to cost. The layout optimisation framework should take a small, almost negligible, fraction of time compared to the time required for physical rearrangement. For a framework structured as Figure 2, the optimisation process is likely to iterate over hundreds, if not thousands, of steps. The processing time should be kept minimum for a single evaluation process. The total processing time of each step/episode consists of socket communication time ($t_{connect}$), relocating time ($t_{relocate}$), and lastly the evaluation time (t_{eval}).

$$t_{\text{ep}} = t_{\text{connect}} + t_{\text{relocate}} + t_{\text{eval}}, \quad \text{where} \quad t_{\text{eval}} \propto N_{\text{pts}} \tag{1}$$

Socket communication time and relocating time are fixed and relatively small. The more computational demanding part are the point cloud based calculation involving thousands of points. Since the evaluation time is directly proportional to total point counts (N_{pts}), two types of down-sampling exercises were used in the proposed approach:

1. **Camera Down-scaling**: Photogrammetry cameras have a large number of pixels, typically between 10 -20 megapixels depending on the measurement volume. For simulating camera FoV, micron-level resolution is unnecessary, yet negatively impacts processing efficiency. Consequently, camera down-scaling is needed, where the numbers of pixels in width and height each are reduced by a scaling factor of F_s, decreasing the total pixel count by F_s^2.
2. **Voxel Down-sampling**: Another level of down-sampling happens when importing point clouds into the evaluation process. Before aligning the point clouds to reconstruct a 3D scene, they are grouped in voxel space, and each occupied voxel generates one point being the average of all points inside, reducing the total point counts in the point cloud. Down-sampling voxel size is chosen based on application geometry. For OOI visibility only analyses, the down-sampling rate can be set at the same level as CAD model sampling rate and should be governed by OOI dimensions. If the resulted overlap mapping needs to be used for further target coverage analysis, the down-sampling rate should be closer to the size of the retro-reflective targets.

Comparing two down-sampling methods, camera down-scaling is more effective, but also more aggressive. Hence, a balancing analysis should be performed, where the scaling factor, pixel counts, resultant fitness ratio and run time should be compared. The balancing analysis for a dual-camera system will be demonstrated in a case study in Section 6.

3.2.2. Point Cloud Sampling from CAD

The CAD point cloud is sampled from an STL (Stereolithography) format using Poisson Disk sampling. The most complete geometry about the OOI is described in CAD data. The STL format, simplifying geometries into triangular meshes, is compatible with most CAD software packages, therefore chosen for the study. The collection of vertices, being the corner points for each mesh, is already a form of point cloud. However, mesh sizing is not uniform across the 3D object, especially when working with a detailed design. This will result in a non-uniform point cloud, which will further affects the point-based calculation of visibility and coverage ratio. Poisson Disk sampling with elimination (Yuksel, 2015) is used to create a uniform point cloud. Poisson disk sampling randomly generate points on mesh surfaces and eliminate points that are too close,

resulting in a uniformly distributed point cloud. More sampling points can capture more details, however, affects the overall processing time for the optimisation. In this paper, CAD sampled point cloud and RGBD scanned OOI has a similar level of point density, providing sufficient features for alignment. Since the visibility and coverage calculation are macro-scale evaluation, finer details, such as fasteners and couplings, can be neglected.

3.2.3. Point Cloud Registration

Point clouds are captured within the camera view frame, and registration is required for scene reconstruction. Registration is the process of aligning point clouds in a common coordinate system, implemented in two steps (2) coarse registration and (3) fine registration, as illustrated in Figure 3. Coarse registration uses extrinsic parameters as layout information provided by the digital twin. However, slight inaccuracy in spatial data can introduce misalignment, especially in orientation terms. A fine registration is needed to correct this computational inaccuracy. Point-to-plane iterative closest point (ICP) registration, proposed by Chen and Medioni (1992), is used to finely align these slightly misaligned point clouds. This is a faster variant compared to the point-to-point registration (Rusinkiewicz and Levoy, 2001). The error function is defined as

$$\varepsilon(\boldsymbol{T}) = \sum_{(\mathsf{p},\mathsf{q})\in\mathcal{S}} ((\mathsf{p} - \boldsymbol{T}\mathsf{q})n_{\mathsf{p}})^2, \tag{2}$$

with $\mathcal{S}$ being the corresponding set of target point cloud p and source point cloud q, n_{p} being the estimated surface normals for points in p, and the goal being finding the transformation matrix $\boldsymbol{T}$ minimising the error function $\varepsilon(\boldsymbol{T})$. As the result of the ICP process, a transformation matrix $\boldsymbol{T}$ that minimised the error function is obtained; a fitness ratio f as an indicator of overlapping area, computed as the ratio between registered points and total point count in the source cloud p; and the inlier RMSE (root-mean-square-error) calculated with all registered points. Fitness ratio f_{cam} resulted from Step 3 is an indicator of common areas covered by different camera view points.

The same registration technique is also used in Step 7 and 8 from Figure 3, to align the nominal CAD point cloud and the captured point cloud, where a fitness ratio f_{CAD} demonstrates the OOI visibility compared to the original CAD.

3.2.4. KDTree Intersection

Having reconstructed the scene, one can use known object position (output from the digital twin) to crop out the bounding box where the OOI resides, obtaining its point-cloud representation. In Figure 3, a visual representation of the reconstructed frame is shown at Step 4, combining the points captured from the left and right camera viewpoints. However, triangulation relies on common features on the OOI, meaning the points unique to a camera viewpoint do not contribute to the measurement. Overlapping points from both point clouds need to be obtained, and a KDTree intersection process is used to obtain the common points in the point cloud set.

Figure 5 shows the process of intersection through querying the KDTree of a source point cloud. When two point clouds (red and green) are aligned, the points do not sit exactly precisely together. Therefore, a contain function over the point cloud set will unlikely to return a reasonable solution. KDTree, a binary data structure proposed in Maneewongvatana and Mount (1999), is used to query nearest neighbors in group green of group red within a radius distance. Once the KDTree is mapped, empty nodes and the corresponding points from group red are removed, returning the intersection of the point cloud set. Comparing with voxelisation intersection approach, where each point from group red is voxelised and queried for whether it contains a point from group green, a spherical node is used in KDTree queries. This removes the corner effect of voxel, where a further diagonal point is included while a closer point isn't, ensuring the robustness of the proposed approach.

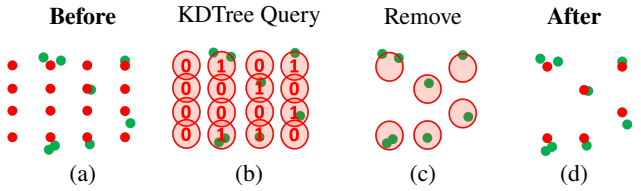


Figure 5: KDTree intersection process to remove non-overlapping points, (a) Before: Two point clouds with points overlapping; (b) KDTree Query: Turn source point cloud into a KDTree querying within a radius distance; (c) Remove: Check if KDTree contains points and remove empty nodes; (b) After: return the overlapping points

3.3. 3D Layout Optimisation

Maximising the visibility and target coverage on an OOI will contribute to the effectiveness of a measurement process, however, it should not be the only concern in cell layout planning. In this section, a holistic approach is introduced considering not only the FoV of cameras, but also the estimated measurement uncertainty and reachability of robot key poses. Moving the OOI far from the camera will help to maintain a good level of visibility/coverage, however, at the expense of unreachable positions and high measurement uncertainty. In our previous work Wang et al. (2022a), cell compactness, as a bounding volume of a cell was used to reduced material handling cost in 3D layout optimisation. In this paper, functional indicators, reachability and measurement uncertainty, are considered instead, and both address cell compactness. In summary, an objective function considering the aforementioned aspects is presented in Eq. 3.

$$\mathbf{f}(\mathbf{s}) = \begin{cases} w_{\text{FoV}} p_{\text{FoV}} + w_{\epsilon} p_{\epsilon} & \text{when } R = True; \\ 0 & \text{when } R = False, \end{cases} \tag{3}$$

where $\mathbf{s}$ being the state of the cell layout, R is a binary indicator of key pose reachability, p_{FoV} and w_{FoV} are the performance indicator and weighting factor for FoV evaluation, and p_{ϵ} and w_{ϵ} are the performance indicator and weighting factor for measurement uncertainty. The following subsection will present more details in obtaining these parameters and why they are important in layout planning.

Reachability

Key robot poses should be queried for reachability, ensuring the robotic task can be performed. Robot poses that involve direct interaction with tooling and parts should be included, such as pick/place position, drilling and welding points etc. In our previous work Wang et al. (2023a,b), collision avoidance was prioritised. However, collision can often be resolved by changing robot path or pose configuration. In addition, collision at key poses reflects major errors in the early mechanical design stage rather than the layout design stage. Although being a safety critical factor, collision should be ensured outside of layout planning.

FoV Evaluation

The FoV effectiveness p_{FoV} is represented by either p_{vis} or p_{cov} depending on the analysis goal. The process of obtaining p_{vis} and p_{cov} are outlined in Section 3.1 and 3.2. Although there are two methods in calculating p_{FoV}, it is the performance indicator for camera FoV and it is presented as a fitness ratio to a baseline point cloud. For OOI visibility, the baseline point cloud is generated from the CAD model ($p_{\text{FoV}} = f_{\text{CAD}}$). For target coverage, the baseline point cloud is the characterisation of coded targets, and p_{FoV} is calculated through KDTree intersection. A positive weighting factor w_{FoV} should be assigned, meaning a better visibility/coverage contributes positively to the measurement effectiveness of the layout.

Estimated Measurement Uncertainty

In practice, all measurements are subject to uncertainty. In photogrammetry applications, visibility/coverage can help reduce the overall measurement uncertainty, but not remove it. Uncertainty inherited from the device itself and its working principles is unavoidable. For simplicity, most metrology manufacturers use a linear relationship to describe the uncertainty expectation with regard to measurement distance. In this paper, a distance-based linear estimation of the measurement uncertainty is used.

$$p_{\epsilon} = A + B \times d_{\max}, \tag{4}$$

where A and B are constants that characterise the uncertainty of the photogrammetry system, and $d_{\max}$ is the maximum distance between the camera view frame and the measured object. This parameter negatively influences the objective function, as higher uncertainty negatively impacts the measurement process.

3.4. Optimisation Strategy

The optimisation strategy depends on the characteristics of the objective function. In this paper, the definition of the proposed 3D layout planning approach has the following attributes:

1. The objective function has minimal noise, since it was captured from a virtual environment.
2. The objective function is a derivative-free/black-box function, in which gradient information is unavailable.
3. The objective function is computationally expensive to run, therefore, a computational budget should be set.
4. The optimisation can be solved with a non-deterministic approach, and a goal threshold can be defined.

Three heuristic optimisation strategies, namely stochastic, population-based and probabilistic, are investigated for the black-box problem, within which three algorithms are chosen and they are Simulated Annealing (SA), Genetic Algorithm (GA) and Bayesian Optimisation (BO). In addition, deep reinforcement learning strategy in optimisation was explored by Wang et al. (2023a,b) These machine learning (ML) models take a long time to train, in the scale of days. In a cost-sensitive RMS environment, they have to be well planned into the reconfiguration process and executed well before physical rearrangement. In addition, the difference between trained layout and new layouts can also affect the effectiveness of the trained ML model. On the other hand, heuristics optimisation is computational-lean, and requires no training, therefore suits on-site operations and can be easily modified and re-run at any stage of the reconfiguration.

For robustness, the initial point is set as random and each algorithm is tested repeatedly for 10 times, obtaining the root-mean-square (rms) and mean absolute deviation (MAD) across all steps and optimisations. The comparison process is illustrated in Figure 6. For each algorithm, exploration and exploitation are balanced to achieve the best optimisation results within 100 function calls. Again this requirement is introduced to achieve rapid reconfiguration. Exploration focuses on discovering the unknown space at the expense of computational budget, while exploitation focuses only on potential improvement, hence utilises the acquired knowledge, however, is at a higher risk to stuck in a local optimum. The rms and max scores for optimal results obtained with balanced hyperparameters for each algorithm is then compared with a baseline. The baseline is calculated by a brute force grid search, in which the bounding space is divided into a finite gird assuming uniform conditions within the node and each node point is consulted forming a global understanding of the bounding space.

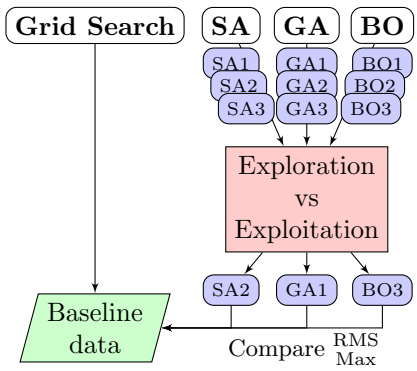


Figure 6: Investigation flow for optimisation algorithms, with purple block representing processes with 10 repeats.

Simulated Annealing

SA randomly searches the bounding space with decreasing step size, and goes towards a better solution with a possibility of accepting a worse solution to avoid local optimum. The

generalised simulated annealing approach proposed by Tsallis and Stariolo (1996) is used here. The step size ($\Delta\mathbf{s}$) is generated by a distorted Cauchy-Lorentz visiting distribution $\mathsf{g}_{q_v}(\Delta\mathbf{s})$ governed by the artificial decaying temperature $T_{q_v}(t)$ and the visiting parameter q_v

$$\mathsf{g}_{q_v}(\Delta\mathbf{s}) \propto \frac{T_{q_v}(t)^{-D/(3-q_v)}}{\left[1+(q_v-1)\frac{\Delta\mathbf{s}^2}{T_{q_v}(t)^{2/(3-q_v)}}\right]^{(1/(q_v-1)+(D-1)/2}} \quad (5)$$

where

$$T_{q_v}(t) = T_{q_v}(0)\frac{2^{(q_v-1)}-1}{(1+t)^{(q_v-1)}-1} \quad (6)$$

and t is time in iteration, D is the number of dimensions. As time t progresses, $T_{q_v}(t)$ decreases. A large initial temperature $T_{q_v}(0) = 5230$ is given to encourage larger exploration across the entire bounding space. As reported by Tsallis and Stariolo (1996), a visiting parameter q_v between 1 to 3 is recommended, therefore a default value of 2.62 is set. The algorithm starts with a random initial state and moves towards better solutions. At each iteration, there is an acceptance probability p_{q_a} that a worse solution can be accepted and it is controlled by the acceptance parameter q_a.

$$p_{q_a} = \begin{cases} 1-\mathsf{A}^{1/(1-q_a)} & \text{when } \mathsf{A} > 0; \\ 0 & \text{when } \mathsf{A} < 0, \end{cases} \quad (7)$$

where $\mathsf{A} = 1-(1-q_a)T_{q_v}(t)\Delta\mathbf{f}/(t+1)$ and $\Delta\mathbf{f}$ is the value difference between the current best solution and the worse solution. In this investigation, q_a is balanced for exploration vs exploitation. The algorithm will stop either when no improvement is discovered in 1000 steps, or when exhausts the computational budget. In our case, the algorithm presents the best solution found within a strict computational budget and follows a structure as presented in Algorithm 1. The generalised SA function is implemented via the SciPy (Virtanen et al., 2020) package.

Genetic Algorithm

GA uses binary representation of the continuous bounding space and evolves through a population set generating good candidate solutions through crossover and mutation. Naturally its explorative/exploitative behaviour can be controlled by population size (n_{pop}), crossover rate (R_c) and mutation rate (R_m).

The optimisation follows a process as illustrated in Algorithm 2. A random set of n_{pop} states (i.e. individuals) are generated to form an initial population, within which good candidates are chosen through a tournament selection process. To form the next generation, each state variable is encoded into n_{bit} bits. Selected candidates are paired up to produce *off-spring*, by cross-over state variables within the pair at a probability of R_c. During the cross-over process, mutation happens at a probability of R_m when a random bit flip its status. The *off-springs* then replace the previous generation and become the base population for the next evolution. Since a computational budget is set, the algorithm presents the best solution found within 100 function calls, equivalent to $100/n_{\text{pop}}$ generations for a population of n_{pop}.

Algorithm 1 Simulated Annealing

```
BS ← Define min/max for each state variable
s ← Generate a random initial state in BS
f(s) ← Compute corresponding initial score
t, t_max ← 0, 100                ▷ Time and computational budget
q_v, q_a ← Set visiting and acceptance parameter
T_qv ← Set initial temperature
while (t < t_max) do
    T_qv ← Update temperate
    p_qa ← Compute acceptance probability
    s_new = s + Δs ← Generate new candidate state by g_qv
    f(s_new) ← Compute corresponding score of s_new
    Δf = f(s) − f(s_new)
    if (Δf > 0) then             ▷ Better solution found
        s = s_new                ▷ Assign new current state
    else i_rand ← Random number between [0,1]
        if (i_rand > p_qa) then
            s = s_new            ▷ Accept the worse solution
        end if
    end if
    t ← t + 1
end while
```

Bayesian Optimisation

BO interprets the objective function with Gaussian processes, and builds a probabilistic surrogate model guiding its search. In this paper, BO follows a structure as shown in Algorithm 3. It starts with n_0 random initial points and uses the Matérn covariant function (kernel) to fit the objective function. The simplified form of Matérn kernel with unit scale can be written as

$$k(\mathbf{s_i},\mathbf{s_j}) = \frac{\sqrt{2\nu}\mathbf{d}(\mathbf{s_1},\mathbf{s_j})}{\Gamma(\nu)2^{\nu-1}}\mathbf{D_{ij}}^{\nu}K_{\nu}(\sqrt{2\nu}\mathbf{d}(\mathbf{s_1},\mathbf{s_j})), \quad (8)$$

where $\mathbf{d}(\mathbf{s_i},\mathbf{s_j})$ is the Euclidean distance between two states, $\Gamma(\nu)$ is the gamma function, K_ν is the modified Bessel functions of the second kind, and ν is the smoothness parameter. Since the objective function originated from a simulation environment, there should be minimal noise, hence a relatively smooth Matérn kernel of $\nu = 5/2$ is applied here. As the Matérn kernel is fitted, the mean $\mu(\mathbf{s})$ and standard deviation $\sigma(\mathbf{s})$ in the unknown area can be computed, effectively forming a confidence bound for the unknown area. The upper confidence bound method was employed as the acquisition function to determine the next point to sample.

$$acq(\mathbf{s}) = \mu(\mathbf{s}) + \kappa\sigma(\mathbf{s}), \quad (9)$$

The standard deviation $\sigma(\mathbf{s})$ is weighted by κ, which controls the exploration vs exploitation behaviour of the optimisation process. Where $acq(\mathbf{s})$ maximises indicates the most likely point for improvement, hence this point is sampled next. Iteratively, the algorithm explores the search space towards the global maxima. In addition, BO is also studied for its initial sample size. When $n_0 = 1$, it has a similar starting point with SA, and when $n_0 = 10$, it resembles the initial population of GA. The BO pipeline is implemented with the bayesian-optimization

Algorithm 2 Genetic Algorithm

$\mathcal{BS} \leftarrow$ Define min/max for each state variable
$n_{\text{bit}}, n_{\text{pop}}, R_c, R_m \leftarrow$ Set evolving strategy
$i, i_{\max} \leftarrow 0, 100/n_{\text{pop}}$ ▹ Iteration count and max budget
$\mathbf{s_1} \ldots \mathbf{s_{n_{\text{pop}}}} \leftarrow$ Generate random initial population
$\mathbf{f}(\mathbf{s_1}) \ldots \mathbf{f}(\mathbf{s_{n_{\text{pop}}}}) \leftarrow$ Compute corresponding initial scores
while $i < i_{\max}$ **do**
  for $\mathbf{s}$ in $\mathbf{s_1} \ldots \mathbf{s_{n_{\text{pop}}}}$ **do** ▹ Tournament Selection
    $\mathbf{s_p}, \mathbf{s_q} \leftarrow$ Randomly pick two states in population
    $\mathbf{f}_{\max} = max\{\mathbf{f}(\mathbf{s}), \mathbf{f}(\mathbf{s_p}), \mathbf{f}(\mathbf{s_q})\}$
    $\mathbf{s} = \mathbf{s}_{\max}$ for $\mathbf{f}_{\max} \leftarrow$ Register corresponding state
  end for
  $\mathbf{S_1} \ldots \mathbf{S_{n_{\text{pop}}}} \leftarrow$ Encode population states into bits
  for $[\mathbf{S_n}, \mathbf{S_{n+1}}]$ in $\mathbf{S_1} \ldots \mathbf{S_{n_{\text{pop}}}}$ **do** ▹ Cross-over
    if rand(0,1) $< R_c$ **then**
      P_c = randint(0,len($\mathbf{s}$)) $\leftarrow$ Pick a cross-over point
      $\mathbf{S_n}[:P_c n_{bit}]$=$\mathbf{S_{n+1}}[:P_c n_{bit}]$
      $\mathbf{S_{n+1}}[P_c n_{bit}:]$=$\mathbf{S_n}[P_c n_{bit}:]$ ▹ Exchange bits
      for $\mathbf{S}$ in $[\mathbf{S_n}, \mathbf{S_{n+1}}]$ **do** ▹ Mutation
        if rand(0,1)$< R_m$ **then**
          P_m = randint(0,len($\mathbf{S}$))
          $\mathbf{S}[P_m]$ = 1-$\mathbf{S}[P_m]$ ▹ Random bit flip
        end if
      end for
    end if
  end for
  $i \leftarrow i + 1$
end while

Algorithm 3 Bayesian Optimisation

$\mathcal{BS} \leftarrow$ Define min/max for each state variable
$\mathbf{s}$=$[\mathbf{s_1}, \ldots \mathbf{s_{n_0}}] \leftarrow$ Generate n_0 random initial states
$\mathbf{f}(\mathbf{s})$=$[\mathbf{f}(\mathbf{s_1}), \ldots, \mathbf{f}(\mathbf{s_{n_0}})] \leftarrow$ Compute corresponding scores
$i, i_{\max} \leftarrow 0, 100 - n_0$ ▹ Iteration count and max budget
while $i < i_{\max}$ **do**
  $k(\mathbf{s}) \leftarrow$ Fit Gaussian Process to fill state gaps
  $\mu(\mathbf{s}), \sigma(\mathbf{s}) \leftarrow$ Predict mean and std. dev
  $acq(\mathbf{s}) \leftarrow$ Assemble the acquisition function
  $\mathbf{s_{i+1}} \leftarrow$ Find next state for max($acq(\mathbf{s})$)
  $\mathbf{f}(\mathbf{s_{i+1}}) \leftarrow$ Sample the corresponding scores
  $\mathbf{s}$+ = $\mathbf{s_{i+1}}$; $\mathbf{f}(\mathbf{s})$+ = $\mathbf{f}(\mathbf{s_{i+1}})$, $\leftarrow$ Add to collection
  $i \leftarrow i + 1$
end while

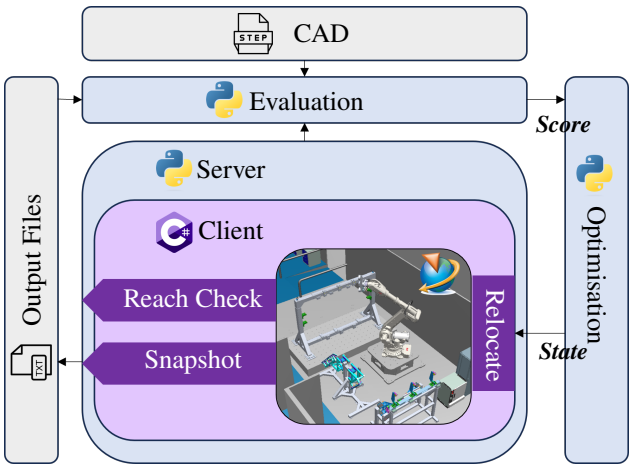


Figure 7: Methodology outline and communication architecture

(Nogueira, 2014) package, and Scikit-learn (Pedregosa et al., 2011) Matérn 2.5 Gaussian Process Regressor.

3.5. Software Architecture

The proposed evaluation and optimisation approach is built upon a set of communicated software environments as shown in Figure 7. The starting point is a simulation environment, built with CAD models, kinematics, and layout information captured by measurements (via metrology and robot inverse calculation). Since the virtual environment takes captured data from a physical system, it has a high level of spatial alignment, hence becomes a 3D digital twin. The digital twin is used for evaluating the camera FoV and layout suitability, through 1) Relocation: Move assets in a defined location, 2) Snapshot: capture point cloud representation of the scene, 3) Reach Check: Confirm robot reachability given a set of poses and points.

Enabling the 3D digital twin environment's automated optimisation capability, a .NET application programming interface (API), configured as a client, is used to automate the generation of point cloud output files and to pass reachability check results. Meanwhile, a server was created, given its available libraries of point cloud analysis and optimisation. The server implements a communication layer for FoV evaluation and layout optimisation.

The evaluation block reads information provided by the server, output files and CAD, then computes a score, describing the suitability of the tested layout arrangement. The suitability score, as a function of layout arrangement ($\mathbf{f}(\mathbf{s})$), can then be optimised.

The 3D digital twin is built in the software environment of Siemens Tecnomatix Process Simulate, where a RGBD camera can be created with photogrammetry camera parameters, capturing point clouds from a defined view frame. Other similar simulation packages, such as Visual Component and Gazebo, also have alternative methods to export point cloud representations from a given view point. In addition, the software, hosting the 3D digital twin, should also have an API interface, allowing the users to define custom functionality for an automated optimisation process.

4. Use Case Study

The use case is set up in Omnifactory® (Sanderson et al., 2024; Omnifactory), a RMS testbed for digital manufacturing technologies. As illustrated, the RMS has a LEGO-like floor construction, where a robot can be placed between any 4 adjacent floor points, and AGVs can move between robotic cells. A multi-product robotic assembly cell setup within Omnifactory is used as an example to demonstrate the evaluation and optimisation framework. As displayed in Figure 9, the OOI is the jig

Figure 8: Omnifactory® - UK national training and research testbed for smart manufacturing system: Reconfigurable floor with calibrated floor points to locate robots within the facility

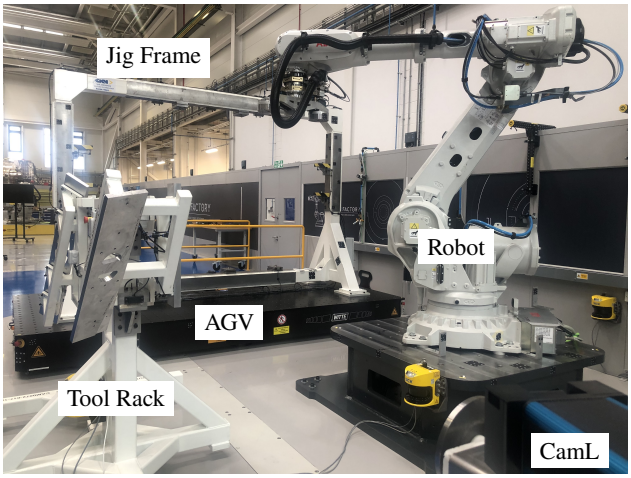


Figure 9: Robotic assembly cell with reconfigurable jig frame mounted on an AGV and photogrammetry system

frame mounted on an AGV. The frame is designed to be reconfigurable between winglets, rudders and elevators, and moves in and out of different robotic cells. This cell is an auto reconfiguration cell, where a highly precise (within 40μm) robotic pick and place process is performed (Wang et al., 2022c,d). Precision of this level is common in aerospace manufacturing and assembly, but difficult to achieve with automated systems. Therefore integrated photogrammetry is required to ensure repeatability of the reconfiguration process. The multi-product robotic cell within Omnifactory provides a RMS environment. The robot can change in the brand, make, configuration and assembly poses. The tooling, with which the robot interacts, can also change with respect to its design and configuration. The optimisation framework should be able to handle the level of variation in the dynamic environment.

Since the jig frame (OOI) is made with box-section beams of 150mm × 150mm, an ICP registration search range was set at 150mm, and a down-sampling rate of 50mm for OOI visibility analysis and 20mm for target coverage analysis was chosen.

Two Geodetic Systems' VSTARS D12 cameras were used and positioned on extendable poles at either side of the robot, having 4DoFs (height and xyz rotations). The minimum number of images required to reconstruct 3D coordinate is two, meaning two cameras are needed in an automated environment. This is also the case where the challenge for FoV is the most prominent. The framework does not limit the number of cameras, but the computational performance will need to be investigated.

Retro-reflective targets were setup on the robot base and the frame, and the cameras use the robot base to establish the working coordinate system, and measure the frame's position relative to the robot base as presented in Figure 10. According to Geodetic Systems' D12 camera manual, it can achieve a measurement accuracy of 10 μm + 10μ/m, corresponding to estimated measurement uncertainty in Equation 4.

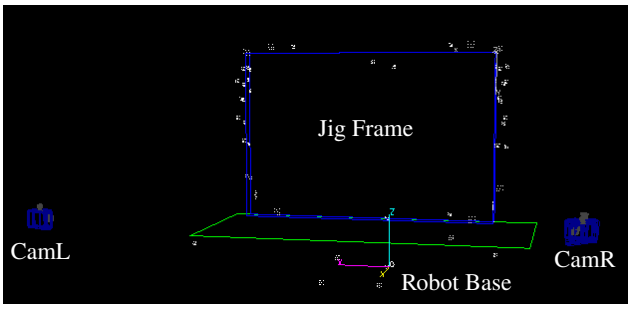
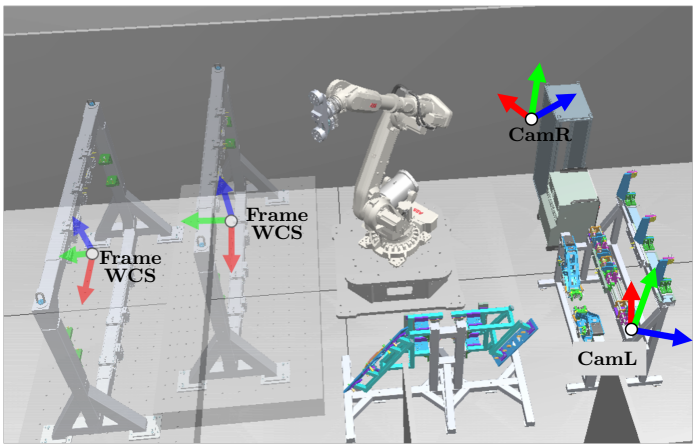


Figure 10: Photogrammetry VSTARS software interface

In practice, tool racks and robot create view obstruction to the jig frame, affecting overall measurement effectiveness. Therefore, the cameras and frame position within the cell are critical for the overall assembly. The visibility/coverage problem can be defined as a function of frame position and camera positions as illustrated in Figure 11, where the AGV can move in x and y, and the cameras can adjust their height (z) and orientations (θ, ρ, ϕ). In this investigation, an assumption is made in the optimisation stage that the camera will always points towards the centre of the jig frame (OOI), effectively decreasing camera's

4DoFs $[z, \theta, \rho, \phi]$ down to 1DoF $[z]$.



State s= {Frame WCS, CamL, CamR}

Figure 11: Optimisation Space in top view and 3D view, where the position of frame, CamL and CamR define the state

4.1. Virtual Camera Scaling

Virtual camera scaling affects the number of points the framework has to process, therefore direct impacting computational power required, as described in Equation 1. A down-scaling balancing analysis should be performed before any camera FoV evaluation and optimisation exercises. In this investigation, the number of points, OOI visibility and processing time are compared under different scaling factors, as computed in Table 1.

Table 1: Down-scaling factor F_s with resultant processing performance

F_s	Img Size	Point Count	Visibility	t_{eval}
1	(4096, 3072)	24×10^6	35.32%	25.93
2	(2048,1536)	6.3×10^6	35.57%	7.27
4	(1024, 768)	1.6×10^6	34.79%	1.10
8	**(512,384)**	**3.9** $\times10^5$	**32.36%**	**0.74**
16	(256, 192)	9.8 $\times10^4$	4.20%	0.49
32	(128,96)	2.5×10^4	NaN	NaN

Instead of target coverage, OOI visibility is used here, as it computes visibility ratio against the OOI CAD, hence is less sensitive to view point, and inherently has less evaluation noise. Since the optimisation process will likely require total function call count to be in the hundreds or thousands, processing time per function call should be within 3 secs to keep an optimisation of 100 search episodes within 5 mins. Each function call consists of socket connection time t_{connect}, relocation time t_{relocate} in addition, so the evaluation time t_{eval} is kept within a second, hence a scaling factor of 8 was chosen.

4.2. Digital Twin Construction

The proposed optimisation framework can only provide meaningful guidance when the virtual layout representation is accurate to the physical factory. To ensure spatial alignment between the twins, the virtual environment is defined with actual spatial measurement from the physical world. Robot's joint angle and pose data are connected and replicated in the Digital Twin. For the frame and cameras, their actual positions can be updated up to 3Hz. Storage tooling are considered stationary within a short production scope. Therefore, their locations are captured through photogrammetry scans of the entire workstation. Alternatively, spatial data can be obtained by Enhanced Reference System (ERS) point (Wang et al., 2022d), and laser scanning. The closeness between the physical and virtual environments are discussed and validated in Section 5.

5. Experimental Validation

Having digital twin as the framework input means the confidence in optimisation results is directly dependent on closeness between the digital and physical system. Therefore, experimental validation between simulated FoV and actual camera output is presented here.

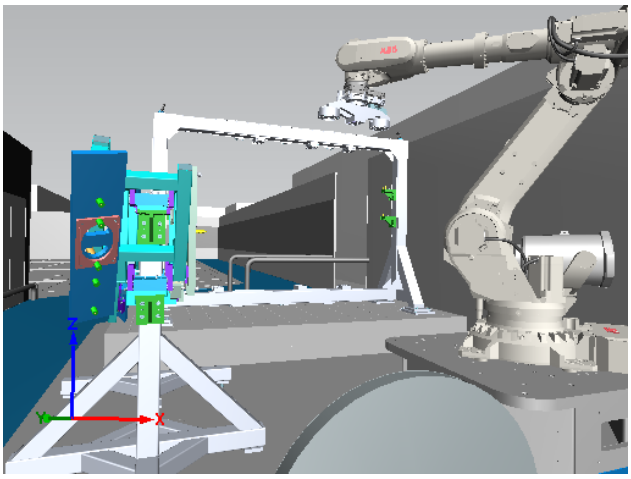
Camera extrinsic parameters and frame positions can be extracted from the VSTARS software and used to spatially align the digital twin with the physical correspondence. The tool rack position is imported via robot probing, however, photogrammetry measurement is a valid alternative.

A comparison between the virtual camera view and the actual camera view can be observed in Figure 12 and 13. There are 47 coded targets, each containing 8 retro-reflective points, therefore, a total of 376 retro-reflective points covering both sides of the frame. From the CamL view point as shown in Figure12, there are 34 (coloured in red) out of 376 points being incorrectly identified for their coverage status, resulting in a 90.96% accuracy. From the CamR view point displayed in Figure 13, there are 25 out of 376 points being incorrectly identified for their coverage status, resulting in a 93.35% accuracy.

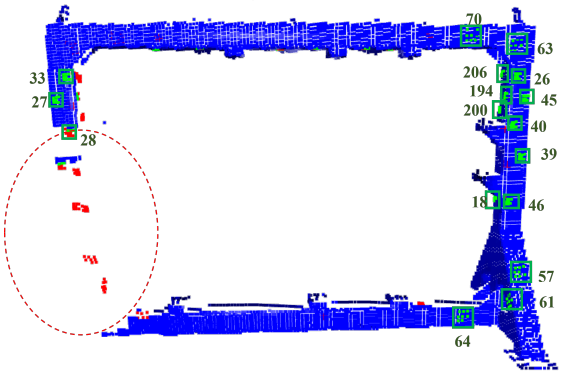
Since KDTree search is based on a spherical space with a defined radius, target points on the edge of the frame or visible area can fall into the spherical query node, resulting in incorrect coverage status. A KDTree query with control of the search direction specified for each target point should be able to resolve this issue, hence improving the accuracy of the proposed method.

For common area seen by both CamL and CamR, the simulated coverage results and the actual experimental coverage ratio are illustrated in Figure 14. The simulated coverage ratio is at 27.05%, and the actual target coverage is at 25.46%. There are 23 points whose coverage status was concluded incorrectly, resulting framework accuracy of 93.88%. These incorrect interpretations also fall on target points that are on the edge of the frame or the visible area. Similarly, a directional adaptive KDTree query should overcome this issue.

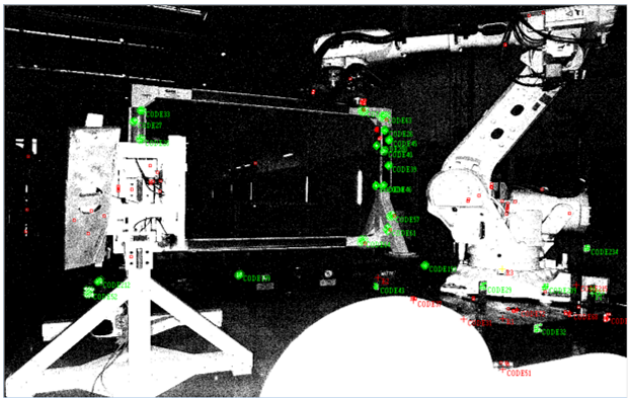
Based on Figure 14 (a), the illustrated visibility can also be used as a guide for target distribution and allocation. Clearly, the current target arrangement can be improved by utilising the front face of the top and bottom beams. Illustrative guidance would not be possible for a manual trial-and-error approach in practise.



(a) digital twin view

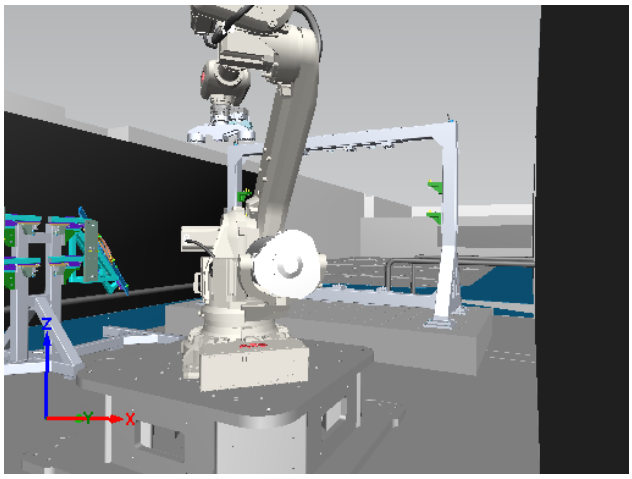


(b) Captured Point Cloud

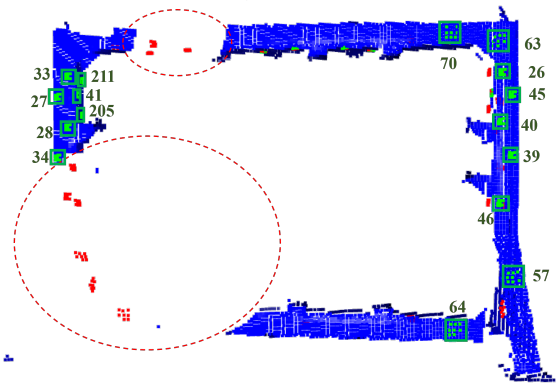


(c) Actual camera view

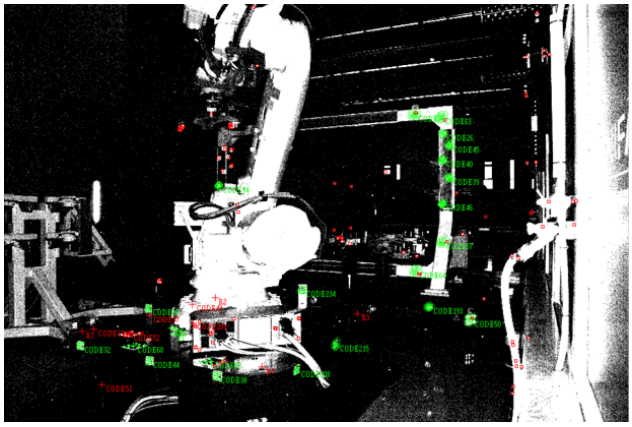
Figure 12: Digital Twin, point cloud and actual camera view from CamL, with aligned obstruction area (highlighted as dashed red circle in (b)). Visible coded targets and their IDs are also illustrated, with a 90.96% simulation accuracy.



(a) digital twin view



(b) Captured Point Cloud

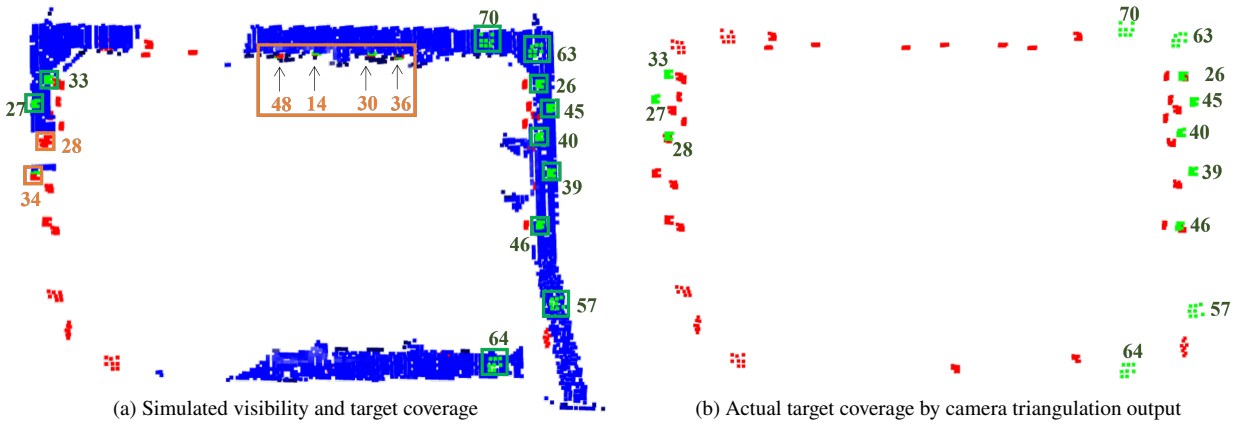


(c) Actual camera view

Figure 13: Digital Twin, point cloud and actual camera view from CamL, with aligned obstruction area (highlighted as dashed red circle in (b)). Visible coded targets and their IDs are also illustrated, with a 93.35% simulation accuracy.

6. Optimisation Results

This section will describe how the baseline best is established and how the optimisation results from heuristics compare to the baseline. The baseline is established through a brute force grid search (GS) where each node in the grid is consulted mapping the entire bounding space. The grid equally divided the jig frame bounding space into a 6×7 grid with a step of 150mm in x and y direction, and camera heights into a 5×5 grid with a step of 5mm for each, resulting in a four-dimensional domain. The AGV is manually driven with a joystick controller and 150mm is an estimation of the manual parking error without any additional metrology feedback. As for the cameras, they are mounted on an extendable pole and a 5mm manual error is assumed. Visibility and coverage for cameras and jig frame within the defined step size were assumed as uniform. The jig frame and camera height grid results in 1050 combinations ($6 \times 7 \times 5 \times 5$) and 1050 corresponding function calls were tested.



(a) Simulated visibility and target coverage

(b) Actual target coverage by camera triangulation output

Figure 14: Target coverage validation: Coded target ID and distribution are compared between the digital twin and the physical correspondence. Targets under obstruction are in red and visible points are in green. Visible coded target position and distribution are highly aligned between (a) the simulated case and (b) the actual camera capture, with a 93.88% of total prediction accuracy. Differences in target coverage sits on code 14, 28, 30, 36, 34 and 48, which are at the edge of the visible area due to KDTree search query.

In addition, hyperparameter selection for balancing exploration and exploitation is performed for each optimisation algorithm. For SA, acceptance probability Q_a was investigated. A lower acceptance probability indicates a lower likelihood to accept a worse score, directing the algorithm to behave more exploitatively. GA has a population size of n_{pop}, and evolves through $100/n_{pop}$ generations. Each state parameter is decoded into 8 bits, a default mutation rate R_m of 1/32 (by number of state variables len($\mathbf{s}$) and decoded bits n_{bit}) is used. State variable cross-over at each iteration/generation happens at a rate R_c, which is analysed and compared. Lastly, BO with the upper confidence bound method is investigated against initial sampling point n_0 the κ parameter, where a high κ value means exploration preferred and vice versa. For robustness, random initial states are used for each algorithm, and 10 optimisation repeats were performed. The rms for each iteration across the 10 repeats are also compared with the GS baseline results.

6.1. OOI Visibility

GS FoV evaluation for OOI visibility is presented as a heatmap as Figure 15. It shows the optimal layout with a yellow box where the frame is positioned around the top right corner with CamL and CamR at 1280 and 1200 respectively. In addition, we observe that reachability is an issue at the top left corner, as the score $\mathbf{f}(\mathbf{s})$ drops to zero for all camera positions. Across the bounding space, the objective function on the right edge is relatively smooth, and less so on the left. This is expected as the tool rack is located near the bottom left which can cause large obstructions as shown in Figure 12 and 13. When the frame is positioned further away, the tool rack obstruction can be minimised. A maximum score of 36.54 is observed at the right edge, where $\mathbf{s}$ = [Frame.x = 4356, Frame.y=25391, CamL = 1280, CamR =1200]. Best scores found by SA, GA and BO are shown in the nearest node in the grid with red, yellow and green dots and their numerical values are summarised in Table 2. They are also concentrated at the right edge, with different camera height configurations. This is coherent with the baseline best found in GS mapping.

Table 2: Optimal layout comparison for OOI visibility

	Frame.x	Frame.y	CamL	CamR	Score
SA	4359	25157	1294	1202	36.99
GA	4362	25161	1290	1216	36.90
BO	4356	25365	1300	1220	36.95
GS	4356	25391	1280	1200	36.54

According to Table 2, best scores found by SA, GA and BO are above the baseline best score. However, this is not a fair comparison as the best score is taken over 10 repeats. Since optimisation algorithm has a random initial state, rms and best scores across 10 repeats are compared in Table 3. One can observe that rms scores for SA, GA and BO are comparable with the GS baseline, however, computed with a fraction of the processing time. As the rapid reconfiguration requirement sets a 5-min limit to the total processing time, all heuristic approaches are proven to be feasible for RMS applications. Since bounding spaces in RMS layout are continuous, GS's assumption of discrete nodes is only valid with sufficient node points, and this is impossible for a four-dimensional layout problem within 100 function calls, demonstrating the advantage of SA, GA and BO in such optimisation tasks.

Table 3: Optimisation algorithm comparison for visibility

	rms Score	Max Score	Process Time
SA	36.60	36.99	4.17 mins
GA	36.63	36.90	4.17 mins
BO	36.50	36.95	4.17 mins
GS	36.54	36.54	43.75 mins

Across 100 function calls, the optimisation behaviour for SA, GA and BO are compared in Figure 16, where rms value at each iteration step is plotted together with their amplified MADs. It

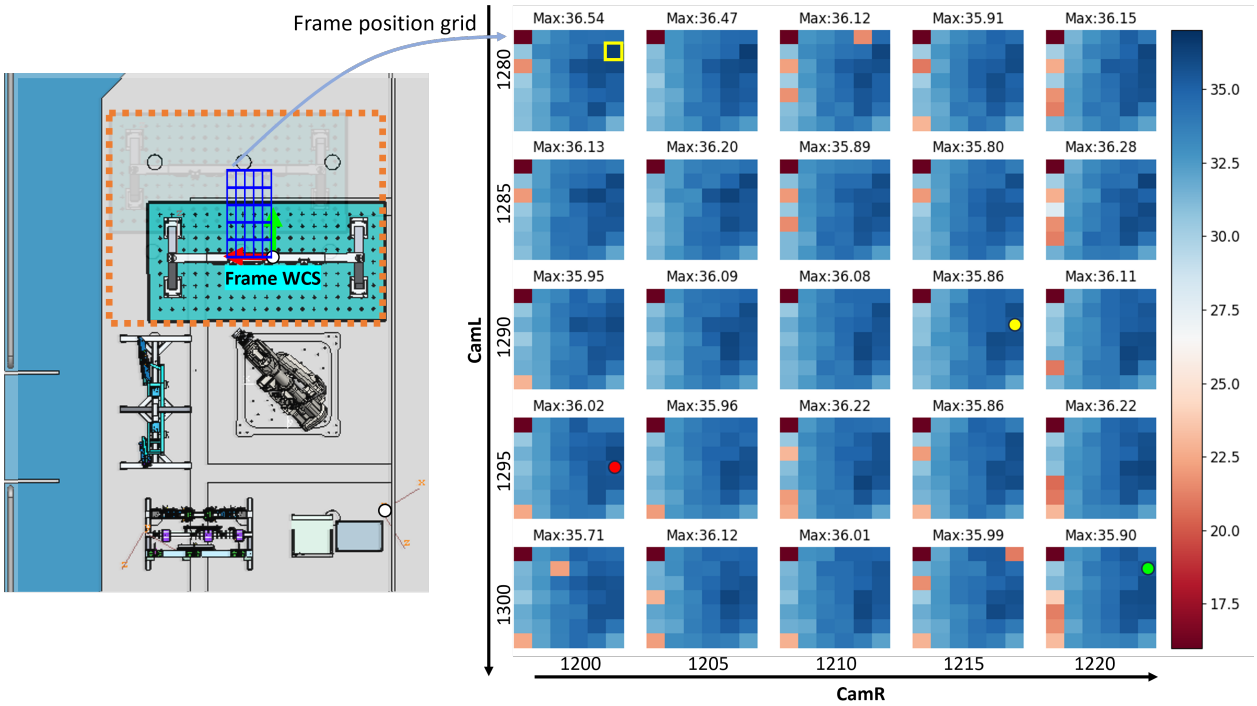


Figure 15: OOI visibility based objective function mapping across GS, each subplot represents the AGV exploration space/bound, while the horizontal and vertical axes represent the height of CamR and CamL, respectively. Highlighted dots indicate the closest node to the optimal state, as found by SA (red), GA (yellow) and BO (green), with the yellow box showing the optimal state found by GS

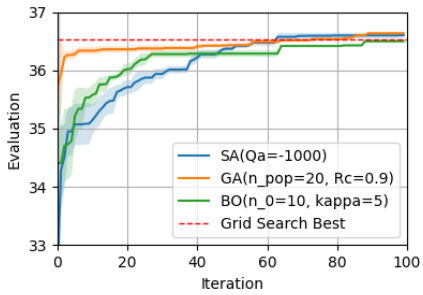


Figure 16: rms and MAD behaviour across 100 function calls for $\mathbf{f}(\mathbf{s})$ based on OOI visibility, note that MADs (shaded area) is 100 times amplified for visualisation, and the dotted red line indicates GS baseline best.

is evident that GA can quickly reach to high scores given the tournament selection and cross-over mechanism. SA although only has one sample to start with, but it is able to climb rapidly, catching up with GA within 60 iterations and overtaking GA after reaching the baseline. However, once a good result is achieved, it struggles to improve given its random nature. GA, on the other hand, is able to find better solutions through continued cross-over. BO also has the ability to climb quickly, yet faces a similar issue to SA that it struggles to improve once plateaued. Given the search is based on a probability model, it only samples one point at a time, making it more difficult to improve quickly. Again, the OOI visibility has a rather smooth profile, therefore, the population-based strategy is more effective. Overall, GA is the best optimiser within 100 function calls and managed to achieve slightly better results than the baseline GS, with a low MAD value.

6.2. *Target Coverage*

The same study is performed on layout suitability based on target coverage, with target arrangement captured by actual camera measurements in the real world. Firstly, a GS baseline is presented as depicted in Figure 17. The observation about reachability stays the same on the top left corners. The top right corner suffers from a low score with a reduced coverage ratio. A good frame position is around the centre towards the lower right bound, and a maximum score of 40.11 is found at $\mathbf{s}$ = [Frame.x = 4656, Frame.y=24941, CamL = 1285, CamR =1220]. Across the grid, the objective function is less smooth compared to the previous study on OOI visibility, indicating that the target coverage is much more sensitive to the layout state. Since coded targets, containing 8 retro-reflective points, are sparsely allocated on the jig frame, each coded target has a much heavier weight on the coverage ratio. The best scores discovered through SA, GA and BO within 100 function calls are highlighted on the nearest grid node. SA and GA are closely located together with the GS best, with the highest score discovered by BO at the bottom edge from a higher CamL position. Numerical results for optimal solutions are presented in Table 4. It is clear that SA and GA landed in the same area for all four state variables, both surpassing GS. On the other hand, BO discovered the highest score in a similar frame location with a higher camera viewpoint.

However, the best solution is not representative for the general behaviour of these algorithms with random initial states. rms

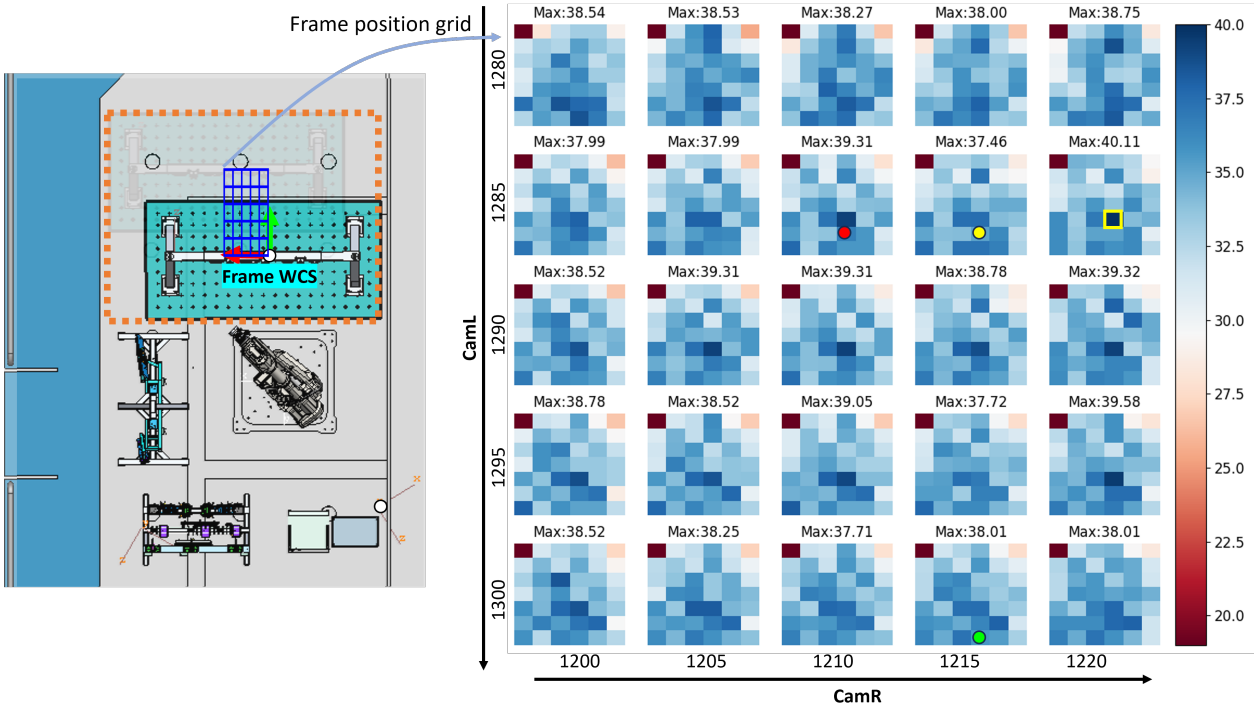


Figure 17: Target coverage based objective function mapping across GS, each subplot represents the AGV exploration space/bound, while the horizontal and vertical axes represent the height of CamR and CamL respectively. Highlighted dots indicate the closest node to the optimal solutions, as found by SA (red), GA (yellow) and BO (green), and the yellow box showing the optimal state from GS.

Table 4: Optimal layout comparison for target coverage

	Frame.x	Frame.y	CamL	CamR	Score
SA	4658	24766	1286	1211	41.19
GA	4661	24771	1287	1213	41.19
BO	4675	24692	1299	1214	41.46
GS	4656	24941	1285	1220	40.11

score, max score and process time across 10 repeats are concluded in Table 5. For rms scores, SA achieved 98.58% of the baseline best; GA matched the baseline ; and BO managed to outperform GS by 0.12%, all with a fraction of total processing time. The heuristic approaches also satisfied the 5-min planning limit, introduced by rapid reconfiguration.

Table 5: Optimisation algorithm comparison for coverage

	rms Score	Max Score	Process Time
SA	39.54	41.19	4.17 mins
GA	40.11	41.19	4.17 mins
BO	40.16	41.46	4.17 mins
GS	40.11	40.11	43.75 mins

Across 100 function calls, the rms and MAD are compared between four optimisation approaches in Figure 18. SA starts with a high slope, as iteration progresses, the artificial temperature is cooling and restricting search steps, making it slower to progress. GA benefited from its population-base strategy, and is able to climb quickly. The gap with baseline best is gradually reducing and at around the 80th iteration, GA managed to match baseline best. In later iterations, when it comes to narrowing down for improvement, GA struggled to generate better solutions. BO, however, progressed at a constant pace, and overtook baseline best just after the 80th iteration.

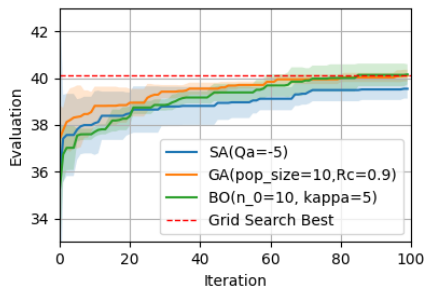
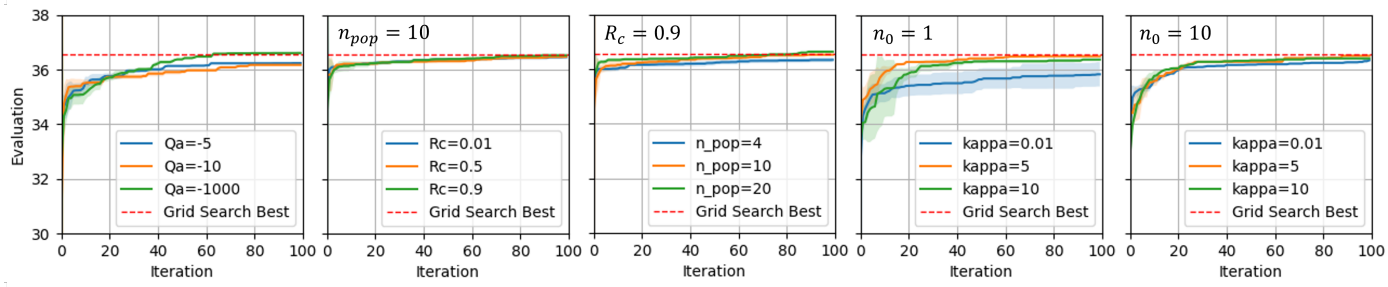


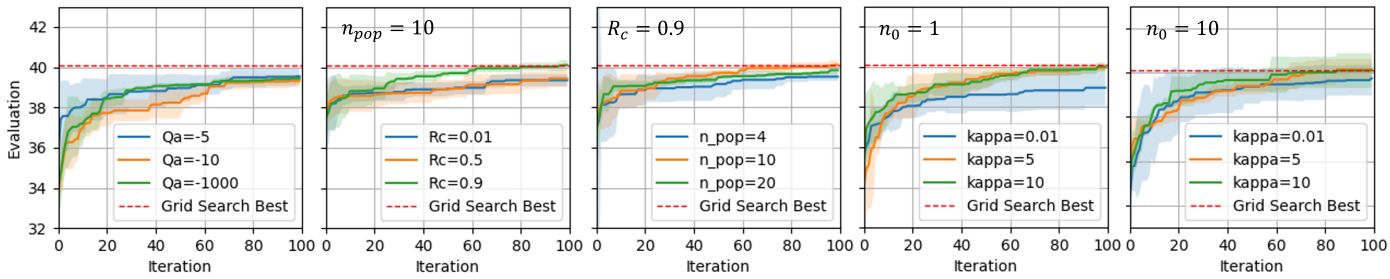
Figure 18: rms and MAD behaviour across 100 function calls for $\mathbf{f}(\mathbf{s})$ based on target coverage, note that MADs (shaded area) is 100 times amplified for visualisation, and the dotted red line indicates GS baseline best.

With targets sparsely allocated across the jig frame (OOI), it is expected to see larger fluctuations between adjacent states, leading to a more noisy profile. This is also reflected in larger MADs across the search process. However, this hinders GA for further improvement. BO, having a guided probability model, can account for these uncertainties through the weighting parameter κ, making it more suitable for solving more realistic target coverage problems. Overall, BO is the best of the tested algorithms for optimising target coverage with the proposed evaluation approach.



(a) SA (b) GA (n_{pop} = 10) (c) GA (R_c = 0.9) (d) BO (n_0 = 1) (e) BO (n_0 = 10)

Figure 19: Exploration vs exploitation study of OOI visibility for (a) SA, (b) GA with a population size of 10, (c) GA with a cross-over rate of 0.9, (d) BO with 1 initial sample, (e) BO with 10 initial samples, note that the red dotted line is showing GS best as a baseline



(a) SA (b) GA(n_{pop} = 10) (c) GA(R_c = 0.9) (d) BO(n_0 = 1) (e) BO (n_0 = 10)

Figure 20: Exploration vs exploitation study of target coverage for (a) SA, (b) GA with a population size of 10, (c) GA with a cross-over rate of 0.9 , (d) BO with 1 initial sample, (e) BO with 10 initial samples, note that the red dotted line is showing GS best as a baseline

6.3. Exploration vs Exploitation Analysis

Hyperparameters used for the above comparison are obtained through the exploration vs exploitation analysis. For SA, the acceptance parameter Q_a is investigated at three different scales of magnitude. A higher Q_a implies a higher degree of exploration. For GA, the population size n_{pop} and the cross-over rate R_c are studied, while R_m is kept at $1/(\text{len}(\mathbf{s})n_{bit})$. Larger n_{pop} and higher R_c means a more explorative search strategy. Lastly, BO is analysed for initial sample points n_0 and the acquisition weighting parameter κ. When $n_0 = 1$, BO has a similar starting point as SA, and when $n_0 = 10$, it resembles the initial population of GA.

For the case of OOI visibility based $\mathbf{f}(\mathbf{s})$ illustrated in Figure 19, MADs (amplified by 100) are only visible at initial iterations, which indicates a good level of overall convergence in this case. An exception for BO at $(n_0 = 1, \kappa = 0.01)$ was observed with high level of MADs. This is due to the overly exploitative search strategy combined with random intial states.

As for target coverage based $\mathbf{f}(\mathbf{s})$, they are plotted against different hyperparameter configurations in Figure 20. Larger MADs are observed across all optimisation processes. The configuration for BO again demonstrates the overly exploitative behaviour for $\kappa = 0.01$. An example of over-exploration on a noisy function profile is also present for $(n_0 = 10, \kappa = 10)$.

Finally, hyperparameters used and their corresponding rmss, max and MADs through 10 repeats are summarised in Table 6. The best optimizer configurations for each algorithm are highlighted in bold text, and the best performances in solving respective objective functions for OOI visibility/target coverage are labelled in green.

Table 6: Optimisation Algorithm Comparison

Category	Hyperparameter	OOI Visibility			Target Coverage		
		rms	Max	MAD	rms	Max	MAD
SA	$Q_a = -5$	36.23	36.64	0.8e-3	**39.54**	**40.66**	**8e-3**
	$Q_a = -10$	36.16	36.44	0.4e-3	39.31	40.91	6e-3
	$Qa = -1000$	**36.60**	**36.99**	**0.9e-3**	39.43	40.11	3e-3
GA	$n_p = 10; R_c = 0.01$	36.45	36.63	0.2e-3	39.35	40.67	6e-3
	$n_p = 10; R_c = 0.5$	36.47	36.65	0.1e-3	39.44	40.65	9e-3
	$n_p = 10; R_c = 0.9$	36.52	36.86	0.5e-3	**40.11**	**41.19**	**5e-3**
	$n_p = 4; R_c = 0.9$	36.33	36.81	2e-3	39.51	40.13	4e-3
	$n_p = 20; R_c = 0.9$	**36.63**	**36.90**	**0.3e-3**	39.83	40.92	4e-4
BO	$n_0 = 1; \kappa = 0.01$	35.82	37.22	9e-3	38.95	40.86	18e-3
	$n_0 = 1; \kappa = 5$	36.48	36.78	0.3e-3	39.94	40.65	4e-3
	$n_0 = 1; \kappa = 10$	36.36	36.79	0.6e-3	40.00	41.18	4e-3
	$n_0 = 10; \kappa = 0.01$	36.33	36.79	0.9e-3	39.73	41.71	16e-3
	$n_0 = 10; \kappa = 5$	**36.50**	**36.95**	**0.6e-3**	**40.16**	**41.46**	**9e-3**
	$n_0 = 10; \kappa = 10$	36.40	36.59	0.3e-3	40.08	41.98	16e-3

6.4. Note on Measurement Accuracy

The authors would like to highlight that the aim of this paper is to simulate and evaluate visibility of points/objects of interest in a metrology-aided manufacturing setup, and provide an efficient and integrated methodology in layout planning and optimisation. The improvement in visibility and coverage inevitably benefits measurement accuracy. However, measurement accuracy for photogrammetry systems is also influenced by other interdependent factors, such as lighting conditions, camera resolution, number of photographs, target response angle and more. The

described study solely focuses on visibility/coverage of common features. How this can be used to improve measurement accuracy falls outside of the scope for this investigation.

7. Conclusions and Future Work

Rapid data acquisition from photogrammetry systems relies on labour-intensive setup processes. Its increasing application in the context of flexible manufacturing leads to a need for metric evaluation for measurement effectiveness, and optimisation methodology for finding optimal layout in integrated systems. This paper proposed an digital-twin based FoV evaluation and 3D layout optimisation framework for reconfigurable manufacturing systems.

With spatial data captured from photogrammetry cameras and robot, a highly aligned virtual representation (digital twin) was constructed. Virtual cameras output RGBD information, which is used to reconstruct the scene in point cloud, representing its FoV. A point cloud based analysis, combining ICP registration and KDTree intersection, computes the visibility of the OOI and coverage ratio of retro-reflective targets. Efficiency and fidelity of the camera outputs were balanced through camera down-scaling and point cloud down-sampling. Experimental validation confirmed digital and physical similarity above 93%. This result means station-level 3D layout design for MAA applications can be perform effectively in digital twin environments.

Robot reachability to key poses, camera FoV and estimated uncertainty were considered for layout optimisation. Due to the characteristics of the objective function, GA, SA and BO were tested for only 100 functions calls and compared against a brute force GS algorithm. All tested algorithms were able to reach within 98% of GS best within 5 mins. GA and SA, in particular, found better solutions (by 0.16% and 0.25% respectively) than GS best for OOI visibility. While for target coverage, BO outperfomed the GS best by 0.12%. All three algorithms had a process time of less than 10% of the baseline GS. These results demonstrated the efficiency of the proposed framework, with the emphasis on its suitability for a wide range of development stages from design to on-site execution.

The proposed methodology also has some limitations, regarding the accuracy between digital and physical FoV. Angle convergence effects are not included in this paper. This may have contributed to deviation between simulated and actual target coverage. Moreover, target points on the edge of the visible area are mistakenly included due to spherical nodes in KDTree intersection. Both can be addressed by a controlled search with adaptable direction, further improving FoV simulation accuracy.

With an emerging trend for in-line/in-process metrology, a similar philosophy of point-cloud scene representation can be adopted for other metrology systems. MAA applications commonly involve laser trackers and laser radars for measuring different surface features, and the extension to these systems and features can be further investigated. Lastly, the proposed approach can be used to plan robot path and pose configurations. Robot arms often can reach one point with different joint configurations, dynamic FoV evaluation can help determine an optimal configuration in MAAs.

Acknowledgements

The authors would like to acknowledge Made Smarter Innovation - Research Centre for Connected Factories (EP/V062123/1) for funding this study.

References

AMRC, A.M.R.C., . Factory 2050. URL: https://www.amrc.co.uk/news/amrc-launches-cutting-edge-factory-2050-advanced-manufacturing-research-facility-in-sheffield.

Arnarson, H., Yu, H., Olavsbrten, M.M., Bremdal, B.A., Solvang, B., 2023. Towards smart layout design for a reconfigurable manufacturing system. Journal of Manufacturing Systems 68. doi:10.1016/j.jmsy.2023.03.012.

Azab, A., ElMaraghy, H.A., 2007. Mathematical modeling for reconfigurable process planning. CIRP Annals 56, 467–472. doi:10.1016/J.CIRP.2007.05.112.

Azamfirei, V., Granlund, A., Lagrosen, Y., Palm, W.J., 2021. Towards fixtureless robotic in-line measurement assisted assembly, a case study. 2021 IEEE International Workshop on Metrology for Industry 4.0 and IoT, MetroInd 4.0 and IoT 2021 - Proceedings , 636–641doi:10.1109/METROIND4.0IOT51437.2021.9488551.

Azamfirei, V., Psarommatis, F., Granlund, A., Lagrosen, Y., 2024. Towards zero-defect manufacturing: a review on measurement-assisted processes and their technologies. Procedia Computer Science 232, 1001–1010. doi:10.1016/J.PROCS.2024.01.099.

BAESystems, . Factory of the future. URL: https://www.baesystems.com/en/factory-of-the-future.

Barazzetti, L., 2017. Network design in close-range photogrammetry with short baseline images. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2.

Benderbal, H.H., Dahane, M., Benyoucef, L., 2018. Exhaustive search based heuristic for solving machine layout problem in reconfigurable manufacturing system design. IFAC-PapersOnLine 51, 78–83. doi:10.1016/J.IFACOL.2018.08.238.

Bensmaine, A., Dahane, M., Benyoucef, L., 2014. A new heuristic for integrated process planning and scheduling in reconfigurable manufacturing systems. International Journal of Production Research 52, 3583–3594. doi:10.1080/00207543.2013.878056.

Besbes, M., Mahjoub, Y.I., Bonte, T., Berger, T., Sallez, Y., Zolghadri, M., 2021. Solving facility layout problem with safety consideration of reconfigurable manufacturing and assembly systems. Procedia CIRP 104, 1942–1947. doi:10.1016/J.PROCIR.2021.11.328.

Catalucci, S., Thompson, A., Piano, S., Branson, D.T., Leach, R., 2022. Optical metrology for digital manufacturing: a review. doi:10.1007/s00170-022-09084-5.

Chaube, A., Benyoucef, L., Tiwari, M.K., 2012. An adapted NSGA-2 algorithm based dynamic process plan generation for a reconfigurable manufacturing system. Journal of Intelligent Manufacturing 23, 1141–1155. doi:10.1007/S10845-010-0453-9/METRICS.

Chen, Y., Medioni, G., 1992. Object modelling by registration of multiple range images. Image and Vision Computing 10, 145–155. doi:10.1016/0262-8856(92)90066-C.

Drouot, A., Irving, L., Sanderson, D., Smith, A., Ratchev, S., 2017. A transformable manufacturing concept for low-volume aerospace assembly. IFAC-PapersOnLine 50, 5712–5717. doi:10.1016/j.ifacol.2017.08.1123. repeatability smaller than 0.05mm, Laser scanner to correct robot position relative to work piece.

Geodetic Systems, I., . Basics of photogrammetry. URL: https://www.geodetic.com/basics-of-photogrammetry/.

Grieves, M., 2014. Digital twin : Manufacturing excellence through virtual factory replication. White Paper .

Grieves, M., Vickers, J., 2016. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches , 85–113doi:10.1007/978-3-319-38756-7-4.

Guan, X., Dai, X., Qiu, B., Li, J., 2012. A revised electromagnetism-like mechanism for layout design of reconfigurable manufacturing system. Computers

and Industrial Engineering 63, 98–108. doi:10.1016/J.CIE.2012.01.016.

Hocine, C., Benaissa, A., 2021. New binary particle swarm optimization algorithm for surveillance and camera situation assessments. Journal of Electrical Engineering and Technology , 1–11doi:10.1007/S42835-021-00961-9/TABLES/1. 2D optimisation.

Konda, K.R., Conci, N., 2013. Global and local coverage maximization in multi-camera networks by stochastic optimization. Infocommunications Journal 5, 1–8. 2D coverage camera position optimistion.

Koren, Y., Gu, X., Guo, W., 2018. Reconfigurable manufacturing systems: Principles, design, and future trends. Frontiers of Mechanical Engineering 13, 121–136. doi:10.1007/s11465-018-0483-0.

Leng, J., Liu, Q., Ye, S., Jing, J., Wang, Y., Zhang, C., Zhang, D., Chen, X., 2020. Digital twin-driven rapid reconfiguration of the automated manufacturing system via an open architecture model. Robotics and Computer-Integrated Manufacturing 63, 101895. doi:10.1016/J.RCIM.2019.101895.

Li, J., Berglund, J., Auris, F., Hanna, A., Vallhagen, J., kesson, K., 2018. Evaluation of photogrammetry for use in industrial production systems. IEEE International Conference on Automation Science and Engineering 2018-August, 414–420. doi:10.1109/COASE.2018.8560496.

Liu, T., Burner, A.W., Jones, T.W., Barrows, D.A., 2012. Photogrammetric techniques for aerospace applications,. Progress in Aerospace Sciences 54, 1–58. doi:https://doi.org/10.1016/j.paerosci.2012.03.002.

Maneewongvatana, S., Mount, D.M., 1999. Analysis of approximate nearest neighbor searching with clustered point sets. Data Structures, Near Neighbor Searches, and Methodology: Fifth and Sixth DIMACS Implementation Challenges.

Maropoulos, P.G., Muelaner, J.E., Summers, M.D., Martin, O.C., 2014. A new paradigm in large-scale assembly-research priorities in measurement assisted assembly. International Journal of Advanced Manufacturing Technology 70, 621–633. doi:10.1007/s00170-013-5283-4.

Mason, S.O., Grün, A., 1995. Automatic sensor placement for accurate dimensional inspection. Computer Vision and Image Understanding 61, 454–467. doi:10.1006/CVIU.1995.1034.

Muelaner, J.E., Cai, B., Maropoulos, P.G., 2010. Large-volume metrology instrument selection and measurability analysis. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 224. doi:10.1243/09544054JEM1676.

Nogueira, F., 2014. Bayesian Optimization: Open source constrained global optimization tool for Python. URL: https://github.com/bayesian-optimization/BayesianOptimization.

Olague, G., 2001. Autonomous photogrammetric network design using genetic algorithms. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 2037, 353–363.

Omnifactory, . Omnifactory: Redefining manufacturing. URL: https://www.omnifactory.com/.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Petroodi, S.E.H., Eynaud, A.B.D., Klement, N., Tavakkoli-Moghaddam, R., 2019. Simulation-based optimization approach with scenario-based product sequence in a reconfigurable manufacturing system (RMS): A case study. IFAC-PapersOnLine 52, 2638–2643. doi:10.1016/J.IFACOL.2019.11.605.

Rangel, E.O., Costa, D.G., Loula, A., 2019. On redundant coverage maximization in wireless visual sensor networks: Evolutionary algorithms for multi-objective optimization. Applied Soft Computing 82, 105578. doi:10.1016/J.ASOC.2019.105578.

Rusinkiewicz, S., Levoy, M., 2001. Efficient variants of the ICP algorithm. Proceedings of International Conference on 3-D Digital Imaging and Modeling, 3DIM , 145–152doi:10.1109/IM.2001.924423.

Sanderson, D., Turner, A., Shires, E., Chaplin, J., Ratchev, S., 2019. Demonstration of transformable manufacturing systems through the evolvable assembly systems project. SAE Technical Papers doi:10.4271/2019-01-1363.

Sanderson, D., Wang, Z., Bainbridge, D., Ratchev, S., 2024. Omnifactory: a national training and research testbed for smart manufacturing systems, in: 14th Conference on Learning Factories, Twente, Netherlands.

Santos, R., Toscano, C., de Sousa, J.P., 2021. A new simulation-based approach in the design of manufacturing systems and real-time decision making. IFAC-PapersOnLine 54, 282–287. doi:10.1016/J.IFACOL.2021.08.033.

Silk, M.L., Andrews, D.L., 2010. Design for measurement assisted determinate assembly (MADA) of large composite structures. Journal of the Coordinate Metrology Systems Conference 2.

Strubel, D., Morel, O., Saad, N.M., Fofi, D., 2017. Evolutionary algorithm for positioning cameras networks mounted on UAV. IEEE Intelligent Vehicles Symposium, Proceedings , 1758–1763doi:10.1109/IVS.2017.7995961.

Torayev, A., Wang, Z., Martínez-Arellano, G., Chaplin, J., Sanderson, D., Ratchev, S., 2023a. Multi-criteria decision-making for optimal manufacturing configuration selection using an object-oriented data model and mathematical formalization, in: Proceedings of ASME 2023 18th International Manufacturing Science and Engineering Conference, MSEC 2023. doi:10.1115/msec2023-105605.

Torayev, A., Wang, Z., Martínez-Arellano, G., Chaplin, J., Sanderson, D., Ratchev, S., 2023b. Optimal selection of manufacturing configurations using object-oriented and mathematical data models, in: Advances in Transdisciplinary Engineering. doi:10.3233/ATDE230025.

Touckia, J.K., Hamani, N., Kermad, L., 2022. Digital twin framework for reconfigurable manufacturing systems (RMSs): design and simulation. International Journal of Advanced Manufacturing Technology 120, 5431–5450. doi:10.1007/S00170-022-09118-Y/FIGURES/12.

Tsallis, C., Stariolo, D.A., 1996. Generalized simulated annealing. Physica A: Statistical Mechanics and its Applications 233, 395–406.

Virtanen, P., Gommers, R., Oliphant, T.e.a., 2020. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nature Methods 17, 261–272.

Wang, L., Wang, Z., Gumma, K., Turner, A., Ratchev, S., 2022a. Multi-agent cooperative swarm learning for dynamic layout optimisation of reconfigurable robotic assembly cells based on digital twin. Journal of Intelligent Manufacturing doi:10.1007/s10845-023-02229-7.

Wang, L., Wang, Z., Kendall, P., Gumma, K., Turner, A., Ratchev, S., 2023a. Deep dynamic layout optimization of photogrammetry camera position based on digital twin. IEEE Transactions on Automation Science and Engineering doi:10.1109/TASE.2023.3323088.

Wang, L., Wang, Z., Kendall, P., Gumma, K., Turner, A., Ratchev, S., 2023b. Digital-twin deep dynamic camera position optimisation for the V-STARS photogrammetry system based on 3D reconstruction. International Journal of Production Research doi:10.1080/00207543.2023.2252108.

Wang, Y., Liu, Y., Chen, H., Xie, Q., Zhang, K., Wang, J., 2022b. Combined measurement based wing-fuselage assembly coordination via multiconstraint optimization. IEEE Transactions on Instrumentation and Measurement 71. doi:10.1109/TIM.2022.3186675.

Wang, Z., Kendall, P., Gumma, K., Smith, A., Turner, A., Ratchev, S., 2022c. Development of an affordable and auto-reconfigurable solution for small box assembly, in: IFAC-PapersOnLine. doi:10.1016/j.ifacol.2022.10.078.

Wang, Z., Kendall, P., Gumma, K., Turner, A., Ratchev, S., 2022d. An adaptive, repeatable and rapid auto-reconfiguration process in a smart manufacturing system for small box assembly, in: IEEE International Conference on Automation Science and Engineering. doi:10.1109/CASE49997.2022.9926429.

Yamada, Y., 2006. Dynamic reconfiguration of reconfigurable manufacturing systems using particle swarm optimization. Proceedings - IEEE International Conference on Robotics and Automation 2006, 1444–1449. doi:10.1109/ROBOT.2006.1641912.

Yuksel, C., 2015. Sample elimination for generating poisson disk sample sets. Computer Graphics Forum 34, 25–32. doi:https://doi.org/10.1111/cgf.12538.

Zhang, H., Eastwood, J., Isa, M., Sims-Waterhouse, D., Leach, R., Piano, S., 2021. Optimisation of camera positions for optical coordinate measurement based on visible point analysis. Precision Engineering 67, 178–188. doi:10.1016/J.PRECISIONENG.2020.09.016.

Zhang, X., Zhang, B., Chen, X., Fang, Y., 2019. Coverage optimization of visual sensor networks for observing 3-D objects: survey and comparison. International Journal of Intelligent Robotics and Applications 3, 342–361. doi:10.1007/S41315-019-00102-6/FIGURES/11.

Zheng, L., Zhu, L., Wang, B., Bai, L., 2013. A simulation analysis of facility layout problems in reconfigurable manufacturing systems. Proceedings - 2013 International Conference on Computer Sciences and Applications, CSA 2013 , 423–427doi:10.1109/CSA.2013.106.

Zhou, J., Deng, H., Zhao, Z., Zou, Y., Wang, X., 2024. Sensor placement optimization of visual sensor networks for target tracking based on multi-objective constraints. Applied Sciences doi:10.3390/app14051722.