Development of Transformations between Designed and Built Structural Systems and Pipe Assemblies

by

Mohammad Nahangi

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Civil Engineering

Waterloo, Ontario, Canada, 2015

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Fabrication of steel assemblies is a challenging process using existing machines to perform the tasks involved such as cutting, drilling, and punching. Due to inaccuracies in the fabrication processes, imperfections will inevitably happen. In addition to the fabrication inaccuracies, errors may occur during transportation or due to the temperature changes on construction sites. These challenges become more important in the offsite construction as it requires sequenced fabrication, transportation and installation. Current approaches for quality inspection, in general, and discrepancy analysis, in particular, lack a sufficient level of automation and are prone to error due to the intensive manual work involved. Hence, a proactive framework is substantially required to systematically monitor the fabrication process and control the accuracy of assemblies in order to expedite the erection and installation processes. Additionally, finding defective assemblies is traditionally done through fitting trials on construction sites, which has always been a key challenge as it is associated with rework. Furthermore, realigning the defective assemblies is currently performed based on the workers' experience and lacks automated planning. Therefore, detecting the defective parts in a timely manner and in a systematic way can expedite the erection process and avoids significant delays in construction projects and huge costs as a consequence.

This research aims to improve the fabrication and installation processes by detecting the incurred inaccuracies automatically and plan for realignment of the defective components systematically. In summary, the required framework to achieve these objectives includes four primary steps:

(1) Preprocessing and basic compliance checking,

(2) Spatial discrepancy detection and characterization,

(3) Calculation of the required alignments and adjustments, and

(4) Generalization of the realignment planning and actuation strategy frameworks for parallel systems.

The automated compliance checking and discrepancy analysis is performed employing advanced 3D imaging technologies which have recently opened up a wide range of solutions to acquire as-built status. Characterization of the detected discrepancies is performed by employing robotics forward kinematics concepts and combining with 3D imaging techniques. The required alignment is calculated accordingly using the robotic analogy and inverse kinematic concept. Although the proposed approach can be applied in any types of construction assembly, this thesis mainly focuses on industrial facilities such as steel pipe modules and pipe spools, in particular. Contributions of developing the described framework include:

(1) Developing a proactive strategy for rework avoidance,
(2) Algorithmic and programmable framework,
(3) Efficiency and robustness of the functions and metrics developed, and
(4) Time effectiveness of the framework.

Acknowledgements

During the journey of my PhD, I was privileged to work with Prof. Carl Haas, an exceptional supervisor, a great mentor, and, beyond all, a true friend to whom I can rely. I would like to express my sincere appreciation to him for his guidance, encouragement, and unconditional support during my PhD.

I am also very grateful for the huge support and great mentorship of Prof. Ralph Haas. I had the great opportunity to take his invaluable advises not only in my PhD, but even beyond in my academic career. A part of the experimental equipment used in this thesis is also supported by his financial grants.

I would also like to thank my PhD committee members, Prof. Fernanda Leite from UT-Austin, Prof. Keith Hipel, Prof. Tarek Hegazy, and Prof. Jeffrey West from the University of Waterloo.

The majority of the thesis contents were peer-reviewed in the form of technical journal papers before thesis submission. I would like to thank Prof. William O'Brien, the editor-in-chief of *Elsevier Journal of Advanced Engineering Informatics* for handling and publishing the work presented in Chapter 3, Prof. Miroslaw Skibniewski, the editor-in-chief of *Elsevier Journal of Automation in Construction* for handling and publishing the work presented in Chapter 4, and Prof. Lucio Soibelman, the editor-in-chief of *ASCE Journal of Computing in Civil Engineering* for handling and publishing the work presented in Chapter 5 and 6, as well as the anonymous reviewers who reviewed these publications.

I would also like to thank Prof. Scott Walbridge for his invaluable comments on my drafts. Prof. Walbridge was one of the co-PI's in the collaborative research and development (CRD) projects related to the objectives of this thesis. These CRD projects were financially granted by the Natural Science and Engineering Research Council (NSERC) Canada, in collaboration with SNC-Lavalin, and Aecon. Acknowledgements are also directed to Mr. Dennis Samolczyk from Aecon, Mr. Peter Freeman from SNC-Lavalin, and Mr. Paul Murray, the Executive VP of Energy at Aecon.

My warm gratitude goes to my fellow graduate students and friends: Yelda, Jamie, Mahdi, Arash, Samin, Behrooz, Hassan, Shahin, Thomas, Chris, and Mohammed. It was also a great pleasure to work with some fantastic coop students and undergraduate research assistants: Ali, Lav, Muhammad Ali, Julia, Felipe, Jane, Lucas, and Justin.

Above all, I would like to thank my beloved wife, Raheleh. Her unwavering love, patience, continuous support and encouragement were undeniably in the end what made this dissertation possible. Thank you for being my best friend. I owe you everything.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Problem statement

Building Information Modeling (BIM), which contains all information pertinent to designing, planning, and managing a construction system and infrastructure project as a digital representation (Ding et al. 2014), is becoming widespread in the construction industry as a critical part of any effective management system. The automation of processes within construction management systems can effectively lead to higher production rates and increased productivity, more efficient use of materials, and better product quality (Akinci et al. 2006a; Tang and Akinci 2012), and as such, automating parts of BIM can lead to these benefits.

One of the goals of construction management systems and BIM is the reduction of the impact of rework. In the construction literature, rework is the wasteful effort involved in reproducing a product that did not adequately conform to contractual requirements in its previous iterations (Hwang et al. 2009; Simpeh et al. 2012). Rework can be a result of the complexity of the fabrication process, the inaccuracy of the tools employed for performing construction, human error, and inadequate inspection and monitoring during construction. Permanent deflections may also occur during transportation and hoisting of components that lead to necessary realignment in order to conform to contractual requirements (Figure 1-1). Rework contributes to cost and schedule overruns on construction projects (Hegazy et al. 2011; Love 2002). Specifically, research published by the Construction Industry Institute (CII) places the cost of rework between 2% and 20% of total cost on any typical project (CII 2011). Rework is inherently a reactive construction practice, and represents a failure of the management system to maintain proactive control. Figure 1-1 shows an example of potential deformations that occur after fabrication, which is a typical source of error.

On industrial construction projects, it is popular for BIM to be used to perform clash detection in an attempt to identify potential conflicting processes and components, and thus to mitigate rework risk (Hu and Mohamed 2012; Hu and Mohamed 2014; Wang and Abourizk 2009). However simulation-based methods for monitoring the geometric status of construction components during assembly remain relatively unexplored and unutilized.

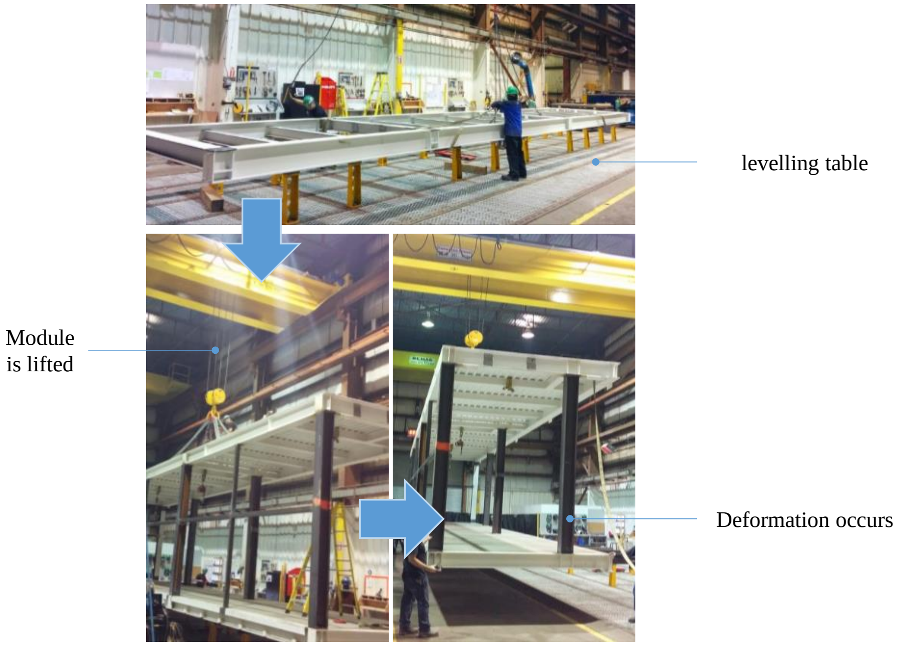


Figure 1-1: Assembly of a parallel system in a fabrication shop. Once module assembly is completed, it is lifted for transportation and shipment. Deformations thereby occur and module assembly on construction site becomes challenging. Photos are taken by the candidate during a site visit in January 2015.

Automating (1) as-built modeling within the BIM framework, (2) dimensional compliance monitoring, and (3) realignment planning, would allow for early detection of out-of-tolerance assemblies, immediate remediation of this out-of-tolerance, and proactive mitigation of rework. When considering realignment, this research makes use of a kinematics by robotics analogy. For better understanding the role of the robotics analogy in this thesis, construction management systems and control systems theory are discussed in the following section.

1.2 Control systems theory

Operation and maintenance (O&M) actions are necessary for a construction management system to flow forward and therefore be operative and productive (Figure 1-2). The forward flow of actions requires a feedback signal to track the status of the construction subsystem each time it is maintained and assessed (Turkan et al. 2012). Automating the tasks involved in the management system is a key to improving productivity which is the ultimate goal of construction management systems (Akinci et al. 2006a; Akinci et al. 2006b; Tang et al. 2010; Xiong et al. 2013). In a construction management system, Building Information Models (BIMs) provide detailed as-designed data (e.g. 3D drawings) that lead the forward flow. On the other hand, the as-built data provide the feedback signal for the operations and maintenance required in the system. Such operations include progress tracking (Golparvar-Fard et al. 2009a; Kim et al. 2013a), safety monitoring (Chi and Caldas 2012; Gong and Caldas 2011; Teizer et al. 2005c), and as-built status modeling and design compliance monitoring (Ahmed et al. 2014; Bosché and Guenet 2014; Golparvar-Fard et al. 2011; Lee et al. 2013). One important action that necessarily needs to be addressed is the assessment of defective construction component alignment for effective and efficient corrective actions (i.e. realignment plans) and therefore refitting solutions. These actions become more important in industrial facilities because they heavily involve complicated component geometries that require sequential fabrication, subassembly, transportation, and installation (Song et al. 2005).

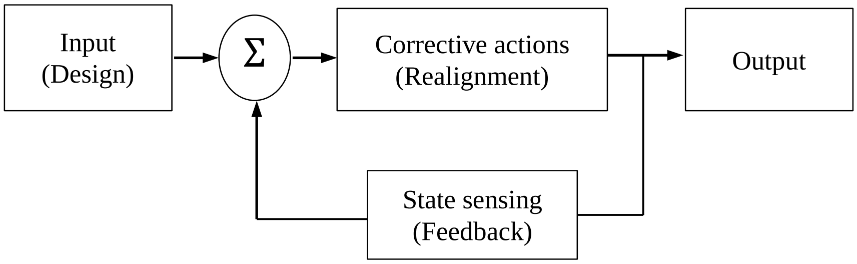


Figure 1-2: Control systems theory and the flow between various components for a fabrication and assembly control system

1.3 Robotics analogy

For finding the required transformations between designed and built assemblies, an analogy of robotics is used. Construction assemblies are analogous to robots with mutual degrees of freedom that make it possible to plan for realignment of an assembly with the kinematics analysis of a robot. Serial and parallel components are therefore the two basic component types from which any larger system can be built (Patel and George 2012). These basic components are briefly introduced in the following section.

1.3.1 Serial vs. parallel systems

Considering kinematics (introduced and explained in the following section), a parallel system is a closed-loop mechanism that is made up of an end-effector with n (n>1) degrees of freedom (DOF's) and a fixed platform, linked together by m (m>1) kinematic chains. These kinematic chains are serial systems of links and joints. Actuation takes place through k (k>n) simple actuators in chosen revolute and prismatic joints (Huang et al. 2012; Liu and Wang 2013). Parallel systems are therefore more complex, because they have multiple kinematics chains and represent multiple end effectors. Mathematically, inverse kinematics solutions to serial systems, such as a construction excavator can be finite or infinite, whereas parallel systems are more commonly indeterminate. In terms of technical applications, parallel manipulators and systems are more applicable in high precision positioning and control applications, while serial manipulators are most useful for gross motion (Patel and George 2012). The type of end effector in parallel manipulators is typically a platform (multiple target flanges/grippers), while the end effector in serial systems is typically a single flange/gripper (Figure 1-3). Based on this definition, construction assemblies can be represented by serial and parallel manipulators according to the key constraints defined. Figure 1-3 shows typical serial and parallel assemblies, and illustrates the application of the robotics analogy for realigning construction components.

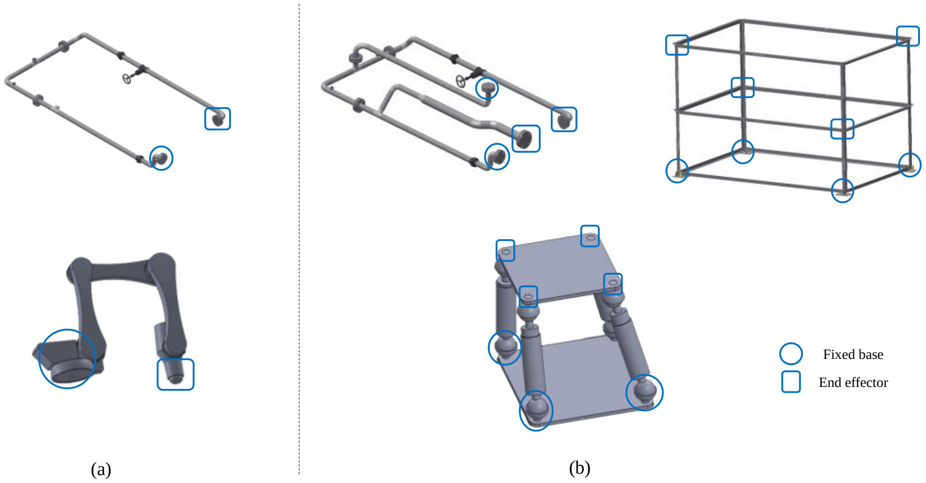


Figure 1-3: (a) Serial, and (b) parallel systems/manipulators

Figure 1-4 shows how serial and parallel assemblies and the analogy of robotics can be used for automated realignment and discrepancy management systems.

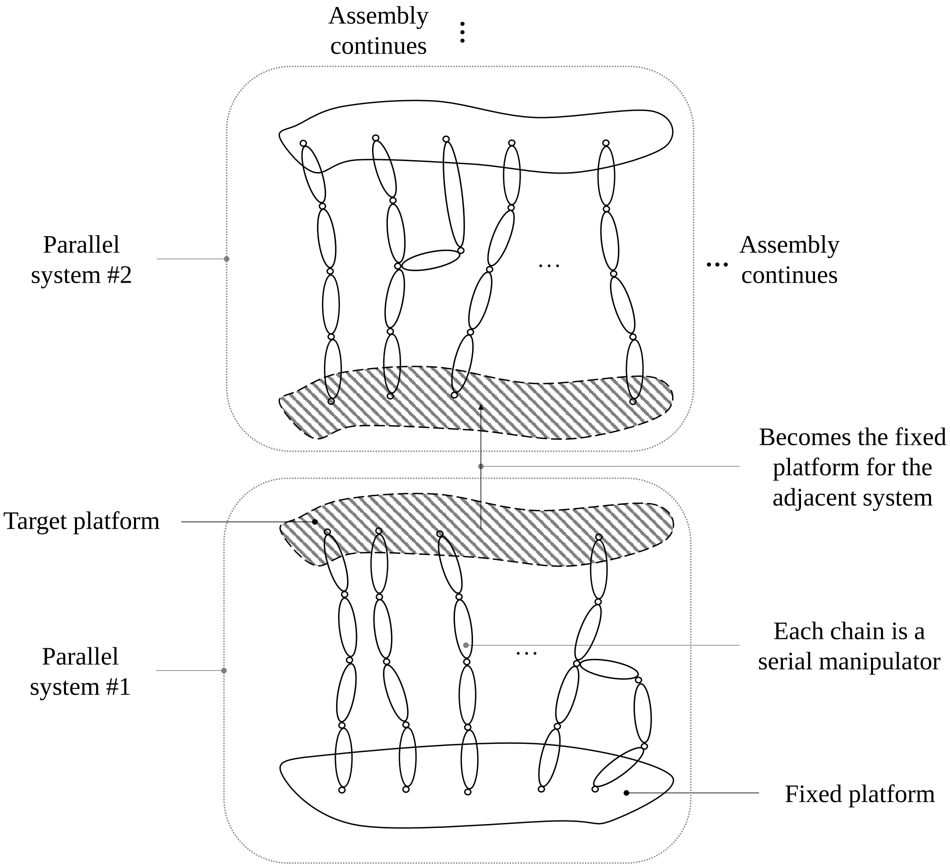


Figure 1-4: Generalized representation of construction components assembly using serial and parallel systems (manipulators)

1.3.2 Forward vs. inverse kinematics

The study of position or pose of robotic manipulators involves two problems: inverse and forward kinematics. Forward kinematics is the problem of finding the unknown pose (position or orientation) of an end effector or mobile platform associated with a specific set of joint and link inputs, which are constrained by the system's degrees of freedom. In construction, forward kinematics has been employed for modeling and investigating construction robots. Moreover, forward kinematics is also employed for chain development of vision mechanisms such as real-time 3D scanning and tracking of an object of interest by (Cho et al. 2012; Talmaki et al. 2015) The inverse kinematics problem involves mapping a known pose of the end effector or mobile platform of the mechanism to a set of input joint and link variables that will achieve that pose (Liu and Wang 2013). Inverse kinematics concepts have been used to solve many challenges in the construction industry. Such applications include position error modeling of automated manipulators (Cho et al. 2004) and automated earthmoving (Singh 1997). As mentioned earlier, the proposed method for realignment planning and corrective action in this thesis makes use of inverse kinematics concepts from robotics.

1.4 Research scope

While this analogy can be made for many types of construction assemblies, this thesis focuses on prefabricated industrial construction assemblies for verification and validation purposes. Industrial assemblies include complex pipe spools and pipe racks and modules that are widely used in industrial plants (see Figure 1-3). Utilization of the desired framework for automated realignment will substantially affect productivity by proactively planning and applying the required changes on defective assemblies. It will then minimize the unfavorable rework on the projects.

1.5 Research objectives

Detailed objectives of utilizing the desired framework for development of transformations between designed status and built status include:

1- Development of a customizable framework for 3D compliance checking of construction assemblies,

2- Automated measurement of the discrepancies incurred for reliable realignment planning,

3- Development of realignment plans for serial and parallel assemblies, and

4- Generalization of the realignment plans and realignment actuation.

Achieving the aforementioned objectives will lead to development of the required components for proactive and automated realignment and actuation of construction assemblies within an integrated construction management systems, as shown in Figure 1-2.

1.6 Thesis organization

This thesis is structured and presented in 7 chapters as shown in Figure 1-5.

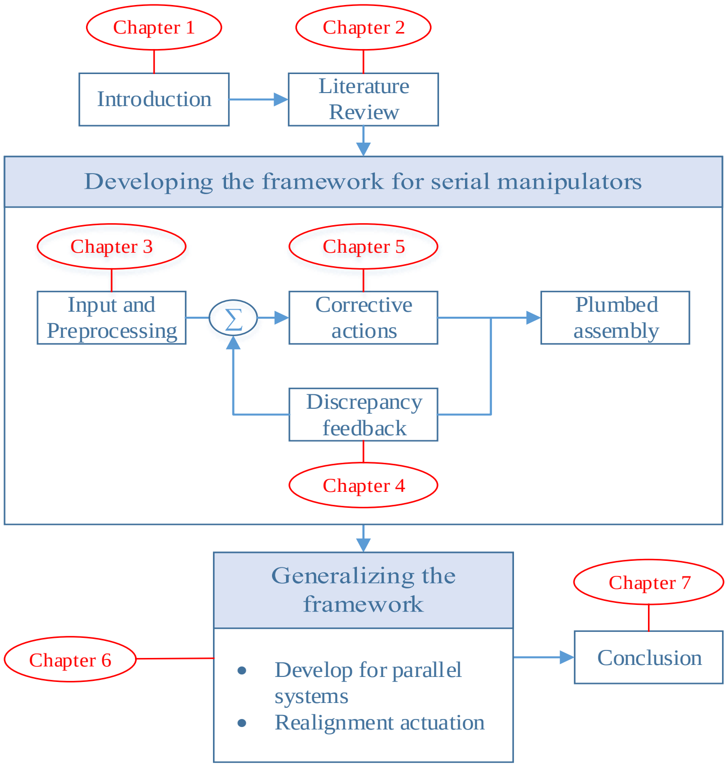


Figure 1-5: Thesis structure within the big picture (control systems theory)

Chapter 1, the current chapter, introduced the challenging problems existing in construction and states the motivation for this research. The research scope and objectives were also identified. The related literature and background is then investigated in Chapter 2, in order to thoroughly frame the knowledge gap. The literature is investigated from different viewpoints toward the frameworks proposed in this research.

In Chapter 3, the preliminary steps required to achieve the objectives of this framework are presented. The metrics and functions for performing the point cloud registration, as well as the principles of 3D compliance checking and existing challenges are formed and developed. The framework developed for the preprocessing step is customizable, so that the effective variants can be easily measured to investigate their impacts and overcome the existing challenges. The framework is tested and validated by an experimental study.

Once the inputs for the proposed framework are appropriately preprocessed, the incurred discrepancies are detected, localized, and quantified in Chapter 4. An automated, reliable, and (semi) real-time feedback is calculated in local coordinates that enables the quantification of incurred discrepancies in the form of required transformations for the realignment frameworks. 3D imaging and robotics kinematics theories are employed for detecting and analyzing the discrepancies.

Chapter 5 focuses on developing the framework for realignment calculation using an inverse kinematics analogy. A linearization approximation based on Taylor series expansion is used for calculating the required changes in the previously defined DOF's. The realignment plans are then offered based on the existing conditions on construction sites and assemblies. The developed framework is tested and validated on serial assemblies (i.e. serial pipe spools).

Chapter 6 generalizes the developed realignment frameworks for any type of construction assembly. A general and algorithmic procedure for realignment calculation and actuation of construction assemblies is developed. A new method for solving the kinematics chain equation of parallel systems is also presented. The developed algorithm is verified and validated for experimental-sized case studies.

Chapter 7 summarizes and outlines the findings and results of the developed frameworks explained. Contributions to the body of knowledge are also identified. Limitations are discussed, and potential research avenues for future work are recommended.

Chapter 2

Literature Analysis

2.1 Summary

In this chapter, the related background is investigated in order to comprehensively frame the knowledge gap. As briefly mentioned in the introduction (Chapter 1), the proposed framework emerges from a combination of various research streams in construction engineering. In summary, the proposed framework uses some tools for automated realignment planning of construction components. The related literature for this framework is investigated in three primary thrusts: (1) 3D imaging, (2) fabrication and assembly control, and (3) robotics. Each thrust is then evaluated from various perspectives, as shown in Figure 2-1.

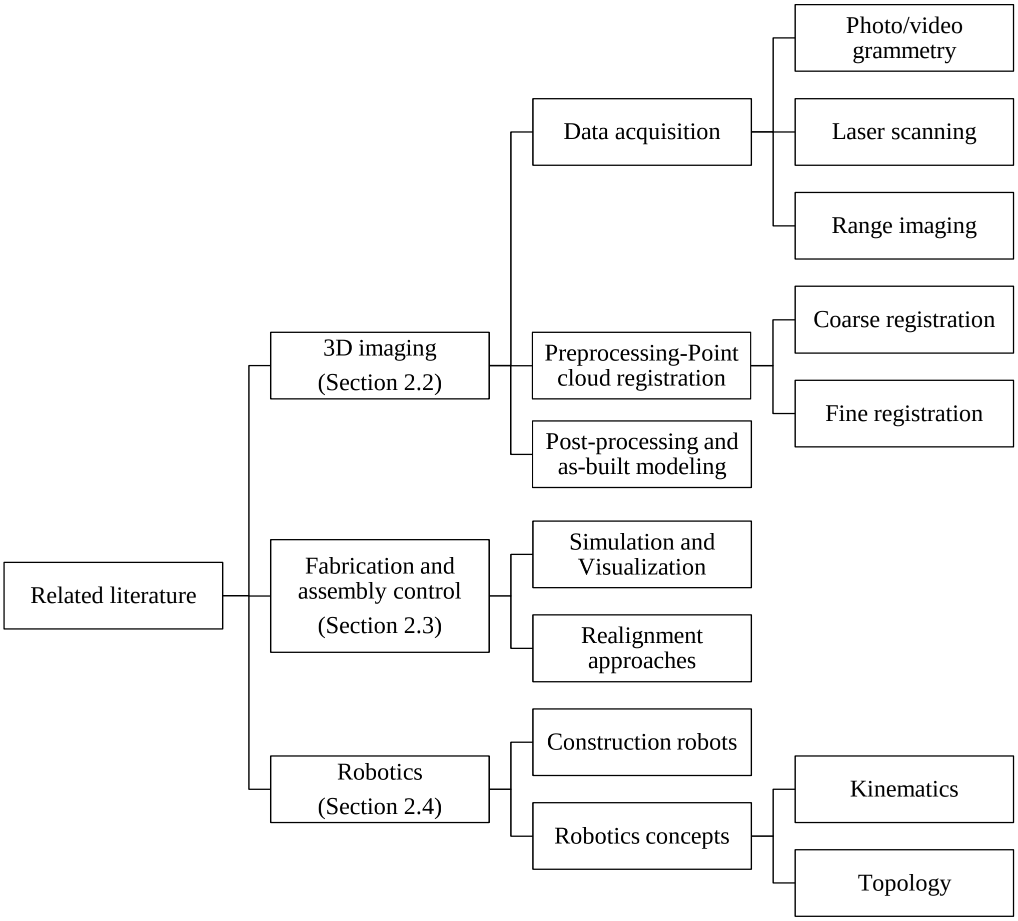


Figure 2-1: The related literature and investigated streams for background analysis

2.2 3D imaging

Generally, depth generation from 2D images is called 3D Imaging or Stereoscopy. Laser scanners or digital photogrammetry can be used to acquire a 3D image. The 3D image produced is also called a 3D point cloud. Photogrammetry was first used for depth generation from aerial photos. However, with significant recent advances in computing and taking the advantage of computers, it has become a very beneficial tool for various applications such as automated inspection, as-built status assessment, safety, and motion tracking. Such methods are used for data acquisition. Once a sufficiently accurate point cloud is acquired, some processing is performed in order to extract meaningful information. This information can then be used for various purposes such as progress tracking and quality control. Particularly, the three main steps for the 3D imaging process used in this thesis are data acquisition, registration, and post processing (as-built modeling). These steps are explained and discussed in the following section.

2.2.1 Data acquisition and point cloud generation

In the construction industry, information which represents coordination of objects at any time is hugely beneficial. Such information can be used in various phases of construction and for various purposes including progress measurement (Kim et al. 2013a; Turkan et al. 2012), equipment and material tracking (Memarzadeh et al. 2013; Yang et al. 2010), safety planning (Chi and Caldas 2012), and productivity tracking (Gong and Caldas 2011). The most commonly used technologies for 3D data acquisition in the construction industry are evaluated in this section. Once data are captured using the appropriate method, it should be compared with 3D CAD models that exist in BIM. Registration is then required to perform this. Thus, different methods of registration are then described extensively in the next section.

Finding and matching the conjugate pairs in different images, which is called photogrammetry, used to be the only efficient method for generating a 3D point cloud. However, with recent advances in related technology, a point cloud can be produced significantly faster and more accurately. Laser scanners are also improving in accuracy and cost. Once three-dimensional information is acquired, it can be used for status assessment. These methods are different in terms of capture speed, purchasing cost, and preprocessing time.

2.2.1.1 Photo/video-grammetry

Photogrammetry includes various mathematical and geometric constraints. In general, all photogrammetric approaches consist of bundle adjustment that is the combination of two main constraints in photogrammetry, which are collinearity and coplanarity. The processing procedure is becoming significantly faster by taking the advantage of powerful computers in the processing steps. This may be a reason for the increase in using photogrammetry for various applications. It was initially offered as an accurate method to visualize 3D as-built status in the construction industry (Memon et al. 2005). In that study, digital photos were used for producing the as-built status and the resulting point cloud was then compared to CAD models to measure the progress of construction projects automatically. Although it was introduced as a robust method, its application in construction projects remained limited due to the significant cost and time of computation.

It was then integrated with laser scanning to automatically measure the progress of construction projects (El-Omari and Moselhi 2008). Studies then revealed its usefulness for damage detection in different areas of civil infrastructure. (Ahmed et al. 2011b) presented a very low cost and efficient method for pothole detection in pavements. Other examples of point cloud generation and reconstruction using photogrammetry in construction projects can be found in (Ahmed et al. 2011a; Ahmed et al. 2012).

Digital photogrammetry can also be used for progress tracking, if a wide range of photos are taken during construction (Golparvar-Fard et al. 2009b). Although preprocessing has become easier and faster using existing software packages, it is still limited since it requires intensive manual interaction in the preprocessing step. It also does not handle homogeneous or poorly lit surfaces well either. All studies point out that photogrammetry is considerably less expensive compared to other methods of data acquisition, however it requires more preprocessing that makes it less applicable in construction projects (Bhatla et al. 2012; Dai et al. 2013; Golparvar-Fard et al. 2011).

In the case that required features are extracted from video cameras and recordings, the technique is called videogrammetry. Since video frames are sequential, pixels existing in each frame are progressively reconstructed based on the previous frame. This characteristic of videogrammetry enhances the reconstruction of civil infrastructure more quickly. The level of accuracy for progressive reconstruction using videogrammetry has become reasonable using high resolution cameras and considering its low cost of purchase.

Several studies have been done recently to investigate the effectiveness and robustness of using videogrammetry in civil infrastructure. It was used as an alternative method for infrastructure reconstruction and was investigated for feasibility of being employed in construction projects (Brilakis et al. 2011). The research shows that it can be used for 3D reconstruction as it provides the desirable accuracy. More studies then showed its particular use in different sectors. Examples of using videogrammetry for damage detection and safety evaluation can be found in (Brilakis et al. 2010; Koch and Brilakis 2011; Koch et al. 2013; Zhu and Brilakis 2010). In a similar study, an algorithm is developed to identify objects from video frames in order to detect moving equipment on construction sites (Chi and Caldas 2011). Most of the above mentioned techniques are machine vision-based, and the object recognition for identification and classification is performed employing different tools.

2.2.1.2 Laser scanning

Three dimensional laser scanning, which is also known as LADAR (Laser Detection and Ranging), has become the most common technology to acquire 3D point clouds in the engineering industry. It was first introduced as an accurate method to capture construction sites and items (Cheok et al. 2001) and was then used as a robust and accurate method to capture the current status of construction projects in different sectors of civil infrastructure such as structural, industrial, and architectural areas (Rabbani et al. 2007; Tang et al. 2010).

The method was used to develop a framework for quality control purposes on construction sites (Akinci et al. 2006a). A framework was presented to use 3D sensors in the quality control process. It was concluded that using 3D sensing technologies and laser scanning in particular can significantly speed up the quality control process and avoid potential errors that cause rework. 3D laser scanning was then used as an innovative method for structural health monitoring (Park et al. 2007). They concluded that the accuracy provided by a laser-based 3D image enhances the ability to monitor and control the performance of construction items from the structural point of view.

In a study by (Turkan et al. 2012), laser scanning was used to track the progress of a construction site by recognizing the built components and comparing them with the project schedule. The method of object recognition was based on Bosche's approach (Bosche 2008; Bosche and Haas 2008; Bosche et al. 2009). The approach uses 3D CAD models, which are available in 3D Building Information Models (BIM), and compares with the acquired point cloud. The method is robust and has a high rate of recall compared to more recent studies for progress tracking that are mostly machine vision- based

(Kim et al. 2013a). As discussed, although laser scanning has become the most common tool for data acquisition in construction projects, it has still remained untapped for remote assessment purposes. Current approaches for object recognition have opened up potential solutions using accurate and robust methods to inspect and control the quality in the construction industry as the components are being fabricated.

2.2.1.3 Range imaging

3D range cameras are useful for tracking moving objects and construction equipment and materials (Bhatla et al. 2012; Teizer 2008). Various applications of 3D range cameras are explained in the mentioned research. A method is then developed to rapidly determine the blind spots using range images to avoid accidents in sites (Teizer et al. 2010). In a similar study, range images are used for obstacle avoidance on construction sites in order to reduce the likelihood of accidents like collision of equipment with workers and power lines (Teizer et al. 2005a). Range cameras are independent of the backlight and can cover a wide field of view. They are inexpensive compared to laser scanners but more expensive than digital cameras used for photogrammetry. Recently, a range camera called Kinect $^{\text{TM}}$ has been used for safety purposes (Ray and Teizer 2012). In this study, a range camera is used for real-time monitoring of construction workers to avoid hazardous postures and gestures. The developed algorithm is sufficiently robust with a high rate of recall and is trained for a wide range of body postures.

2.2.1.4 3D data acquisition summary

As discussed earlier, each data acquisition method used in 3D imaging and reconstruction has advantages and disadvantages. Table 2-1 shows a brief comparison of the described techniques for data acquisition.

Table 2-1: Comparison of different methods of data acquisition

Data acquisition method	Accuracy*	Cost	Ease of use	Potential applications
Laser scanning	High	High	Less portable, requires preprocessing for merging separate scans	Very accurate applications such as quality monitoring of sensitive infrastructure
Photo/video-based	Medium-high	Low	Easily applicable, can be used on a mobile phone or tablet	Progress monitoring, object and material tracking
Range imaging	Low-medium	Low	Easily applicable	Close range object recognition and tracking

* See (Bhatla et al. 2012; Dai et al. 2013; Golparvar-Fard et al. 2011)

As illustrated in Table 2-1, despite being expensive and less portable, laser scanning is the most accurate and least time consuming method among all common techniques for reconstruction of civil infrastructure. However, depending on the budget and required level of accuracy, the appropriate method can be selected in different construction projects. In this thesis, laser scanning is used for data acquisition.

2.2.2 Preprocessing- point cloud registration

In general, registration is the process of transforming two sets of data into one coordination system (Zitová and Flusser 2003). For 2D images, registration is used for several purposes such as 3D image (point cloud) generation, object recognition, object tracking, and motion tracking. Registration can be performed using either visible features like corners edges or non-visible features that should be calculated using the scale invariant feature transform (SIFT) (Lowe 2004). Registration of 2D images in construction projects have been mostly used for 3D reconstruction of civil infrastructure.

Construction sites retrieval using shape-based (Brilakis and Soibelman 2008) and material-based (Brilakis et al. 2005) techniques is performed to reconstruct as-built status of the projects. The researchers developed machine vision-based tools for automated particular features recognition and registration consequently. Having found the common features in different 2D photograph, it becomes simple to find the transformation for registration. More recently, 3D image data from construction sites are retrieved from 2D time lapsed photographs using SIFT features in order to track the progress of projects (Golparvar-Fard et al. 2009b).

Registration techniques in 3D can be classified from different points of view. Determination of matching features in two different datasets used to be performed manually. These common features can be sphere targets in different scans used for occlusion avoidance. Although the traditional method was more accurate in some cases, it is not reliable due to an insufficient level of automation for practical application. Additionally, traditional methods are time consuming and not applicable for huge practical point clouds. From this perspective, registration is classified into automated versus interactive methods. Based on the level of automation that the method of registration provides and the level of achieved accuracy, different techniques could be compared. Although automated registration techniques are used to merge different point clouds taken from different viewpoints, the main purpose of registration in this framework is to compare as-built status with 3D CAD models. In other words, it is assumed that an accurate 3D image is generated based on the described methods using the appropriate data acquisition tool.

Point cloud registration is summarized here as consisting of two main steps: coarse registration and fine registration. Coarse registration is an initial step that is performed to roughly align the two point clouds. This rough alignment results in acceleration of the subsequent steps and prevents getting stuck in local minima. Consecutively, fine registration finds the best fit between the point clouds by minimizing the error which is the Euclidean distance between correspondences (Salvi et al. 2007). Different methods for coarse and fine registration and their advantages versus disadvantages are investigated in the following sections.

2.2.2.1 Coarse registration

The main goal in the coarse registration is to find a proper alignment for two datasets. Some of the registration techniques that are not preceded by the coarse registration step may not work robustly and correctly. The existing methods for coarse registration can be classified based on the method used for finding the matching correspondences and the transformations required for the registration. According to the comprehensive study done by (Salvi et al. 2007), most common correspondences used for coarse registration are point-to-point (Chua and Jarvis 1997; Johnson 1997; Johnson and Hebert 1998). However, lines (Vanden Wyngaerd and Van Gool 2002) or even surfaces (Chung et al. 1998) may be used for the coarse registration step.

Having found the correspondences, it is required to transform the matching features. For this purpose, the least-squares adjustment method is the most common technique to transform the models.

The method of least-squares finds the best transformation by minimizing the error between the objective point clouds. The rapidness of the method depends on several factors such as the size of the point clouds, number of correspondences used, and the method of optimization for solving the minimization problem in the least square adjustment. However, statistical approaches that dominantly use a covariance matrix for finding resemblance between two datasets have been used for coarse registration. Although they may not be sufficiently robust for all kinds of distributions and scatters, which are point clouds in this context, they work significantly faster. Random sample consensus (RANSAC) (Chen and Hung 1999) and principle component analysis (PCA) (Chung et al. 1998) are the examples of statistical methods used for registration. The first method searches for the best fit of a given sample in a database containing outliers and the latter one performs the alignment calculating the covariance matrix of the datasets and then the required transformation employing eigenvectors.

The PCA registration method for the rough alignment step is performed linearly which makes it very fast in contrast with the other methods that are performed iteratively. Although the registration may be more accurate for the iterative approaches, the achieved level of accuracy may not be required, since a rough alignment is desired in this step and the rapidness of the approach plays a more important role.

As discussed earlier, the main purpose of the coarse registration is to expedite the process of registration. It will be shown that if coarse registration is not performed and ignored, the distance minimization may get stuck in local minima which leads to incorrect registration which precludes further steps. Based on the necessity and its relative rapidness and simplicity in terms of computation, coarse registration is an important step and should be performed to achieve correctly registered models.

2.2.2.2 Fine registration

Once the datasets are roughly aligned, the fine registration step is performed to find a more accurate fit between the point clouds. The majority of the methods presented for fine registration are iterative, and the associated error is monotonically decreasing. The methods presented here are evaluated in terms of efficiency as a combination of robustness and rapidness for different point clouds. Different methods are explained in this section according to the comprehensive study done by (Salvi et al. 2007).

A very pioneering study was presented using the closest points as the correspondences for registration (Besl and McKay 1992). The approach presented is called Iterative Closest Points (ICP) and aims to register closest points in each of the iterations. It starts with searching for the nearest points in a point cloud from the other dataset. In other words, one subset of a point cloud is chosen; then nearest points in the other dataset are determined. The subset chosen for the nearest-neighbour search is called *“control points”*. Depending on the method used for sub-sampling or selecting the control points, ICP can be performed more robustly considering different distribution in point clouds. The control points can be sampled uniformly, randomly, or normally (Rusinkiewicz and Levoy 2001). Depending on the distribution of points along the point cloud, the proper method for subsampling is selected.

Nearest-neighbor search methods also play a key role in the ICP approach. Various search types were thus used to find the nearest neighbors of the control points (Greenspan and Godin 2001). In later years, several studies were attempted to improve the robustness of the ICP approach and extend it for any types of point clouds in any situation. Different techniques are used to determine the correspondences in ICP. For instance, ICPIF (ICP with invariant features) uses invariant features as the correspondences. In addition to these invariant features, color and curvature in range images are also invaluable to facilitate the ICP registration (Godin et al. 2001). Overall, ICP is the most commonly used and the most robust method for fine registration. Although the rate of convergence is slow compared to other methods for fine registration, its simplicity for implementation can benefit the users. Several variants affecting the performance of ICP are extensively investigated by (Rusinkiewicz and Levoy 2001). Different methods used in each step are evaluated and their performances are compared for different sizes of point clouds and different distributions.

Simultaneously with developing the ICP approach, a similar method was developed for range image registration (Chen and Medioni 1992). Their approach is primarily based on point-to-plane Euclidean distance minimization. While the performance of Chen’s method is quite different compared to ICP, it has been considered as an improvement to ICP method in the literature. In later years, different attempts were done to improve the performance of ICP by adjusting different parameters during the registration process and employing new tools for optimization, transformation, and correspondence registration. A novel method presented by (Masuda 2001) is an example of significant improvements to ICP method. Having formulated the registration as a high dimensional optimization problem, it can be resolved using modern optimization techniques such as Genetic Algorithms (Chow et al. 2004). Although the presented method is faster for specific point clouds than the previously discussed registration methods, it is known as a difficult approach to be formulated and implemented.

Summarily and adopted from the comparison studies done by (Salvi et al. 2007), modified ICP is the most efficient algorithm among the presented methods. Although in some cases the accuracy achieved by Chen's method is slightly higher than ICP, Besl's implementation is significantly faster. For the coarse registration step, since PCA is a linear transformation in contrast with the other methods that are iterative, it performs the rough alignment faster in the order of a couple of magnitudes.

In construction technology, automated registration is dominantly used for progress tracking. Recently, (Bosché 2012) presented a method for registration of construction laser scans with 3D/4D building models. The coarse registration is plane-based and it is followed by ICP-based iterations for fine registration. More recently, a fully automated registration method is presented for scheduling and progress tracking (Kim et al. 2013b). In the previously discussed methods, the process is semi-automated as the noise removal is being performed manually. However, Kim's approach for automated registration includes an automated noise removal using a decay function. The method is followed by an ICP-based registration in the fine registration step as in Bosche's algorithm. ICP algorithm plays a key role in various components of this framework.

2.2.3 Post processing and as-built modeling for industrial assemblies

As mentioned earlier, a processing step is required to extract meaningful information from massive point clouds acquired on construction sites. Such techniques for as-built status assessment are called *Scan-to-BIM* (RW.ERROR - Unable to find reference:462; Ahmed et al. 2014; Son et al. 2014a) and *Scan-vs.-BIM* (Bosché et al. 2014b; Son et al. 2015; Son et al. 2014b) based on Bosché's definition (Bosché et al. 2014b). Scan-vs.-BIM results in the recognition and identification of the objects and it thus becomes possible to measure the discrepancies by comparing to the as-designed models existing in the BIM. (Bosché et al. 2014b) showed the possibility of tracking MEP (Mechanical/Electrical and Piping) works using the Scan-vs.-BIM registration. Although their approach potentially improves the quality control process in the MEP industry, the need for systematic realignment and adjustment of defects and discrepancies still has to be addressed.

In the industrial sector, in general, and pipelines and pipe spools, in particular, as-built status reconstruction is a key challenge because of the complexity of the components involved (Kim et al. 2013c; Son et al. 2013). (Lee et al. 2013) have recently introduced a skeleton-based approach for automated reconstruction of cylindrical objects including elbows and T-sections. Wang et al. (Wang et al. 2014) have recently integrated augmented reality with building information models (BIMs) to track the progress and assess the built status on piping projects. More recently, an automated approach has been developed for detection and identification of cylindrical objects in the construction industry (Ahmed et al. 2014). As thoroughly discussed by (Rabbani et al. 2007), the application of the Hough transform in 3D space is not time effective and cannot be employed for real-time monitoring of facilities because of the intensive calculation and processing involved. To overcome this limitation, (Ahmed et al. 2014) took the advantage of using a 2D Hough transform and investigated point clouds projected into sequential slices in orthogonal directions. The resulting Hough circles are then consolidated and merged to reconstruct the objects in 3D and detect the diameter of the cylindrical objects. Bosché et al. (Bosché et al. 2014a) integrated Scan-vs.-BIM defined by (Bosché et al. 2014b) with Ahmed's approach for detection of cylindrical components (Ahmed et al. 2014). Although their approach is significantly faster than higher order Hough transforms and has a high rate of recall and precision, it is directed toward automated detection of as-built status of cylindrical objects and is much less well suited for identifying defects and discrepancies in local coordinates as described here.

From the deviation analysis point of view, a recent study by Anil et al. (Anil et al. 2013) showed the benefits of performing deviation analysis for the as-built status assessment of residential buildings. The benefits of their method include time and cost effectiveness compared to conventional methods (e.g. tape measuring and manual data acquisition), which ultimately improves productivity. However, using deviation analysis software packages results in overlooking existing constraints on construction sites, because they are based on point-to-point distance calculation, as discussed earlier. This limitation becomes more important in sequential erection procedures such as industrial facilities where on-site deviations strongly rely on the situations of adjacent components and tolerances. In current approaches, on-site situations are not properly modeled, and the deviations calculated are therefore inaccurate and unreliable for further considerations required to be given to the defective assemblies. In other words, the feedback signal calculated based on current approaches to deviation analysis is unreliable to be used for developing the required realignments for fitting the assemblies. Figure 2-2 illustrates these ideas.

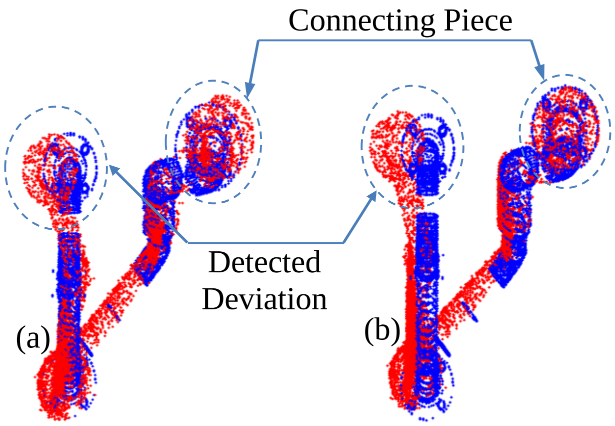


Figure 2-2: Unconstrained (a) vs. constrained (b) registration. The deviation analysis results without (a) and with (b) considering onsite situations are shown. The connecting piece is fixed in (b) while these situations are overlooked in (a). As a result, the detected deviations vary as depicted above. Setting the connecting piece as fixed, constrained registration (b) results in a larger deviation than regular registration (a), and creates an origin for subsequent refitting and realignment planning

As shown in Figure 2-2, in constrained registration, both global and local situations are modeled, while in the regular registration, the situations are only modeled globally and conditions in adjacent parts are overlooked.

2.3 Fabrication and assembly control

During the past few decades, the advantages provided by the enhancement of computer processors have enabled the increasing application of construction simulation. During the 1990's, simulation tools and their use were limited to academia and an insignificant number of contractors because they were neither affordable nor efficient. In later years and with the development of graphical programming interfaces, construction simulation became an inseparable part of a construction project, with consequent key improvements in scheduling and productivity (Kang et al. 2008). The significant efforts of Abourizk and his collaborators (AbouRizk 2010) have resulted in simulation becoming a highly beneficial tool with a long-term impact on prediction and visualization of real projects.

2.3.1 Visualization and simulation

Simulation was employed to control the process of modular construction by considering the constraints in the simulation. The delivery of materials and the process schedule are the key challenges that have been dominantly used in the simulation for modular construction. Planning and scheduling in construction was firstly introduced as an optimization problem considering the project constraints and cost minimization by (Hegazy 1999a; Hegazy 1999b). For the modular construction and staged fabrication, simulation combined with scheduling considerations has been used for optimization of resource allocation and project scheduling (Sharma et al. 2006). Mobile crane and other equipment localization are other important issues in the industrialized building construction solved by simulation (Lei et al. 2013).

Recent 3D visualization applications have seen particular use in offsite construction modeling (Han et al. 2012b). According to (Han et al. 2012b), required BIM input is used in order to represent individual elements for the visualization, some of which require predefinition: project schedule, geometry, required tasks, and component specifications. Visualization facilitates the process of assessing projects after changes are applied in the simulated model. It can also assist with the prediction of both the impact of the changes on other processes and potential interference with other tasks assigned to the crews. The results underline the ability of visualization to expedite project schedule and prevent rework, both of which lead to improved productivity and cost saving. Conflicting processes and tasks can be easily determined using 3D visualization, and potential errors are automatically detected before they result in significant increases in project costs. A number of scenarios are investigated through visualization, and a favourable schedule can then be automatically generated (Alvanchi et al. 2012). However, most of the studies in this area have been limited to schedule planning and resource allocation at construction sites (Azimi et al. 2011; Kamat and Martinez 2003; Kamat et al. 2011).

Despite all of the benefits that simulation can provide for modular construction, it is still limited due to the high level of knowledge required by the users to be applied and implemented in real-world projects. Simulation users should be familiarized with the terminology and software interface prior to employing the software, which makes it harder to be applied by project managers. Applying schedule changes in the simulation tools requires processes by users and is not simple to be implemented by users. In order to address these limitations 3D visualization has been introduced recently. In general, visualization is an extended type of simulation followed by visual demonstration of the results. Once the staged fabrication tasks are visualized, it is much more understandable for users to see the impact of applied changes compared to the simulation results that require higher level of knowledge for processing. Since 3D geometry and construction components are visualized, workspace requirements and other geometrical considerations are consequently identified. This early identification plays a key role in prevention of probable stoppages in projects. Hence, using 3D visualization would be more advantageous considering the resulted animation to show the processes and changes.

Providing geometric information for 3D visualization is on the basis of the existing information in the simulated construction model. Thus, the first step is to integrate the simulated model and define how to extract required information for 3D visualization. (Kamat and Martinez 2001) presented a framework for visualization from a simulated model and existing information structures. The presented model was then validated by applying on real construction projects as a complementary tool for construction simulation (Kamat and Martinez 2003). Based on the mentioned limitations that simulation cannot address, 3D visualization was widely approved by project managers and became a common decision tool for the processes during construction jobs. One of the values that 3D visualization added to the construction industry was the optimum localization of the equipment such as crane towers in complex construction sites (Al-Hussein et al. 2006; Han et al. 2012a).

Recently, 3D visualization has been particularly used for offsite construction modeling (Han et al. 2012b; Moghadam et al. 2012). In the model presented, required inputs that exist in the Building Information Model (BIM) are used to represent different elements for the visualization. These elements that require being predefined include project schedule, geometry, required tasks, and components specifications. The attempted research facilitates the process of assessing projects after applying changes in the simulated model. Additionally, it can assist in prediction of the impacts on other processes and potential interferences with the rest of the tasks assigned to the crews. The results outline that the method expedites the projects schedule and prevents rework, which both lead to improved productivity and saved costs. In addition to the minimized risk of rework, post-simulation and visualization minimize the idleness time of construction workers and make the method more efficient that can result in saving costs as a secondary effect. The conflicting processes and tasks are simply determined using 3D visualization, and potential errors are automatically detected before causing significant costs to projects. Various scenarios are investigated by visualization, and the favorable schedule is then automatically generated. As mentioned earlier, 3D visualization significantly improves productivity in the modular construction industry; however most of the attempted studies in this stream are limited to schedule and resource location planning on construction sites.

2.3.2 Current realignment approaches

Due to the complexity of the tasks involved in the fabrication process, such as welding, punching, and drilling, fabricated assemblies are unavoidably distorted. This distortion is associated with the concept of residual strain (and stress in some cases) in materials. Generally, there are two existing methods for correcting the distortion in fabricated assemblies: (1) applying force, and (2) heating (AISC 2005).

Applying force for correcting the fabrication errors is feasible for light assemblies. The applied force can be in the form of hydraulic presses, local jacking and welding, turn buckles, chain falls and come-alongs. However, the applied force method is not practical for heavier assemblies. They should be heated and then cooled when they require correction. As the defected area is heated it expands, but since it is surrounded by cold unheated area, the expansion is constrained. Then the heated area is left to be cooled and the expanded elements become shorter due to the constraint. The heating method is applicable to large steel sections and complicated pipe spools. This process is very sensitive and if not evenly applied, as unwanted curvature occurs which requires further correction. As seen, the process requires skilled workers availability on construction sites and accurate tools for the heating method (AISC 2005).

Offsite construction requires the fabricated modules to be assembled on sites. The modules should be leveled accurately to avoid causing extra loads on structural elements and escalated tolerances. Fitting modules together on site is usually performed using bolts. Thus, the modules should be aligned when it is lifted to be permanently installed. However, due to some unavoidable effects on modules that occur during fabrication, transportation, or installation, fitting modules is always a key challenge for installers and requires experienced workers (Taylor 2009; Velamati 2012). These effects include human errors during fabrication and changes in weather conditions. For instance, panelized assemblies may become defected by high weather humidity. Hence, modularization in severe weather conditions such as many parts of the North America and Canada, in particular, requires special considerations. In addition, lifting heavy modules may affect their capacity to resist structural loads.

Despite all of considered specifications for weather-proofing and transporting the assemblies, perfectly controlling the quality of modules and prefabricated assemblies is unfeasible in real projects. In practice, experienced and professional installers are essential to fit and align the marrying adjacent modules. The mentioned process for aligning defected modules is considerably time-consuming and not practical for huge modules. Finding the best fit among a critical number of panels in different locations is also a challenge that requires a thorough automated deviation analysis to monitor the tolerances. Currently, such an automated system to check whether or not the module fits to its adjacent modules is missing on construction sites. In cases where the item does not fit well, it has to be realigned manually, and that is a source of rework and causes delay in projects. Thus, monitoring, identifying and characterizing defective modules ahead of time and in such a systematic way that prevents rework can save huge costs.

In general, six consecutive tasks are involved in the erection process of steelwork:

1. Receiving material from fabrication shop,
2. Temporarily lifting and holding material in place,
3. Bracing and supporting temporarily to assure the stability,
4. Permanently aligning the connections by bolting or welding,
5. Connection cladding, and
6. Final coat application.

The delivery of materials on construction sites should be somehow managed to avoid out-of-sequence delivery that causes significant delays and costs to projects. Besides managing the delivery of materials, the assemblies should be securely braced in order to ensure that they withstand temporary loadings such as wind, snow, and temperature changes. Thus, the erection sequence of the delivered materials should be performed based on the consideration of critical parts and elements. Figure 2-3 shows the importance of staged erection of steel assemblies. As illustrated here, the bays that are braced should be erected and installed earlier than the other bays; otherwise the structure will collapse or require severe bracing that is costly. For example, in the case shown in Figure 2-3, AB and CD bays should be erected and installed earlier.

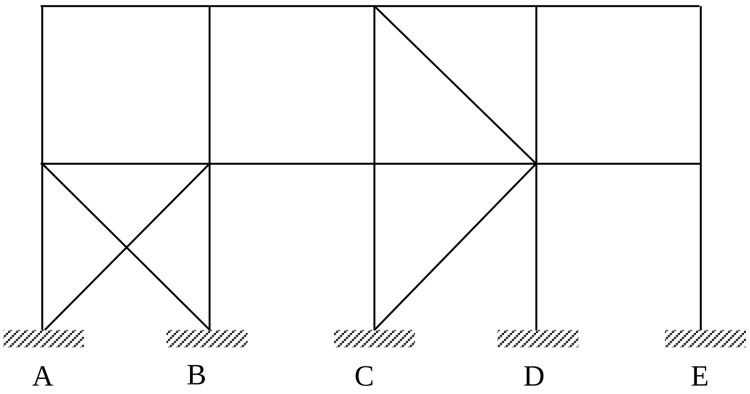


Figure 2-3: The braced bays and staged erection process

To ensure that the assembly imperfections are within the acceptable tolerance limit, the accuracy of fabrication should be monitored from the very beginning steps. Considering the structural role of the assembly, the tolerance limit is identified. The assembly dimensions should fall within the required tolerance limit in order to be functionally and safely installed (AISC 2005).

However, the most challenging part of the erection is to permanently align the adjacent modules (AISC 2005; Sarah Slaughter and Eraso 1997). Once the assembly is lifted, a trial for fitting to the adjacent modules is performed. In case it is aligned, it fits the adjacent modules and further steps are performed accordingly, otherwise another position should be tried to see whether or not it is plumbed. Different positions are tried using heavy chain falls and winches. This procedure is done a few times, and if the assembly does not fit, realignment and repair should be planned. After inspecting the defected assembly, the realignment plan is applied on it and if realignment is not effectively and economically performed, it is replaced (Sarah Slaughter and Eraso 1997). Typically, the erection and plumbing on construction sites are performed using the flow diagram in Figure 2-4. This diagram is originally used for simulating the erection process on construction sites.

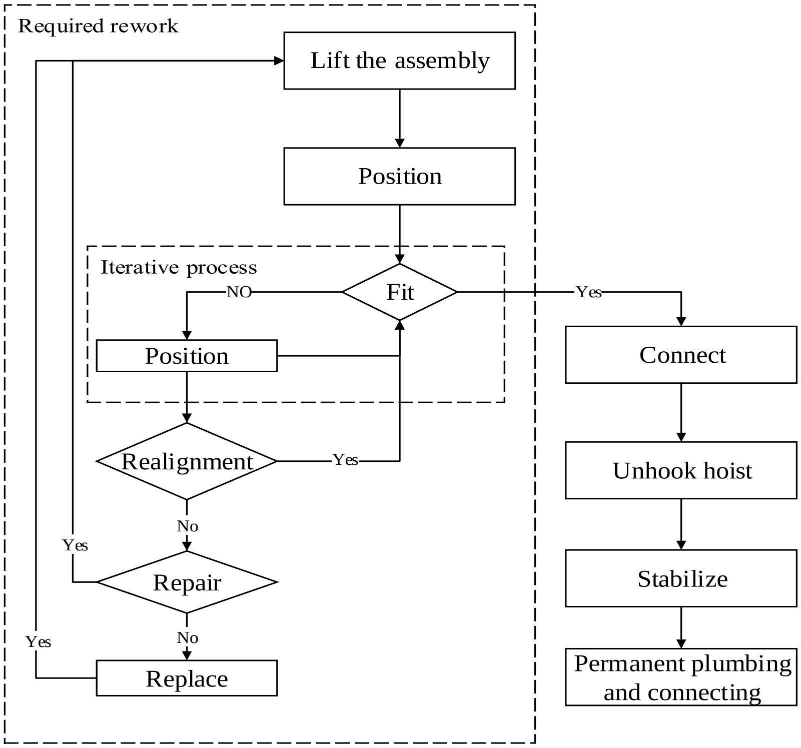


Figure 2-4: Steel assemblies erection and installation flowchart

The iterative part is performed based on the importance of the assembly on the construction site. A cost analysis may be performed to find the number of iterations required for running the process reasonably economically. Basically, the tasks required for aligning the defected assembly including the iterative part, realignment, and repair are associated with the "*rework*" concept in construction projects that causes significant delays on sites. The delay in installation and erection conflicts with the delivery of materials on sites as well as the construction schedule, which both cause huge costs to projects. As illustrated in Figure 2-4, after running a few alternative positioning trials, if the assembly does not fit the adjacent modules, realignment is performed and the fitting trial (rework) continues. Simulating the erection process using the diagram in Figure 2-4, enhances the planning for delivery of materials in a timely manner; however, the main aim of the description provided here is to figure out the complicated process involved with typical assemblies on construction sites.

In summary, alignment of marrying pieces and assemblies is currently performed by skilled workers and managers, and an opportunity is presented for automation in the defect detection and realignment application steps. In other words, the defects are identified with fitting trials on sites, which is costly and time consuming. Having found the occurred errors in the shipped assemblies, workers try to achieve the design state based on the experience they have gained in that particular project. Thus, due to unavailability of skilled workers and unreliability of current processes, project managers are eager to find a systematic way to identify errors and the required transformation to achieve the best fit to the design state. Such a framework enables minimization of the rework associated with the realignment. Furthermore, automating these tasks will help the decision makers to choose the best option; either the fabrication error is going to be realigned, or it is going to be compensated by (consecutively) moving the adjacent assemblies. Such a framework for automated alignment of defective assemblies in the construction industry still needs to be addressed.

2.4 Robotics in construction

In this research, generation of the realignment plan is achieved using inverse kinematics procedures by representing an assembly such as pipe spool, as a robot arm with degrees of freedom (Cho et al. 2004; Craig 1989). In this section an overview of the literature related to both construction robots and the application of robotic concepts in the construction industry is investigated. Construction robots that have been developed over the past few decades have significantly facilitated the process of construction by increasing the level of automation. Beyond that, using the algorithms and concepts that are originally developed in this area has increased the level of automation as well. For example, using the *"Path Planning"* concept, which is originally developed in robotics, expedites the construction process (e.g. heavy lifts) and prevents interference of conflicting tasks and consequently increases the level of safety. Investigation and evaluation of the literature from these points of view in the construction industry is the main subject of this section.

2.4.1 Robotics concept application in construction

Two general concepts are investigated in this section: (1) robotics kinematics and (2) serial vs. parallel systems/manipulators.

2.4.1.1 Kinematics

Generally, kinematics refers to the science of motion which evaluates the motion of a particle or rigid body without consideration of the source of motion. Computing the position, velocity, and acceleration are all included within the science of kinematics. In robotics, the science of kinematics is classified into two main categories: (1) Forward Kinematics, and (2) Inverse Kinematics that are defined as follows.

2.4.1.1.1 Forward kinematics

Forward or direct kinematics refers to the application of kinematics in robots to compute the position and orientation of the end effector from given geometry and required joint characteristics and parameters. In simpler words, given a set of joint angles and link lengths, forward kinematics is to compute the position of the end effector relative to the origin. This is also known as direct kinematics (Craig 1989).

2.4.1.1.2 Inverse kinematics

Inverse Kinematics refers to the use of kinematics to find the joints situation in order to reach a specified coordination posed by the end effector. Calculating all possible sets of joint angles that lead to a given position of the end effector is not as simple as forward kinematics since the kinematics parameters are nonlinearly related to each other. Thus, a closed form solution may not exist in practical applications, and heuristic approaches may be used (Cho et al. 2004). The questions that arise here also include the existence of solutions, the singularities investigation, and the multiple sets of acceptable solutions on the basis of the kinematics equations. Figure 2-5 schematically shows forward and inverse kinematics concepts in robots.

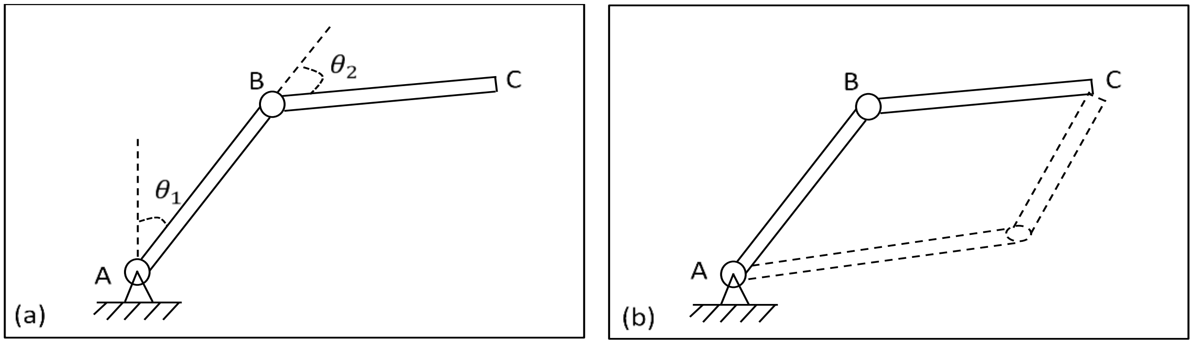


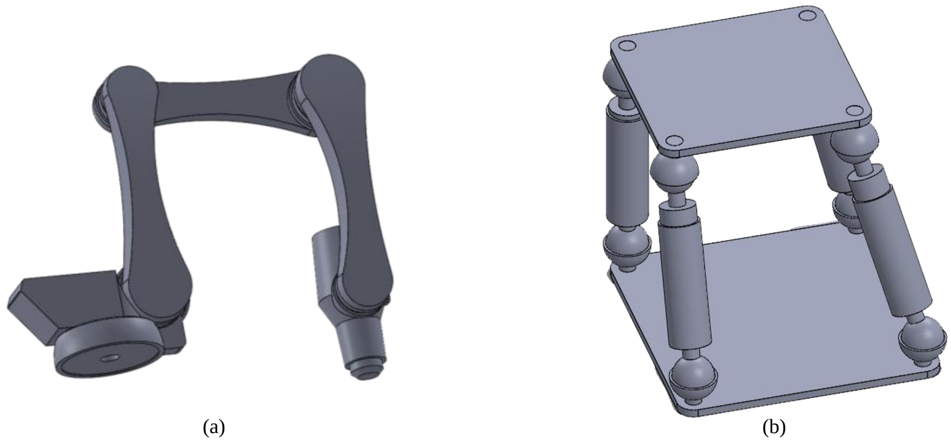
Figure 2-5: Schematic demonstration of forward (a), and inverse (b) kinematics.

As illustrated in Figure 2-5, the joint angles are given values of θ_1 and θ_2. Thus the position of the point *C* (end effector) is calculated considering the member lengths are given as well. In Figure 2-5 (b), the position of the end effector and the geometry (lengths) are known and the joint angles are desired to be calculated. Based on the existing conditions in the particular case shown here, two possible solutions are demonstrated.

2.4.1.2 Serial vs. parallel systems

Serial or sequential systems or manipulators, also known as serial manipulators, are the most common robots. They are composed of a series of links connected by joints from a base to end effector. Since six degrees of freedom (DOF's) are theoretically required to place an object in a specified position and orientation, sequential systems usually have six joints. An example of a serial manipulator with six DOF's is demonstrated in Figure 2-6 (a). Sequential systems are widely used in construction equipment.

Parallel systems have key differences with serial manipulators. Generally, a parallel manipulator is composed of several serial systems that simultaneously control the end effector. In comparison to serial manipulators, parallel systems are significantly more rigid. However, their workspace is limited compared to serial systems. Additionally, their nonlinear behaviour makes their use limited in high precision applications. Instead, they are widely used in movement simulators, such as flight and automobile simulation in order to investigate the movement effect. Figure 2-6 (b) shows a typical parallel system. A key assumption in such systems is that the links are rigid and movement occurs at joints. This is in contrast to a finite element model of an assembly or structure.


(a)

(b)

Figure 2-6: (a) Serial (sequential), and (b) parallel systems/manipulators

2.4.2 Robotics hardware application in construction

The necessity of automation and decreasing human interaction (Atkinson 1999) in construction processes is demonstrated by the increasing use of robots in different construction jobs such as fabrication, installation, and inspection. Using construction robots improves safety in harsh conditions and hazardous circumstances. Additionally, it improves productivity as the main objective by improving the automation level and promptness of construction tasks. Thus, using construction robots has become common in various types of construction jobs, such as *"stakeless earth moving"* which is in common practice now. Since the initiation of robotics in controllable construction equipment and Large Scale Manipulators (LSM), it has been improving by tremendous advances in the technologies involved such as computer processors and materials used. In this section, different sectors of construction that are improved using construction robots are investigated, and related studies are evaluated.

The idea of using robots for automating construction tasks was developed in the early 1980's (Haas et al. 1995). Health and safety were the most challenging concerns that drove the use of robotics in difficult construction tasks. Excavators, fork-lifts, and LSM on a crane are the common construction equipment that actually perform tasks as robots. Since then, the challenges involved in such equipment are the subjects of the most of studies by researchers. An example of using robots in civil infrastructure is the automated crack sealer used for pavement maintenance (Haas 1996). This study presents the development of an automated machine (robot) that recognizes and seals the cracks in pavements automatically. The performance of the machine was improved later by adding automated path planning to the crack sealer (Kim et al. 1998a; Kim et al. 1998b).

Simultaneously with tremendous advances in developing new robots for different construction tasks, researchers attempted to improve the existing systems. In other words, fusing the existing construction robots with other technologies such as 3D imaging or laser scanning opened up new solutions to improve the productivity in construction projects and increase the level of automation. (Cho et al. 2002; Kim et al. 1998a; Kim et al. 1998b) presented a framework to automatically model construction sites using a large scale manipulator carrying a laser scanner to acquire data for real time modeling. The framework was then used for modeling unstructured construction workspaces (Cho and Haas 2003). The developed method can also be used in as-built status assessment by the real-time information provided automatically. The accuracy of the framework was then improved by real-time error modeling and applying an inverse kinematics to the large scale manipulator (Cho et al. 2004).

Robots are particularly used in the offsite construction industry. After turning out to be a beneficial tool in the construction industry, the use of robotics became more common in the offsite construction tasks. Due to the repetitive nature of the jobs and the necessity of maintenance in prefabricated modules, a strategy called Robot-Oriented Design (ROD) was introduced to take the benefits of robots on construction sites (Bock 2007). This strategy mainly aims to increase the use of robotics in the construction industry and investigates the feasibility of using robots for different jobs. Precast concrete panel production, prefabricated masonry walls, wooden panel production, and three dimensional steel units are the examples of using robots in the offsite construction industry that are thoroughly investigated by (Bock 2007). In addition to the huge use on offsite construction sites, robots are also used for repetitive onsite tasks such as road construction and maintenance, tower and bridge building, dam construction, and tunneling. Mobile material distributors that are used on sites, mobile tunneling machines, and façade detectors are the examples of beneficial robots used in construction. Using robots in the offsite construction also enhances the builders to achieve customized products faster, which improve the productivity.

2.5 Knowledge gap and contribution

The analysis of the related literature lead to the conclusion that automatic realignment and adjustment of defective assemblies is an emerging need for rework avoidance and therefore cost saving. Although, significant advances have been made for automated as-built modeling, potential approaches for automated realignment and actuation planning have remained relatively unexplored. Developing and utilizing such automated frameworks could proactively mitigate the rework associated with heuristic and unsystematic realignment approaches that are currently being used. This thesis aims to address the above-mentioned challenges in construction. For development of the required transformations between as-designed and as-desired (could be as-built) configurations, the proposed framework here makes construction assemblies analogous to robot arms, and therefore uses the concepts that are originally developed in robotics. While the algorithms developed are to some extent generalizable, validation experiments in this thesis are focused on pipe spool fabrication and simple steel frame alignment given their lead roles in industrial and commercial construction respectively. The experimental results show that the method is sufficiently accurate to be employed for realignment planning and actuating strategies. Required components and metrics are explained in the following chapters.

Chapter 3
Inputs, Preprocessing, and Basic Compliance Checking

A similar version of this chapter is published as the following peer-reviewed journal article. Only minor editorial changes are applied for being consistent with the University of Waterloo thesis format.

Nahangi, M., Haas, C. (2014). "Automated 3D compliance checking in pipe spool fabrication", *Advanced Engineering Informatics,* Volume 28 (4), 360-369.

3.1 Summary

While the focus of experimental work in this thesis is on piping fabrication and steel framing because of their central roles in industrial and commercial construction, any framework for the transformations required to align these assemblies must start with a treatment of the acquisition of the data used to detect and measure misalignment, and it must describe how that data will be preprocessed to facilitate compliance checking. In the industrial construction sector, pre-fabrication errors inevitably occur due to the complexity of the tasks involved in the pipe spool fabrication process, the inaccuracy of the tools employed for performing these tasks, human error, and inadequate inspection and monitoring during the process. Permanent deflections may also occur during shipment and transportation. After delivery at construction sites, defective spools must be detected and further consideration given to the erection of the spools to tolerance levels specified; otherwise, the repair and realignment associated with rework can cause schedule delays and consequent substantial costs increases. This chapter presents an automated approach for monitoring and assessing fabricated pipe spools using automated scan-to-BIM registration. Defects are detected through a neighbourhood-based Iterative Closest Point (ICP) approach for the registration process. Experiments show that the proposed approach can be employed for the automatic and continual monitoring of such assemblies throughout fabrication, assembly and erection to enable timely detection and characterization of deviations. The main objective of the framework presented in this chapter is an automated 3D inspection framework and algorithms for construction assemblies, in general, and pipe spools, in particular.

3.2 Problem statement

Fabrication imperfections in construction assemblies are often unavoidable due to the intensive level of manual work involved. Such imperfections and errors can add significant costs to construction projects, especially when they are not detected early. According to (Akinci et al. 2006a; Burati Jr. et al. 1992; Hwang et al. 2009; Josephson and Hammarlund 1999; Patterson and Ledbetter 1989), up to 10 % of construction costs are attributable to rework due to the late and untimely identification of defects, and 5 % are the result of rework during maintenance (Akinci et al. 2006a; Patterson and Ledbetter 1989). A major risk associated with construction projects is thus the cost of rework due to defects that occur as a result of human error or material flaws. It has been noted that more than 50 % of rework cases arise from human errors due to an insufficient level of inspection or lack of regard for the importance of quality assessment in the construction phase (Akinci et al. 2006a). Approximately 10 % of the cost of construction rework is attributable to material defects (Akinci et al. 2006a; Yates and Lockley 2002). Continual inspection and assessment to detect human error and the quality of construction materials can therefore prevent rework and save substantial associated costs.

Even minor errors in the fabrication process may have a significant effect on the functionality of modular and fabricated parts. For example, in a steel structure, a wrongly fabricated beam can impose significant forces or bending moments in its structural system. In modular construction, to mate adjacent parts during installation they are often moved in order to compensate for fabrication errors, which can cause significant accumulated displacement, deflection, residual stress, and exceeded tolerances. Weather conditions at construction sites and temperature changes during welding are other examples of sources of error that produce inaccurate results (Kassimali 1999). To avoid rework, fabrication errors should therefore be systematically controlled.

Current approaches for the inspection and assessment of materials, however, are time consuming, inadequate, and inefficient (Section 2.3.2). Craft workers typically use conventional manual measurement devices, and most contractors or managers employ paper-based assessment procedures for the inspection of the quality of materials. These methods are flawed because of the inexact, infrequent and inadequate communication of the manual data among inspectors, contractors and managers. The management of traditional methods of communication between different interfaces is challenging due to the complexity involved. Surveyors and inspectors are too expensive to track continual changes while items are being fabricated, assembled or built. This may result for example in tolerance stacking of assemblies and spools in manifolds. The broader result is that the lack of an integrated and rapid monitoring system for tracking changes means that errors that could be fixed prior to significant rework are overlooked. Hence, providing the inspectors and superintendents (O'Brien et al. 2011) with a fast and automated monitoring framework for detecting fabrication errors or inaccuracies would make it possible to plan for the potential realignment, repair or replacement of a defective part in a timely manner.

An automated approach should thus be developed to avoid conditions that can lead to inaccuracy, ineffectiveness, and inefficiency during the inspection procedure. Recent advances in electronic technology for data acquisition and processing have been fused for use in construction quality control procedures (Akinci et al. 2006a). However, a fully automated procedure that is reasonably accurate with respect to the assessment of the dimensional quality of individual fabricated construction items is still lacking.

As reviewed in Section 2.2.1, recent advances related to data acquisition for as-built assessment have provided beneficial information about real-time monitoring and inspection, and fabricated assemblies can now be scanned and accurately reconstructed within an acceptable time frame (Brilakis et al. 2012). Of all available reconstruction approaches and data capturing devices, the laser scanner is the tool most commonly employed due to its ease of use, robustness, and accuracy (Bosche and Haas 2008; Tang et al. 2011; Tang et al. 2010; Turkan et al. 2012). Although it has become the most appropriate tool for retrieving the as-built status on construction sites, its cost of purchase is a major limitation. In contrast, photogrammetry and other 3D imaging techniques are significantly less expensive than laser scanners and are also effective sources of data for assessing the as-built status (Brilakis et al. 2012; Cho and Haas 2003; Golparvar-Fard et al. 2009b; Golparvar-Fard et al. 2011; Teizer et al. 2005b). The determination of as-built status in construction projects has thus become possible using laser or image-based tools that are sufficiently accurate and reasonably rapid.

This chapter presents an automated approach to continual assessment of the fabrication process of steel assemblies in general and pipe spools in particular. This approach includes automated registration of scanned-to-designed data, followed by an interactive comparison of the fabricated state at any point in the fabrication process with the desired states in order to identify fabrication accuracy. The 3D CAD models available in building information modeling (BIM) represent the data about the designed state, and laser-scanned point clouds indicate the as-built status of industrial assemblies and pipe spools. Enhancing the comparison through the use of automated registration, as proposed here for pipe spools that are randomly located and oriented on construction sites or in fabrication facilities, is challenging.

3.3 Compliance checking methodology

As comprehensively discussed in Section 2.2, 3D imaging approaches have been applied predominantly for the detection of damages to civil infrastructure, and their impact on the quality control process has remained relatively limited, because they generally detect damage that has already occurred and indications such as cracks in concrete structures and potholes in pavement. They are generally incapable of detecting potential errors. On the other hand, in the fabrication process, the use of 3D images and associated approaches has been limited to simulation and visualization based on identifying conflicting tasks and resolving related issues. The framework proposed here is directed instead toward defect detection and prevention through process control using 3D imaging techniques in order to address the deviations highlighted here. Fabrication errors are therefore caught early, prior to incurring substantial costs and creating significant delays in construction projects. Figure 3-1 depicts a summary of this framework and the flows between different components.

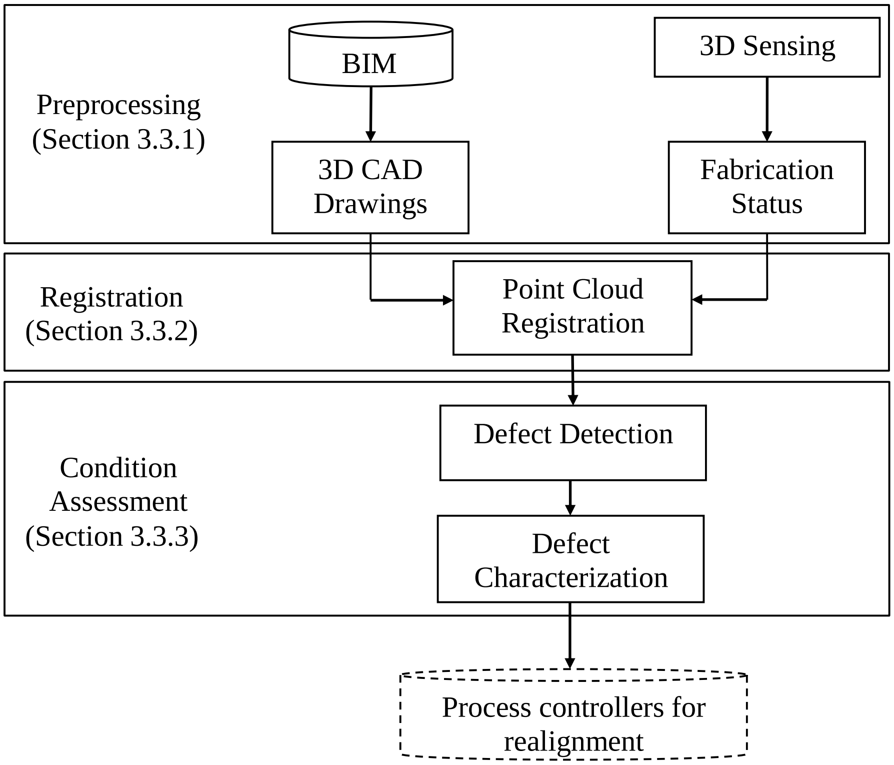


Figure 3-1: Proposed model for automated compliance checking and feedback for process control

A laser-based approach is used for the acquisition of the as-built status of prefabricated assemblies, and the comparison is based on point cloud registration with 3D CAD models. Once two models are registered, the final root mean square (RMS) value measured locally is the metric used for measuring the performance of the registration. Defective parts are then detected based on the quantification of the registration results. The proposed method includes the three primary steps stated above, which are described extensively in the following section.

3.3.1 Preprocessing-data preparation

3.3.1.1 Format conversion

Building information models (BIMs), including the coordination of three-dimensional objects and additional information such as time scheduling, are available in a variety of formats. As a basic rule, an integrated file known as BIM should embody descriptors that thoroughly represent objects. These descriptors can be solid objects, currently the most common type defined in CAD software packages. One of the major challenges of working with this type of object is that they are not easily comparable with acquired data due to their different natures and levels of complexity. These solid objects must therefore be converted to a dataset that can be compared to the captured data, which is simply a 3D point cloud. For that purpose, STereoLithography (STL) is the most appropriate and commonly used format for representing 3D solid objects in the form of 3D point clouds. It is an open source file format available either in binary or ASCII and provides coordination of vertices and vectors normal to the surface of triangulated objects. The conversion of 3D solid objects to point clouds is achieved through the estimation of facets with multiple triangles. The vertices are then considered the comprising points for generating the point cloud format. Similar file formats, such as object (OBJ), were later developed in order to address the deficiencies and limitations of previously developed methods. In the work presented in this chapter, the available design information is converted and implemented using STL because it is currently the only extension supported.

3.3.1.2 Noise, artifacts and clutter removal

It is almost impossible to avoid the presence of materials and equipment on construction sites and unwanted objects are also being moved or added to the region of interest (ROI). Such unfavorable objects are also scanned as well as other objects in the ROI, and they must be removed. Synonyms for such objects in the literature are “clutter” and “unwanted artifacts”. In practice, they result in noise or points superfluous to our purpose. For that reason, superfluous points are manually detected in point cloud processing software packages, such as FARO Scene (FARO 2008), Trimble RealWorks (Trimble 2006), or AutoCAD (AutoCAD 2012). This step is the only manual task in the proposed approach; however, a recent study has revealed that full automation of this procedure including noise removal, is now possible (Kim et al. 2013b). Sample results of the preprocessing step are shown in Figure 3-2.

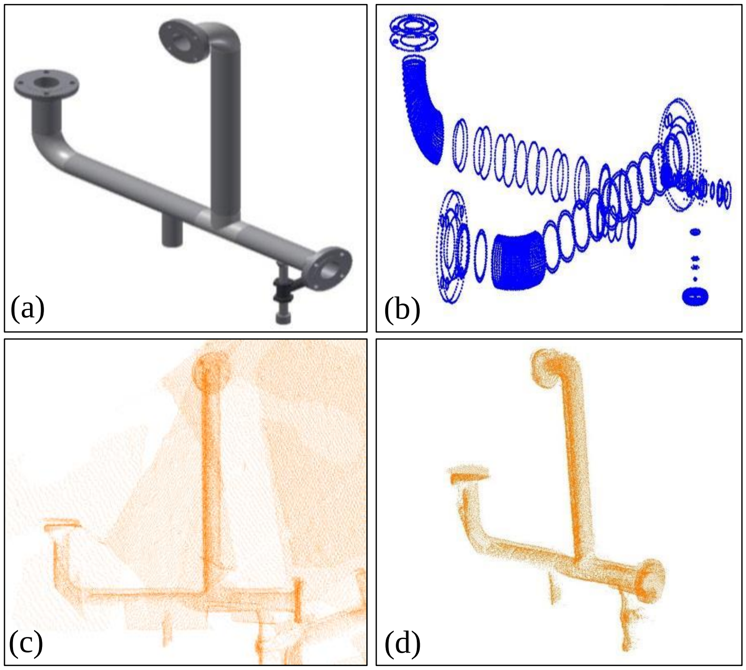


Figure 3-2: (a) 3D CAD drawing of one of studied pipe spools in solid objects format; (b) Using vertices in the STL format for converting to a point cloud format (*M*), rotated angle; (c) Original scanned point cloud, top view; and (d) Filtered point cloud (S), isometric view.

3.3.2 Registration

Once the preprocessing step has been performed, the point clouds are merged in order to enhance the comparison between the as-built and designed states of the assemblies. During the past couple of decades, considerable work has been directed at the evaluation of different approaches to registration (Jaw and Chuang 2008), (Rabbani et al. 2007), (Sharp et al. 2004), (Gruen and Akca 2005). As mentioned, registration is generally performed through two primary steps: coarse registration and fine registration (Section 2.2.2). Briefly, coarse registration is a rough alignment in order to expedite the fine registration step. Finding the best fit (global minimum) is almost guaranteed when the coarse registration is performed correctly (Section 2.2.2.1). Fine registration is then performed in order to find the best fit between the model and the scanned scene (Section 2.2.2.2). Figure 3-3 shows the proposed approach to the point cloud registration, which commences with principal component analysis (PCA) for the coarse registration, and followed by ICP for the fine registration. The PCA and ICP techniques are described in the following section.

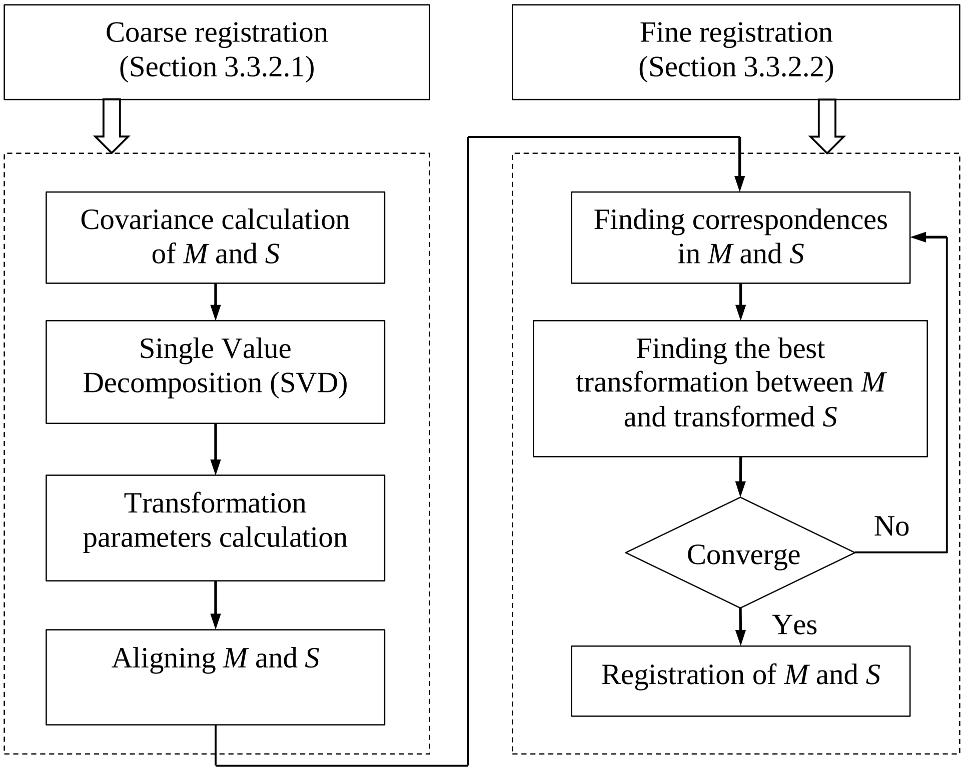

Coarse registration
(Section 3.3.2.1)
Fine registration
(Section 3.3.2.2)
Covariance calculation of *M* and *S*
Single Value Decomposition (SVD)
Transformation parameters calculation
Aligning *M* and *S*
Finding correspondences in *M* and *S*
Finding the best transformation between *M* and transformed *S*
Converge
No
Yes
Registration of *M* and *S*

Figure 3-3: Proposed registration algorithm

3.3.2.1 Coarse registration

To roughly align the 3D CAD model (*M*) and the scanned as-built status (*S*), coarse registration is performed. Chapter 2 included a discussion of coarse registration methods. Of these methods, PCA is sufficiently quick and robust due to its linearity of performance and the simplicity of the computation of the parameters involved. The accuracy provided by PCA is also adequate compared to that offered by the iterative methods. Briefly, PCA determines the principal axes in the two databases and then aligns them. The resulting registration is roughly aligned and significantly expedites the fine registration step, also guaranteeing that the iterative process of finding the best fit for the fine registration ascertains finding the global minima without becoming stuck in local minima.

The first step is to calculate the covariance. Given a 3D scatter, the covariance matrix is defined as follows (Kim et al. 2013b; Salvi et al. 2007):

$$K = \frac{1}{N}\sum_{i=1}^{N}\{(x_i - \bar{x})(x_i - \bar{x})^T\}, \quad i = 1,2,3,\dots,N, \tag{3-1}$$

where *K* is the covariance matrix, *N* is the number of points contained in the point cloud, and x_i is the i^{th} point in the point cloud. Once the covariance matrix is calculated for both *M* and *S*, the eigenvalues and eigenvectors of the two point clouds are calculated based on single value decomposition (SVD) of the covariance matrix (*K*):

$$K = U\Lambda U^T, \tag{3-2}$$

where Λ is a diagonal matrix, and U is the eigenvector. It should be noted that the eigenvector represents the principal axes of the point cloud, based on consideration of the distribution of the points in the point cloud. Following the calculation of U_M and U_S, which represent eigenvectors for *M* and *S*, the required translation components are defined. The rotation required (R_c) to align the principal axis of *S* with the principal axis of *M* is calculated as follows:

$$R_c = U_M \times U_S^{-1}, \tag{3-3}$$

and the required translation (T_c) is defined as follows:

$$T_c = \bar{x}_M - R_c \times \bar{x}_s \tag{3-4}$$

After *T* and *R* are applied as the required transformation components, the two point clouds are aligned based on the calculation of the directions of the principal axes in both point clouds. The results of the application of PCA show that it resolves the problem of becoming stuck in the local minima by causing the point clouds to be roughly aligned. Figure 3-4 shows sample results of the fine registration for both with and without the application of PCA. As can be clearly seen in Figure 3-4-(a), although ICP has been performed, the global minimum is not detected, and the registration results are insufficiently reliable to provide the level of accuracy desired for remote quality assessment. As indicated in Figure 3-4-(b), the global minimum is detected because of the initial coarse registration. Pseudo code and MATLAB program for PCA are provided in Appendix A.

3.3.2.2 Fine registration

After the coarse registration step, the point clouds must be registered more accurately to enable the evaluation of the scanned point cloud that represents the as-built status of the fabricated assemblies. As discussed extensively in Section 2.2.2.2, during the past couple of decades, a considerable number of studies have evaluated a variety of approaches to registration (Gruen and Akca 2005; Jaw and Chuang 2008; Rabbani et al. 2007; Sharp et al. 2004). Of the previously discussed methods, those that perform the registration more robustly and converge to the best fit more quickly for 3D point clouds have become valuable tools in automatic prototyping and remote assessment. On the other hand, feature-based methods work less efficiently and are thus unsuitable for cases in which significant accuracy is desired. Since ICP uses existing correspondences that are points, it converges more quickly in 3D than similar approaches. According to (Salvi et al. 2007), ICP is sufficiently fast and robust to be employed for noisy point clouds considering the mathematical computation required (Besl and McKay 1992). Due to all these advantages coupled with the simplicity of its implementation, ICP has been employed for the fine registration step in this research.

The iterative procedure registers two datasets Model (*M*) and Scene (*S*), by finding the points in the *Scene* closest to predefined subset of *M*. The robustness and swiftness of the ICP process is affected by a number of variants: the method of subset selection, the nearest point searches for matching, and the optimization techniques for error minimization (Rusinkiewicz and Levoy 2001). The Scene is iteratively translated and rotated as follows:

$$S^{*} = R_f \times S + T_f \ , \qquad \text{3-5}$$

where R_f and T_f are required rotation and translation matrices for updating S to S^{*}. Finding the required rotation and translation matrices involves an optimization problem for minimizing the error between the transformed points in S^{*} and *M*. The error represents the summation of the Euclidean distances of corresponding points in the two point clouds. The objective function for the optimization problem posed in this research can be expressed as follows:

$$Minimize\text{:} \qquad d = \frac{1}{N_S} \sum_{i=1}^{N_S} \|M_i - S_i^{*}\|^2 \ , \qquad \text{3-6}$$

where N_S is the number of points in the *Scene* used for finding the correspondences in the *Model* and d is the error that is to be minimized. It can be mathematically proven that the error (d) decreases monotonically during ICP (Besl and McKay 1992). However, it may converge to a local minimum rather than searching for the global minimum. Hence, an appropriate sufficiently fast optimization method that guarantees the global minimum will be found is essential here. Additional challenges associated with ICP are discussed in the following paragraph.

In the presented work in this chapter, fine registration is based on three primary assumptions related to solving the challenges involved in ICP:

1. For determining the correspondences that are the closest points, all points contained in the scene are considered to be the target subset,
2. A brute-force search is used for finding the nearest neighbors. The basic concept of this search method is the calculation of all distances followed by the selection of the nearest neighbors, which is also called an exhaustive search or naïve approach (Kjer and Wilm 2010). The search method can be modified for large point clouds by using more efficient techniques such as a KD-tree (Rusinkiewicz and Levoy 2001).
3. To avoid convergence to a local minimum, the Levenberg-Marquardt (LM) approach is employed for error minimization. Table 3-1 provides a summary of the advantages and disadvantages of common nonlinear optimization techniques that are in accordance with the principles expressed by (Bates 1988).

Table 3-1: Comparison of commonly used methods for nonlinear optimization

Method	Convergence Rate	Global minimum finding guaranteed
Steepest Descent (SD)	Slow	No
Gauss-Newton (GN)	Fast	No
Levenberg- Marquardt (LM)	Slow	Yes

As indicated in Table 3-1, of the commonly used nonlinear optimization methods, although it is slower than Gauss-Newton method, LM is the only approach that guarantees global minimum convergence. Like other nonlinear optimization methods, the LM method is based on a Taylor series expansion for the prediction of the next incremental value. The following are the primary equations for the calculation of the values required for determining the global minimum:

$$d\left(p+\delta_p\right) \approx d(p) + J\delta_p \,, \tag{3-7}$$

where J is the Jacobian matrix ($J = \frac{\partial d}{\partial p}$) and δ_p is defined as follows:

$$\delta_p = (J^T J + \lambda I)^{-1} J^T (d - d^*), \tag{3-8}$$

in which I is the identity matrix, λ is the non-negative damping factor adjusted at each iteration and $(d - d^*)$ is the residual function that is the difference between the real (d) and estimated values (d^*).

Once the optimization process is completed, transformation parameters are obtained and the model is updated. The minimization of the error enables the best transformation parameters to be calculated. The resulting rotation and translation matrices are used to update S and find a better match. This procedure is performed iteratively until the convergence criterion is satisfied. Figure 3-4 shows sample results of the ICP registration step with and without the previous coarse registration step.

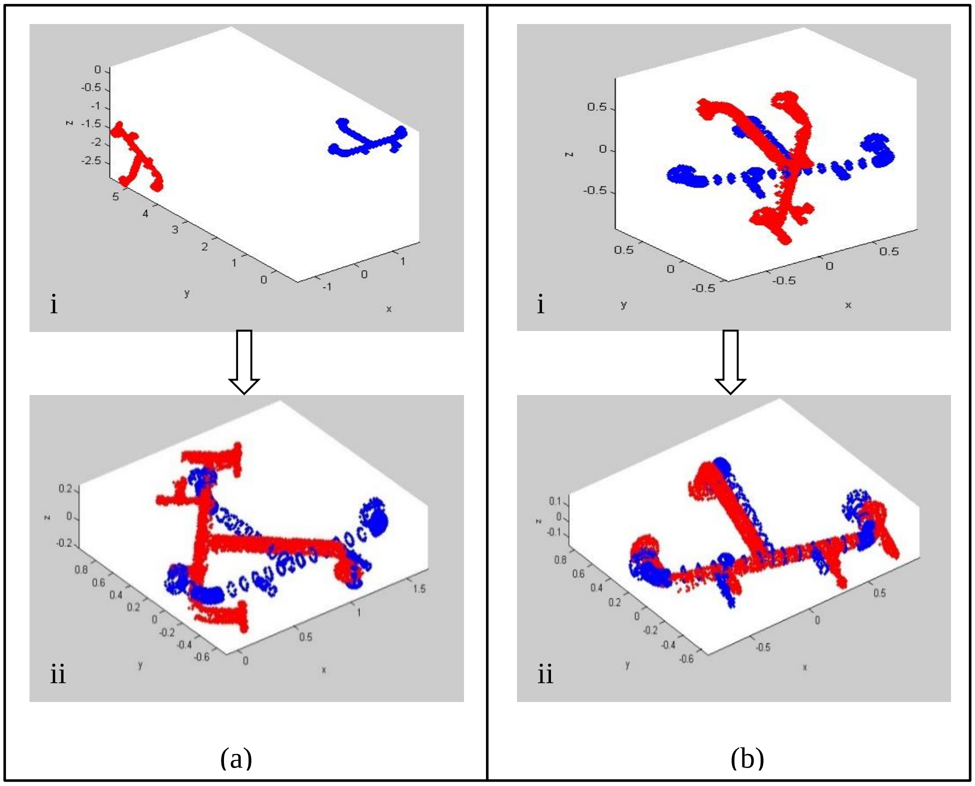


Figure 3-4: Sample registration results (a) without, and (b) with the coarse registration. (i: initial and ii: final positioning of the point clouds, red: Scanned and blue: 3D CAD model)

3.3.3 Condition assessment

Once the two point clouds are reliably registered, it then becomes possible to evaluate the dimensional accuracy of the assemblies. A condition assessment of the steel pipe spool assemblies is performed in two primary phases: detection of existing deviations and quantification of the deviations detected. The first step is performed through visualization of the error between the existing and desired condition, as provided by the registration process. The errors are then quantified based on an examination of the local root mean square (RMS) values. These steps are discussed in the following sections.

3.3.3.1 Automated deviation analysis

In the manufacturing industry, deviation analysis is a key technique for finding the defects in assemblies. Commercial software packages for performing deviation analysis have been widely used for the assessment of the as-built status of residential buildings (Anil et al. 2013). Researchers have held that using computer-aided techniques facilitates the productive detection of deviations in buildings. An automated distance-based deviation analysis is therefore used here as a means of visualizing potential errors and existing defects in assemblies. For this research, the method has been fully automated and implemented in MATLAB. Sample deviation analysis results for one of the assemblies studied are provided in Figure 3-5.

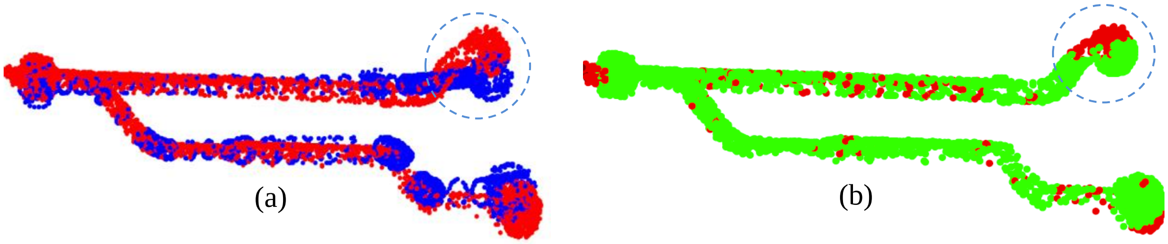


Figure 3-5: (a) Registration results for a wrongly fabricated pipe spool as shown. (Blue: 3D CAD in point cloud format, Red: laser-scanned data), and (b) Deviation analysis results to detect the defective parts (Green: well fabricated, and Red: incurred defects). Colors are edited in order to have the defective parts significantly highlighted. Red points that exist in the well fabricated parts are due to the registration results affected by the discrepancy that occurred at the remarked parts.

3.3.3.2 Local error investigation

The definition of a neighborhood-based metric is required because the total RMS value calculated in the registration step is inadequate for characterizing the deviations, a deficiency due to additional factors that also play a role in the determination of RMS value. These factors include assembly shape and point distribution in both the *Scene* and *Model* point clouds. For the purposes of this research, the *Scene* and *Model* point clouds are divided into corresponding clusters so that the local RMS investigation can be conducted. A 3D cube with a user-defined size is moved automatically along the assembly, as shown in Figure 3-7-(c), and an algorithm searches for points contained in both point clouds within the neighborhood defined by the cube. The local RMS is the error associated with the corresponding pair of point clouds in *M* and *S*, called *m* and *s*, respectively. The resulting RMS value enhances the characterization and quantification of the previously detected deviations. Figure 3-6 presents the proposed algorithm for the quantitative condition assessment of pipe spool assemblies (see Appendix B).

Input	**Processing**	**Output**
Registered *M* and *S* (Registration outputs)	1- Establish local clustering by moving a 3D cube along the assembly that has a user-defined size. 2- For each assembly position (*P*) find the containing points of *M* and *S* and assign the corresponding points as m_P and s_P. 3- For each pair of *m* and *s* calculate the RMS value.	Locally characterized as-built status

Figure 3-6: Proposed algorithm for automated condition assessment

3.4 Experimental verification and validation

3.4.1 Data

To validate the proposed methodology and measure the performance of the condition assessment approach for pipe spool assemblies, a set of experiments were conducted at the University of Waterloo’s Infrastructure Sensing and Analysis Laboratory. An assembly was designed and fabricated for the experiments and consists of a set of reconfigurable pipe spools, as shown in Figure 3-7-(a), and a pipe rack (not shown). The reconfigurability of the pipe spool makes it possible to set known displacements and rotations from a notional design state and then to validate and measure the performance of the proposed method. For this set of experiments, a specific branch of the pipe spool, as shown in Figure 3-7-(b), was studied. The end flange of the spool was rotated at a variety of rotation angles, as shown in Figure 3-7-(b).

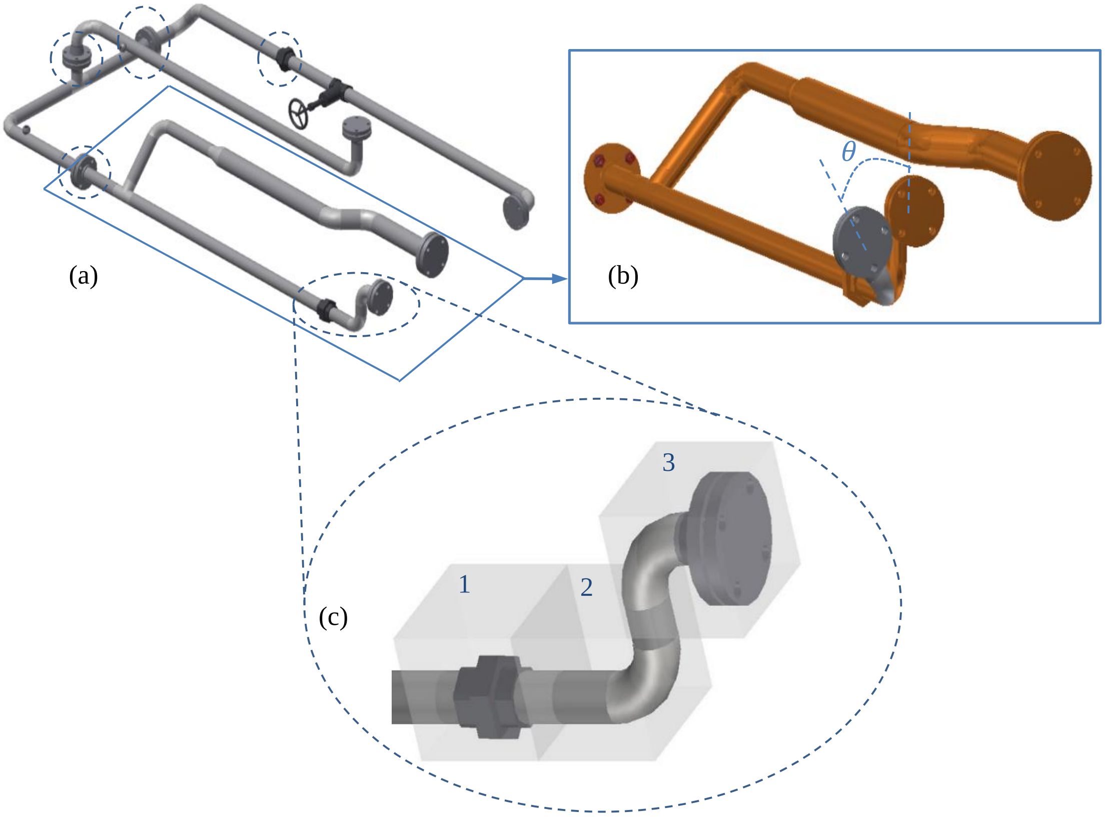


Figure 3-7: (a) Reconfigurable pipe spool in UW's Infrastructure Sensing Laboratory. The remarked joints are adjustable and make it possible to set known rotations and displacements at different point. (b) The branch that is investigated in the experimental study conducted in this chapter. The end flange is rotated at different angles (θ) and the incurred error is then quantified (θ is exaggerated to better illustrate the incurred error). (c) Critical cube positions for the experimental study.

Data were acquired using a *FARO LS 840 HE*, which is a phase-based laser scanner that measures the phase shifting between the shot and the captured ray. Based on the acquired phase shift, it then calculates the object coordination. Additional technical specifications for the laser scanner used in the experimental study are provided in Table 3-2. Clearly, a newer model would yield a more precise point cloud and improve results commensurately; however, available resources did not permit this.

Table 3-2: Technical specifications of FARO LS 840 HE

Laser Properties		Power: 10.5 *mW*; Wavelength: 785 *nm*
Distance	Range	0.6 *m* to 40 *m*
	Accuracy	0.6 *mm* (maximum resolution)
Angle	Range	Horizontal: 360º; Vertical: 320º
	Accuracy	Horizontal: 0.009º ; Vertical: 0.00076º
Maximum Resolution		Up to 700000000 points
Measurement Speed		120000 *Hz*

3.4.2 Results

To apply the proposed algorithm (Figure 3-6) for different alterations of the pipe spool, the end flange angle of rotation θ was changed, shown in Figure 3-7-(b). The original state ($\theta = 0°$) was tested initially and RMS values for the critical cubes, shown in Figure 3-7-(c), were extracted using the proposed algorithm. The same procedure was then performed for other alterations of θ. Two types of fabrication error typically occur in practice: incorrect fitting and welding, which usually leads to a few degrees of error, and inaccurate readings of the drawings which cause more substantial errors. To cover both of these cases, a wide range of rotation angles were therefore considered. For the implementation of the proposed algorithm for the local registration, $30 \times 30 \times 30$ *cm* was selected as the cube size. A summary of the results is provided in Table 3-3 and Figure 3-8.

Table 3-3: Experimental results for different alterations and at critical cube positions

Angle of Rotation	RMS (*m*) for different cube positions shown in Figure 3-7-(c)		
	#1	#2	#3
0°	0.028369	0.029423	0.024574
5°	0.030433	0.027646	0.026358
10°	0.033062	0.03202	0.033339
15°	0.035997	0.03577	0.035725
30°	0.040289	0.040421	0.055234
45°	0.045668	0.046307	0.063474
90°	0.059326	0.063373	0.116397

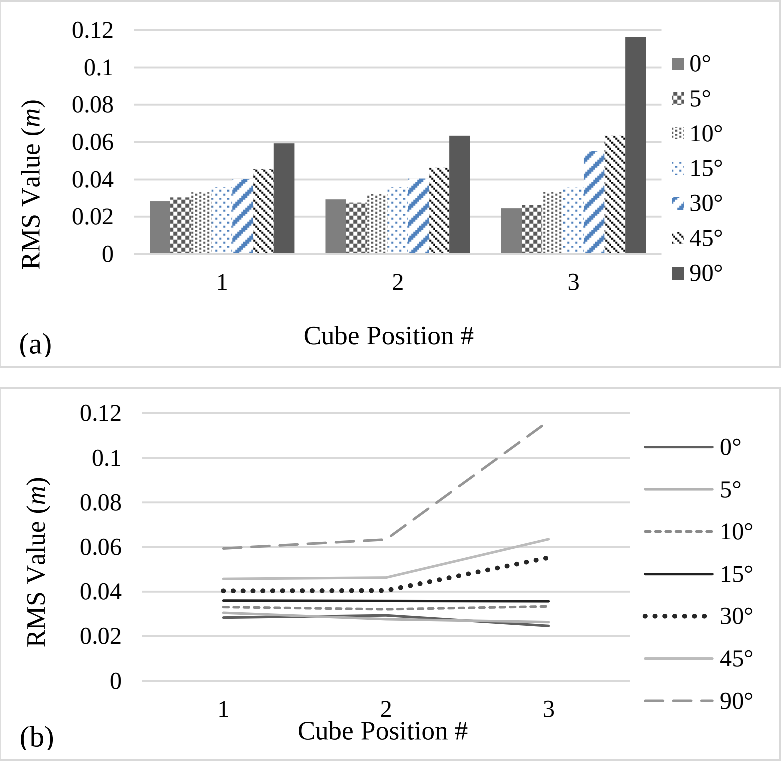


Figure 3-8: (a) RMS values at critical points and, (b) RMS values for different alterations.

In Figure 3-8-(b), it can be clearly seen that the RMS value, which represents the average error, increases as the position of the cube moves closer to the end flange for high angular displacements. This finding could lead to the conclusion that, for a defective pipe spool, due for example to human error in interpreting the design drawings, the RMS increases as the position of the investigated cube approaches the defective part. However for angular displacements less than 15°, the error at each location is relatively constant with respect to the displacement angle. This is due to the degradation of fine registration of the assembly as a whole due to the end-piece displacement. As shown in Figure 3-8, in the assemblies studied, the local RMS value is an indicative descriptor of defect detection in the studied assemblies. Automated localization, without resorting to visualization, of low angular deflections requires further study, and some ideas are presented below.

One notable advantage of the methodology proposed here is that it detects the whole assemblies autonomously and individually, in contrast with traditional deviation detection methods, in which a distance-based analysis is used for each point. An approach for subsequent precise alignment deviation analysis that will enable automated realignment planning for defective assemblies is illustrated in Figure 3-9. Assuming a virtual serial manipulator kinematic model for the spool (Cho et al. 2004; Craig 1989), discrepancy and deviation analysis at this point can be considered as determination of required transformation parameters, rotation and translation, to assign to different sections of the previously registered point clouds corresponding to links in the virtual manipulator representing the fabricated static spool. This is going to be described extensively in Chapter 4. Automated feature and shape recognition techniques, such as the Hough Transform and local PCA, are the examples of methods for deviation and discrepancy quantification that have been shown to work for this purpose (Ahmed et al. 2014; Rabbani et al. 2007).

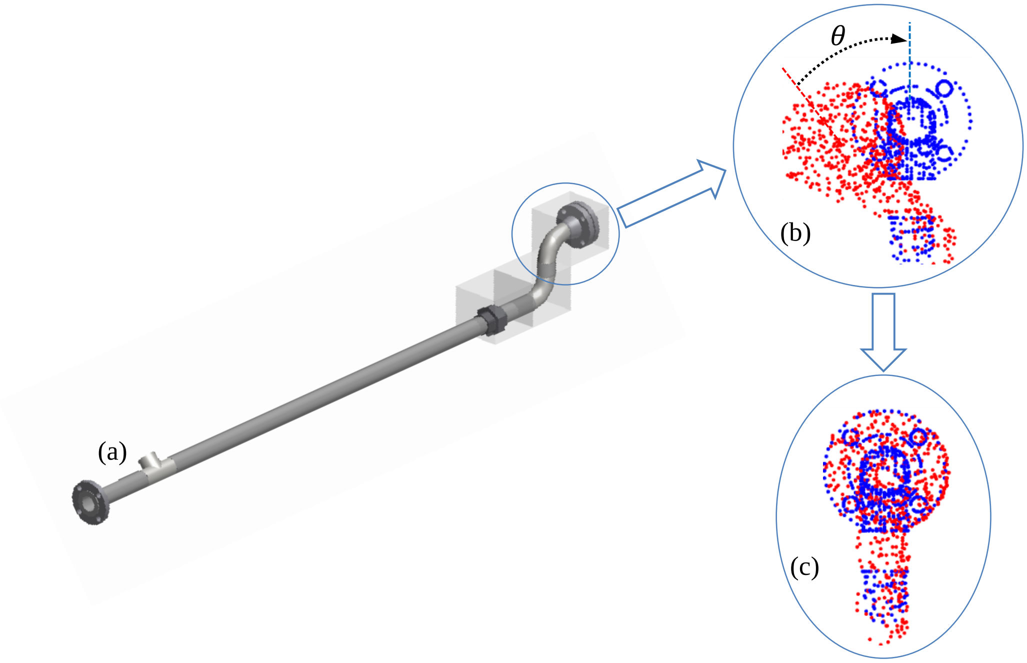


Figure 3-9: (a) Local investigation of assemblies. Containing point clouds in the remarked cube, viewed from the bottom of the end piece, before (b), and after (c) determining the incurred deviation (Red: as-built status acquired by laser scanning, blue: as-designed state converted to the point cloud format). θ: calculated angular deviation.

To investigate the relationship between the angle of rotation and the RMS value for each position, a regression analysis was conducted. For this analysis, $\theta = 30°$ was eliminated for the regression analysis, and was used for the validation of the regression analysis. However, the correlation coefficient shown in Table 3-4 implies that the regression analysis was performed reliably. Investigating a rotation angle that has been physically measured during the experiments is a more practical means of determining the level of accuracy provided by the developed approach. The analysis takes a linear regression ($\theta = a_0 + a_1 RMS$). Using the estimating function calculated by the regression analysis and the measured RMS for $\theta = 30°$ extracted from the local registration algorithm results, the angle of rotation for each cube position was estimated and compared to the real value, as shown in Table 3-4. The results confirm that the proposed method can be used for automated defect detection and preliminary characterization with an adequate degree of accuracy (i.e. discrepancy analysis).

Table 3-4: The estimated value for $\theta = 30°$ using regression analysis results and the accuracy of the approach

Cube position	Angle (deg)	a_0	a_1	R^2	RMS Value (m)	Measured Angle (deg)	Error (%)
1	30	-86.072	2926.4	0.9925	0.04029	31.83	6.09
2	30	-70.839	2515.7	0.9883	0.04042	30.85	2.83
3	30	-20.964	969.71	0.9941	0.05523	32.60	8.66

It should be noted that the size of the cube is unchanged during the experiment; however it can be defined by users, as mentioned earlier. Because RMS represents the average error for the contained points in the cube, the change in the size of the cube will affect the RMS value. For example, in case where the cube is moved along a defective part of the assembly, the more it covers the defects and deviations, the larger RMS value. On the other hand, if making the cube size larger results in covering a well fabricated part, the RMS value may either become smaller or stay unchanged (slightly changed), depending on the size of the cube and the size of assemblies. Its orientation is not influential. The general intent behind sizing the cube is to be able to localize discrepancies, so in most cases its size will be a matter of judgment related for example to the diameter of the pipes which form the spool.

3.5 Summary of findings on preprocessing and basic compliance checking

An automated approach has been presented for defect detection and characterization with respect to pipe spools. An ICP-based approach, preceded by PCA for coarse registration, is employed for the automated registration step. The coarse registration initiated in the algorithm proposed here not only guarantees the determination of the global minimum but also accelerates the registration process (Figure 3-4). The accuracy of the registration method proposed here strongly relies on the accuracy of the acquired point cloud. Scanning resolution and accuracy of the point cloud play an important role in acquiring a reliable point cloud to be used in the registration. Once reliable data are acquired, the proposed registration method ensures the best fit with as-designed data. Inaccurate and unreliable scan data and registration result in larger RMS values. Automated distance-based deviation analysis has also been performed as a means of detecting, localizing, and visualizing the incurred defects present in the assemblies studied.

An experimental study was conducted in order to examine the feasibility of automatically quantifying the deviations thus detected in the defective parts of the assemblies. The algorithm for defect detection and the characterization of the errors encountered includes sub-clustering and local RMS investigation. For investigating and establishing the relationship between the RMS value and the level of deviation present in the assembly a pure rotation has been applied in discrete steps to the critical flange. The results signify a robust linear relationship between the local RMS and the angle of rotation. The experimental results show that the proposed method is promising and has the potential to be employed for characterizing deviations in a preliminary fashion that occur in steel assemblies such as pipe spools with a reasonable degree of accuracy (less than 10 % error). However, this is insufficient accuracy for input to a realignment planning model, so subsequent, automated, precise deviation analysis is also required.

One of the limitations of the presented framework in this chapter is that for symmetrical geometries, it might not operate properly during the registration phase. The registration should also be customizable so that it can indicate and fix in space previously identified perfectly fabricated parts, a factor that obviously plays a role in the registration results. These assumptions have been disregarded in the current study and will be investigated later as potential avenues for the improvement and customization of the registration process. However, significantly improved and more reliable results can certainly be expected if such assumptions are considered in the model and incorporated in new algorithms. An additional consideration is that pure angular displacement is the only deviation investigated in this research. Deviations combined with translations thus represent an additional possibility of extending the scope of this framework. This is discussed in the following chapter.

Chapter 4

Automated Spatial Discrepancy Feedback

A similar version of this chapter is published as the following peer-reviewed journal article. Only minor editorial changes are applied for being consistent with the University of Waterloo thesis format.

Nahangi, M., Yeung, J., Haas, C., Walbridge, S., West, J. (2015). “Automated assembly discrepancy feedback using 3D imaging and forward kinematics”, *Automation in Construction*, DOI: 10.1016/j.autcon.2015.04.005.

4.1 Summary

Compliance checking for gross errors is quite useful in the fabrication of piping and steel assemblies in order to minimize rework. However, assembly of steel structures, modules, and pipe spools requires cycles of finer fitting and alignment in fabrication facilities and on construction sites. To minimize this work, good discrepancy feedback for automated refitting and realignment is required. A framework for such feedback is presented in this chapter that overcomes current limitations. It commences with a constrained registration step to overcome the limitations of the current discrepancy analysis approaches including the one developed in the preceding chapter. By borrowing concepts from robotic kinematics and 3D image alignment theories, forward kinematics models are generated link by link, and thus provide the means for a local discrepancy analysis for quantifying the deviations autonomously. Experiments show that the proposed approach is suitably accurate and sufficiently fast to be employed for real-time feedback in order to systematically and automatically develop the realignment plans required for refitting and realigning assemblies, which is the key objective of the work presented in this chapter.

4.2 Problem statement

Cyclic fitting and realignment trials during assembly and plumbness of steel pipe spools and modules are critical because imperfections unavoidably occur during fabrication and transportation (Sarah Slaughter and Eraso 1997). A series of studies by Love et al. (2012), notes that 10-20 percent of the total project cost is wasted (i.e. direct cost and indirect cost) due to rework of defective components in the industrial sector (Love 2002; Love et al. 2004; Simpeh et al. 2012). The material and assembly interface mediation is difficult to control in standard practice, and therefore problematic, because: (1)

detection of discrepancies occurs predominantly during fitting trials in the construction phase and provides minimal time and opportunity to mitigate detrimental cost and schedule effects (Akinci et al. 2006a; Arditi and Gunaydin 1997), (2) current surveying and quality control approaches generate discontinuous data that are not integrated electronically with decision making tools or remotely available to project managers (Akinci et al. 2006a), and (3) the efficiency and effectiveness of the discrepancy detection and resolution strongly relies on the ability of the construction professional performing the intensively manual work (AISC 2005).

In industrial construction, erection and installation of assemblies and segments become more complicated, since geometrically sophisticated facilities require staged fabrication and sequential erection. Industrial facilities and assemblies are mostly fabricated offsite because of the more controlled environment in fabrication shops (Haas and Fagerlund 2002). The fabricated facilities are then shipped to construction sites and sequentially erected. Although an inspection procedure is performed before shipping the materials to construction sites, the shipped segments may be deformed or damaged during shipment. When an attempt is made to erect and install a defective segment on a construction site, it will cause rework, because it requires realignment and repair (Hwang et al. 2009). If the defective segment is not realigned, it will cause escalated and exceeded tolerances that will ultimately cause more severe problems.

Current approaches for inspection of facilities and investigation of the as-built status of segments to avoid causing huge costs are not sufficiently effective and accurate (Nahangi et al. 2014). Most of the approaches deal with intensively manual interaction which is inaccurate. More than 50 % of the total rework cases are caused by human errors (Akinci et al. 2006a). Other than the inaccuracy of existing manual approaches, conventional inspection frameworks do not provide continuous, electronic, and automated feedback for realigning and refitting of defective assemblies. Therefore, automated approaches have emerged as advantageous tools for quality assurance and as-built status assessment purposes (Bhatla et al. 2012; Fathi and Brilakis 2013; Park et al. 2007; Tang et al. 2010).

Although the required accuracy for the as-built status is provided using the automated approaches mentioned above, the use for these approaches is still limited as they can only detect defects that have already occurred. Current approaches are limited to the *"after-the-fact"* quality control that is associated with rework as an unfavorable effect on construction sites, as mentioned earlier. In addition to the matters discussed, existing constraints and limitations of sequential installation on construction sites, and industrial projects in particular, are overlooked in the deviation analysis approaches. For example, in the case where a pipe spool is to be installed as a part of a complex module, the constraints in the adjacent part(s) are disregarded, and the pipe spool is investigated separately by comparing the as-designed status, existing in the BIM (building information model), with the acquired as-built data.

Realignment and adjustment of defective segments is currently performed by expert practitioners, which is suboptimal and costly (Section 2.3.2). Fitting trials are performed until the tolerances are within the acceptable range based on the deviation analysis. However, encountered discrepancies need to be quantified early in order to investigate the possibility of realignment in a timely manner. In other words, current approaches for discrepancy detection are incapable of providing effective information to be used for realignment and repair. Such information is associated with the feedback loop in control systems theory (Turkan et al. 2012). A feedback loop is essential for operation, maintenance and related actions for a system to be operative and productive. If a system is well maintained, the likelihood of falling behind schedule and of rework occurrence is minimized. It also results in significant cost savings as a secondary consequence. Once accurate and reliable feedback is provided, it can be applied in different phases of construction such as automated pipefitting during spool fabrication, alignment during module fabrication, and steel erection, fitting, and alignment after transportation or other disruptive events.

In summary, 3D discrepancy feedback is required for iron workers, welders, and pipefitters to fix the misfit and discrepancies already encountered. Appropriate and efficient discrepancy feedback should identify and localize the discrepancies in a timely fashion. The feedback should be provided on a moment-to-moment, hourly, daily, weekly or on an arbitrary frequency basis to perform realignment and adjustment whenever it is necessary. When staged fabrication is used and assemblies are erected and installed in a sequence, local existing situations should also be modeled. In other words, current approaches are only capable of considering global situations and conditions on fabricated assemblies, which is insufficient for appropriate and accurate modeling. Situations and conditions on construction sites should be modeled both locally and globally in order to have an effective, reliable, and efficient feedback loop.

This chapter presents an automated framework to address the limitations discussed that current approaches have not yet overcome for quantifying the discrepancies encountered in steel assemblies. It ultimately aims to support a realignment solution in a systematic way for defective assemblies and segments with a quantified discrepancy as a feedback signal that can be electronically mediated between various interfaces involved. While this approach can be broadly used for any type of fabrication, this chapter focuses on industrial facilities, in general, and pipe spools, in particular. As explained in previous chapters, the proposed method employs 3D imaging techniques for obtaining the as-built status and robotic kinematics theory for autonomous discrepancy detection, characterization, and quantification which leads to automated realignment planning. The proposed methodology is described here, and required functions and metrics are formed. An experimental study then follows to validate the proposed model and measure the performance of the method. The results of the experiments conducted are presented with extensive discussions and potential avenues for improving the framework follow in a summary of findings and suggested future work.

4.3 Automated discrepancy analysis

The proposed framework includes four primary steps, as shown in Figure 4-1 (dashed boxes). The approach commences with a Scan-to-BIM registration using one of several available approaches discussed in Section 2.2.2, which then feeds into the investigation of the as-built status assessment. The registration is modified considering local existing situations on construction sites, as discussed earlier. It is followed by the development of a forward kinematics model for the assembly to enhance the local discrepancy analysis. Figure 4-1 illustrates the proposed framework for the quantification of discrepancies for the realignment planning. The realignment loop shown in Figure 4-1 requires the feedback signal discussed earlier and will be fully explained in Chapter 5.

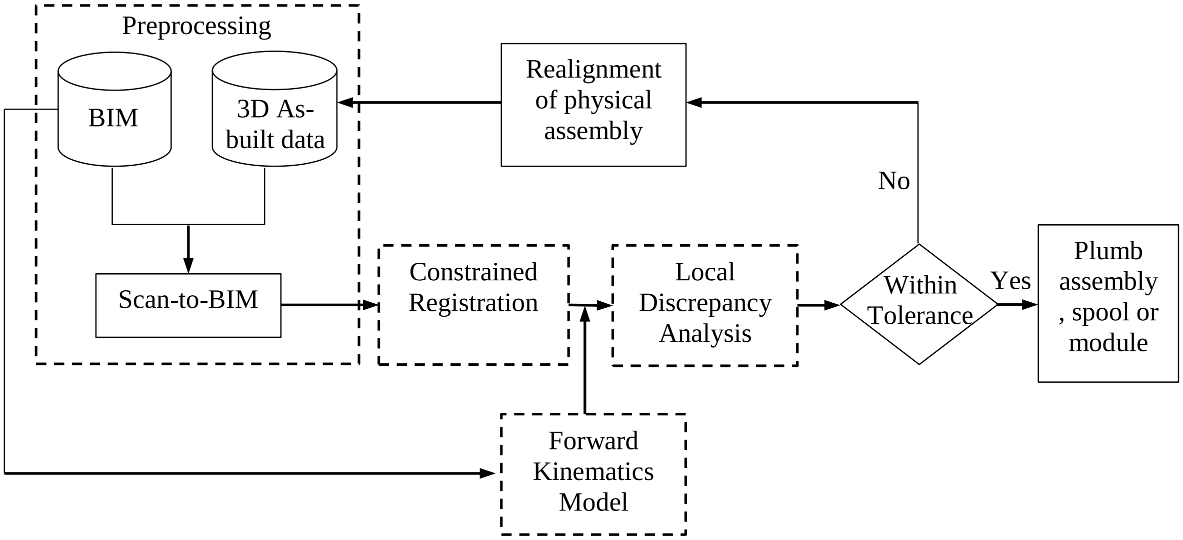


Figure 4-1: Proposed framework for autonomous discrepancy quantification and realignment planning.

4.3.1 Preprocessing

Preparing the point clouds in a proper format and performing a rough Scan-to-BIM registration is the required preprocessing for the proposed method. Various data acquisition techniques can be employed to acquire the as-built status. Point cloud registration is the solution used here to align the point cloud roughly. A review of registration techniques in addition to their performance evaluation can be found in Section 2.2.2. In the work presented in this chapter, an ICP (Iterative Closest Point)-based method (Besl and McKay 1992; Bosché 2012; Kjer and Wilm 2010) is employed for alignment of point clouds without considering the local situations. A limited number of iterations were used in order to expedite the process considering the level of accuracy required for this step. The output of the preprocessing step is the aligned as-built status (S) with BIM (M), however, local situations are yet to be considered and modeled.

4.3.2 Constrained registration

Once the point clouds are roughly aligned, existing on-site situations must be modeled to effectively measure the discrepancies. As discussed earlier, a key constraint that is inappropriately modeled is the situation of incorporating parts into existing assemblies. To overcome this limitation in the quality control process, a modified registration technique is proposed in Figure 4-2. Constrained registration is the term that is defined here for the initial registration step as it is constrained to the consideration of immediately adjacent assemblies and connecting parts. Implementation of constraints is performed using a selectable region that identifies the connecting piece to the adjacent assemblies (Appendix C), Figure 4-2-(a) and (b). This region will become the origin for assigning the local axes in the development of the forward kinematics chain (4.3.3 and Figure 4-2). Once the connecting piece is identified, a local registration yields into the calculation of the required transformation for registering the as-built data to BIM, contained in the selected region (Figure 4-2-c).

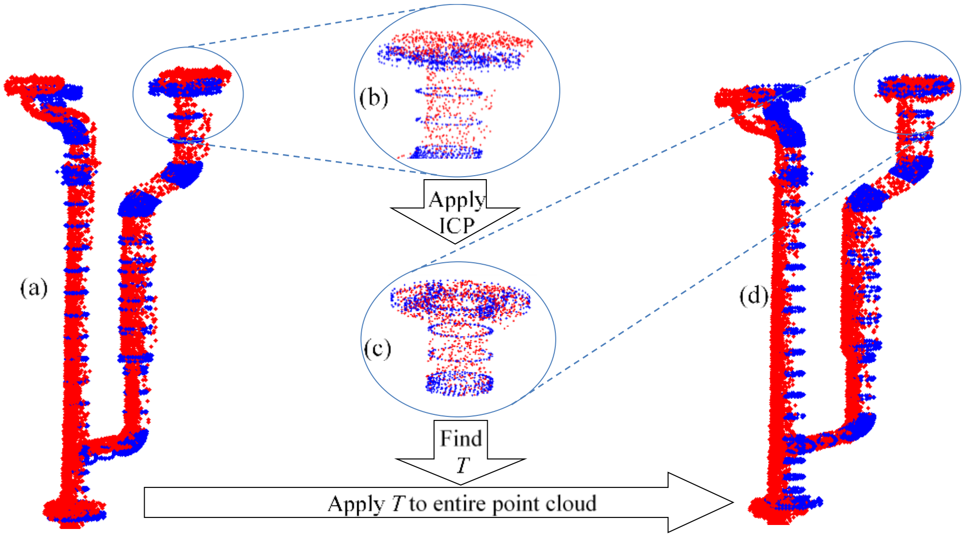
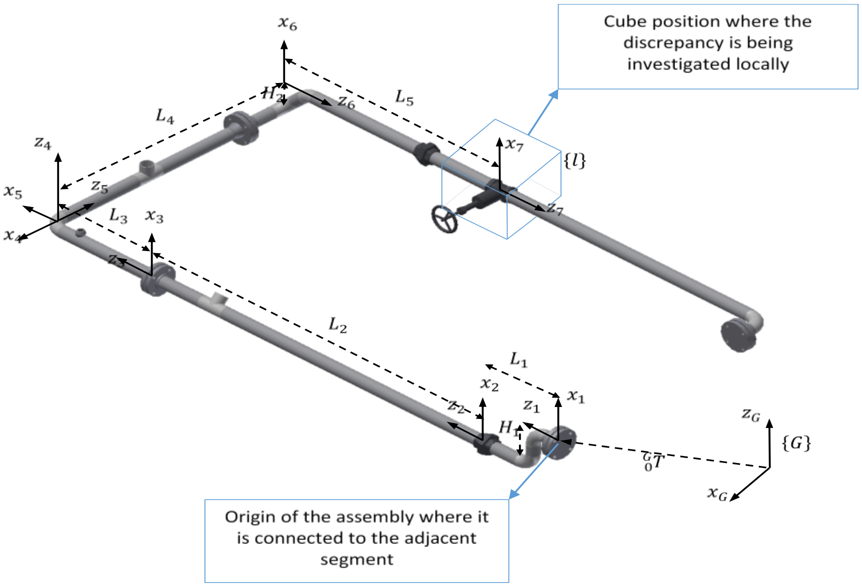


Figure 4-2: Proposed approach for constrained registration. Input: regular registration results (a); selectable region which is connected to the adjacent part (origin) (b). Apply ICP and find the transformation (T) for the selected region (c). Output: constrained registration by applying T on the entire assembly (d).

After finding the required transformation (T) for the connecting piece, Figure 4-2, the transformation is applied to the entire assembly (Appendix D). The resulting point cloud represents the actual as-built status of the fabricated assembly on the construction site as the existing local and global conditions have been taken into consideration. Having performed the constrained registration, the discrepancies can be calculated autonomously in local coordinates using a forward kinematics analogy described in the following section in detail.

4.3.3 Forward kinematics model for pipe assemblies

As discussed in the background section, this chapter employs forward kinematics to generate the relationships between different parts of an assembly. For establishing the relationships between various local axes, Denavit-Hartenberg (D-H) notation has been used (see Appendix E for more explanation). In general, D-H is an approach to systematically model the position and orientation of each part of the assembly with reference to a global coordinate system. Each part of the assembly is assigned to the corresponding local axis that is defined based on the members and joints characteristics. Figure 4-3 illustrates a typical assembly with local axes and the reference coordinate system $\{G\}$. An analogy with robot links is then applied. In a robot, *"links"* are defined as standard sequences (*"twists"*) of D-H transformations from one degree-of-freedom (rotational or prismatic joint) to the next of a manipulator in serial and parallel combinations. Here, *"links"* are defined pragmatically from one flange, weld, or joint (for frames) location to the next, where those locations are feasible for effecting subsequent realignment actions such as loosening and reconnecting bolts, or even re-welding. Where the analogy with robotics diverges, is that actions such as re-welding can result in any combination of rotation and translation and thus any transformation. However for an inverse kinematics solution for such repairs, feasible degrees of freedom must be pre-assigned using experience and judgment. Elastic and plastic imposed deformations (e.g. heat straightening) are actions to be considered in future research.



Cube position where the discrepancy is being investigated locally

Origin of the assembly where it is connected to the adjacent segment

Figure 4-3: Hypothetical illustration of forward kinematics model for a typical assembly

In Figure 4-3, a typical assembly is shown. $\{G\}$ refers to the global coordinates of the assembly. ${}^{G}_{0}T$ is the required transformation from the global coordinates into the origin of the assembly to assign local axes based on D-H notation. Each part of the assembly is then investigated and the discrepancy is measured autonomously. The following transformation is thus required for a given local position and orientation of the cube, shown typically in Figure 4-3.

$$ {}^{G}_{l}T = {}^{G}_{0}T \times {}^{0}_{l}T \qquad \text{4-1} $$

Where, ${}^{0}_{l}T$ is calculated by D-H notation based on the geometry. ${}^{G}_{l}T$ is the required link between global and local coordinates of a given assembly that is used in the next section for converting the required transformation calculated in the local coordinates. The local coordinates $\{l\}$ can be located at any arbitrary position where discrepancies are investigated. For the typical assembly shown in Figure 4-3, $\{l\}$ coordinates is located at joint 7. D-H parameters are then defined based on the assigned local axes and the kinematics chain is thus generated as shown in Equation 4-2.

$$ {}^{0}_{l}T = {}^{0}_{7}T = {}^{0}_{1}T\ {}^{1}_{2}T\ {}^{2}_{3}T\ {}^{3}_{4}T\ {}^{4}_{5}T\ {}^{5}_{6}T\ {}^{6}_{7}T \qquad \text{4-2} $$

For more detail on D-H parameters and required calculations see (Craig 1989) and Appendix E. The resulting matrix ${}^{0}_{l}T$ is then back substituted in Eq. 1 for the calculation of ${}^{G}_{l}T$.

It should be noted that understanding the link transformation from global to local coordinates in real complex assemblies is a challenging task. Employing the method presented here makes it simpler to investigate real, complex, industrial assemblies autonomously and automatically. It also provides a mathematically formulated framework that can be used for further actions for automating the tasks involved.

4.3.4 Local discrepancy analysis

Once the model and the as-built status are registered considering the onsite situations, and the forward kinematics model is developed for the assembly, it becomes possible to investigate the accuracy of fabrication and quantify the discrepancies to facilitate automated refitting and realignment. The location of a 3D cube is defined by users where discrepancy is desired to be quantified. The cube encompasses a local realignment origin location as well. The cube is stopped automatically along the spool section from the origin. The points from each point cloud (M and S) contained in the defined cube are then extracted automatically. The required transformation for achieving the designed state from the as-built status is calculated performing local ICP on the region identified previously. The proposed method for quantifying the discrepancies is similar to constrained registration. The local discrepancy analysis has three primary steps: (1) local registration, (2) global-local transformation, and (3) discrepancy quantification. Figure 4-4 shows the algorithm developed for local discrepancy analysis.

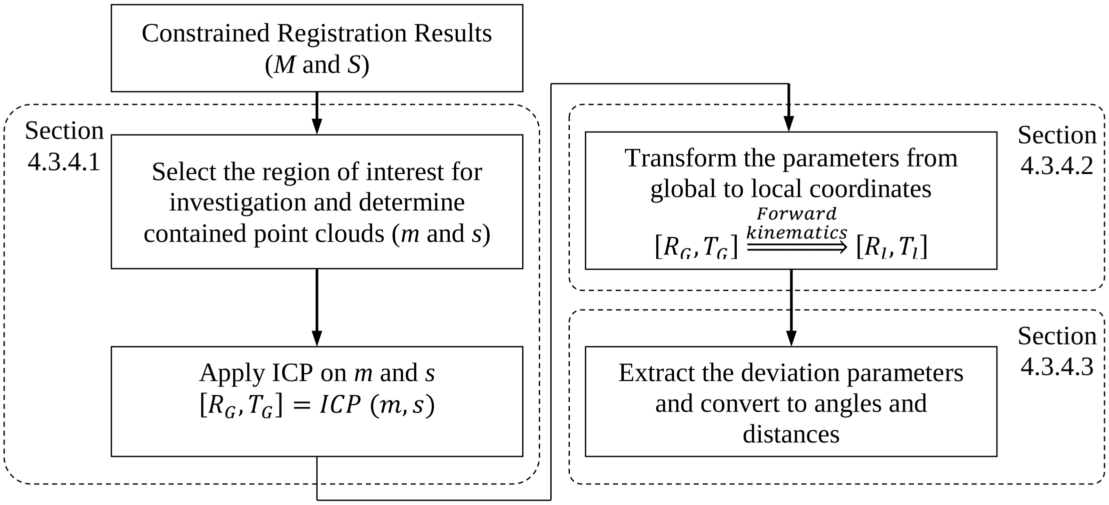

Constrained Registration Results
(M and S)
Section 4.3.4.1
Select the region of interest for investigation and determine contained point clouds (m and s)
Apply ICP on m and s
$[R_G, T_G] = ICP\ (m, s)$
Section 4.3.4.2
Transform the parameters from global to local coordinates
$[R_G, T_G] \xRightarrow[kinematics]{Forward} [R_l, T_l]$
Section 4.3.4.3
Extract the deviation parameters and convert to angles and distances

Figure 4-4: Proposed approach for local discrepancy analysis

4.3.4.1 Local registration

The input parameters are the registered point clouds, *M* and *S*, which are the outputs of the constrained registration step (Figure 4-2). Various parts of the assembly are then evaluated autonomously. For this purpose, a 3D cube, which has user-defined sizes, is selected in the assembly and the contained points from *M* and *S* are determined (Figure 4-5). The contained points from each point cloud *M* and *S* are clustered as *m* and *s* respectively. Each pair of clustered point clouds is then registered by applying an appropriate method of registration on *m* and *s*. Iterative closest point (ICP) is the approach employed for this step as well (Appendix F).

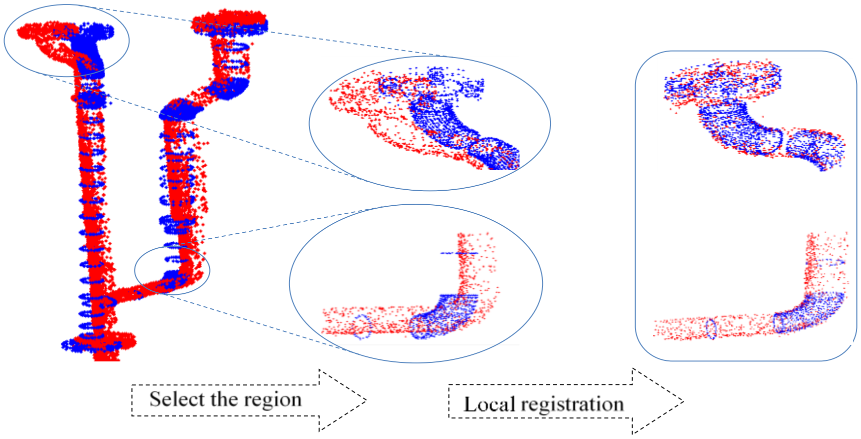


Figure 4-5: Illustration of the proposed approach for local registration (red: scanned data; blue: 3D model in point cloud format)

Once each cube and the contained points are identified, ICP is applied in order to perform the local analysis. The transformation required for local registration is extracted based on the ICP results. In summary, this step results in the calculation of required rotation and translation matrices to register the containing points in each selected cube autonomously. The transformation parameters calculated are in the form of matrices, while the discrepancies must be in the form of angles and distances. Therefore, the resulting parameters must be converted to a metric that specifies the rotation angles and translations.

4.3.4.2 Global-local transformation

While the registration is performed locally, the parameters are measured in the global coordinates. In order to measure the transformation parameters at each link's local coordinates, a transformation from global to local coordinates is required. To enhance the transformation a forward kinematics chain is used as discussed earlier. Figure 4-6 illustrates details on this discussion.

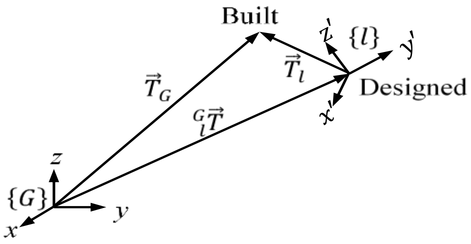


Figure 4-6: Required transformation for the calculation of local registration in the local coordinate

As illustrated in Figure 4-6, the required transformation to measure the rotation angles in the global coordinate system is calculated as follows:

$$\vec{T}_G = {}^{G}_{l}\vec{T} \times \vec{T}_l \tag{4-3}$$

in which, $\vec{T}_G$ is the transformation measured in the global coordinates $\{G\}$ (local registration results), ${}^{G}_{l}\vec{T}$ is the required transformation to transfer the global axes to local axes based on the cube position in Section 4.3.3 using the forward kinematics model developed previously in Equation 4-1. $\vec{T}_l$ is the required transformation measured in the local coordinate $\{l\}$ which is subjected to be calculated here. Pre-multiplying both sides of Equation 4-3 by ${}^{G}_{l}\vec{T}^{-1}$ results in the calculation of $\vec{T}_l$ as follows:

$$\vec{T}_l = {}^{G}_{l}\vec{T}^{-1} \times \vec{T}_G \tag{4-4}$$

4.3.4.3 Discrepancy quantification

The calculated transformation $\vec{T}_l$ consists of a rotation $\vec{R}$ and a translation $\vec{T}$ as follows:

$$\vec{T}_l = \begin{bmatrix} \vec{R} & \vec{T} \\ \vec{0} & 1 \end{bmatrix} \tag{4-5}$$

in which, the rotational part $\vec{R}$ is a 3×3 matrix with entries as follows:

$$\vec{R} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \tag{4-6}$$

On the other hand, the rotation matrix consists of three consecutive rotations around *x*, *y*, and then *z* axes (Euler angles) as follows:

$$\vec{R} = R(\phi_z) \times R(\phi_y) \times R(\phi_x) \tag{4-7}$$

in which, $R(\phi_i)$ is the rotation matrix about the *i-axis*.

After substituting the rotation matrices and calculating the generalized rotation matrix, the Euler angles (ϕ_x,ϕ_y,ϕ_z) are calculated as follows:

$$\phi_y = -\,sin^{-1}(R_{31}) \tag{4-8}$$

$$\phi_z = -\,cos^{-1}\left(\frac{R_{11}}{cos(\phi_y)}\right) \tag{4-9}$$

$$\phi_x = -\,cos^{-1}\left(\frac{R_{33}}{cos(\phi_y)}\right) \tag{4-10}$$

For more details, see Appendix G.

4.4 Experimental verification

In order to validate the proposed methodology, a set of experiment was conducted using a small scale pipe module existing in the Ralph Haas Civil Infrastructure Sensing (CIS) Laboratory. The specimen facilitates conducting the required experiment for measuring the performance of the proposed method. The data acquisition method and the experimental specimen are discussed in the following sections. A comprehensive analysis of the experiments is then provided to derive some conclusions.

4.4.1 Data collection

The approach described here made use of a laser-based 3D sensing technology to acquire continuous as-built data of assemblies as they are fabricated, transported and installed. A *FARO LS 840 HE* was used in this chapter for as-built status acquisition. The scanner used here was a phase-based device that can be reliably employed in indoor environments. More technical details of the laser scanner used can be found in Table 3-2 and (FARO 2014a; FARO 2014b). Clearly, more accurate data results in more reliable analyses; however, available resources did not permit access to a more precise device.

4.4.2 Experimental specimen

A small scale set of pipe spools (module), which can be assembled in various configurations, was employed in this experimental study. The joints and connections were designed in a way that makes it possible to arbitrarily apply various angular and translational deflections. The experimental pipe module including 3D model is depicted in Figure 4-7.

As shown in Figure 4-7-c and Figure 4-7-d, two branches are specifically investigated in this set of experiments. Given the capability of the experimental specimen (Figure 4-7-b), the spool branches can be rotated for various alterations. These arbitrary alterations are measured manually. The rotational and translational discrepancies are then measured using the proposed method. In order to perform the quantitative deviation analysis autonomously and locally, the forward kinematics model is developed for the investigated branches of the module. The results are then compared with the manually measured values (Table 4-1) in order to determine the accuracy and precision of the proposed methodology.

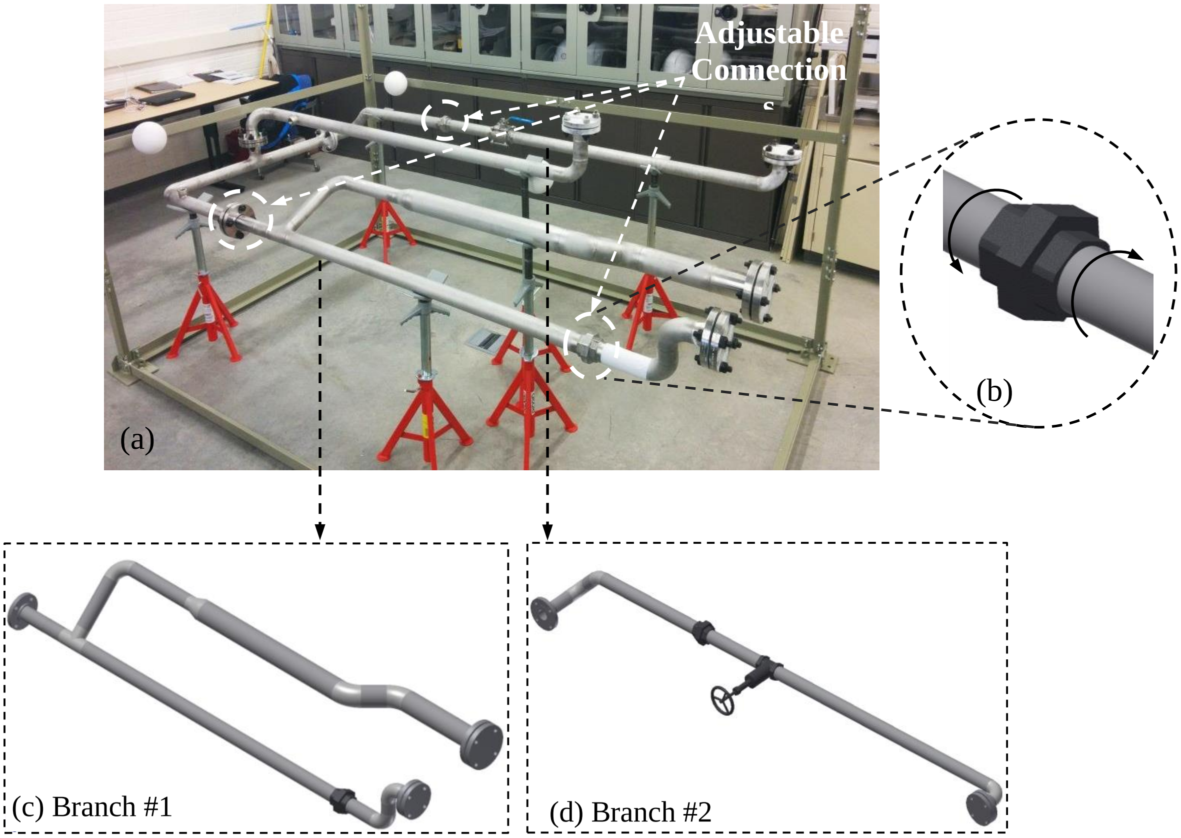


Figure 4-7: Small scale pipe module in University of Waterloo’s Infrastructure Sensing Laboratory (a) and 3D models (c, d); adjustable connections that make it possible to reconfigure the pipe spool (b).

4.4.3 Results and discussion

The results of the experiments performed are shown in Table 4-1. The required functions and metrics defined in the methodology section were programmed and implemented in MATLAB. The computing machine used has a 3.7 GHz ×12 processor unit and a 32 GB RAM. Each module branch was investigated under various deviational configurations. Rotational discrepancy was applied in three alterations of 5°, 10°, and 15°. Translational discrepancy was then examined by increasing the branch length by the two amounts of 2*cm* and 5*cm*. Thirty iterations were performed for each ICP registration where it was required for either fine or local registration. The implemented method took approximately 13 minutes for Branch #1, and 4 minutes for Branch #2. ICP point cloud registration is the predominant part (i.e. approximately 95% for both case studied spools) of the processing time. The difference in processing time for Branch #1 and Branch #2 is because of the different sizes of the point clouds used for registration.

Table 4-1: Experimental results for the spool branches under various configurations (pure rotation, pure translation, and combined deflections)

Branch #	Applied Discrepancy		Detected discrepancy				Root Mean Square (*cm*)	
	Rotational (*deg*)	Translational (*cm*)	Rotational (*deg*)	Error (*deg*)	Translational (*cm*)	Error (*cm*)	Fabricated state	Realigned assembly
1	* 5°	-	4.985	-0.015	-	-	6.89	4.24
	* 10°	-	9.763	-0.237	-	-	7.49	4.22
	* 15°	-	15.08	0.08	-	-	8.53	3.98
	-	** 2 cm	-	-	1.987	-0.013	6.27	4.17
	-	** 5 cm	-	-	5.02	0.02	9.07	4.32
	+ 5°	10 cm	4.893	-0.107	10.02	0.02	13.66	4.16
2	* 5°	-	4.976	-0.024	-	-	4.83	3.77
	* 10°	-	10.013	0.013	-	-	6.89	3.68
	* 15°	-	14.986	-0.014	-	-	7.24	3.83
	-	** 2 cm	-	-	1.983	-0.017	5.83	3.52
	-	** 5 cm	-	-	4.897	-0.103	8.97	3.67
	+ 5°	10 cm	4.978	-0.022	10.01	0.01	14.18	3.79

* Pure rotation

** Pure translation

+ Combined rotational and translational discrepancy

As seen in Table 4-1, the proposed method for quantification of the incurred discrepancies is suitably accurate for both pure and combined discrepancies. The measured error might be due part to inaccuracies resulting from the manual measurement. The results show that the proposed method has the potential to be employed for fabrication and inspection processes to avoid the associated rework on construction sites. In addition to the reported accuracy, the proposed method was found to be insensitive to point cloud registration for a wide range of investigated resolution levels. This might have been anticipated with reference to Kwon's work (Kwon et al. 2004). In order to show this insensitivity, the tests were performed with different point cloud sizes.

For resampling the point cloud with a different number of points, a random resampling algorithm was performed (Algorithm 4-1). In Algorithm 1, P^* represents the resampled point cloud for which the components are randomly selected from the original point cloud P.

Algorithm 4-1: Random resampling algorithm

```
P*=[ ]
for i=1 to an arbitrary # of points
    j= random(1 < j< # of point contained in P)
    P*(i)=P(j)
end for
end
```

The proposed approach for discrepancy quantification was then applied on the resampled point clouds to investigate the effect of point cloud resolution on the discrepancy results. In other words, a sensitivity analysis was performed with different numbers of points contained in the acquired as-built status to investigate how point cloud resolution affects the discrepancies detected. Quantified discrepancies using various resolutions are shown in Table 4-2 and Table 4-3.

Table 4-2: Measured discrepancy values for branch #1 with different point cloud resolutions

Branch #1							
Discrepancy Type	# Points					mean	standard deviation
	85978*	2000	5000	10000	50000		
Rotation	4.985° 9.763° 15.08°	4.971° 9.658° 15.12°	4.992° 9.643° 15.011°	5.001° 9.953° 14.993°	4.962° 9.882° 15.235°	4.982° 9.784° 15.09°	0.016° 0.136° 0.097°
Translation	1.987cm 5.02cm	1.964 cm 4.887 cm	1.997 cm 4.926 cm	1.958 cm 5.121 cm	1.979 cm 5.036 cm	1.975 cm 4.993 cm	0.015 cm 0.092 cm
Combined	4.893° + 10.02cm	4.887° + 10.023 cm	4.879° + 10.018 cm	4.863° + 10.029 cm	4.924° + 10.030 cm	4.892° + 10.020 cm	0.020° + 0.010 cm

* Original point cloud size used for discrepancy calculation in Table 4-1.

Table 4-3: Measured discrepancy values for branch #2 with different point cloud resolutions

Branch #2						
Deviation Type	# Points				mean	standard deviation
	46324*	2000	5000	10000		
Rotation	4.976° 10.013° 14.986°	4.938° 10.009° 14.877°	4.856° 10.043° 14.943°	4.974° 10.038° 14.968°	4.923° 10.03° 14.929°	0.049° 0.015° 0.038°
Translation	1.983cm 4.897cm	2.011 cm 5.021 cm	1.974 cm 5.014 cm	1.965 cm 4.864 cm	1.983 cm 4.966 cm	0.02 cm 0.072 cm
Combined	4.978° + 10.01cm	4.887° + 10.002 cm	4.903° + 9.93 cm	4.912° + 10.021 cm	4.92° + 9.991 cm	0.035° + 0.036 cm

* Original point cloud size used for discrepancy calculation in Table 4-1

As clearly seen in Table 4-2 and Table 4-3, scan resolution has an insignificant effect on the discrepancy values detected and quantified. Because the required processing time is an exponential function of the number of points in the point cloud (Besl and McKay 1992), this relative independency results in an expedited process that reasonably estimates the discrepancy values. For example, for branch 1, processing time for the original point cloud containing approximately 86000 points was 756 seconds; while for the resampled point cloud that contains 10000 points, it took 19 seconds on the same processing unit (Table 4-4). However, both point cloud sizes resulted in similar detected discrepancies (Table 4-2). Using sparse point clouds therefore saves computational time and cost, thus improving the applicability of the proposed framework (Table 4-4). It should be noted that the dominant part of the processing time shown in Table 4-4 is the required time for the rough registration, which is a preprocessing step. Using the proposed method for discrepancy quantification is sufficiently quick that it can be employed in real-time modeling and inspection processes if roughly aligned point clouds are previously given.

Table 4-4: Processing time for quantifying the discrepancies with different point cloud resolution

#1	number of points	85978*	2000	5000	10000	50000
	Time (sec)	756	1	5	19	253
#2	number of points	46324*	2000	5000	10000	50000
	Time (sec)	227	1	5	18	N/A

* Original size

For the purpose of realignment, the quantified discrepancy is inversely applied in the region for which the inaccuracy is already detected. In order to evaluate the feasibility of automated realignment proposed here, the root mean square (RMS) between the designed and built status is then investigated. RMS is a metric that represents the average error between the realigned assembly and the desired state, which is designed in 3D models. RMS is calculated as follows:

$$RMS = \left[\frac{1}{n}\sum_{i=1}^{n}(d_i)^2\right]^{\frac{1}{2}} \qquad \text{4-11}$$

where, *n* is the number of corresponding points and *d* is the Euclidean distance between corresponding points in *M* and *S*.

RMS value is calculated before and after applying the realignment on the fabricated assembly. Calculated RMS values are shown in the last two columns of Table 4-1. Fabricated assembly is the original fabricated state acquired by the appropriate 3D imaging technique, and realigned assembly is the state after applying the detected discrepancy to realign the assembly and achieve the designed state.

As noted in Table 4-1, RMS value for all of the investigated configurations is decreasing after the realignment application, which signifies that the average error between designed and built status is decreasing. In other words, smaller RMS values mean that the fabricated assembly is closer to the designed state and the fabrication errors detected, and quantified using this approach are being corrected. Theoretically, an RMS value of zero is ideally desired for a well fabricated assembly; however, due to the unpreventable limitations and errors that exist during the data acquisition stage, achieving a zero RMS value is extremely unlikely. It should be noted that the RMS value for the realigned configuration of each branch is a constant value, desirably converging to zero (~4.2 *cm* for branch 1 and ~3.7 *cm* for branch 2). More details and discussions regarding the RMS value can be found in Chapter 3.

4.5 Summary of automated spatial discrepancy analysis

A framework has been presented for automated defect and discrepancy quantification of industrial fabricated facilities. The proposed method employed promising 3D imaging techniques for acquiring as-built status in addition to the building information model (BIM) to enhance a comparison for detecting and localizing the potential discrepancies. Encountered discrepancies that might occur during fabrication, transportation, and installation are localized and characterized using point cloud registration methods. A constrained registration is then defined to overcome the limitations that previous registration techniques are not able to address. Existing situations on construction sites are then appropriately modeled using the constrained registration developed. A forward kinematics model is then developed for the investigated assemblies using an analogy of pipe spools with robot arms. Taking advantage of the kinematics theory enables the identification of discrepancies and the calculation of required transformations from global to local coordinates. This transformation results in autonomous investigation of the fabricated state of assemblies and the required considerations for realigning and repairing the defective segments are thus developed.

An experimental study was conducted to validate the proposed method. A reconfigurable pipe spool model was used to perform tests for various alterations. The local discrepancy analysis method for quantifying the encountered discrepancies includes: (1) local registration to find the incurred fabrication errors, (2) transformation of the discrepancies to be measured in the local coordinate axes, and (3) calculation of the incorporated discrepancy matrices. The proposed approach was implemented and programmed in MATLAB. To evaluate the model for various deviational deficiencies, pure rotation, pure translation, and combined transformations were made on the pipe spools. Some noticeable insights were observed including the following:

- The discrepancies imposed in the laboratory experiments were accurately quantified (max error is 0.237º and 0.02 *cm* for rotational and translational discrepancies respectively).
- The acquired point cloud resolution was found to have little impact on the accuracy of the discrepancy detection, meaning that the possibility exists for considerable savings in computation time by using lower resolution scans.

- The proposed methodology provides accurate, timely, and local discrepancy feedback for pipefitters in order to facilitate realignment and fitting of defective assemblies.
- The proposed methodology offers the possibility for enhanced, systematic and automated realignment planning, which is the ultimate intent of the dimensional 3D discrepancy analysis performed.

It should be noted that in real, sophisticated pipe modules, the kinematic chain is complicated. Therefore, employing the mathematically formulated kinematics chain introduced here in such a challenging and complicated environment will certainly facilitate the inspection process and improve the level of automation, which is necessary for electronic integration. However, the simple experimental case study conducted here, shows the power of the proposed method and opens up potential solutions for complicated industrial facilities being erected on construction sites. Such a framework should then result in favorable secondary effects such as improvement in productivity.

One of the limitations of the developed framework is that it requires accurate and reliable acquired point clouds as inputs. Inaccurate inputs will cause errors in the quantification process, and the results would thus be unreliable for further realignment considerations. In such cases, the encountered discrepancies detected are less reliable because there is an error associated with both the fabrication process and framework implementation. Practical solutions for minimizing the impact of the framework implementation error include automated noise filtering, point cloud surface smoothing and regeneration, and temporarily attached objects removal using appropriate methods. Handheld laser scanners are available for close range (0-2 m) applications that have accuracy in the range of ±0.03 mm, but this is an expensive approach (typical $ 100K cost). To overcome this limitation, one possible solution is to employ other identification methods such as automated 3D reconstruction employing machine vision techniques followed by automated geometrical identification methods. Such methods may include a Generalized Hough Transform performed in 3D, which is capable of identifying as-built geometries (Ahmed et al. 2014). However, such processing techniques are computationally costly. Developing an image-based framework that might be more time and cost effective is also considered as a potential avenue for the improvement of the discrepancy quantification process. An additional possibility for extending the developed model is to apply it on real-sized pipe spools and modules integrated with other project interfaces in order to investigate the effects it will have on other construction tasks.

A method has been described in this chapter for discrepancy analysis characterized and quantified in terms of forward kinematics. This provides the foundation for inverse kinematics solutions for automated realignment planning. One such solution for serial assemblies is presented in the next chapter.

Chapter 5
Calculation of Realignment Plans

A similar version of this chapter is published as the following peer-reviewed journal article. Only minor editorial changes are applied for being consistent with the University of Waterloo thesis format.

Nahangi, M., Haas, C., West, J., Walbridge, S. (2015). "Automatic realignment of defective assemblies using an inverse kinematics analogy", *ASCE Computing in Civil Engineering*, CP.1943-5487.0000477.

5.1 Summary

Alignment and plumbness of construction assemblies is a challenging and fundamental problem because it relies on manual solutions to the underlying geometric feedback control problems inherent in practices such as pipefitting and steel structures erection. Where defective components and segments are not well controlled, the errors propagate in larger components and therefore cause more severe challenges and large costs consequently. This chapter presents a framework for automatic and systematic development of realignment actions required to achieve a desired state by employing and combining two theories: (1) 3D imaging that enables the identification of the as-built status and then quantification of incurred discrepancies as a feedback signal (developed in Chapter 4) by comparing the captured as-built status with the designed state existing in the building information model (BIM); and (2) an inverse kinematics analogy that results in the calculation of required changes in the degrees of freedom defined where manipulations and changes can be applied. Experimental results show that the framework can generate the required actions for achieving a desired state systematically and with a high level of accuracy.

5.2 Problem statement

Operation and maintenance (O&M) actions are necessary for a construction management system to flow forward and therefore be operative and productive (Figure 5-1). The forward flow of actions requires a feedback signal to track the status of the construction subsystem each time it is maintained and assessed (Turkan et al. 2012). Automating the tasks involved in the management system is a key to improving productivity which is the ultimate goal of construction management systems (Akinci et al. 2006a; Akinci et al. 2006b; Tang et al. 2010; Xiong et al. 2013). One important action that necessarily needs to be addressed is the assessment of defective construction component alignment for effective and efficient corrective actions (i.e. realignment plans) and therefore refitting solutions. These actions become more important in industrial facilities because they heavily involve complicated component geometries that require sequential fabrication, subassembly, transportation, and installation (Song et al. 2005).

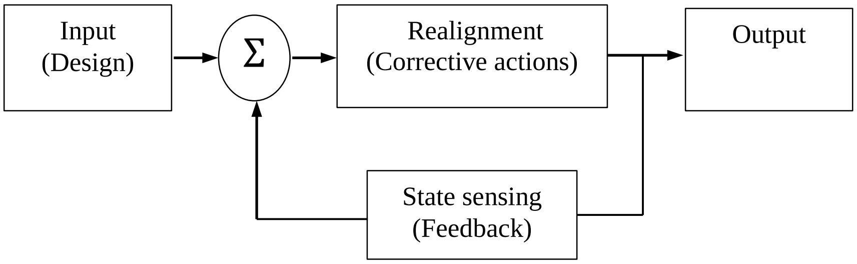


Figure 5-1: Control systems theory and the flow between various components

Although significant advances in dimensional and tolerance control have been achieved in the manufacturing and fabrication industry, there are still challenges that cause rework or slow assembly and therefore lead to delays in construction projects. Such challenges include fabrication errors that may occur because of the complicated geometry, as mentioned earlier, limited capacity of the manufacturing machines, or the deformations and movements caused by transportation. Fabrication errors and discrepancies therefore occur unavoidably during the process of assembly, transportation, and construction. Realignment, repair or replacement are solutions, whereby each solution should be timely and accordingly selected (ASME B31.3 2009). For example, *"cold fit-up"* is the most common way to solve the misalignment problem; however, it requires investigations on strain/stress to avoid exceeding the allowable values. Moreover, fit-up will mechanically affect the adjacent components, which will cause misalignment in another region. Such actions are associated with the unfavourable rework concept.

The rework associated with the unsystematic and indiscriminate repair and realignment of defective assemblies on construction sites is currently one of the most challenging and costly consequences of fabrication errors. According to (Hwang et al. 2009), rework sources and consequences should be identified as early as possible to effectively minimize the resulting costs. Currently, the required actions for realigning and refitting the defective segments and assemblies are derived based on the experience that craft workers have on a specific assembly, and therefore require experienced and expert practitioners. Moreover, the corrective actions are constrained by a list of criteria that vary from one person to another. Such repair actions may solve the installation and erection problems temporarily; however, they may cause more severe challenges that require more time and money to be solved (Milberg and Tommelein 2003; Milberg and Tommelein 2005).

With recent advances in 3D sensing technologies, acquisition of reliable as-built data has become possible. Promising technologies such as laser scanning and digital photogrammetry have provided a sufficient level of accuracy and automation for the built status acquisition and as-built modeling (Bhatla et al. 2012; Dai et al. 2013; Golparvar-Fard et al. 2011). Despite the fact that accurate acquisition tools are available for continual monitoring and inspection of fabricated assemblies, their potential is yet to be fully achieved, as their output is not effectively manipulated to enhance the realignment and repair plans. This is the result of overlooking the situations created by previously installed segments, and specifically the situations in the adjacent region. In other words, accurate as-built status may be acquired in a reasonably time-effective way; however, once they are compared to the originally designed state, the detected deviations are complex in 3D and may not be reliable or complete because of overlooking real on-site situations. Even, when reliable deviations are obtained, they are typically not systematically and effectively used to derive the realignment plans that would minimize construction alignment and refit work.

This chapter presents an automated framework for developing systematic realignment and refit plans based on the discrepancy feedback acquired by employing 3D imaging and robotics theories, which is the key paradigm shift as a key contribution of this research. The construction objects and assemblies themselves are treated as systems of objects with mutual degrees of freedom that are controllable through construction actions that are optimized and planned automatically in the framework presented in this chapter. In other words, the framework presented here aims to increase automation and improve the efficiency of assembly and construction alignment work substantially. Such an automated framework has the potential to be electronically integrated with the interfaces and components involved in order to provide craft workers and decision makers with a timely and effective solution for defective or not yet fully aligned assemblies and a plan for required corrective actions. The research methodology then follows along with the required functions and metrics to be formed and implemented. A set of experiments is described that validates the proposed framework and measure its performance. Conclusions and recommendations for future research are finally presented for identifying the potential avenues for extending this research.

5.3 Proposed methodology

The approach presented in this chapter employs 3D imaging techniques for the built status identification required in the feedback loop discussed previously (Figure 5-1). It is then combined with robotic kinematics theories for achieving the designed state given in BIM. An inverse kinematics analogy is used for automated generation of required corrective transformations. Figure 5-2 shows the proposed realignment framework and the flow between constitutive components.

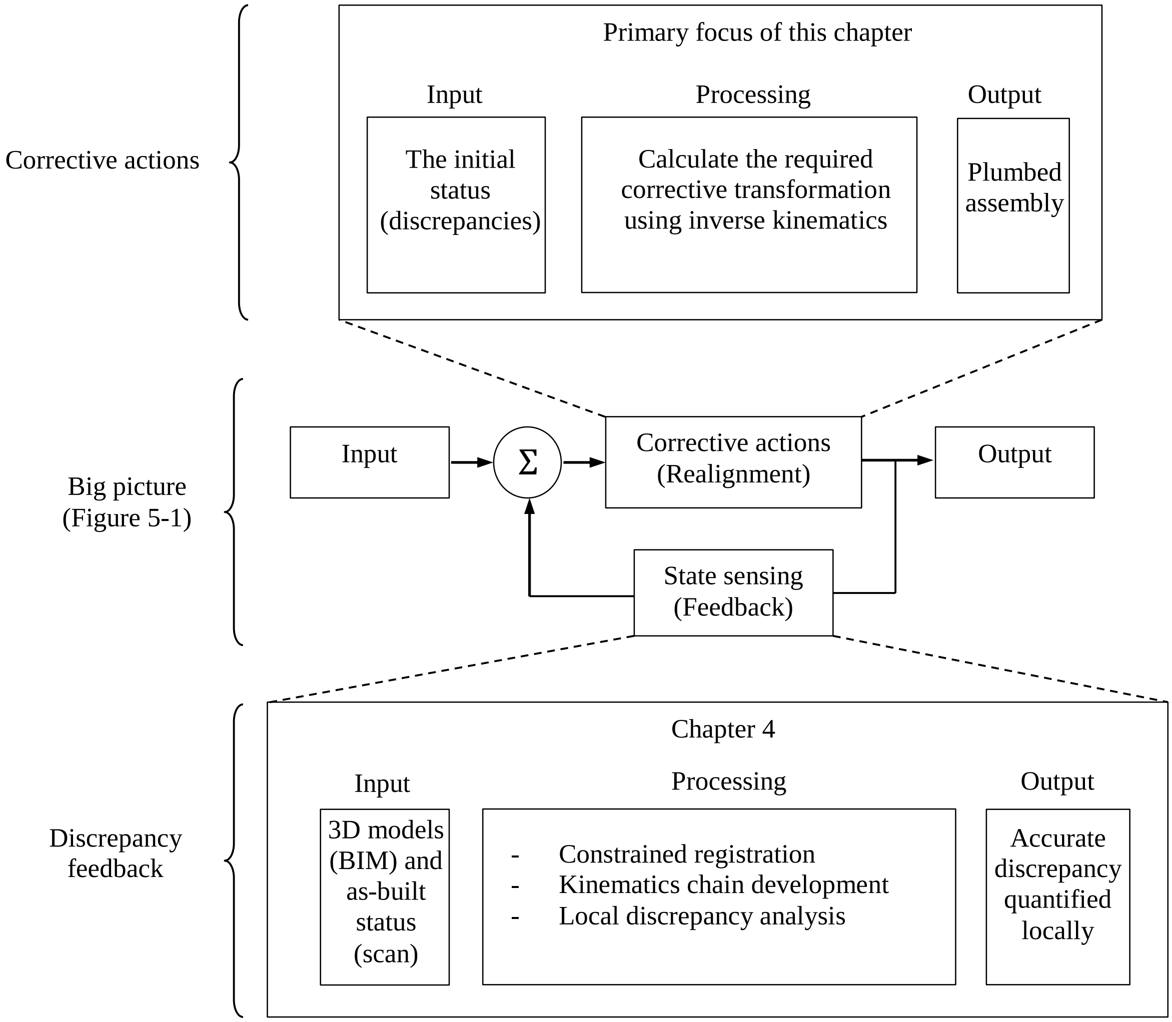

Corrective actions
Primary focus of this chapter
Input
Processing
Output
The initial status (discrepancies)
Calculate the required corrective transformation using inverse kinematics
Plumbed assembly
Big picture (Figure 5-1)
Input
Σ
Corrective actions (Realignment)
Output
State sensing (Feedback)
Discrepancy feedback
Chapter 4
Input
Processing
Output
3D models (BIM) and as-built status (scan)
- Constrained registration
- Kinematics chain development
- Local discrepancy analysis
Accurate discrepancy quantified locally

Figure 5-2: Automated realignment framework and the constitutive components within the big picture

As shown in Figure 5-2, the automated realignment framework consists of two primary steps: (1) as-built status identification, which results in the calculation of incurred discrepancies in their local coordinates, and (2) the inverse kinematics framework, which is used to achieve the designed state, given in BIM. Using the analogy of an assembly with a robot arm, defined in Chapter 3 and Chapter 4 makes it possible to calculate the joint angles and member lengths that result in a given target position of the critical region, which is being investigated to be refitted and realigned. The primary steps for the realignment plan development and required functions and metrics are detailed in the following sections.

5.3.1 As-built status identification

As extensively discussed in the introduction and summarized in Figure 5-2, necessary corrective actions require a feedback signal for identifying the current (initial) status in general. Utilizing the analogy described earlier, current 3D compliance status and incurred discrepancies are considered as the required feedback for generating the corrective actions. Such a feedback must handle the situations both locally (i.e. adjacent segments) and globally (i.e. geometry) in order to be effective, efficient, and reliable. The feedback should also be frequently achievable by users who are typically the pipefitters and welders. For calculating the discrepancy feedback, the method developed in Chapter 4 is employed. In summary, the acquired as-built status is compared with the designed state (BIM), while real on-site situations are appropriately modeled. The discrepancy feedback is then calculated in the local coordinate system.

The method for quantification of discrepancies, which fits within the big picture (i.e. control systems theory), consists of four primary steps:

1- Preprocessing: 3D models and acquired as-built status are refined and processed in order to perform a Scan-to-BIM registration. An iterative closest point (ICP)-based registration (Kim et al. 2013b; Kjer and Wilm 2010; Rusinkiewicz and Levoy 2001) is employed to find the best fit between the fabricated state and designed state (Section 4.3.1).

2- Constrained registration: overlooking the local situations in the previous step results in the conclusion that the best fit is not necessarily the correct fit. In order to overcome such limitations, a constrained registration is defined by assigning the origin of the assembly and restricting it to the constraint that it must be connected to the adjacent piece. Such modifications result in correct registration output, considering the actual procedure for installation and erection of the industrial facilities and handle the fact that the pieces are being sequentially erected. An ICP-based registration is employed in this step as well as the registration in the preprocessing step (Section 4.3.2).

3- Kinematics chain development: by using the previously discussed analogy and borrowing robotics concepts, a forward kinematics chain is developed for each assembly in order to establish the relationship between the global and local coordinates. For this purpose, joints and members are defined where realignment can feasibly be applied. For example, a revolute joint is defined where realignment can be applied in the form of twist, or a prismatic joint is considered where the members can be cut or re-welded for shortening or lengthening of members. Such a key analogy leads to the definition of degree of freedom (DOF). From the mathematical point of view, kinematics chain development results in a function whereby the variables are the degrees of freedom (DOF's) defined here and the output is the position of the end flange/member (Craig 1989). A typical output function of the kinematics chain development using Denavit-Hartenberg (D-H) notation (Denavit and Hartenberg 1955) follows:

$$P = f(\Omega) \tag{5-1}$$

where P is a vector representing the position of the end flange/member in the global coordinate system and f is the kinematics chain developed. Ω is a vector that represents the DOF's defined based on the considerations discussed earlier, i.e.:

$$\vec{\Omega} = (\theta_1, \theta_2, \dots, \theta_i, L_1, L_2, \dots, L_j) \tag{5-2}$$

in which, the θ's are the rotational (revolute joints), the L's are the translational (prismatic joints) degrees of freedom and actual states identified after discrepancy analysis, and i, j are the number of defined rotational and translational degrees of freedom respectively (Section 4.3.3).

4- Local discrepancy analysis: a sliding cube moves along the fabricated assembly and contained points from the 3D model and 3D image are locally registered. It results in the calculation of the required transformation for realigning the region contained by the sliding cube. However, the parameters are calculated and seen in the global coordinates and they must be converted to the local coordinate system. The previously generated kinematics chain is the link for converting the transformation parameters to be seen in the local axes assigned where the sliding cube is located. Local discrepancy analysis aims to identify the initial state (P^1), referring to Equations 5-1 and 5-2 (Section 4.3.4). The initial state identification is also associated with the required feedback discussed previously (Figure 5-1 and Figure 5-2). In summary, the output of this step is the discrepancy feedback required for calculating the necessary corrective actions (i.e. forward flow), discussed in the following section.

5.3.2 Inverse kinematics analogy

The discrepancy calculated in the previous step is fed as an input for calculating the required corrective actions. Once the built status is identified and the desired configuration is provided in 3D models (BIM), the subsequent challenge yields to an inverse kinematics problem whereby the designed configuration is targeted from the initial state, which is already captured by conducting 3D imaging and applying kinematics theories. Previously developed frameworks are the auxiliary toolboxes utilized in various steps of the inverse kinematics algorithm. These toolboxes are: (1) a kinematics model developed based on feasibility of physical realignment in the assembly, and (2) a 3D imaging toolbox used for measuring the discrepancies and identifying the built status. The toolboxes may be employed solely or in combination with each other to facilitate the identification and calculation of the required functions. Figure 5-3 shows the proposed inverse kinematics algorithm. A numerical approach is used to solve the inverse kinematics problem rather than employing geometrical methods, because numerical approaches provide a more general solution space that can then be extended for real, complex cases and ultra-

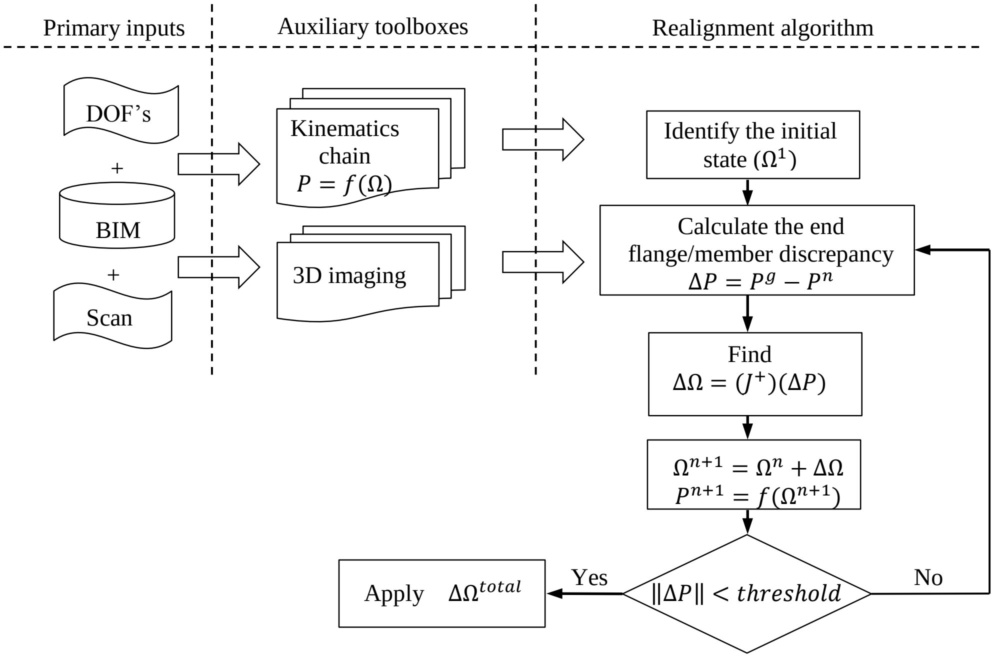

Primary inputs
Auxiliary toolboxes
Realignment algorithm
DOF's
+
BIM
+
Scan
Kinematics chain
$P = f(\Omega)$
3D imaging
Identify the initial state (Ω^1)
Calculate the end flange/member discrepancy
$\Delta P = P^g - P^n$
Find
$\Delta\Omega = (J^+)(\Delta P)$
$\Omega^{n+1} = \Omega^n + \Delta\Omega$
$P^{n+1} = f(\Omega^{n+1})$
$\|\Delta P\| < threshold$
Yes
No
Apply $\Delta\Omega^{total}$

Figure 5-3: Summary of the inverse kinematics algorithm

general models. A detailed description of the required functions, shown in Figure 5-3, is provided in the following sections.

Once the initial state (Ω^1) is captured, the end flange/member position is calculated, using Equation 5-1 as follows:

$$P^1 = f(\Omega^1) \tag{5-3}$$

Therefore, the deviation of the target configuration (P^0) from the initial state (P^1) is calculated ($\Delta P = P^0 - P^1$). Using Taylor series for investigation of the position function defined by the kinematics chain (Equation 5-1), results in the linearization of the kinematics chain as:

$$f(\Omega^{n+1}) \approx \; f(\Omega^n) + \frac{\partial f}{\partial \Omega} \times (\Omega^{n+1} - \Omega^n) \tag{5-4}$$

where, $\frac{\partial f}{\partial \Omega}$ is the Jacobian matrix (J) of the kinematics chain calculated for the initial state in a multi-variable space. n and $n+1$ represent the initial and target state at each iteration respectively. As a result, Equation 5-45-1 can be rewritten as follows:

$$f(\Omega^{n+1}) - f(\Omega^n) = \Delta P = J\;(\Omega^n) \times \Delta\Omega \tag{5-5}$$

in which, $J\;(\Omega^n)$ is the Jacobian matrix calculated for the initial state of each iteration represented by Ω^n. Pre-multiplying both sides by J^{-1} results in the calculation of $\Delta\Omega$, which is the required transformations in the defined DOF's (i.e. changes in joints' angle or length). Instead of using the regular inverse of the Jacobian matrix (J^{-1}), the pseudoinverse (J^+) is used as follows:

$$\Delta\Omega = J^+ \times \Delta P \tag{5-6}$$

The pseudoinverse is the generalized form of an inverse matrix (Ben-Israel and Greville 2003; Weisstein 2014). For an invertible matrix, the pseudoinverse equals the regular inverse; however, because the Jacobian matrix (J) is very likely to be non-invertible (i.e. non-square in most cases) in a real problem and ultra-general models, a generalized pseudoinverse is used instead (Weisstein 2014). The most common form of the generalized inverse matrix is called Moore-Penrose (pseudo)inverse (Penrose 1955). The Moore-Penrose inverse of a matrix (J) is calculated as follows:

$$J^+ = (J^T J)^{-1}\; J^T \tag{5-7}$$

where, J^T is the transpose matrix of J.

After calculating the required changes ($\Delta\Omega$), a new configuration is derived (P^{n+1}). The achieved state of the assembly and the target state are then compared, in order to check the impact of the transformations calculated and identify how close the end flange/member is reached to the desired configuration. Such an iterative procedure continues until the convergence criteria in achieving the designed configuration is satisfied. It will be shown mathematically and physically, through an example and experiments, that only a few iterations are required to satisfy a reasonably chosen threshold. The threshold must be chosen based on the allowable tolerances defined in the project specifications and standards defined in BIM. For example, in the US standard, allowable erection tolerances for piping are provided in (ASME B31.3 2009). Table 5-1 shows a summary of the allowable tolerances for the erection phase of a typical assembly.

Table 5-1: Maximum allowable tolerances in different situations of a typical assembly with diameter of *D*=10 *inch* (25 *cm*) and length of *L*=32 *feet* (10 *m*), extracted from ASME B31.3 (2009)

Description	**Maximum allowable tolerance**
Typical tolerance for erection (field installation)	- *D/2* or 15 *cm* for safety related piping (12.5 *cm*) - D or 30 *cm* for non-safety related piping (25 *cm*)
Erection tolerance for each assembly to avoid tolerance accumulation	- 2.5 *cm* for vertical plane - 7.5 *cm* for horizontal plane
Fit-up tolerances for realignment	- ~1 *cm* for strain sensitive alignment - ~1.8 *cm* for non-strain sensitive alignment

A hypothetical example is shown in Figure 5-4 in order to illustrate the detailed procedure involved with the inverse kinematics calculation and application. A 2-DOF pipe spool branch is shown in Figure 5-4-(a). The defined DOF's are one rotational degree of freedom as a revolute joint, and one translational degree of freedom as a prismatic joint.

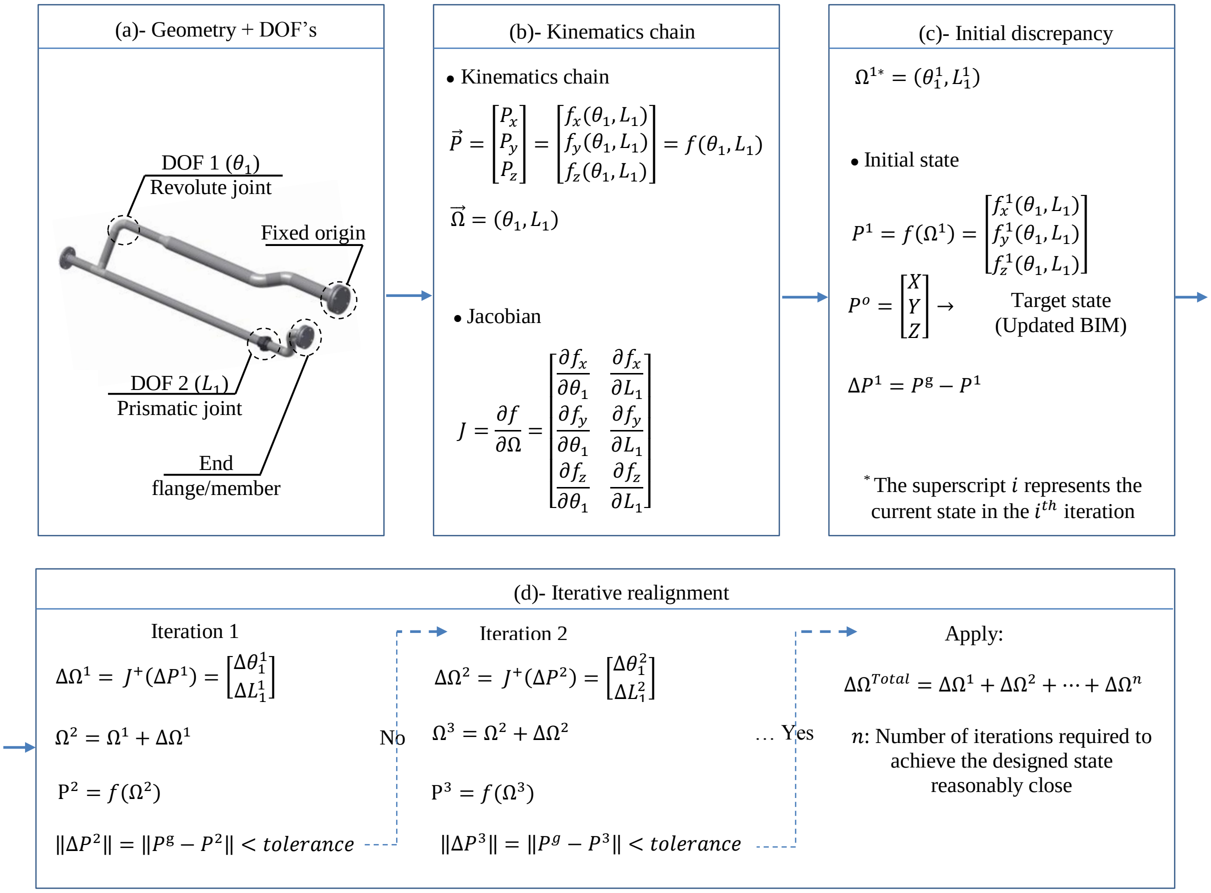

(a)- Geometry + DOF's
DOF 1 (θ_1)
Revolute joint
Fixed origin
DOF 2 (L_1)
Prismatic joint
End
flange/member
(b)- Kinematics chain
• Kinematics chain
$\vec{P} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} f_x(\theta_1, L_1) \\ f_y(\theta_1, L_1) \\ f_z(\theta_1, L_1) \end{bmatrix} = f(\theta_1, L_1)$
$\vec{\Omega} = (\theta_1, L_1)$
• Jacobian
$J = \frac{\partial f}{\partial \Omega} = \begin{bmatrix} \frac{\partial f_x}{\partial \theta_1} & \frac{\partial f_x}{\partial L_1} \\ \frac{\partial f_y}{\partial \theta_1} & \frac{\partial f_y}{\partial L_1} \\ \frac{\partial f_z}{\partial \theta_1} & \frac{\partial f_z}{\partial L_1} \end{bmatrix}$
(c)- Initial discrepancy
$\Omega^{1*} = (\theta_1^1, L_1^1)$
• Initial state
$P^1 = f(\Omega^1) = \begin{bmatrix} f_x^1(\theta_1, L_1) \\ f_y^1(\theta_1, L_1) \\ f_z^1(\theta_1, L_1) \end{bmatrix}$
$P^o = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \rightarrow$
Target state
(Updated BIM)
$\Delta P^1 = P^g - P^1$
* The superscript i represents the current state in the i^{th} iteration
(d)- Iterative realignment
Iteration 1
$\Delta\Omega^1 = J^+(\Delta P^1) = \begin{bmatrix} \Delta\theta_1^1 \\ \Delta L_1^1 \end{bmatrix}$
$\Omega^2 = \Omega^1 + \Delta\Omega^1$
$P^2 = f(\Omega^2)$
$\|\Delta P^2\| = \|P^g - P^2\| < tolerance$
No
Iteration 2
$\Delta\Omega^2 = J^+(\Delta P^2) = \begin{bmatrix} \Delta\theta_1^2 \\ \Delta L_1^2 \end{bmatrix}$
$\Omega^3 = \Omega^2 + \Delta\Omega^2$
$P^3 = f(\Omega^3)$
$\|\Delta P^3\| = \|P^g - P^3\| < tolerance$
… Yes
Apply:
$\Delta\Omega^{Total} = \Delta\Omega^1 + \Delta\Omega^2 + \cdots + \Delta\Omega^n$
n: Number of iterations required to achieve the designed state reasonably close

Figure 5-4: Hypothetical example to indicate the detailed procedure for inverse kinematics

As shown in Figure 5-4-(a), the defined DOF's and information in the 3D models (BIM) results in the calculation of the kinematics chain. The end flange position function, derived directly from the kinematics chain, is a 3×1 matrix that consists of three different functions representing the components in 3D (x, y, z) wherein each has two variables (2 DOF's). The Jacobian matrix is then calculated as shown in Figure 5-4-(b). In this particular example, J is a 3×2 matrix whereby the rows represent the three dimensions in the coordinate space and the columns represent the two DOF's. The initial discrepancy is thus detected and quantified at each degree of freedom using the previously developed functions, shown in Figure 5-4-(c). It therefore results in the identification of the initial state [Ω^1 and therefore $P^1 = f(\Omega^1)$]. Finally, the iterative realignment action is calculated in an iterative procedure, and applied using the Jacobian (J) and the locally quantified discrepancy. As shown in

Figure 5-4-(d), the resulting configuration after applying the calculated transformation is checked at the end of each iteration to determine whether more iterations are required or if the desired state has been already achieved. In the case that more iterations are required, the same procedure is repeated considering the resulting configuration as the initial state of the new iteration.

5.4 Experimental verification

While the proposed method for automated realignment can be broadly applied to various construction sectors, this work focuses on the industrial facilities as they are predominantly involved with sequential fabrication and erection and suitably fit within the scope and objective of this research. For validating the theories developed and the analogy used, an experimental study was designed. The experiments were performed in Infrastructure Analysis and Sensing Laboratory at the University of Waterloo. The laboratory has a reconfigurable set of pipe spools (Figure 5-5) that makes it possible to test various configurations in a more controlled environment for accurate performance measurement of the method developed.

5.4.1 Data acquisition

The approach described uses laser scanning for capturing the as-built status of assemblies tested in the experimental study. A *FARO LS 840 HE* is employed for generating the 3D image of the built status for the automated realignment framework. More specifications and detail about the laser scanner used in this set of experiments can be found in Table 3-2 and (FARO 2014a; FARO 2014b).

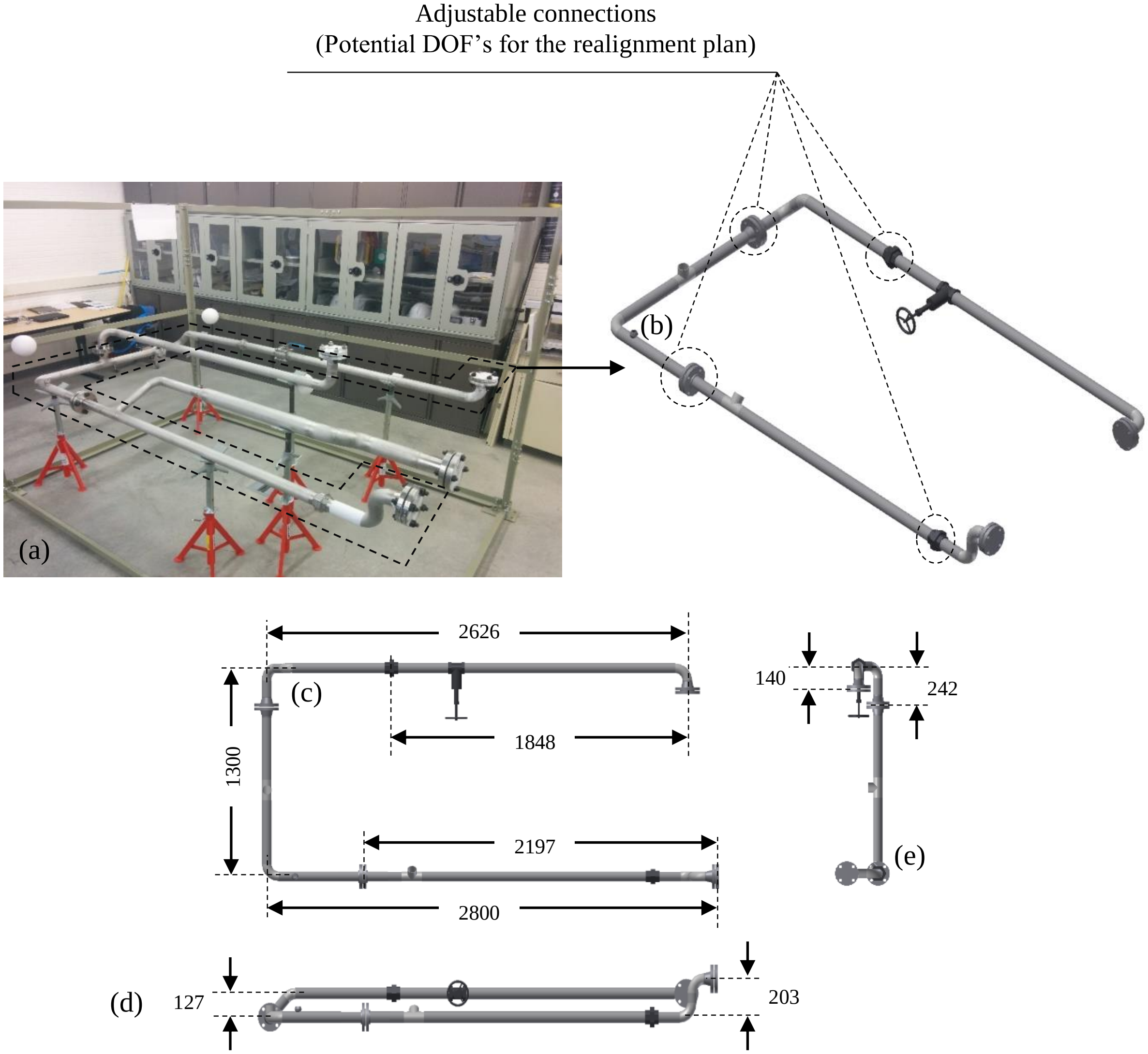


Figure 5-5: The experimental specimen in Infrastructure Analysis and Sensing Laboratory at University of Waterloo (a), and isometric view of the tested configuration 3D model (b), isolated by the dashed lines in (a). Plan view (c), side view (d), and end view (e) along with geometric information needed in the kinematics chain development (originally designed drawings) are also shown (dimensions in mm).

5.4.2 Investigated configurations

For measuring the performance of the automated realignment framework, a spool branch of the laboratory specimen, shown in Figure 5-5-(b), was isolated and used with some branches removed, resulting in a simpler model. The adjustable connections make it possible to define various DOF's at different positions and investigate the potential realignment solutions that would normally be implemented in practice by cutting, refitting, and rewelding. The branch was tested under two different configurations: (I) two revolute joints as shown in Figure 5-6-(a), and (II) one revolute joint and one prismatic joint as shown in Figure 5-6-(b). A similar discrepancy was applied at the end flange/member for both configurations tested. For Configuration I, the realignment consists of rotational potential transformations only; while for Configuration II, the realignment consists of both rotational and translational potential transformations as shown. The convention of the positive sign for the required alignments, shown in Figure 5-6, is handled by the D-H notation (Appendix E). The required geometric parameters are already provided in the 3D drawings existing in the BIM (Figure 5-5). The parameters identifying geometry are the constant variables in the kinematics chain. The inverse kinematics calculation and application results of the experiments conducted are fully described and discussed in the following section (see Appendix H for kinematics chains of Configurations I and II)

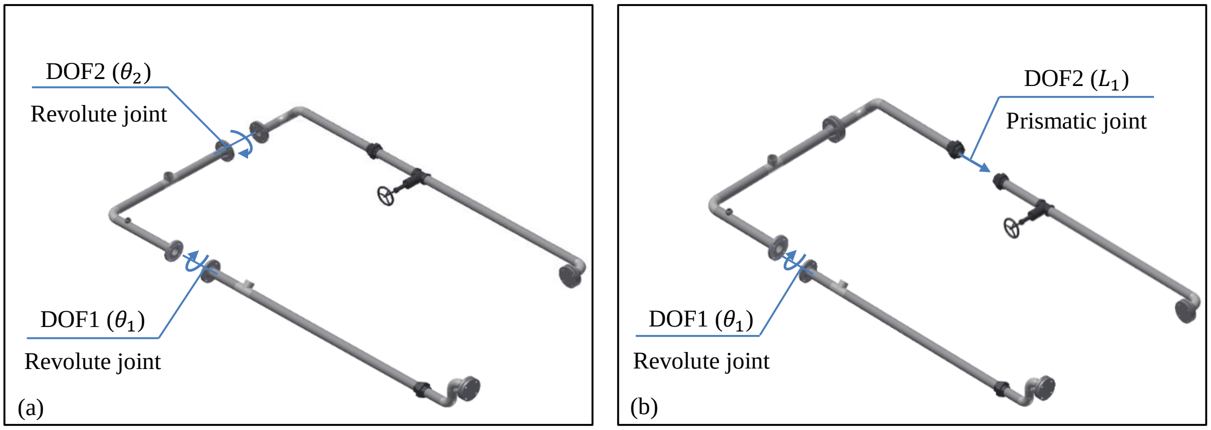


Figure 5-6: Defined degrees of freedom (DOF's) and required geometric information for Configuration I (a), and Configuration II (b).

5.5 Results and discussions

The designed experiments were performed utilizing the framework proposed, and the results are summarized in Table 5-2 and Table 5-3. The required metrics and functions defined in the methodology section were programmed in MATLAB on a computing machine that had a 3.7 GHz×12 processor unit and a 32 GB RAM. The investigated pipe spool branch was tested under the designed configurations shown in Figure 5-6.

In order to evaluate the effectiveness of each configuration and the DOF's defined, a similar target position at a similar initial state was applied for both configurations. The target position might be either the originally designed coordinates or an imposed position resulting from the already installed segments in the neighbourhood of the assembly tried to be fitted. For the designed experiments, the target position P^o was taken as (-100, 1280, 180) (mm) measured in the reference coordinate system of the assembly. Clearly, the position can be converted and measured in the local axes defined at an arbitrary point (e.g. in particular, at the previously defined DOF's) using the kinematics chain developed for each configuration.

Table 5-2: Inverse kinematics iterations and the resulting realignment solution for Configuration I

Iteration	DOF1 (θ_1) (deg)	DOF2 (θ_2) (deg)	P^1 (initial state) (mm)			P^o (target position) (mm)			$\|\|\Delta P\|\|$ (mm)	$\Delta\theta_1$ (deg)	$\Delta\theta_2$ (deg)
			x	y	z	x	y	z			
#1	8.43º	-6.58º	-115	1273	174	*-100*	*1280*	*180*	18	-2.12 º	1.38 º
#2	6.31º	-5.21º	-99.8	1280	174				6	Not required	

Table 5-3: Inverse kinematics iterations and the resulting realignment solution for Configuration II

Iteration	DOF1 (θ_1) (deg)	DOF2 (L_1) (mm)	P^1 (initial state) (mm)			P^o (target position) (mm)			$\|\|\Delta P\|\|$ (mm)	$\Delta\theta_1$ (deg)	ΔL_1 (mm)
			x	y	z	x	y	z			
#1	1.78º	0	-115	1273	174	*-100*	*1280*	*180*	24	-0.65º	6
#2	1.13º	6	-101	1298	174				18	0.00	0.00
#3	1.13º	6	-101	1298	174				18	Closest possible position is achieved	

As seen in Table 5-2 and Table 5-3, the target position is achieved after a couple of iterations and the deviation is within the allowable tolerance. $||\Delta P||$ is the deviation (Euclidean distance) between the position achieved after applying the joint angles at each iteration, and the target position imposed by BIM. $\|\Delta P\|$ is calculated as follows:

$$\|\Delta \mathrm{P}\| = \|P^0 - P^1\| = [(P^0 - P^1)^T (P^0 - P^1)]^{1/2} \tag{5-8}$$

Since only a small number of iterations is required to find the joint changes to achieve the target position, the proposed approach is computationally efficient and time effective. The implemented and programmed procedure on MATLAB took less than one second to calculate the joint changes on a processing machine described earlier. The allowable tolerance considered was 10 *mm*, based on the typical information provided in Table 5-1. It was observed that the position achieved for Configuration I, is within the allowable tolerance value considered ($\|\Delta P\| < 10\, mm$), after performing only one realignment iteration. However, for Configuration II, the closest possible position is achieved as shown (θ_1 is rotated 1.13° and L_1 is lengthened 6 *mm*). The reason that the end member/flange cannot reach closer to the target is because of the "*workspace*" defined based on the selected DOF's. In robotics, workspace analysis results in the calculation of the achievable positions of the end member/flange based on the kinematics chain model and the DOF's defined. Such an analysis is therefore essential to determine if a target position is achievable before designing the realignment options in assemblies. To some extent, the DOF's for any pipe spool may be based on the judgement of a craft worker involved such as a pipefitter or welder. The workspace analysis is a potential avenue for future research to generalize the models developed in this chapter.

Clearly, one immediate solution for the originally designed target position from which the discrepancy is detected and quantified, is the opposite application of the quantified discrepancies (i.e. $\theta_1 = -8.43°$, $\theta_2 = 6.58°$ for configuration I and $\theta_1 = -1.78°$ for configuration II). However, the target position usually varies from the originally designed situations. The change in the target position is due to the changes after the design phase, in general, and during the construction phase, in particular. Such changes may be due to the discrepancies that have occurred. This information is stored in the BIM according to the changes occurred, and is standardized how to be tracked, based on for example industry foundation class (IFC) standards. Having such information available in the BIM, enables to identify the updated target position.

For measuring the realignment impact on the defective assembly and comparing the realigned state with the initial state, both situations are scanned and registered in a global coordinate system. The resulting RMS values, scan parameters, and registration information are reported in Table 5-4.

Table 5-4: Scan parameters, registration information, and resulted RMS values for the tested configurations

Parameters	Configuration I	Configuration II
Size of the as-designed (3D drawings) in point cloud format[*]	53922	53922
Size of the as-built (scanned) point cloud[*]	414140	389566
RMS value of the initial state (fabricated assembly)	8.83 *cm*	9.24 *cm*
RMS value of the realigned assembly	3.41 *cm*	4.18 *cm*
Registration method	PCA[**]-ICP with 30 iterations	PCA-ICP with 30 iterations

[*] Number of points used for 3D image registration

[**] Principle Component Analysis for the rough alignment phase. More details can be found in Nahangi and Haas (2014), Kim et al. (2013b).

One important observation in the conducted experimental study is that the root mean square (RMS) is decreasing (see Table 5-4), which signifies that the assembly is acceptably and well assembled and aligned based on the continually updated information existing in the BIM (Nahangi and Haas 2014). The RMS is the average distance error between the corresponding points in the designed and fabricated states (two investigated point clouds). The RMS is calculated as follows:

$$RMS = \left[\frac{1}{n}\sum_{i=1}^{n}(d_i)^2\right]^{1/2} \qquad 5\text{-}9$$

where, n is the number of corresponding points and d_i is the Euclidean distance between the corresponding points in the two states, designed and fabricated. It can be mathematically proven that the discrepancy of the end member is monotonically converging to zero (Besl and McKay 1992; Kjer and Wilm 2010; Rusinkiewicz and Levoy 2001). As evidence of achieving the target position, the RMS must ideally converge to zero as well; however, a zero RMS value is not feasible because of the variance between the designed point cloud (3D drawings converted to point clouds) and the built status (scanned point cloud). A dominant portion of the residual RMS value (3.41 *cm* for configuration I and 4.18 *cm* for configuration II) is due to the attachments such as valves that are originally designed in 3D drawings, but do not really exist to be scanned (see Figure 5-5). Noise is another source of error and therefore results in larger residual RMS value. More details on this discussion can be found in Nahangi and Haas (2014). Results and the effect of the designed experiments on Configuration I are shown in Figure 5-7.

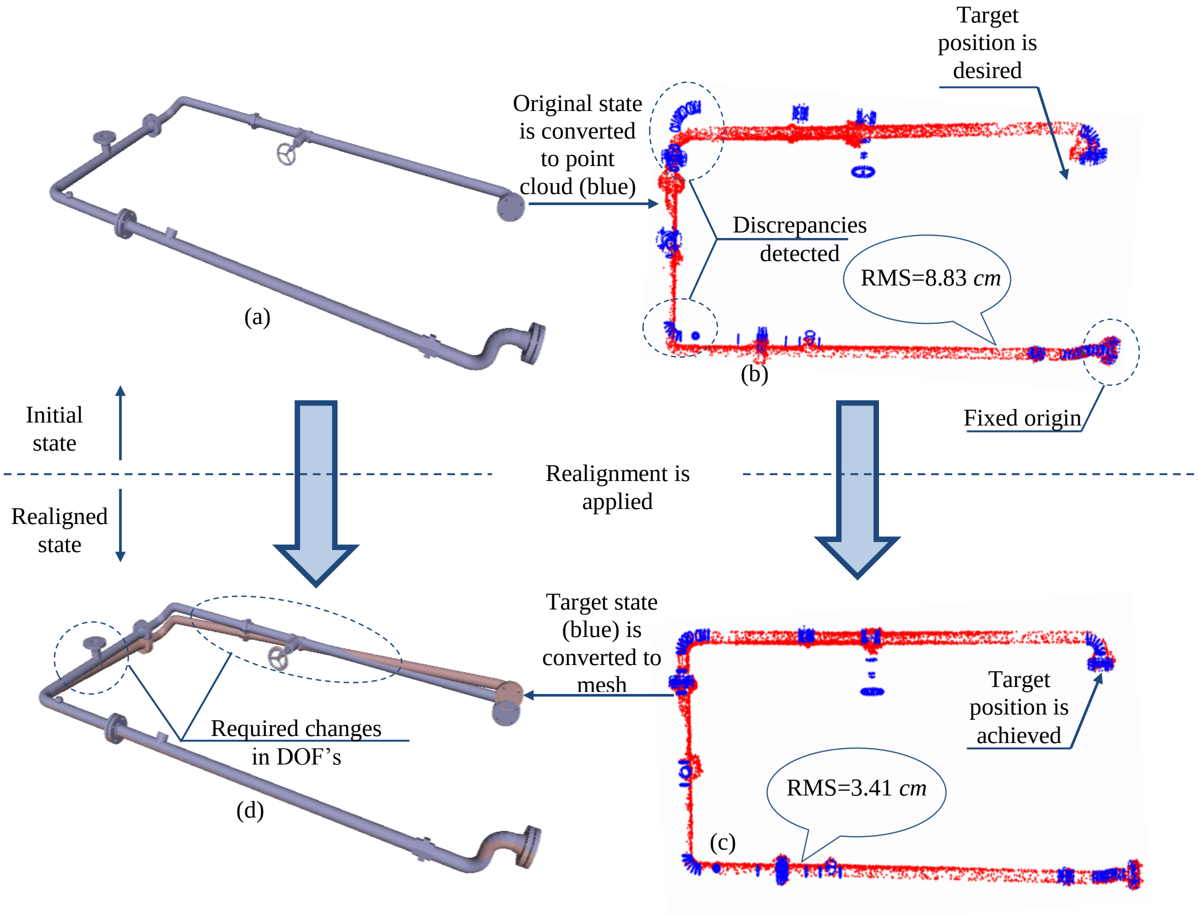


Figure 5-7: Illustration of Configuration I. Originally designed state (a); Original state is converted to point cloud format and is registered with the sensed state (laser scanned) and discrepancies are thus detected (b). Red is scanned and blue is the designed state in point cloud format; Realignment is performed and the changes are applied considering the target position desired (c); The realigned state is converted to mesh format and it is superimposed with the originally designed state in order to identify the changes required for realignment (d). The changes are applied in the DOF's defined (see Figure 5-6). For better interpretation of the results, scanned point clouds are depicted from above while the mesh formats are depicted with an isometric view.

5.6 Summary of realignment planning and realignment strategies

An approach is presented in this chapter that makes use of 3D imaging for capturing as-built status, and robotics concepts for generating potential realignment plans for defective assemblies and segments detected. The proposed approach is performed in two primary steps: (1) using the analogy of a pipe spool as a robot arm, and thus developing the kinematics chain based on previously defined and feasible degrees of freedom (Chapter 4) resulting in the identification of the discrepancies that have occurred in each DOF; and (2) feeding the construction management system with such information to calculate potential solutions for repairing and realigning the defective assemblies and segments. An inverse kinematics analogy is used for this purpose. A detailed algorithm to solve the resulted inverse kinematics problem was discussed and summarized in Figure 5-3 and Figure 5-4.

An experimental study to measure the performance of the method developed and evaluate the accuracy of the resulting realignment plan is then described. The proposed framework is found to be capable of calculating the required realignments for defective assemblies and pipe spools accurately and time effectively. Some remarks and insights from utilizing the approach described in this chapter are summarized as follows:

- The developed approach was found to be significantly time effective. Once the DOF's are defined, and therefore the kinematics chain is generated, the processing phase for finding the joint changes took less than 1 second for the two configurations tested. Although the mathematical approach used was an iterative procedure, it was noted that only a few iterations are required to reasonably reach the target position, which improves the time-effectiveness of the framework developed.
- Using the numerical approach for solving the inverse kinematics problem, which was detailed previously, enhanced the real-time modeling of general cases and complicated geometries and assemblies. In contrast with the geometrical solutions existing for solving inverse kinematics, which are less suited for complicated geometries, numerical approaches are more extendable and generalizable. By taking advantage of recent advances in powerful computing and processing units, numerical approaches can be used for complicated geometries time and cost effectively (near real-time, $t < 1sec$), and robustly as observed in the experimental study.
- A Taylor series approximation was made to investigate the linearized kinematics chain instead of the analysis of the real workspace for simplification purposes; however, it was found that

the method can reach the target position with a sufficient level of accuracy, considering the allowable tolerance values used in the construction industry (samples are shown in Table 5-1).

- The decreasing RMS value also showed that the desired target position was reasonably reached. The RMS values, reported in Table 5-4, commensurately reveal that the average error is decreasing; and the target position is therefore being achieved. The initial error substantially decreased (61% for Configuration I, and 54% for Configuration II) after applying the required changes.

As discussed, discrepancies must be detected and fixed early and before causing huge impact on project cost and schedule. In the cases where the discrepancies are not well handled, they escalate and cause more severe problems. Using the approach formed here enables the users to detect, localize, quantify, and correct an incurred discrepancy in a timely manner with minimal changes on the interacting segments. The application of such a solution is practically feasible as well. The realignment application can also be improved considering the fact that the resulting joint changes required are measured locally, which makes more sense while applying on the defective assembly. Designing and utilizing virtual connectors that represent sequences of craft worker actions such as cutting, refitting, and rewelding or the use of chain falls, come along winches, or turn buckles not only enables pipefitters to realign and refit assemblies, but also can be used for self-aligning (adjusting) applications, which requires further investigation for developing and conducting automated intelligence-based frameworks.

Complicated geometries can be mathematically modeled, and corresponding realignment plans can be automatically generated. However, the developed approach in its current form is limited to realignment calculation of serial assemblies (i.e. pipe spools or serial manipulators using the robotics analogy in general). It can also be utilized for real, sophisticated pipe modules, which are parallel manipulators and robots using the robotics analogy. Developing the realignment framework for 3D steel frames is a potential for extending this research, which requires implementation of parallel systems, development of their kinematics chain, and integration with the existing functions for solving the inverse kinematics.

Another limitation of the developed framework is that the realignment solution currently offered is not the best option considering the energy (work) consumed. The required realignment work must be optimized in order to come up with the most efficient action plan. Such an optimization problem is not trivial and requires structural investigation of the assemblies and the costs of craft workers and automated actions. Such structural investigation includes the development of physical properties such as stiffness or flexibility that should be combined with the previously defined geometric parameters (DOF’s and kinematics chain). Energy or work minimization for finding the optimum realignment solution is another potential avenue for future work.

In the next chapter, the ideas developed in this chapter are generalized and extended to parallel systems.

Chapter 6

Generalization of the Realignment Planning and Actuation Strategy

A similar version of this chapter is submitted to be peer-reviewed as the following journal article. Only minor editorial changes are applied for being consistent with the University of Waterloo thesis format.

Nahangi, M., Czerniawski, T., Haas, C., Walbridge, S., West, J. "Parallel systems and structural frames realignment planning and actuation strategy". *ASCE Journal of Computing in Civil Engineering*, Submitted in April 2015, (under review).

6.1 Summary

Parallel structural systems and assemblies are challenging to erect, align and plumb on construction sites due to their complex geometries and current heuristic realignment strategies. Examples of parallel systems include complicated pipe modules and pipe racks in the industrial construction sector. This chapter presents a generalized approach analogous to robotics and inverse kinematics for parallel systems realignment planning building developed on the series approach in Chapter 4 and Chapter 5. In addition to the calculation of a realignment strategy, feasible applications of such a strategy are also investigated in this chapter. The framework for realigning parallel systems has two primary steps: (1) as-built status identification by capturing the geometric state of construction assemblies using 3D imaging theories, and (2) realignment calculation and actuation based on degrees of freedom (DOF's) defined during the development of the kinematics chains of assemblies. The Quasi-Newton-Raphson (QNR) method is employed for solving the kinematics equation of the inverse kinematics analogy. Experimental results show that the developed algorithms are sufficiently accurate to capture any incurred geometrical discrepancies in parallel construction assemblies and proactively calculate and plan for efficient realignment strategies. Generalization of realignment calculation for parallel systems and realignment actuation are the key contributions of the work presented in this chapter.

6.2 Problem statement

Parallel systems are of significance in construction because they play a key role in most types of construction assemblies. Virtually every building's structural system is parallel. However, analyzing sub-components in isolation of the larger system may result in the identification of sub-systems as serial or parallel depending on level of segmentation or abstraction. Parallel systems such as complicated pipe spools with multiple connecting segments and modules are copious in the industrial sector, in particular. Prefabricated building modules and frames can be modeled as parallel systems as well. As far as structural stiffness is concerned, parallel systems are also important because they improve general systems reliability. For example, in order to maximize reliability, a structural system must have several parallel ways of carrying its load. Each of these defines a partial failure mode. As these parallel loops are added to the system, the degree of system redundancy goes up, and the probability of system failure goes down (Ditlevsen and Bjerager 1986).

As mentioned earlier in the Introduction (Section 1.1), automating (1) as-built modeling within the BIM framework, (2) dimensional compliance monitoring, and (3) realignment planning, would allow for early detection of component out-of-tolerance, immediate remediation of this out-of-tolerance, and proactive mitigation of rework. When considering realignment, the problem involves kinematics and therefore these methods must be able to effectively analyze two classes of components. Serial and parallel components are the two basic component types from which any larger system can be built (Patel and George 2012). Figure 6-1 shows how larger components are built using these two basic systems.

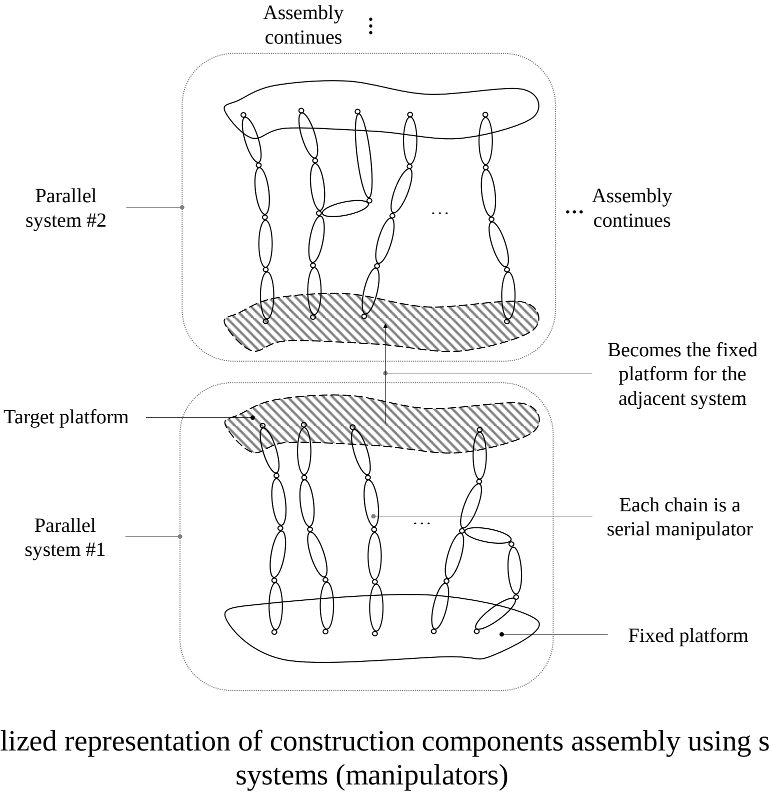


Figure 6-1: Generalized representation of construction components assembly using serial and parallel systems (manipulators)

Current methods for adjusting and iteratively realigning parallel structures and frames involve aligning, levelling and plumbing using such tools as survey instruments (see Figure 1-1), diagonal guy cables, chain falls, turnbuckles, come-alongs, wedges, jacks, pull-lifts, and proprietary pulling devices such as Tirfors (Tractel 2015). The site engineer or iron worker foreman provides instruction using the survey equipment, and the erection crew manipulates the frame to a position compliant with requirements and then attaches it up firmly (Sarah Slaughter and Eraso 1997). If the frame exhibits adverse dimensional non-compliance due to fabrication error that has gone undetected until erection, then local corrections in the form of rework, such as cutting with a welding torch are performed.

To address the inefficiencies of these approaches, this chapter presents an automated and proactive method for systematic realignment of parallel systems. Developing such a framework will avoid the unfavorable rework occurrence associated with the current approaches for realigning and fitting. The new parallel system analysis methodology is presented along with associated functions and metrics. A set of experiments is described that validates the proposed framework and measures its performance. Finally, conclusions and recommendations for future research identify the ways in which this analysis framework may lead to further progress.

6.3 Summary of the previous chapters for serial assemblies

As thoroughly discussed in Chapter 4 and Chapter 5, a framework was developed for automated realignment of serial assemblies such as a single-branched pipe spool. The realignment framework developed is summarized in Figure 6-2. To initialize the realignment procedure, accurate and reliable detection and characterization of the initial state is required. The initial state of an assembly is captured using a combination of robotics analogy and 3D imaging theories. The geometry is defined by a kinematics chain using the Denavit-Hartenberg (D-H) convention (Craig 1989; Denavit and Hartenberg 1955). The realignment algorithm was based on a Taylor-based linearization approximation of the kinematics chain using the gradient of the kinematics chain. The required changes were represented by the Jacobian of the kinematics chain in the joint-space coordinate system.

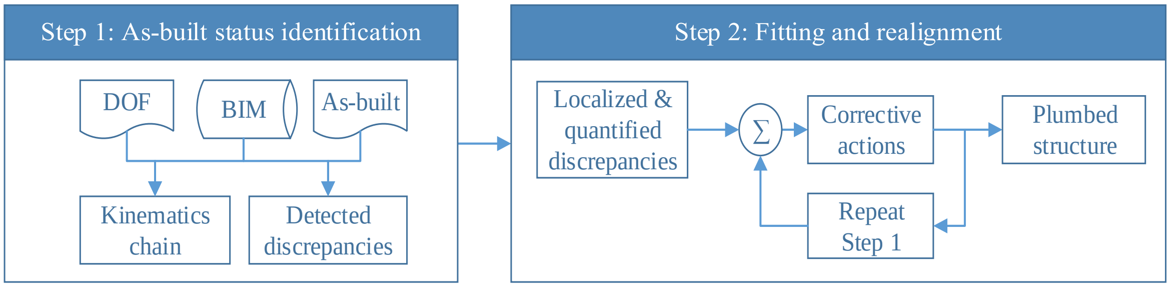

Step 1: As-built status identification
DOF
BIM
As-built
Kinematics chain
Detected discrepancies
Step 2: Fitting and realignment
Localized & quantified discrepancies
∑
Corrective actions
Plumbed structure
Repeat Step 1

Figure 6-2: General framework for realignment of defective assemblies

6.4 Research objectives for parallel systems realignment

Automatic and systematic realignment and adjustment of defective assemblies is an emerging need for rework avoidance and therefore cost saving. While the results of the realignment framework were significantly promising for “proactive” rework mitigation, the needs for modeling parallel systems and generalizing the developed framework for any types of construction assemblies remained unanswered. This chapter aims to generalize the previously developed serial systems framework for parallel systems and also introduces potential ways of realignment application on construction assemblies. Such a systematic framework enables proactive diagnosis of discrepancies, and it offers potential realignment strategies based on the existing geometric and physical status of assemblies. Required functions and metrics for generalizing the framework for parallel systems are developed. A general approach for defining the required parameters is explained.

6.5 Parallel systems realignment framework

In principle, the methodology for parallel systems realignment is similar to the previously developed framework for serial structures and pipe assemblies (Figure 6-2). The key difference between the realignment frameworks for parallel and serial systems is that the kinematics chain for parallel systems represents both inverse and forward kinematics. In general, the kinematics chain of a serial system is a special case of the general form developed for parallel systems. Similar to the serial systems, the framework for realignment of parallel systems consists of two primary steps (Figure 6-2): (1) as-built status identification feedback that is captured and fed into (2) fitting and realignment strategies for determining the required actions for aligning and adjusting defective assemblies. A detailed methodology along with the required functions and metrics are provided in the following section.

6.5.1 Step1: As-built status identification

The first step for a control system is to capture the initial geometric arrangement of that system. In construction, the current geometric arrangement is also known as the as-built status. For identifying the as-built status of the assemblies and structural frames, a combination of 3D imaging and robotics theories is used. In summary, the as built status identification includes preprocessing, registration, and local discrepancy identification.

6.5.1.1 Preprocessing

The algorithm presented in this work takes a 3D image and 3D model of an assembly as inputs, quantifies the discrepancies incurred, and characterizes the initial status. Preparation of the inputs for the primary processing engine is generally called preprocessing. In particular, this framework requires the following inputs and the corresponding preprocessing steps:

1- As-built status: The framework requires a 3D point cloud representing the as-fabricated status of the assemblies. The point cloud (3D image) is acquired using one of the many 3D imaging technologies such as laser scanning, photo/video grammetry, range imaging, etc. This framework relies on accurate input data, meaning that more robust results will be achieved by employing more accurate input data. According to recent studies (Bhatla et al. 2012; Golparvar-Fard et al. 2011; Jahanshahi et al. 2011; Liu et al. 2014), modern photo/video-based techniques for 3D reconstruction have provided the required level of accuracy for applications in construction. Laser scanning is known to be the most accurate technique, however, the relatively high cost of purchasing equipment hinders adoption by contractors and managers. The as-built status of the desired assembly should be isolated from the laser scan acquired. This step is usually performed manually, however automated methods can also be employed in order to automatically resample and isolate the desired object such as decay function (Kim et al. 2013b).

2- A building information model (BIM) integrated with 3D drawings: Computer aided design (CAD) models that are integrated with building information models are also required in order to identify the desired design state of the assemblies being investigated. 3D CAD models are usually in the form of solid objects and therefore should be converted to a point cloud (e.g. *.STL format in most cases) for performing the comparison required for capturing the initial state. In order to have a resampled point cloud that appropriately represents the 3D model, an evenly distributed point cloud should be generated. For this purposed, a Poisson disk-based resampling method is employed

(Corsini et al. 2012). Figure 6-3 shows an example of 3D-solid-objects conversion to a point cloud using a Poisson-disk based resampling approach. The resulting point cloud (Figure 6-3-c) is then used for the comparison required for discrepancy quantification.

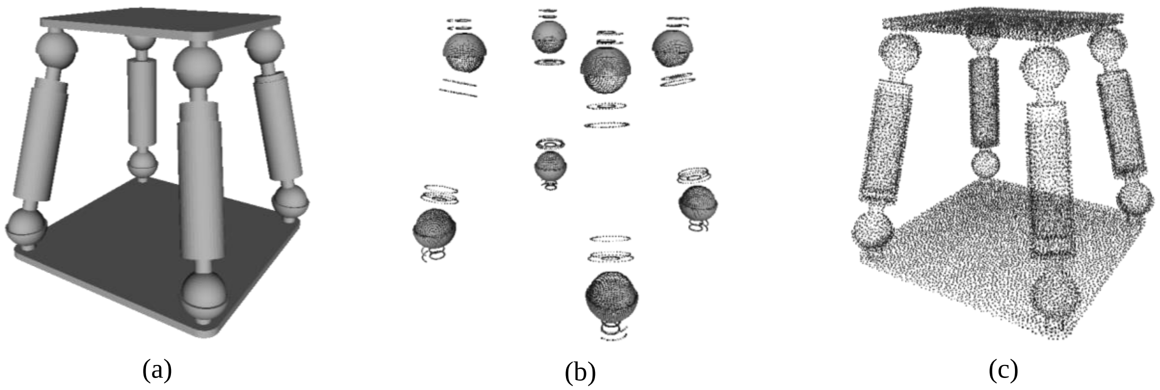


Figure 6-3: Converting solid 3D objects (a) into 3D point clouds. Solid objects are illustrated in STL format that are vertices and normal vectors (b). Vertices are resampled using a Poisson disk-based resampling for a uniformly and evenly distributed point cloud (c).

3- Degrees of freedom (DOF’s): For actuation and realignment planning, a set of degrees of freedom (DOF’s) are required. The DOF’s are defined as joints where manipulations can be applied in the form of either a twist for rotational or lengthening/shortening for translational DOF’s (joints). The potential DOF’s are identified based on the geometry, feasible realignment, and feasible applicable actions. Then the analogy of robotics is applied and the kinematics chain is developed. The development of the kinematics chain is based on Denavit-Hartenberg (D-H) convention (Denavit and Hartenberg 1955).

The outputs of the preprocessing step are fed as inputs into the following primary processing steps.

6.5.1.2 Registration

For the purpose of comparing the point clouds representing the as-built and as-designed states, a modified iterative closest point (ICP) algorithm is used. In principle, the modified ICP algorithm used here is similar to the methods explained in Section 4.3.2 (Appendix D). In contrast with the modified ICP algorithm used for serial manipulators, the registration approach used in this work considers the possibility of existing multiple origins. The summary of the modified ICP algorithm is shown in Figure 6-4.

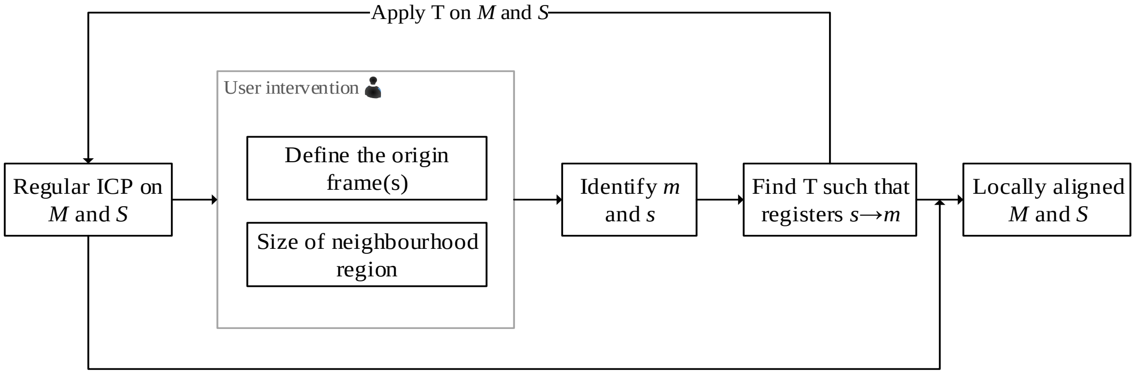

Apply T on *M* and *S*
User intervention
Regular ICP on *M* and *S*
Define the origin frame(s)
Size of neighbourhood region
Identify *m* and *s*
Find T such that registers *s*→*m*
Locally aligned *M* and *S*

Figure 6-4: Overview of the modified ICP algorithm for local registration

Typical results of the constrained registration compared with regular registration are shown in Figure 6-5. Rather than the point cloud format which is required for the described registration method, solid object format is used in Figure 6-5 for better presentation of the results.

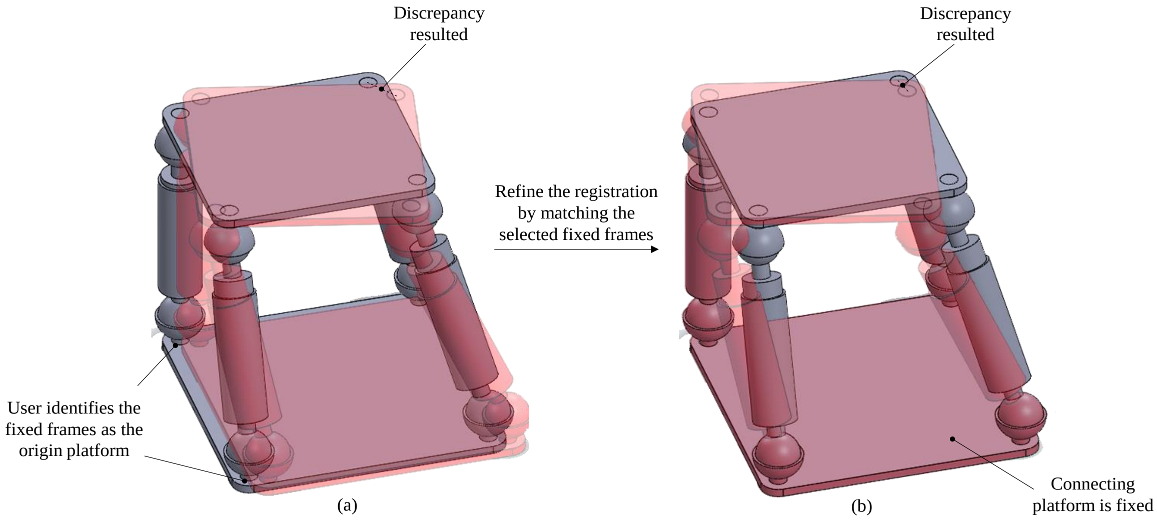

Discrepancy resulted
Discrepancy resulted
Refine the registration by matching the selected fixed frames
User identifies the fixed frames as the origin platform
(a)
(b)
Connecting platform is fixed

Figure 6-5: The impact of using constrained registration. Regular registration is initially performed (a). The origin frames (origin platform) is fixed based on the consideration that the origin platform is connecting to the adjacent module (b). Discrepancies resulting from two registration steps are compared. (Gray is the target state and red is the as-built state)

The algorithm shown in Figure 6-4 and illustrated in Figure 6-5 can be employed for two general cases:

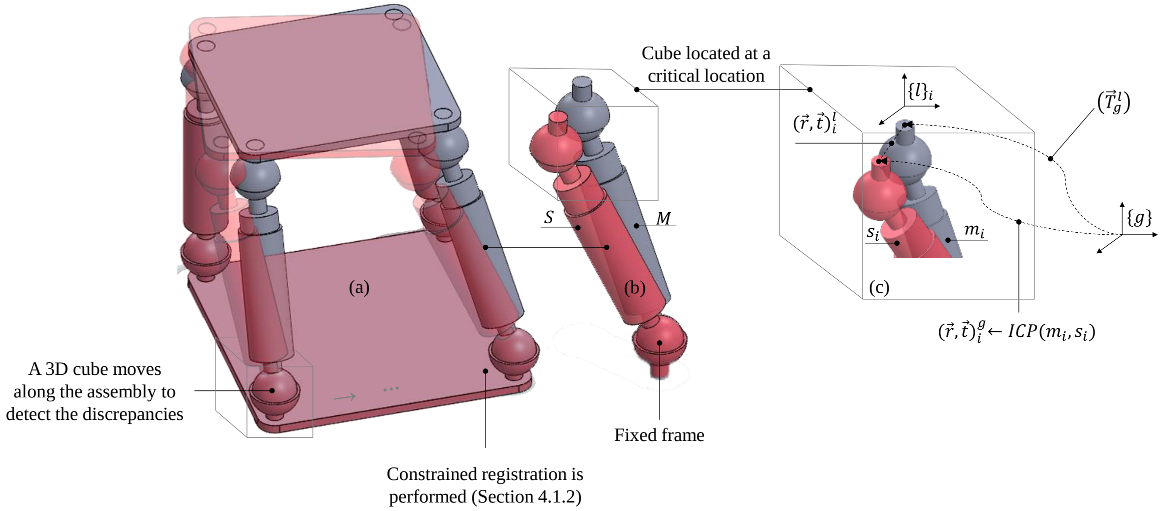
1- Single origin frame: In the most simplified case, a single origin is taken as the fixed frame for developing the kinematics chain. This fixed frame is also taken as the base frame for establishing the relationship between local and global frames using the D-H representation.
2- Multiple origin frames: In parallel frames and structural assemblies, it is often the case that there are multiple origin frames. This is also known as the origin *platform* in parallel frames. The procedure for modified ICP in the case that multiple origin frames exist is fully explained as follows:
    - Perform regular ICP on the preprocessed point clouds M and S.
    - The user identifies the base frames which become the region platform from which the assembly is connected to adjacent components for the *Model* (M). In other words, a subset $m \subseteq M$ is defined by users that indicates the connection points (platform) to the adjacent component(s). The user selects one point from each connecting region and a cube size for the neighbourhood region identification. The size of the cube should be reasonably selected based on geometric size of the assembly and the joint. Once the region representing the connecting platform is defined by the contained points in the cubes identified earlier corresponding points in the *Scene* (S) can also be detected. The resulting subset of the *Scene* is denoted by $s \subseteq S$.
    - The connecting platform design file (m) is now aligned with the as-built status (s). For aligning the point clouds, a local ICP is now applied on m and s. The output of the local ICP applied here is a rigid transformation $\vec{T}$ that matches the representing point clouds such that the error is minimized.
    - Once the transformation is calculated, the entire dataset is transformed by $\vec{T} = (\vec{r}, \vec{t})$ in order to calculate the discrepancies incurred.

$$S^* \leftarrow (\vec{r}, \vec{t}) \times S \tag{6-1}$$

The transformed *Scene* (S^*) can now be compared with the as-designed status represented by M based on the consideration of the local position and orientation (pose) of the platform connecting to the adjacent piece/module. While the local poses are overlooked using regular ICP, the modified ICP registration explained here enables the identification of the connecting regions as local poses.

6.5.1.3 Local discrepancy quantification

Once correct registration is applied considering the local positions and orientations, discrepancies can be reliably quantified. For quantifying the discrepancies incurred, a local registration is performed (Section 4.3.4 and Appendix B). The local discrepancy quantification procedure is illustrated in Figure 6-6.



Cube located at a critical location
$\{l\}_i$
$(\vec{T}_g^l)$
$(\vec{r},\vec{t})_i^l$
S
M
$\{g\}$
s_i
m_i
(a)
(b)
(c)
$(\vec{r},\vec{t})_i^g \leftarrow ICP(m_i,s_i)$
A 3D cube moves along the assembly to detect the discrepancies
Fixed frame
Constrained registration is performed (Section 4.1.2)

Figure 6-6: Local discrepancy quantification. (a) A 3D cube moves along the assembly; (b) identifies the contained points $m_i \subseteq M$ and $s_i \subseteq S$; (c) the local discrepancies are transformed to the local frames.

The procedure for calculating the local discrepancies consists of two primary steps:

1- Local registration (Section 4.3.4.1): for this purpose, a sliding cube of a reasonable size defined by users (depends on the size of the assembly and the required accuracy), moves along the frames and finds the discrepancies as shown in Figure 6-6-(a). For finding the discrepancies a local ICP is performed on the sub-clusters $m_i \subseteq M$ and $s_i \subseteq S$. The output of local ICP on each sub-cluster is stored in an array as $(\vec{r},\vec{t})_i^g = ICP(m_i,s_i)$ for quantifying the discrepancies as shown in Figure 6-6-(c) (i refers to the cube position). The superscript g refers to the global frame; meaning that the calculated rotational and translational discrepancies are being measured in the global frame Figure 6-6-(b) and (c).

2- Global-local transformation (Section 4.3.4.2): while the local ICP is performed on the sub-clusters defined by each cube position, the discrepancies are measured in the global frame. The rotational and translational discrepancies are transformed and measured in the local frames using a transformation chain. The transformation chain will then be used in the calculation of the eventual required joint manipulations as well. The link transformation between the local and global frames ($\vec{T}_g^l$) applies as follows:

$$(\vec{r},\vec{t})_i^l = \left(\vec{T}_g^l\right)^{-1} \times \ (\vec{r},\vec{t})_i^g \qquad \text{6-2}$$

where, the superscript l refers to the discrepancies measured in local frames Figure 6-6-(c).

6.5.2 Step 2: Fitting and realignment

As discussed earlier, parallel systems differ from serial systems in the number of end effectors or target points. In a serial system (e.g. an open loop pipe spool), only one end flange is desired to be realigned and adjusted; while in parallel systems, there are at least two end flanges for realignment. Given a hypothetical parallel system as shown in Figure 6-7, the geometry of a target point (TP) can be defined as a function of the DOF's using D-H representation by reaching the target points via various feasible paths.

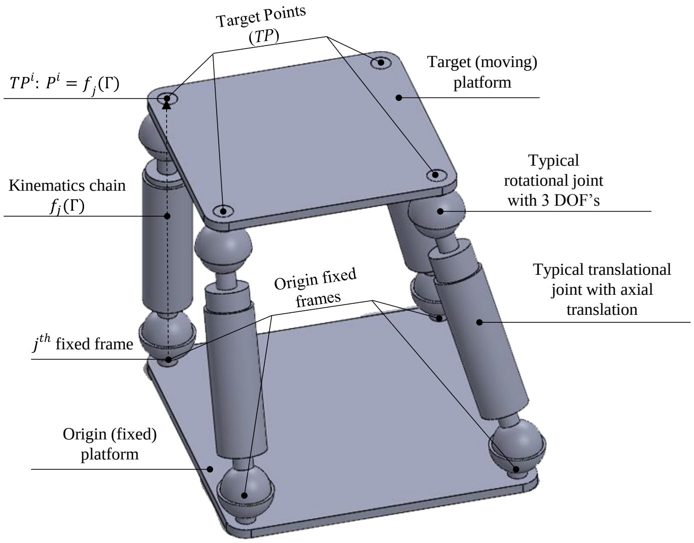


Figure 6-7: Parameters of a typical parallel system

For example (see Figure 6-7), the position (i.e. coordination components x, y, z) of the target point i (TP^i) can be defined using the j^{th} feasible path as:

$$TP^i : p^i = (x, y, z)^i = f_j(\Gamma) \tag{6-3}$$

where, Γ is the DOF vector that $\vec{\Gamma} = (\vec{\phi}, \vec{L})$, in which, $\vec{\phi} = (\theta_1, \theta_2, \cdots, \theta_r)$ is the vector representing rotational DOF's, $\vec{L} = (l_1, l_2, \cdots, l_t)$ is the vector representing translational DOF's, and r, t are total numbers of rotational and translation joints (DOF's) defined. Consolidating the equations resulting from all other feasible paths yields to the following system of equations:

$$TP^i : \begin{cases} f_1(\Gamma) = p^i \\ f_2(\Gamma) = p^i \\ \quad\vdots \\ f_J(\Gamma) = p^i \end{cases} \tag{6-4}$$

where, the subscript J refers to the number of feasible paths identified. Such a system of equations is constructed for each target point defined in the kinematics chain. A new vector $\vec{\mathbb{P}}$ is defined that represents the position of each target point as $\vec{\mathbb{P}} = (p_1, p_2, \cdots, p_I)$, in which, I is the number of target points defined. The geometric relationship of the hypothetical parallel frame can then be generalized as follows:

$$\vec{\mathbb{P}} = \mathbb{F}(\Gamma) = \begin{bmatrix} f_1(\Gamma) \\ f_2(\Gamma) \\ \vdots \\ f_{I \times J}(\Gamma) \end{bmatrix} \tag{6-5}$$

It should be noted that for a serial assembly, the general form $\vec{\mathbb{P}} = \mathbb{F}(\Gamma)$ shown in Eq. 6-5, is a 1×1 matrix (scalar), which results in a simplified special case that requires solving only one equation. This single equation defines the geometry of the assembly. As thoroughly discussed in Chapter 5, the position of an end flange in serial assemblies can be defined as a function of the DOF's defined [$p = f(\Gamma)$], which is also associated with forward kinematics chain development Chapter 4. The inverse kinematics therefore indicates the DOF's for a given end flange (target point) position [$\Gamma = f^{-1}(p)$]. The distinctive difference between serial and parallel systems is that, for parallel systems, there is a system of equations (Eq. 6-5) that defines the geometric relationship, rather than the single equation for serial assemblies. The resulting system of equations (Eq. 6-5) is also nonlinear. Such a difference in the kinematics chain development and investigation makes parallel systems mathematically and computationally more challenging than serial assemblies.

6.5.2.1 Calculation of required changes

The system of equations resulting from defining the DOF's, fixed platform, and target points, has no general and closed form solution (Tsai 1999). Numerical approaches should be employed for solving the resulting nonlinear system of equations. The general approach to solve the resulting nonlinear systems of equations is similar to the approach for serial systems developed in Chapter 5. The linearization approach makes use of Taylor series-based expansion of the kinematics chain equation $\vec{\mathbb{P}} = \mathbb{F}(\Gamma)$. A new function $\mathbb{E}$ is defined for the discrepancy function as follows:

$$\mathbb{E}(\Gamma) = \mathbb{F}(\Gamma) - \mathbb{P}_g, \tag{6-6}$$

in which, $\mathbb{P}_g$ is the desired configuration that contains the desired target positions at each target point defined. The inverse kinematics problem is then simplified to solving the system of equations $\mathbb{E}(\Gamma) = 0$. For solving this nonlinear system, the discrepancy function is approximated using linearization by the Taylor-based expansion employed. An initial start point is required to start the iterative solution. The equation is then iteratively solved for the unknown vector which is $\vec{\Gamma}$. The explained procedure is also known as the Newton-Raphson method (Broyden 1967). Figure 6-8 hypothetically illustrates the Newton-Raphson method for a simplified discrepancy function in 2D.

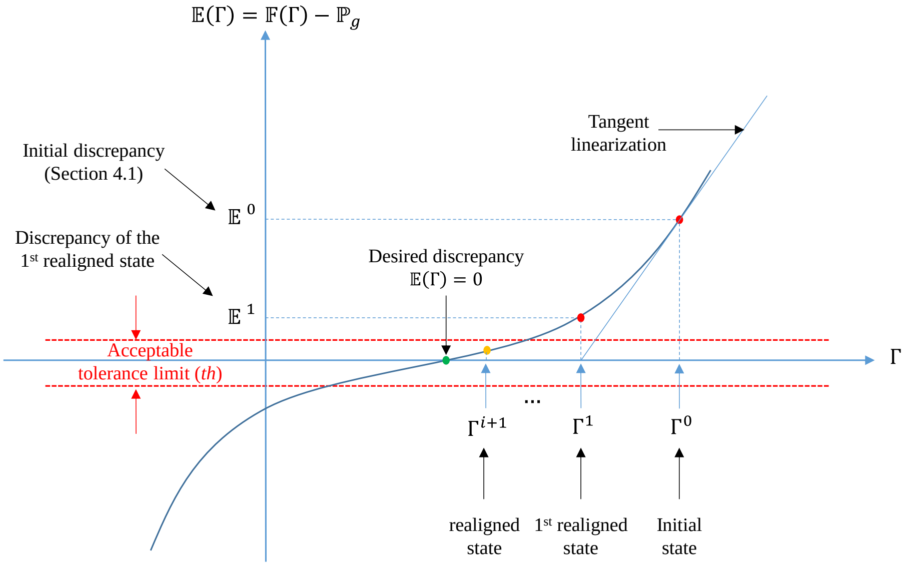

$\mathbb{E}(\Gamma) = \mathbb{F}(\Gamma) - \mathbb{P}_g$
Tangent linearization
Initial discrepancy (Section 4.1)
$\mathbb{E}^0$
Discrepancy of the 1st realigned state
$\mathbb{E}^1$
Desired discrepancy $\mathbb{E}(\Gamma) = 0$
Acceptable tolerance limit (th)
Γ
...
Γ^{i+1}
Γ^1
Γ^0
realigned state
1st realigned state
Initial state

Figure 6-8: Graphical procedure of Newton-Raphson method for a simplified hypothetical function $\mathbb{E}(\Gamma) = 0$

A form of linearization in a multi-dimensional space suitable for ultra-general cases is used in this chapter. The numerical solution for the inverse kinematics problem is summarized below. As shown in Figure 6-8, the kinematics function defining the target positions is linearized using Taylor expansion as follows:

$$\mathbb{E}(\Gamma_{i+1}) = \mathbb{E}(\Gamma_i) + \frac{\partial \mathbb{E}}{\partial \Gamma}(\Gamma_{i+1} - \Gamma_i) = 0 \tag{6-7}$$

in which, Γ_i is the initial state of the assembly in iteration i. Substituting $\partial\mathbb{E}/\partial\Gamma = \partial\mathbb{F}/\partial\Gamma$ with $\mathbb{J}$, where $\mathbb{J}$ is the Jacobian matrix, and rewriting Eq. 6-7 results in:

$$\mathbb{E} = -\mathbb{J} \times \Delta\Gamma \tag{6-8}$$

Pre-multiplying both sides of Eq. 6-8 with the pseudoinverse of the Jacobian matrix ($\mathbb{J}^+$) results in:

$$\Delta\Gamma = -\mathbb{J}^+ \times \mathbb{E} \tag{6-9}$$

Among existing forms and solution for pseudoinverse matrices, the Moore-Penrose pseudoinverse (Penrose 1955) is used here to implement the mathematical solution of the inverse kinematics problem. The Moore-Penrose pseudoinverse is defined as follows:

$$\mathbb{J}^+ = (\mathbb{J}^T\mathbb{J})^{-1}\mathbb{J}^T \tag{6-10}$$

in which, T is the transposition operator. More detail and explanation on this discussion and pseudoinverse matrices can be found in (Penrose 1955; Weisstein 2014). As mentioned earlier, the explained procedure is also known as Newton-Raphson method, in general. In the ultra-general cases where the Jacobian matrix is very likely to be non-invertible (i.e. non-square), a pseudoinverse is used rather than the regular inverse in the Newton-Raphson method. In general cases for solving the resulting nonlinear system of equations being investigated, the method is called the Quasi-Newton-Raphson (QNR) method (Broyden 1967; Dennis and Moré 1977). Other methods to solve the nonlinear system of equations $\mathbb{E}(\Gamma) = 0$, include the "*continuation*" and "*homotopy*" methods. For more detail on these methods and solving the nonlinear system of equations resulting from the inverse kinematics problem, see (Garcia and Li 1980; Li 1997; Sommese et al. 2004; Tsai 1999; Uchida and McPhee 2011; Wampler et al. 1990). Figure 6-9 shows a flowchart for solving the resulting equation for the inverse kinematics problem using the QNR method.

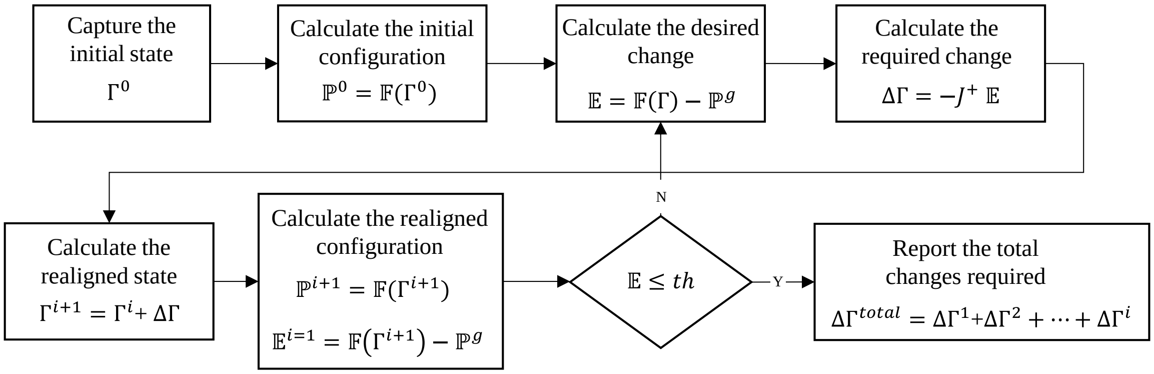

Capture the initial state
Γ^0
Calculate the initial configuration
$\mathbb{P}^0 = \mathbb{F}(\Gamma^0)$
Calculate the desired change
$\mathbb{E} = \mathbb{F}(\Gamma) - \mathbb{P}^g$
Calculate the required change
$\Delta\Gamma = -J^+ \, \mathbb{E}$
N
Calculate the realigned state
$\Gamma^{i+1} = \Gamma^i + \Delta\Gamma$
Calculate the realigned configuration
$\mathbb{P}^{i+1} = \mathbb{F}(\Gamma^{i+1})$
$\mathbb{E}^{i=1} = \mathbb{F}\left(\Gamma^{i+1}\right) - \mathbb{P}^g$
$\mathbb{E} \leq th$
Y
Report the total changes required
$\Delta\Gamma^{total} = \Delta\Gamma^1 + \Delta\Gamma^2 + \cdots + \Delta\Gamma^i$

Figure 6-9: Flowchart to solve the inverse kinematics problem $\mathbb{E}(\Gamma) = 0$ for parallel systems

As a mathematical note, one obvious solution of E. 6-6 is $(-\Gamma_0)$, when $\mathbb{P}_g$ equals the originally design state input based on the discrepancies calculated in Section 6.5.1. In other words, the required change of a system (both serial and parallel) to achieve the original design state equals the reverse of the initial discrepancies $(-\Gamma_0)$ calculated based on the comparison with the original design state. However, the desired state and thereby desired configuration are continually getting updated based on the changes in the interfaces involved. Such an unavoidable and continual change motivates the challenging concepts of "as-built modeling" and "as-built BIM" in construction, and therefore necessitates continual maintenance and monitoring of construction components. Hence, it is unlikely that the specific case explained here exists in real projects; however, the mathematical solution explained can handle that case as well.

6.5.2.2 Actuation of the required changes

After calculating the required changes as a mathematical gradient in the DOF's, realignment strategies should be planned for achieving the desired state. In practice, realignment and readjustment of structural frames and construction components is performed by applying forces, torques, and moments. Such adjustments are applied where realignment is possible to be performed. Employing tools in the form of secondary attachments such as turn buckles, chain falls, and come-alongs can facilitate the realignment application. Such realignment application is defined by the term "*actuation*" in robotics. In this section, actuation of the structural frames is generally explained and discussed.

For explaining the actuation of structural frames and systems, a hypothetical example is shown in Figure 6-10. A turn buckle is used for realignment application (actuation) in a typical construction site configuration. The required change is then projected on the local frame and resulting equivalent changes on DOF's are then calculated Figure 6-10-(c). An actuator can also be included in the kinematics model which adds to the complexity of the kinematics equation.

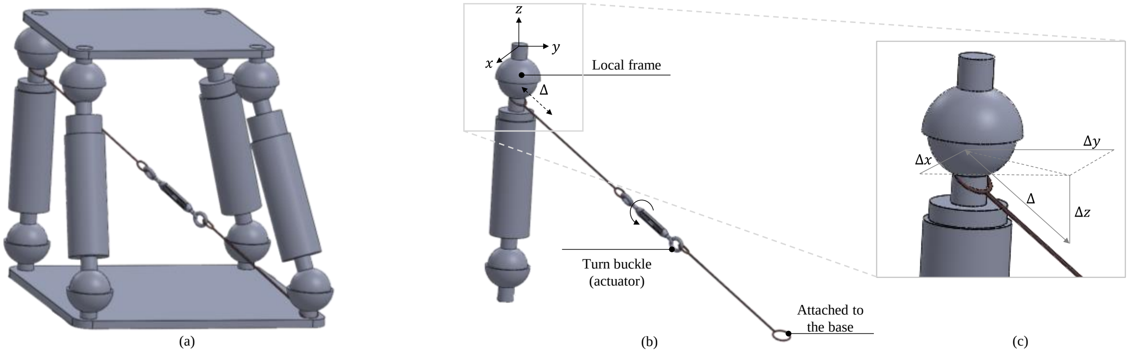


Figure 6-10: Realignment application (actuation) of a hypothetical construction assembly (a). Turn buckle is actuated based on the required changes calculated (Section 6.5.1) (b). Required changes are transformed to the local frame and the $\Delta\Gamma_{act}$ is therefore calculated.

Two key assumptions are made for mathematical modeling of realignment actuation:

1- The origin platform is plumbed; meaning that the DOF's defined for the origin platform experience no changes, and

2- The members are rigid, and deformations only take place in the form of displacement and rotations in the corresponding DOF's (joints).

Making these key assumptions results in the calculation of the changes required by the actuators and measures their impact on the DOF's defined previously based on which the kinematics chain was developed. The closest possible equivalent change possible by the actuators ($\Delta\Gamma_{act}$) is then calculated, as shown in Figure 6-10-(c). For measuring the impact of the actuators used, the actuated configuration is calculated $\mathbb{P}_{act} = \mathbb{F}(\Gamma_{act})$, in which Γ_{act} is the actuated state $\Gamma_{act} = \Gamma^0 + \Delta\Gamma_{act}$. The actuated discrepancy $\mathbb{E}(\Gamma_{act})$ is thereby calculated to measure the resulting discrepancy of the actuated assembly.

6.6 Verification and Validation

In order to verify the framework developed and assess its applicability on industrial assemblies, a set of experiments is performed in the Civil Infrastructure Sensing (CIS) Laboratory at University of Waterloo.

6.6.1 Data collection

The CIS Laboratory at the University of Waterloo has a selectable DOF frame (Figure 6-11) that enables experimental verification of the framework developed in this chapter. The selectable DOF frame is designed in a way that arbitrary configurations can be made using various possible joint options (Figure 6-11). In addition to the selectable DOF frame, a small-scale, multiple-branched pipe spool, which has been used in Chapter 4 and Chapter 5, is tested for verifying the potential applications of the developed framework on complex pipe spool branches.

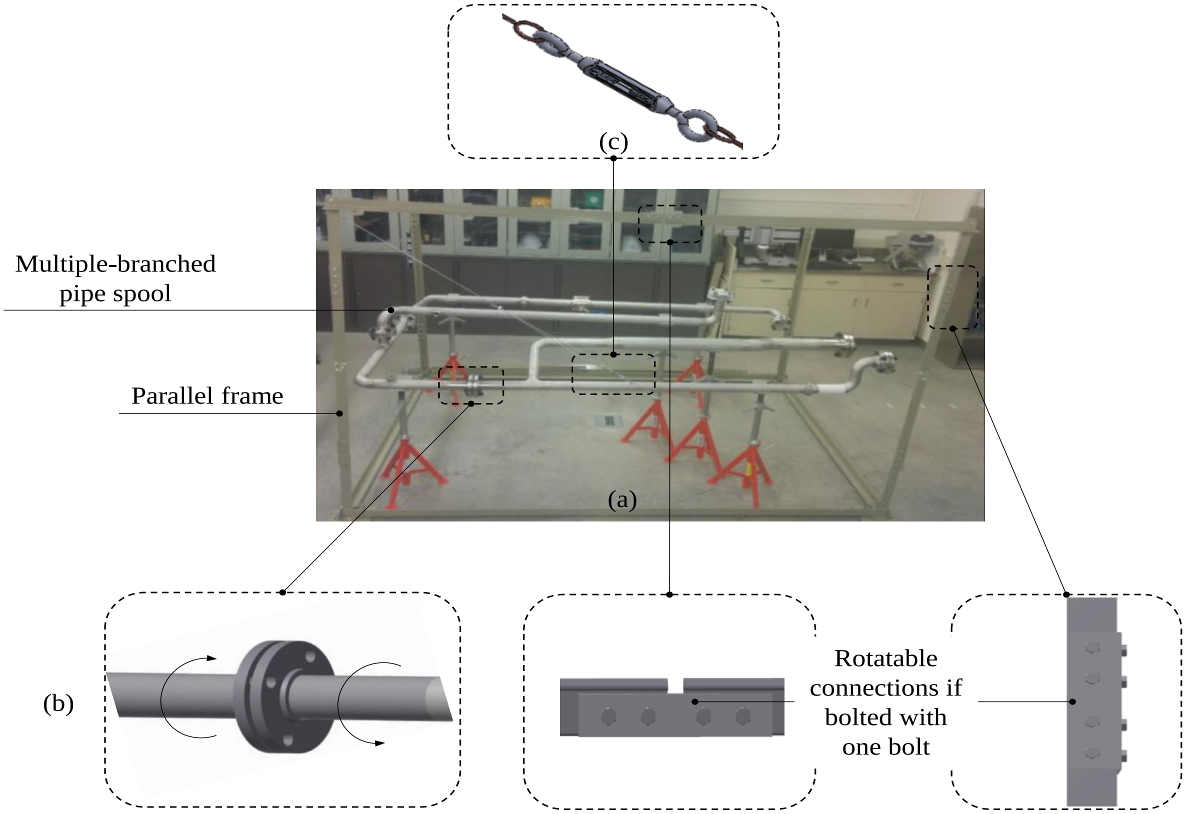

(c)
Multiple-branched pipe spool
Parallel frame
(a)
(b)
Rotatable connections if bolted with one bolt

Figure 6-11: (a) Experimental setup used for verifying and validating the framework. (b) Rotatable connections on both pipe spool and parallel frame enable multiple DOF’s for realignment calculation and application. (c) Turn-buckle used for realignment actuation.

As explained earlier, this method requires a 3D point cloud as the as-built status that can be generated using one of the promising approaches, comprehensively compared in Section 2.2.1.4. The accuracy of this method strongly relies on the accuracy of the point cloud employed for the primary processing step. A laser-based point cloud is used in this work. The required data have been collected using a phase-based *FARO LS-840 HE*. Technical details about the scanner can be found in Table 3-2.

6.6.2 Experiments and results

The previously described methodology for potential realignment calculation and actuation is implemented and programmed in MATLAB on a computing machine that has a 3.7 GHz×12 processing unit and a 32 GB RAM. Two general case studies are then evaluated to measure the performance of the framework and investigate the effective variants.

6.6.2.1 Case I: structural frame/module

The first set of experiments is performed on the selectable degree of freedom frame in the CIS Laboratory. For simplification of the math involved, a planar portion of the frame is isolated and investigated as shown in Figure 6-12. The geometry and dimensions of the configuration used for the experiments is shown in Figure 6-12. As seen in Figure 6-12, the origin platform is defined by four points ($J = 2$) as the fixed frames. The target platform is also defined by two end flanges ($I = 2$) as the target points. There are 4 equations (i.e. kinematics chains) representing the geometry of the described configuration. Additionally, among the potential DOF's existing in the frame, six DOF's are selected. Of these six DOF's, three are translational (prismatic) and three are rotational (revolute) joints, shown in Figure 6-12. It should be noted that although there are four rotational DOF's, there is a geometric relationship between these four angles in the closed-loop frame as $\theta_2 = \theta_1 + \theta_3 + \theta_4$. In other words, there are three unknowns in the mathematical chain and the fourth angle is derived from the geometric relationship explained above. Translational joints are taken where alignment and adjustment is possible in the form of shortening/lengthening. On the other hand, rotational joint are taken where alignment is possible in the form of a local rotation. The DOF vector for Case I is $\vec{\Gamma} = (\theta_1, \theta_2, \theta_3, l_1, l_2, l_3)$.

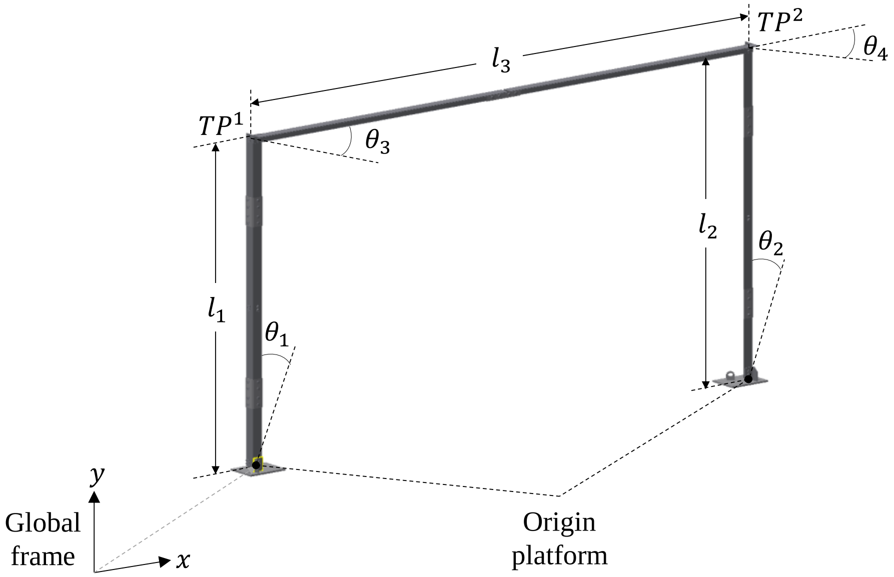


Figure 6-12: Case I: the origin and target platforms along with the DOF's (Γ) are defined (isometric view). The designed state is considered as $\theta_1 = \theta_2 = \theta_3 = \theta_4 = 0$, $l_1 = l_2 = 2000$, and $l_3 = 2992$.

Key matrices identifying the geometry of the frame are summarized in Table 6-1.

Table 6-1: Matrices identifying the geometry of the Case I

Matrix/Vector	Mathematical description	Dimension
DOF	$\vec{\Gamma} = (\theta_1, \theta_2, \theta_3, l_1, l_2, l_3)$	1×6
Position	$\mathbb{F} = \begin{bmatrix} TP^1 \\ TP^1 \\ TP^2 \\ TP^2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$	8*×1
Jacobian	$\mathbb{J} = \frac{\partial \mathbb{F}}{\partial \Gamma} = \begin{bmatrix} \frac{\partial f_1}{\partial \theta_1} & \cdots & \frac{\partial f_1}{\partial l_3} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_4}{\partial \theta_1} & \cdots & \frac{\partial f_4}{\partial l_3} \end{bmatrix}$	8*×6

* Each element has two components (x, y)

The kinematics chains $f(\Gamma)$ are developed based on the definition of the parallel configuration. The global frame is considered coincident with one of the fixed frames, as shown in Figure 6-12-(a). An arbitrary discrepancy is applied on the configuration depicted in Figure 6-12, and the initial state is captured using the method explained in Section 6.5.1. The initial state is fed in the form of discrepancies into the DOF's defined. The iterative realignment procedure (explained in Section 6.5.2.1 and summarized in Figure 6-9) is then applied to find the required changes in the DOF's defined.

The target position is defined as $\mathbb{P}_g = (p_1^1, p_2^1, p_1^2, p_2^2)^T$, in which p_j^i refers to the i^{th} target point calculated by starting from the j^{th} fixed origin (see the second row of Table 6-1). For calculating joint parameters resulting from the inverse kinematics problem, $p_j^1 = p_g^1 = (14, 1900)^T$ and $p_j^2 = p_g^2 = (2980, 1954)^T; \forall j$, ($T$ is the transposition operator). As noted in Table 6-1, each target point's position is calculated using various kinematics chains. Calculation of a single point using various feasible paths results in a discrepancy $\Delta\mathbb{P}$ between the position function and the target position for each path that are not necessarily equal. One immediate output of solving the inverse kinematics problem is to obtain unique positions via various feasible paths. This is mathematically and physically handled by this framework. In order to check the tolerance limit, maximum discrepancy resulted from feasible paths are used for checking the stop criteria for the QNR method explained. The acceptable tolerance is user-defined and is considered 5 mm according to the design codes such as (ASME B31.3 2009). For more detail about the tolerance limit criteria see Table 5-1. The procedure took less than one second on the processing machine used for programming. Key experimental results and metrics are shown in Table 6-2.

Table 6-2: Key experimental results for Case I

Iteration	DOF						$\|\|\mathbb{E}\|\|_{max}$		State
	θ_1 (deg)	θ_2 (deg)	θ_3 (deg)	l_1 (mm)	l_2 (mm)	l_3 (mm)	TP^1 (mm)	TP^2 (mm)	
0*	-1.40	1.40	2.22	1929	2049	3091	67.36	112.90	Out-of-tolerance
1	0.37	-0.29	1.05	1900	1954	2966	5.08	2.41	Out-of-tolerance
2	0.41	-0.36	1.02	1900	1954	2966	0.73	0.71	In-tolerance

*initial state (Γ_0)

6.6.2.2 Case II: multiple-branched pipe spool

A multiple-branched pipe spool is also investigated for verification of the framework developed. The pipe spool has both rotatable and prismatic connections that make it possible to set and investigate various configurations. As shown in Figure 6-13, the investigated configuration has one fixed points ($J = 1$) identifying the fixed platform, and three target points ($I = 3$) identifying the target (moving) platform. A physical interpretation of the target state us either to achieve as-built configuration or to mate properly with facility "tie-in" points in the field. There are 3 equations (i.e. kinematics chain), in total, defining the geometrical relationship. Two rotational and one translational DOF's are also defined based on similar considerations described for Case I. The DOF vector for Case II is $\vec{\Gamma} = (\theta_1, \theta_2, l_1)$.

A similar procedure for developing the kinematics chain and position vector is applied on Case II. The initial state is captured using the local discrepancy analysis framework (Section 6.5.1.3) and the potential realignment plans are calculated (Section 6.5.2.1) based on the defined DOF's. Each element of the kinematics chain of Case II has three components of (x, y, z) rather than Case I that was investigated in 2D. For Case II, the target position is defined as $p_g^1 = (681, -626{,}508)^T$, $p_g^2 = (762{,}196{,}127)^T$, and $p_g^3 = (1160{,}196{,}77)^T$measured in the global frame. The processing for this case also took less than one second on the same processing machine.

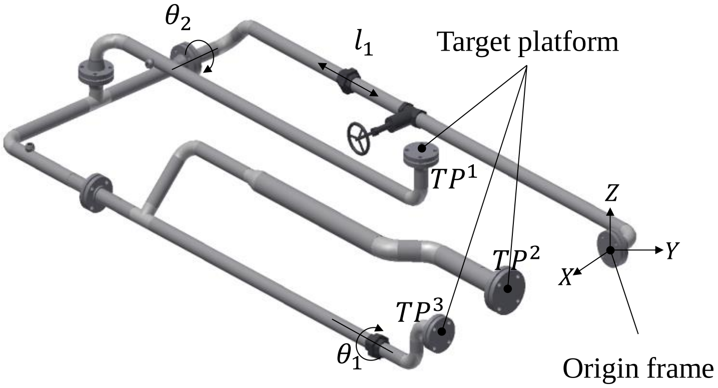


Figure 6-13: Case II: origin and target platforms along with the DOF's are illustrated. Geometric dimensions are provided in Figure 5-5.

The key results for Case II are shown in Table 6-3.

Table 6-3: Key experimental results for Case II

Iteration	DOF			$\|\|\mathbb{E}\|\|_{max}$			State
	θ_1 (deg)	θ_2 (deg)	l_1 (mm)	TP^1 (mm)	TP^2 (mm)	TP^3 (mm)	
0*	5.40	7.80	1898	298.89	393.45	391.60	Out-of-tolerance
1	-0.04	-0.12	1972	23.76	24.59	24.69	Out-of-tolerance
2	0.00	0.00	1848	0.00	0.00	0.00	In-tolerance

*initial state (Γ_0)

6.6.3 Actuation

A turn buckle is used for investigating the possibility of realignment actuation using the mathematical model developed. As explained in Section 6.5.2.2, the actuated change is transformed to the corresponding DOF's in order to measure the impact of actuation. The turn buckle has been installed on the planar frame diagonally, as shown in Figure 6-14. Table 6-2 summarized the required change for realigning the configuration described in Case I. Each complete turn of the turn buckle (2×pitch length of the bolt) is 2.71 *mm* which is applied along the diagonal direction of the side frame, see Figure 6-12 and Figure 6-14. Converting the turn buckle actuation to the corresponding DOF (θ_1, θ_2), see Figure 6-12, results in $\Delta\Gamma_{act} = (\Delta\theta_1, \Delta\theta_2, \Delta\theta_3 = 0, \Delta l_1 = 0, \Delta l_2 = 0, \Delta l_3 = 0)$ as the actuated change and the actuated discrepancy $\mathbb{E}(\Gamma_{act} = \Gamma_0 + \Delta\Gamma_{act})$ is thereby calculated. In contrast with serial manipulators in which forward kinematics is trivial, calculation of the resulting configuration in parallel systems requires solving the nonlinear system of equation $\mathbb{F}(\Gamma) - \mathbb{P} = 0$ as well. In the forward kinematics calculation, the joint parameters Γ are known, while the end flange/member position vector $\mathbb{P}$ is unknown. The procedure described for the QNR method can be employed for forward kinematics, under the condition that the derivatives for the Jacobian are calculated with respect to the position vector, rather than joint angles for inverse kinematics.

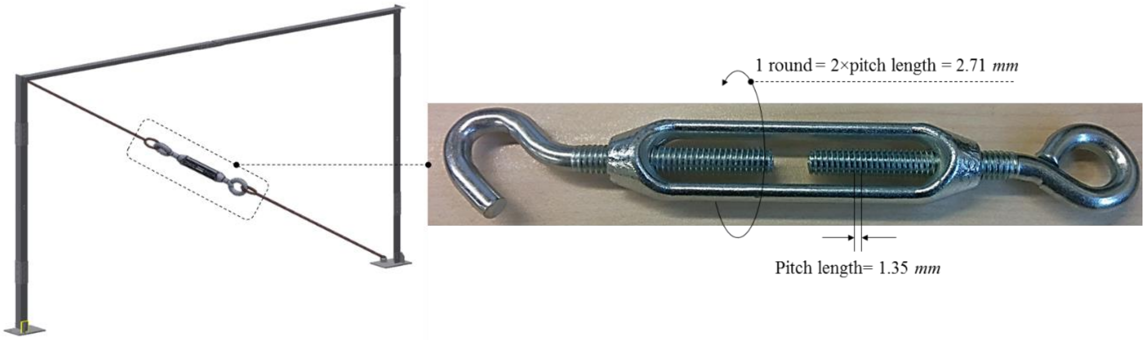


Figure 6-14: Properties of the turn buckle used for the realignment actuation

In order to mathematically measure the impact of actuation on the assembly, the discrepancy configuration resulted from actuation ($\mathbb{P}_{act}$) is compared with the desired configuration ($\mathbb{P}_{g}$). The Euclidean distance ($\Delta\mathbb{P}$) is calculated between these two resulting configurations at TP^{i} as follows:

$$\left(\Delta\mathbb{P}^{i}\right)^{2} = \left({\mathbb{P}_{act}}^{i} - {\mathbb{P}_{g}}^{i}\right)^{T}\left({\mathbb{P}_{act}}^{i} - {\mathbb{P}_{g}}^{i}\right) \tag{6-11}$$

The root mean square (RMS) is then calculated based on the number of target point defined. The RMS is the quadratic mean of the resulting error of the actuation of the assembly at each target point and is calculated as follows:

$$RMS = \left[\frac{\sum_{i=1}^{I}\left(\Delta\mathbb{P}^{i}\right)^{2}}{I}\right]^{1/2} \tag{6-12}$$

where, I is the total number of target points and $I = 2$ for Case I being investigated. Figure 6-15 shows how the RMS changes in applying various numbers of turns of the turn buckle actuator.

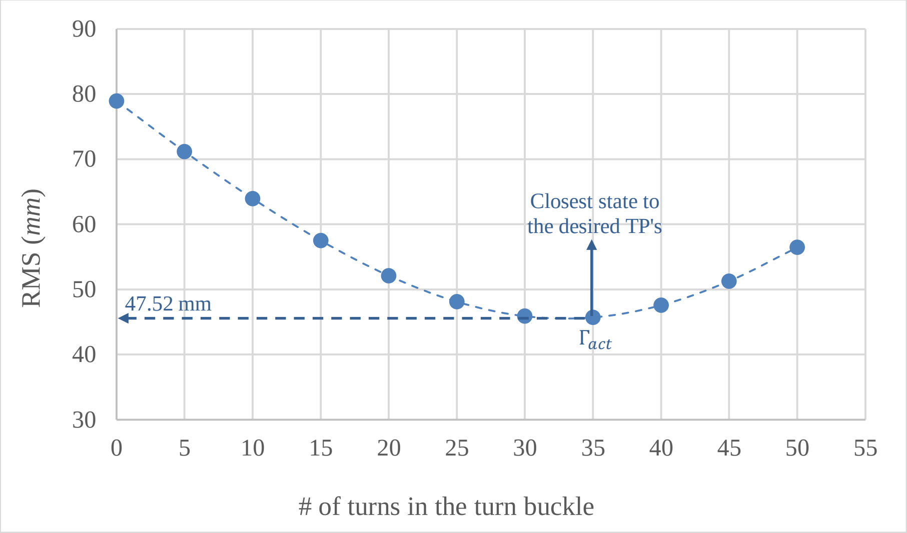


Figure 6-15: Actuated path using a diagonal turn buckle for Case I. Closest possible state to the desired TP's (p_1^g, p_2^g) is shown.

As shown in Figure 6-15, the closest actuated configuration to the desired target points defined for Case I is 35 turns of the turn buckle and the average error (RMS defined in Eq. 6-12) for this realignment is 47.52 *mm*. The RMS must ideally converge to zero in the case that all DOF's are actuated. It should be noted that the described actuation may not be effective for any target points desired. In other words, other actuation scenarios may also be investigated to assess the actuation impact on achieving the target points. For example, the turn buckle, shown in Figure 6-14, can only actuate tension in its direction. Two diagonal turn buckles can then facilitate the actuation process because more target states are covered in the resulting workspace.

6.7 Summary and findings on parallel systems realignment and actuation strategies

A generalized framework is presented in this chapter for autonomous and automated realignment planning and actuation of parallel systems. The analogy of robotics for automatic realignment of construction assemblies introduced in Chapter 4, and developed for serial systems in Chapter 5 is extended and generalized in this work for any type of construction assemblies. The Quasi-Newton-Raphson (QNR) method is used for solving the kinematics equation. The framework is tested and validated on experimental assemblies. The results showed that the method is sufficiently accurate to identify the incurred discrepancies and proactively offer a realignment strategy. Some insights from the experimental study can be summarized as follows:

- Although the kinematics chain for parallel systems is more complex than serial chains, the mathematical solution is quick and robust to achieve the target state within an acceptable tolerance range in less than one second. Such an efficient method can be improved and extended for real-time realignment and assembly. Real-time monitoring and assembly guiding frameworks become possible when accurate 3D image data are obtained in real-time.
- The results for achieving arbitrary target points by solving the resulting inverse kinematics problem is sufficiently accurate and therefore reliable. The kinematics modeling of the parallel systems investigated in the experiments showed that target points are mathematically and systematically achieved considering all site and assembly conditions.
- Systematic and automatic actuation and realignment applications were also presented and tested. The actuation results show that realignment applications can also be mathematically modeled and then can be controlled by the pipefitters or welders. Potentially, rework will be substantially minimized by using this and similarly guided assembly or pipe fitting strategies.

Although the method is capable of calculating the realignment strategy required for any type of construction assemblies, the efficiency challenge still remains unanswered and uninvestigated. In other words, realignment strategies can be developed using the frameworks developed; however, identifying the optimum strategy from a structural perspective requires investigation of structural models based on physical properties of members and joints and of the craft work required for various fixes. Finite element analysis would probably be the suitable tool for identifying the optimum realignment strategy by modeling and investigating the work (energy) required for actuating assemblies. For this purpose, structural and physical properties such as stiffness of the assembly must be incorporated with the model mathematically. This is currently being investigated by the authors and will be covered in subsequent studies.

Potential avenues for extending this framework include applications in modular assembly, staged fabrication, and time effectiveness improvement for real-time monitoring of construction components. Using the introduced robotics analogy in active structural control (Housner et al. 1997; Spencer and Nagarajaiah 2003) and applying it in smart self-adjusted structures, such as tensegrity structures (Fest et al. 2004) are the potential extensions of this research toward structural control. Smart assembly using the general method developed here becomes possible as well. Given the as-built status of an assembly or module and updated BIM, utilizing this framework can provide (semi) real-time feedback for welders and pipefitters to apply the most efficient strategy for achieving the target points. The geometric and local situations are taken into account mathematically and physically using the strategy calculated by this method. Moreover, real-time 3D imaging tools that have been substantially improved recently such as Microsoft Kinect V2 (http://www.microsoft.com/en-us/kinectforwindows/) and Google Tango (https://www.google.com/atap/projecttango/) can also be employed for real-time assembly and monitoring of construction components. It should be noted that the framework explained in this chapter strongly relies on the accuracy of the point cloud acquired for representing the as-built status. Employing the mentioned devices is only effective if the acquired point cloud is guaranteed to have a certain level of accuracy, otherwise the discrepancy analysis results and thereby the resulting realignment are neither reliable nor efficient. In general, time effectiveness improvement to move toward real-time monitoring is the other research subject currently being investigated in CIS laboratory at the University of Waterloo.

Chapter 7
Summary, Conclusions and Recommendations for Future Work

7.1 Thesis summary

This thesis aimed to address the emerging need for continual inspection and assessment of assemblies while they are being fabricated or assembled. Referring back to the control loop in systems theory (Figure 1-1), each systems requires a backward loop as a signal in order for the forward flow to be operative. Likewise, a construction management system for proactive realignment planning and actuation requires the as-built status and the incurred discrepancies as the feedback signal. The required realignment actions can then be calculated based on the calculated discrepancy. Utilizing such a framework may substantially reduce the rework occurrence in construction projects. For calculating the required changes between the designed and built status of assemblies, this thesis borrowed concepts from 3D imaging and robotics kinematics. Making the analogy of construction assemblies with robot arms and mutual degrees of freedom makes it possible to use kinematics (i.e. forward and inverse) concepts for planning the potential realignment actuations. Each component of this research required to develop the explained framework (Figure 1-1 and Figure 1-5), was extensively formed and developed. All frameworks were validated on experimental-sized assemblies in the University of Waterloo's CIS Laboratory.

7.1.1 Integrated, customizable preprocessing system

A customizable and user-controlled preprocessing step for the registration step and automated compliance control of industrial assemblies was first developed in Chapter 3. An ICP (Iterative Closest Point)-based registration preceded by a PCA (Principal Component Analysis) rough alignment was developed for customizable compliance control in order to measure the effective variants. An automated deviation analysis as well as an autonomous approach for discrepancy detection was presented and the method was validated. Experimental results showed that the method enables the identification and detection of incurred discrepancies based on the local error measured by the root mean square (RMS). Although the errors were detected and localized, accurate quantification required more investigation on the registration step and a solution to the challenges for correct deviation analysis (Nahangi and Haas 2014).

7.1.2 Automated local discrepancy feedback calculation

In Chapter 4, the drawbacks of the regular ICP algorithm for local discrepancy quantification were handled by modifying the ICP registration algorithm. These drawbacks included overlooking the local situation (i.e. installation segment) for staged-fabricated construction. A modified registration algorithm was introduced to appropriately consider the on-site situations. Moreover, a forward kinematics loop based on the mutual (Degrees of Freedom) DOF's, existing in the assemblies for realignment, was developed to locally measure and quantify the incurred discrepancies. The measured discrepancies were transformed to the local axis using the kinematics chain developed. Experimental results showed that the developed framework is sufficiently accurate for capturing the initial (as-built) status and feeding as the feedback signal (information) required for calculating the realignment actions (Nahangi et al. 2015b).

7.1.3 Realignment calculation of serial assemblies

Once accurate and localized discrepancies were quantified, it became possible to calculate the required realignment actions (Chapter 5). Making the analogy of linear (serial) pipe spools with serial robots and developing the kinematics chain based on the potential realignment feasibilities led to the calculation of the required changes on the previously defined DOF's. The geometry was identified by the kinematics chain as a function of the defined DOF's. A Taylor-based approximation of the kinematics chain was then used to linearize the kinematics function. A Jacobian (matrix of differentials) was then used as the key metric for describing the relationship between the Cartesian space and the joint space. A pseudoinverse approach based on Moore-Penrose algorithm was also used for solving the linear matrix equation resulting from the indeterminate proposed methodology (Nahangi et al. 2015a).

7.1.4 Generalization of the realignment calculation and actuation

The framework developed in Chapter 4 and Chapter 5 was only applicable to serial assemblies such as a serial pipe spool. Generalization and application to more complex assemblies and parallel systems was developed and validated in Chapter 6. Parallel systems are computationally more intensive because of the multiple branches identifying the geometry. A general procedure for all types of construction assemblies was developed. Moreover, a more general approach, quasi-Newton-Raphson (QNR), for solving the resulting mathematical equation was introduced. Implementation of the generalized framework led to the conclusion that the robotics analogy is applicable for realignment calculation as well as actuation of defective construction assemblies.

7.2 Conclusions and contributions

The key contributions and associated conclusions of the work presented in this thesis are summarized below.

7.2.1 Proactive strategy for rework avoidance

Developing a proactive strategy for rework minimization was a key contribution of this research. Evaluating the related literature in Section 2.2 and 2.3 identified two key knowledge gaps: (1) the potentials of 3D imaging techniques for the as-built status assessment are not yet fully achieved, and (2) existing approaches for fabrication and assembly control are "reactive". Utilizing the framework presented in this work enables a proactive strategy that is ideally applicable for pipe fabrication and steel erection rework minimization. It could also improve productivity as secondary impact.

7.2.2 Algorithmic and programmable framework

Another notable contribution of this work is that it enables an algorithmic and programmable approach for fabrication and assembly control. As mentioned earlier, although significant advances are achieved in 3D imaging and model-based techniques for the as-built status assessment, extracting meaningful information and efficient communication of the extracted information are still challenging. It therefore motivates the need for an algorithmic and programmable framework in order to utilize a fully integrated electronic management system. The set of frameworks developed in this work is a step toward the automation of pipe spool, steel frames and modules fabrication and assembly control tasks.

7.2.3 Efficient and robust framework

An efficient, accurate and robust tool is necessary considering the importance of maintenance and inspection on construction projects. The experimental verification signified that the frameworks are sufficiently accurate and robust for efficient assessment of the as-built status or assembly in construction or fabrication site. Investigating the effective variants such as the size of the sliding cube and point cloud density represent potential paths for improving the performance and robustness of this set of frameworks.

7.2.4 Time effective processing method

With the recent advances in real-time and augmented reality based tools for data acquisition, time-effectiveness is a key factor for the assessment frameworks developed. Investigating the time-related aspects of various components of the developed frameworks showed that the processing method can be effectively applied for assembly and fabrication control. Using faster (near real-time) tools for data acquisition is a potential avenue for extending this research.

7.3 Limitations

One of the limitations of the discrepancy quantification (Chapter 4) and realignment (Chapter 5) frameworks is that they require accurate and reliable inputs. Where point clouds acquired are inaccurate, preprocessing steps will not perform well and then the clouds are useless for the required realignment planning. Another limitation is that the secondary objects such as supports, brackets, and jacks must be appropriately removed from the point cloud in a preprocessing step. The presence of the secondary objects in the acquired point cloud will cause inaccurate discrepancy quantification and therefore unreliable realignment planning. Thus approaches to automated removal of this clutter are required.

Another limitation of the work presented in this work is that it is only capable of finding the feasible changes required for realigning defective assemblies based on the defined DOF’s. Appropriate generalized principles and approaches to assigning DOF’s must be explored. In addition, for efficient realignment planning, best options must be calculated by minimizing the energy (work) consumed. Such an optimization problem is not trivial and requires structural investigation of the assemblies and the cost of craft workers and automated actions. Such structural investigation includes the development of physical properties such as stiffness or flexibility that should be combined with the previously defined geometric parameters (i.e. DOF’s and kinematics chain).

Another step to achieve a fully automated framework is automated noise removal. Developing and implementing (semi-) automated methods for object isolation can result in significant time improvement as the predominant part of the (pre-) processing time of the framework in its current form is spent on accurate object-of-interest isolation. Kinematics chain development is algorithmic and programmable, however, it requires user-defined location and types of DOF’s based on the local and physical situations of construction site and assemblies. Their automated formulation into forward kinematics equations which are feasible but not trivial must also be explored.

7.4 Recommendations for future research

In this section, some potential research thrusts for extending the work initiated in this research are discussed.

7.4.1 Augmentation with BIM for an integrated realignment framework

Augmenting and integrating the developed set of frameworks with building information models (BIM) can facilitate construction management systems for realignment and adjustment of defective assemblies. Integrating DOF's with building information models and developing a 4D BIM (3D drawings + DOF's for realignment) can have significant impacts on minimizing rework and therefore improving productivity. On the other hand, as-built BIM modeling can be generated based on the discrepancies found in this research. Automated BIM reconfiguration for automated as-built modeling is another potential area for extending this work.

7.4.2 Optimization of the developed realignment planning

As mentioned earlier, the described framework is currently offering feasible realignment strategies based on the DOF's defined. It is desirable to calculate the best option among the feasible strategies already calculated. This will require more investigation on physical and structural modeling of construction assemblies and integrating such models with the developed frameworks. The work (energy) consumed for realignment that also includes labor cost for physical actuation can then be optimized for finding the best realignment option.

7.4.3 Fully automated realignment planning

The kinematics chain for the realignment framework is currently developed (semi) manually. The DOF's are defined based on the experts' and practitioners' suggestions whereby realignment is feasible to take place physically and practically. A more automated strategy to develop the kinematics chain will facilitate electronic and automated integration of the realignment strategy with existing construction management systems. Graph-based representation of assemblies (McPhee 1996) seems to be a potential solution for automated development of kinematics chain.

7.4.4 Reduction of preprocessing time

The less processing time spent for calculating realignment plans, the more efficient realignment framework for applying on real-time applications. Although the processing time is already sufficiently fast, and therefore acceptable for real-time applications, however, there is an opportunity to make the implemented computer program faster and more efficient. This may require implementation of the described method in other programming languages such as C++ that provide applicable and useful features such as object-oriented programming.

7.4.5 User interfaces and visualization

These aspects of the framework must be further developed for practical implementation and adoption. An interface for better visualization of the procedure is desired to be developed in future works.

7.4.6 Development of an automated real-time monitoring system

Given the as-built status of an assembly or module and updated BIM, utilizing this framework can provide (semi) real-time feedback for welders and pipefitters to apply the most efficient strategy for achieving the target points. The geometric and local situations are taken into account mathematically and physically using the strategy calculated by the robotics analogy developed and implemented in this thesis. Moreover, real-time 3D imaging tools that have been substantially improved recently such as Microsoft Kinect V2 (http://www.microsoft.com/en-us/kinectforwindows/) and Google Tango (https://www.google.com/atap/projecttango/) can also be employed for real-time assembly and monitoring of construction components. Real-time and smart assembly of construction components are the ultimate extensions of this research.

7.5 Publications

The peer-refereed publications, directly related to the scope of this thesis, and authored by the candidate are listed below:

7.5.1 Peer-refereed journal articles

1. **Nahangi, M.**, Haas, C. (2014). “Automated 3D compliance checking in pipe spool fabrication”, *Advanced Engineering Informatics,* Volume 28 (4), 360-369.
2. **Nahangi, M.**, Yeung, J., Haas, C., Walbridge, S., West, J. (2015). “Automated assembly discrepancy feedback using 3D imaging and forward kinematics”, *Automation in Construction*, DOI: 10.1016/j.autcon.2015.04.005.

3. **Nahangi, M.**, Haas, C., West, J., Walbridge, S. (2015). "Automatic realignment of defective assemblies using an inverse kinematics analogy", *ASCE Computing in Civil Engineering*, CP.1943-5487.0000477.
4. **Nahangi, M.**, Czerniawski, T., Haas, C., Walbridge, S., West, J., (2015). "Parallel systems and structural frames realignment planning and actuation strategy". *ASCE Journal of Computing in Civil Engineering*, Submitted in April 2015, (under review).

7.5.2 Peer-refereed conference papers

1. **Nahangi, M.**, Ahmed, M., Turkan, Y., Haas, C., Haas, R. (2012). "Automated Progress Tracking of Construction Projects Using Sensing and 3D Imaging Technologies". *Modular and Off-site Construction Summit*, November 2012, Edmonton AB, Canada.
2. Yeung, J., **Nahangi, M.**, Shahtaheri, Y., Walbridge, S., Haas, C. (2014). "Comparison of Methods Used for Detecting Unknown Structural Elements in Three Dimensional Point Clouds". *ASCE Construction Research Congress (CRC),* May 2014, Atlanta GA, USA.
3. **Nahangi, M.**, Safa, M., Shahi, A., Haas, C. (2014). "Automated Registration of 3D Point Clouds with 3D CAD Models for Remote Assessment of Staged Fabrication". *ASCE Construction Research Congress (CRC),* May 2014, Atlanta GA, USA.
4. **Nahangi, M.**, Yeung, J., Amaral, J., Freitas, F. N., Walbridge, S., Haas, C. (2014). "Automated Deviation Analysis for As-Built Status Assessment of Steel Assemblies and Pipe Spools". *ASCE Conference of Computing in Civil Engineering*, June 2014, Orlando FL, USA.
5. **Nahangi, M.**, Czerkniawski, T., Yeung, J., Walbridge, S., Haas, C., West, J. (2015). "An image-based framework for automated realignment of industrial assemblies". *CSCE International Construction Specialty Conference (ICSC),* June 2015, Vancouver BC, Canada. **Nominated for the "Best Paper Award" by the technical review panel. Invited for publication in Canadian Journal Civil Engineering special issue.**
6. Czerkniawski, T., **Nahangi, M.**, Walbridge, S., Haas, C., West, J. (2015). "Automated dimensional compliance assessment with incomplete point cloud". *CSCE International Construction Specialty Conference (ICSC),* June 2015, Vancouver BC, Canada. **Invited for publication in Canadian Journal of Civil Engineering special issue.**
7. **Nahangi, M.**, Yeung, J., Haas, C., Walbridge, S., West, J. (2015). "Robotics kinematics analogy for realignment of defective assemblies". *International Symposium on Automation and Robotics in Construction (ISARC),* July 2015, Oulu, Finland.

8. **Nahangi, M.**, Chaudhary, L., Yeung, J., Haas, C., Walbridge, S. (2015). "Skeleton-Based Registration of 3D Laser Scans for Automated Quality Assurance". *ASCE International Workshop of Computing in Civil Engineering (IWCCE)*, June 2015, Austin TX, USA.
9. **Nahangi, M.**, Czerniawski, T., Haas, C. (2015). "Automated 3D shape detection and outlier removal in cluttered laser scans of industrial facilities". *International Conference of Innovation in Construction (CIGOS)*, May 2015, Paris, France.
10. Czerniawski, T., **Nahangi, M.**, Haas, C., Walbridge, S. (2015). "Real-time assembly and fabrication monitoring of industrial assemblies". 15th *International Conference on Construction Application of Virtual Reality (CONVR)*, October 2015, Banff AB, Canada.

Appendix A

Principal Component Analysis (PCA)

```
function [NewPC1, NewPC2] = PCA (PC1, PC2,rotate);


j=1;

while j<=2

    if j==1
            B=PC1;

    end
    if j==2
            B=PC2;

    end

n = length(B);

Mean(1,1) = sum (B(:,1)) / n;
Mean(1,2) = sum (B(:,2)) / n;
Mean(1,3) = sum (B(:,3)) / n;

i=1;
while i<=n
B(i,:) = B(i,:)-Mean;
i=i+1;
end

S = (1/n)*B'*B;

[V,D] = eig(S);

 aux = V;

if D(2,2)>D(1,1) && D(2,2)>D(3,3)
    V(:,1) = aux(:,2);
    if D(1,1)>D(3,3)
        V(:,2) = aux(:,1);
        V(:,3) = aux(:,3);
    else
        V(:,2) = aux(:,3);
        V(:,3) = aux(:,1);
    end
```

```
end



if D(3,3)>D(1,1) && D(3,3)>D(2,2)
    V(:,1) = aux(:,3);
    if D(2,2)>D(1,1)
        V(:,2) = aux(:,2);
        V(:,3) = aux(:,1);
    else
        V(:,2) = aux(:,3);
        V(:,3) = aux(:,2);
    end
end

if D(1,1)>D(2,2) && D(1,1)>D(3,3)
    V(:,1) = aux(:,1);
    if D(3,3)>D(2,2)
        V(:,3) = aux(:,2);
        V(:,2) = aux(:,3);
    else
        V(:,3) = aux(:,3);
        V(:,2) = aux(:,2);
    end
end

R = inv(V)*eye(3);

if rotate==1 && j==1
    R=R*-1;
end

New = [];
i=1;
while i<=n
New (i,:) = B(i,:)*R;
i=i+1;
end

if j==1
    NewPC1 = New;

end
if j==2
        NewPC2 = New;

end

    j=j+1;
```

```
end

n=length (NewPC1);
Mean1(1,1) = sum (NewPC1(:,1)) / n;
Mean1(2,1) = sum (NewPC1(:,2)) / n;
Mean1(3,1) = sum (NewPC1(:,3)) / n;

n=length (NewPC2);
Mean2(1,1) = sum (NewPC2(:,1)) / n;
Mean2(2,1) = sum (NewPC2(:,2)) / n;
Mean2(3,1) = sum (NewPC2(:,3)) / n;

T(1,1) = Mean1(1,1) - Mean2(1,1);
T(1,2) = Mean1(2,1) - Mean2(2,1);
T(1,3) = Mean1(3,1) - Mean2(3,1);

n=length (NewPC1);
i=1;
while i<=n
NewPC1(i,:) = NewPC1(i,:)-T;
i=i+1;
end

end
```

Appendix B

Local error investigation (sliding cube)

```
function [cubeA, cubeB, pos] = cube (a, b, d, minpoint)
%a and b are the point clouds, d is the cube size in meters and minpoint
is
%the minimum number of point in the cube not to be disregarded

cubeA = [];
cubeB = [];

a=a';
b=b';

%Define minimum and maximum values for the X, Y and Z coordinates for both
%point clouds

lengtha = length(a);
lengthb = length(b);

minXa = min(a(:,1));
minYa = min(a(:,2));
minZa = min(a(:,3));

minXb = min(b(:,1));
minYb = min(b(:,2));
minZb = min(b(:,3));

maxXa = max(a(:,1));
maxYa = max(a(:,2));
maxZa = max(a(:,3));

maxXb = max(b(:,1));
maxYb = max(b(:,2));
maxZb = max(b(:,3));

minX = min(minXa, minXb);
minY = min(minYa, minYb);
minZ = min(minZa, minZb);

maxX = max(maxXa, maxXb);
maxY = max(maxYa, maxYb);
maxZ = max(maxZa, maxZb);

X = minX;
Y = minY;
Z = minZ;
pos=[];
```

```
%count the number of cubes
count_cube = 1;

%Main loop: X and maxX define boundaries for each cube in the X axe, the
%same for Y and Z
while X<maxX
    Y = minY;
    Z = minZ;
    while Y<maxY
        Z = minZ;
        while Z<maxZ

            npoints=0;
            aux = [];
            count_aux = 1;
            p=1;

            %test if points are within the desired boundary
            for p=1:lengtha
                if a(p,1)>X && a(p,1)<=X+d && a(p,2)>Y && a(p,2)<=Y+d &&
a(p,3)>Z && a(p,3)<=Z+d
                    aux(count_aux,:) = a(p,:);

                    count_aux = count_aux+1;
                end
            end

            npoints = length(aux);
            cubeA{count_cube} = aux;

            aux = [];
            count_aux = 1;
            p=1;

            for p=1:lengthb
                if b(p,1)>X && b(p,1)<=X+d && b(p,2)>Y && b(p,2)<=Y+d &&
b(p,3)>Z && b(p,3)<=Z+d
                    aux(count_aux,:) = b(p,:);
                    count_aux = count_aux+1;
                end
            end

            npoints = npoints + length(aux);
            cubeB{count_cube} = aux;

            %only increase cube counter if the combined number of points
is
            %greater than the requested
            if npoints>=minpoint
```

```
                pos(count_cube,:)=[X;Y;Z];
                count_cube = count_cube+1;
            end


            %increase boundaries according to specified cube size for next
iteration
            Z = Z+d;

        end
        Y=Y+d;
    end
    X=X+d;
end

%check if last cube has the minimum number of points required
l = length(cubeA);
npoints = length(cubeA{l}) + length(cubeB{l});

auxa=[];
auxb=[];
if npoints<minpoint
    auxa=cubeA;
    auxb=cubeB;
    cubeA=[];
    cubeB=[];
    l=l-1;
    for i=1:l
        cubeA{i} = auxa{i};
        cubeB{i} = auxb{i};
    end
end

end
```

Appendix C

Local neighborhood identification

```
global selectedIndex;
h = clickA3DPoint(M);
d=input('Please select point of interest on plot, then press Enter ');
d=input('Enter the size of cube(default:0.5m): ');
if isempty(d)
    d=0.5;
end
```

Appendix D

Constrained registration

```
[CubeM, CubeD, pos] = selected_cube(M, Dicp, selectedIndex, d);

%get rotation angles necessary to go from as-is to as-designed
[constrRot, constrTrans] = icp( CubeM{1}', CubeD{1}', 30 );
%with results from constrained registration, transform rest of point cloud
Dconstr = constrRot * Dicp + repmat(constrTrans, 1, length(Dicp));
figure( h );
plot3(M(1,:),M(2,:),M(3,:),'b.',Dconstr(1,:),Dconstr(2,:),Dconstr(3,:),'r.
');
%h = clickA3DPoint(Dconstr);
```

Appendix E

Denavit-Hartenberg (D-H) representation

The notation used in this proposal to transform the adjacent local axes is based on Denavit-Hartenberg convention which is summarized in six consecutive steps as follows:

Step 1: Locate and label the joint axes $z_0, z_1, \dots, z_i$ along the axis of rotation of each joint.

Step 2: Locate the origin frame and establish x_0 and y_0 to form a right-hand frame with z_0.

Step 3: Locate the origin O_i where the common normal to z_i and z_{i-1} intersects z_i. If z_i and z_{i-1} are intersecting, locate the origin at this intersection. If z_i and z_{i-1} are parallel locate the origin in any convenient position.

Step 4: Locate x_i along the common normal between z_i and z_{i-1} through O_i.

Step 5: Establish y_i to form a right hand frame for each joint.

Step 6: Calculate the D-H parameters as follows:

a_i=distance along x_i form O_i to the intersection of the intersection of x_i and z_{i-1} axes.

α_i=angle between z_i and z_{i-1} measured about x_i.

d_i= distance along z_{i-1} form O_{i-1} to the intersection of the intersection of x_i and z_{i-1} axes.

θ_i= angle between x_{i-1} and x_i measured about z_{i-1}.

Having found the D-H parameters, required transformation for each frame is then calculated using the following function.

```
syms alpha a d theta h1 l1 l2 theta1 theta2
T=[cos(theta) -sin(theta)*cos(alpha) sin(theta)*sin(alpha) a*cos(theta);
    sin(theta) cos(theta)*cos(alpha) -cos(theta)*sin(alpha) a*sin(theta);
    0 sin(alpha) cos(alpha) d;
    0 0 0 1];
```

Appendix F

Local registration

```
%after constrained registration, analyze specific selected cubes
exit = false;
counter = 1;
while exit == false;
    d=input('Select point of interest on plot, or type "exit", then press
Enter: ', 's');
    if strcmp(d,'exit')
        exit = true;
        break;
    end
    d=input('Enter the size of cube(default:0.5m): ');
    if isempty(d)
        d=0.5;
    end
    [selM, selD, pos] = selected_cube(M, Dconstr, selectedIndex, d);
    figure( h+1 );
    plot3(selM{1}(:,1),selM{1}(:,2),selM{1}(:,3),'b.', ...
        selD{1}(:,1),selD{1}(:,2),selD{1}(:,3),'r.');
    CubeRotation(counter,:) = RotationAngles(selM, selD)
    counter = counter + 1;
end
```

Appendix G

Rotation matrix

```
%provides rotation angles (deg) required to go from a s-is to as-designed
%in x,y,z axis. Input are cube point clouds (output from cube function)
function [ RotAng, RotArray ] = RotationAngles( m, s )
RotAng = zeros( length(m), 3 );
RotArray = zeros( 3,3, length(m) );
for i = 1:length(m)
    [TR, TT] = icp(m{i}',s{i}',50); %TR: Rotation Matrix. TT: Translation
Matrix
    %RotArray{ 1, 1, i } = TR( 1,1 );
    RotArray(:,:,i) = TR;
    %num2cell(TR)
    RotAng( i,2 ) = -asin( TR( 3,1 ) );
    RotAng( i,3 ) = -acos( TR( 1,1 ) / cos( RotAng( i,2 ) ) );
    RotAng( i,1 ) = -acos( TR( 3,3 ) / cos( RotAng( i,2 ) ) );
end
RotAng = RotAng*180/pi;
end

%To test, run finalcode.m once, and then run:
%RotAng = RotationAngles(CubeM, CubeD)

%The results are in the following format. Same order as CubeM:
%phix  phiy  phiz
%phix  phiy  phiz
%phix  phiy  phiz
%phix  phiy  phiz
%...
```

Appendix H

Kinematics chains of Configurations I and II used in the experiments

Config I:

```
clear all
syms alpha a d theta h1 l1 l2 theta1 theta2
T=[cos(theta) -sin(theta)*cos(alpha) sin(theta)*sin(alpha) a*cos(theta);
    sin(theta) cos(theta)*cos(alpha) -cos(theta)*sin(alpha) a*sin(theta);
    0 sin(alpha) cos(alpha) d;
    0 0 0 1];
T1=subs(T,{alpha a d theta},{0 -h1 l1 theta1});

T2=subs(T,{alpha a d theta},{-pi()/2 0 l2 theta2});

T=T1*T2;

f=T*[127; 2626; 1300; 1];
f=f(1:3,:);
f=subs(f,{h1 l1 l2},{203 2197 603})
j=jacobian(f,[theta1 theta2]);
jinv=((j'*j)^-1)*j';
ptarget=[-100 1280 180]';
i=1;
dof(1:2,i)=[8.43*pi()/180,-6.58*pi/180];
pcurrent(1:3,i)=subs(f,{theta1 theta2},{dof(1,i) dof(2,i)});
deltap(1:3,i)=(ptarget-pcurrent);
vdeltap(:,i)=(deltap'*deltap)^.5;
tol=10;
while vdeltap(:,i)>= tol        %tol is the maximum allowable tolerance
    deltadof(1:2,i)=(subs(jinv,{theta1 theta2},{dof(1,i)
dof(2,i)}))*deltap(1:3,i);
    dof(1:2,i+1)=dof(1:2,i)+deltadof(1:2,i);
    pcurrent(1:3,i+1)=subs(f,{theta1 theta2},{dof(1,i+1) dof(2,i+1)});
    deltap(1:3,i+1)=ptarget-pcurrent(1:3,i+1);
    vdeltap(:,i+1)=(deltap(1:3,i+1)'*deltap(1:3,i+1))^.5;
    i=i+1;
end
```

Config II:

```
clear all
syms alpha a d theta h1 l1 l2 theta1 d1
%define all constant variable as symbols to be used for parametric and
symbolic manipulation
T=[cos(theta) -sin(theta)*cos(alpha) sin(theta)*sin(alpha) a*cos(theta);
    sin(theta) cos(theta)*cos(alpha) -cos(theta)*sin(alpha) a*sin(theta);
    0 sin(alpha) cos(alpha) d;
    0 0 0 1];
T1=subs(T,{alpha a d theta},{0 -h1 l1 theta1});
%Denavit-Hartenberg parameters and the transformation generated

T2=subs(T,{alpha a d theta},{pi() l2 d1 pi()/2});

T=T1*T2;


f=T*[0; 127; 1848; 1];
f=f(1:3,:);
f=subs(f,{h1 l1 l2},{203 2197 1300});
j=jacobian(f,[theta1 d1]);
jinv=((j'*j)^-1)*j';
ptarget=[-100 1280 180]';
i=1;
dof(1:2,i)=[1.782*pi()/180,-175];
pcurrent(1:3,i)=subs(f,{theta1 d1},{dof(1,i) dof(2,i)});
deltap(1:3,i)=(ptarget-pcurrent);
vdeltap(:,i)=(deltap'*deltap)^.5;
tol=5;
while vdeltap(:,i)>= tol        %tol is the maximum allowable tolerance
    deltadof(1:2,i)=(subs(jinv,{theta1 d1},{dof(1,i)
dof(2,i)}))*deltap(1:3,i);
    dof(1:2,i+1)=dof(1:2,i)+deltadof(1:2,i);
    pcurrent(1:3,i+1)=subs(f,{theta1 d1},{dof(1,i+1) dof(2,i+1)});
    deltap(1:3,i+1)=ptarget-pcurrent(1:3,i+1);
    vdeltap(:,i+1)=(deltap(1:3,i+1)'*deltap(1:3,i+1))^.5;
    i=i+1;
end
```

Bibliography

AbouRizk, S. (2010). "Role of Simulation in Construction Engineering and Management." *Journal of Construction Engineering and Management*, 136(10), 1140-1153.

Ahmed, M., Haas, C., Haas, R. (2014). "Automatic Detection of Cylindrical Objects in Built Facilities." *Journal of Computing in Civil Engineering*, 04014009.

Ahmed, M., Haas, C. T., Haas, R. (2012). "Using Digital Photogrammetry for Pipe-Works Progress Tracking." *Canadian Journal of Civil Engineering*, 39(9), 1062-1071.

Ahmed, M., Haas, C., Shahi, A., Aryan, A., West, J., Haas, R. (2011a). "Rapid tracking of pipe-works progress using digital photogrammetry." *Proc., Proceedings, Annual Conference - Canadian Society for Civil Engineering*, 2455-2465.

Ahmed, M., Haas, C. T., Haas, R. (2011b). "Toward Low-Cost 3D Automatic Pavement Distress Surveying: The Close Range Photogrammetry Approach." *Canadian Journal of Civil Engineering*, 38(12), 1301-1313.

AISC. (2005). *Code of Standard Practice for Steel Buildings and Bridges*, American Institute of Steel Construction, Chicago, Illinois.

Akinci, B., Boukamp, F., Gordon, C., Huber, D., Lyons, C., Park, K. (2006a). "A Formalism for Utilization of Sensor Systems and Integrated Project Models for Active Construction Quality Control." *Automation in Construction*, 15(2), 124-138.

Akinci, B., Kiziltas, S., Ergen, E., Karaesmen, I. Z., Keceli, F. (2006b). "Modeling and Analyzing the Impact of Technology on Data Capture and Transfer Processes at Construction Sites: A Case Study." *Journal of Construction Engineering and Management*, 132(11), 1148-1157.

Al-Hussein, M., Athar Niaz, M., Yu, H., Kim, H. (2006). "Integrating 3D Visualization and Simulation for Tower Crane Operations on Construction Sites." *Automation in Construction*, 15(5), 554-562.

Alvanchi, A., Azimi, R., Lee, S., Abourizk, S. M., Zubick, P. (2012). "Off-Site Construction Planning using Discrete Event Simulation." *Journal of Architectural Engineering*, 18(2), 114-122.

Anil, E. B., Tang, P., Akinci, B., Huber, D. (2013). "Deviation Analysis Method for the Assessment of the Quality of the as-is Building Information Models Generated from Point Cloud Data." *Automation in Construction*, 35, 507-516.

Arditi, D., and Gunaydin, H. M. (1997). "Total Quality Management in the Construction Process." *International Journal of Project Management*, 15(4), 235-243.

ASME B31.3. (2009). "Section D20-B31.3-G, ASME B31.3 Process Piping Guide." *Engineering Standards Manual PD342*, Revision 2 Ed.

Atkinson, A. R. (1999). "The Role of Human Error in Construction Defects." *Structural Survey*, 17(4), 231-236.

AutoCAD. (2012). "AutoCAD 2013 user's guide." <http://docs.autodesk.com/ACDMAC/2013/ENU/PDFs/acdmac_2013_users_guide.pdf> (March/6, 2013).

Azimi, R., Lee, S., Abourizk, S. M., Alvanchi, A. (2011). "A Framework for an Automated and Integrated Project Monitoring and Control System for Steel Fabrication Projects." *Automation in Construction*, 20(1), 88-97.

Bates, D. M. (1988). *Nonlinear Regression Analysis and its Applications*, New York: Wiley, New York; New York; Toronto.

Ben-Israel, A., and Greville, T. N. (2003). *Generalized Inverses*, Springer.

Besl, P. J., and McKay, H. D. (1992). "A Method for Registration of 3-D Shapes." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2), 239-256.

Bhatla, A., Choe, S. Y., Fierro, O., Leite, F. (2012). "Evaluation of Accuracy of as-Built 3D Modeling from Photos Taken by Handheld Digital Cameras." *Automation in Construction*, 28, 116-127.

Bock, T. (2007). "Construction Robotics." *Autonomous Robots*, 22(3), 201-209.

Bosche, F. (2008). "Automated Recognition of 3D CAD Model Objects in Dense Laser Range Point Clouds". PhD Thesis, University of Waterloo.

Bosche, F., Haas, C. T., Akinci, B. (2009). "Automated Recognition of 3D CAD Objects in Site Laser Scans for Project 3D Status Visualization and Performance Control." *Journal of Computing in Civil Engineering,* 23(6), 311-318.

Bosche, F., and Haas, C. T. (2008). "Automated Retrieval of 3D CAD Model Objects in Construction Range Images." *Automation in Construction,* 17(4), 499-512.

Bosché, F., Ahmed, M., Turkan, Y., Haas, C. T., Haas, R. (2014a). "The Value of Integrating Scan-to-BIM and Scan-Vs-BIM Techniques for Construction Monitoring using Laser Scanning and BIM: The Case of Cylindrical MEP Components." *Automation in Construction.*

Bosché, F., Guillemet, A., Turkan, Y., Haas, C. T., Haas, R. (2014b). "Tracking the Built Status of MEP Works: Assessing the Value of a Scan-Vs-BIM System." *Journal of Computing in Civil Engineering,* 28(4).

Bosché, F. (2012). "Plane-Based Registration of Construction Laser Scans with 3D/4D Building Models." *Advanced Engineering Informatics,* 26(1), 90-102.

Bosché, F., and Guenet, E. (2014). "Automating Surface Flatness Control using Terrestrial Laser Scanning and Building Information Models." *Automation in Construction,* 44(0), 212-226.

Brilakis, I., Dai, F., Radopoulou, S. -. (2012). "Achievements and Challenges in Recognizing and Reconstructing Civil Infrastructure." *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics),* 7474 LNCS, 151-176.

Brilakis, I., Fathi, H., Rashidi, A. (2011). "Progressive 3D Reconstruction of Infrastructure with Videogrammetry." *Automation in Construction,* 20(7), 884-895.

Brilakis, I., Soibelman, L., Shinagawa, Y. (2005). "Material-Based Construction Site Image Retrieval." *Journal of Computing in Civil Engineering,* 19(4), 341-355.

Brilakis, I. K., and Soibelman, L. (2008). "Shape-Based Retrieval of Construction Site Photographs." *Journal of Computing in Civil Engineering,* 22(1), 14-20.

Brilakis, I., Lourakis, M., Sacks, R., Savarese, S., Christodoulou, S., Teizer, J., Makhmalbaf, A. (2010). "Toward Automated Generation of Parametric BIMs Based on Hybrid Video and Laser Scanning Data." *Advanced Engineering Informatics,* 24(4), 456-465.

Broyden, C. G. (1967). "Quasi-Newton Methods and their Application to Function Minimisation." *Mathematics of Computation,* 368-381.

Burati Jr., J. L., Farrington, J. J., Ledbetter, W. B. (1992). "Causes of Quality Deviations in Design and Construction." *Journal of Construction Engineering and Management,* 118(1), 34-49.

Chen, C., and Hung, Y. -. (1999). "RANSAC-Based DARCES: A New Approach to Fast Automatic Registration of Partially Overlapping Range Images." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 21(11), 1229-1234.

Chen, Y., and Medioni, G. (1992). "Object Modelling by Registration of Multiple Range Images." *Image and Vision Computing,* 10(3), 145-155.

Cheok, G. S., Stone, W. C., Lipman, R., Witzgall, C., Bernal, J. (2001). "Laser scanning for construction metrology." *Proc., Proceedings of the American Nuclear Society 9th International Topical Meeting on Robotics and Remote Systems.*

Chi, S., and Caldas, C. (2012). "Image-Based Safety Assessment: Automated Spatial Safety Risk Identification of Earthmoving and Surface Mining Activities." *Journal of Construction Engineering and Management,* 138(3), 341-351.

Chi, S., and Caldas, C. H. (2011). "Automated Object Identification using Optical Video Cameras on Construction Sites." *Computer-Aided Civil and Infrastructure Engineering,* 26(5), 368-380.

Cho, Y. K., Haas, C. T., Sreenivasan, S. V., Liapi, K. (2004). "Position Error Modeling for Automated Construction Manipulators." *Journal of Construction Engineering and Management,* 130(1), 50-58.

Cho, Y. K., and Haas, C. T. (2003). "Rapid Geometric Modeling for Unstructured Construction Workspaces." *Computer-Aided Civil and Infrastructure Engineering,* 18(4), 242-253.

Cho, Y. K., Haas, C. T., Liapi, K., Sreenivasan, S. V. (2002). "A Framework for Rapid Local Area Modeling for Construction Automation." *Automation in Construction,* 11(6), 629-641.

Cho, Y. K., Wang, C., Tang, P., Haas, C. T. (2012). "Target-Focused Local Workspace Modeling for Construction Automation Applications." *Journal of Computing in Civil Engineering,* 26(5), 661-670.

Chow, C. K., Tsui, H. T., Lee, T. (2004). "Surface Registration using a Dynamic Genetic Algorithm." *Pattern Recognition*, 37(1), 105-117.

Chua, C. S., and Jarvis, R. (1997). "Point Signatures: A New Representation for 3D Object Recognition." *International Journal of Computer Vision*, 25(1), 63-85.

Chung, D. H., Yun, I. D., Lee, S. U. (1998). "Registration of Multiple-Range Views using the Reverse-Calibration Technique." *Pattern Recognition,* 31(4), 457-464.

CII. (2011). "A Guide to Construction Rework Reduction." *Construction Industry Institute (CII) Implementation Research Team 252*.

Corsini, M., Cignoni, P., Scopigno, R. (2012). "Efficient and Flexible Sampling with Blue Noise Properties of Triangular Meshes." *Visualization and Computer Graphics, IEEE Transactions On*, 18(6), 914-924.

Craig, J. J. (1989). "Inverse Manipulator Kinematics." *Introduction to Robotics Mechanics and Control,* Second Ed., Addison Wesley, 113.

Dai, F., Rashidi, A., Brilakis, I., Vela, P. (2013). "Comparison of Image-Based and Time-of-Flight-Based Technologies for Three-Dimensional Reconstruction of Infrastructure." *Journal of Construction Engineering and Management*, 139(1), 69-79.

Denavit, J., and Hartenberg, R. S. (1955). "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices." *Journal of Applied Mechanics*, 215-221.

Dennis, J., J., and Moré, J. (1977). "Quasi-Newton Methods, Motivation and Theory." *SIAM Review,* 19(1), 46-89.

Ding, L., Zhou, Y., Akinci, B. (2014). "Building Information Modeling (BIM) Application Framework: The Process of Expanding from 3D to Computable nD." *Automation in Construction,* 46, 82-93.

Ditlevsen, O., and Bjerager, P. (1986). "Methods of Structural Systems Reliability." *Structural Safety,* 3(3), 195-229.

El-Omari, S., and Moselhi, O. (2008). "Integrating 3D Laser Scanning and Photogrammetry for Progress Measurement of Construction Work." *Automation in Construction,* 18(1), 1-9.

FARO. (2014a). "Designed for high performance: The FARO laser scanner LS." <http://www2.faro.com/FaroIP/Files/File/Techsheets%20Download/SEA_Laserscanner840&880.pdf> (April 8, 2014).

FARO. (2014b). "FARO laser scanner LS 840/880." <http://www2.faro.com/FaroIP/Files/File/Techsheets%20Download/IN_LS880.pdf> (April 8, 2014).

FARO. (2008). "FARO SCENE." <http://www.faro.com/downloads/files/scene/E1020_SCENE_5.1_Manual_EN.pdf> (3/6/2013, 2013).

Fathi, H., and Brilakis, I. (2013). "A Videogrammetric as-Built Data Collection Method for Digital Fabrication of Sheet Metal Roof Panels." *Advanced Engineering Informatics*, 27(4), 466-476.

Fest, E., Shea, K., Smith, I. (2004). "Active Tensegrity Structure." *Journal of Structural Engineering*, 130(10), 1454-1465.

Garcia, C., and Li, T. (1980). "On the Number of Solutions to Polynomial Systems of Equations." *SIAM Journal on Numerical Analysis*, 17(4), 540-546.

Godin, G., Laurendeau, D., Bergevin, R. (2001). "A method for the registration of attributed range images." *Proc., 3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference On*, 179-186.

Golparvar-Fard, M., Bohn, J., Teizer, J., Savarese, S., Peña-Mora, F. (2011). "Evaluation of Image-Based Modeling and Laser Scanning Accuracy for Emerging Automated Performance Monitoring Techniques." *Automation in Construction*, 20(8), 1143-1155.

Golparvar-Fard, M., Pena-Mora, F., Savarese, S. (2009a). "D4AR – A 4-Dimensional Augmented Reality Model for Automating Construction Progress Monitoring Data Collection, Processing and Communication." *Journal of Information Technology in Construction (ITCon)*, 14, 129-153.

Golparvar-Fard, M., Pena-Mora, F., Arboleda, C. A., Lee, S. (2009b). "Visualization of Construction Progress Monitoring with 4D Simulation Model Overlaid on Time-Lapsed Photographs." *Journal of Computing in Civil Engineering*, 23(6), 391-404.

Gong, J., and Caldas, C. H. (2011). "An Object Recognition, Tracking, and Contextual Reasoning-Based Video Interpretation Method for Rapid Productivity Analysis of Construction Operations." *Automation in Construction,* 20(8), 1211-1226.

Greenspan, M., and Godin, G. (2001). "A nearest neighbor method for efficient ICP." *Proc., 3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference On,* 161-168.

Gruen, A., and Akca, D. (2005). "Least Squares 3D Surface and Curve Matching." *ISPRS Journal of Photogrammetry and Remote Sensing,* 59(3), 151-174.

Haas, C. (1996). "Evolution of an Automated Crack Sealer: A Study in Construction Technology Development." *Automation in Construction,* 4(4), 293-305.

Haas, C., Skibniewski, M., Budny, E. (1995). "Robotics in Civil Engineering." *Microcomputers in Civil Engineerings,* 10(5), 371-381.

Haas, C. T., and Fagerlund, W. R. (2002). *Preliminary Research on Prefabrication, Pre-Assembly, Modularization and Off-Site Fabrication in Construction,* Construction Industry Institute (CII), University of Texas at Austin, Austin, Texas.

Han, S., Al-Hussein, M., Hasan, S., Gökçe, K. U., Bouferguene, A. (2012a). "Simulation of mobile crane operations in 3D space." *Proc., Proceedings - Winter Simulation Conference.*

Han, S. H., Al-Hussein, M., Al-Jibouri, S., Yu, H. (2012b). "Automated Post-Simulation Visualization of Modular Building Production Assembly Line." *Automation in Construction,* 21(0), 229-236.

Hegazy, T. (1999a). "Optimization of Construction Time - Cost Trade-Off Analysis using Genetic Algorithms." *Canadian Journal of Civil Engineering,* 26(6), 685-697.

Hegazy, T. (1999b). "Optimization of Resource Allocation and Leveling using Genetic Algorithms." *Journal of Construction Engineering and Management,* 125(3), 167-175.

Hegazy, T., Said, M., Kassab, M. (2011). "Incorporating Rework into Construction Schedule Analysis." *Automation in Construction,* 20(8), 1051-1059.

Housner, G., Bergman, L., Caughey, T., Chassiakos, A., Claus, R., Masri, S., Skelton, R., Soong, T., Spencer, B., Yao, J. (1997). "Structural Control: Past, Present, and Future." *Journal of Engineering Mechanics,* 123(9), 897-971.

Hu, D., and Mohamed, Y. (2012). "Pipe Spool Fabrication Sequencing by Automated Planning." American Society of Civil Engineers, 495-504.

Hu, D., and Mohamed, Y. (2014). "A Dynamic Programming Solution to Automate Fabrication Sequencing of Industrial Construction Components." *Automation in Construction*, 40(0), 9-20.

Huang, Z., Li, Q., Ding, H. (2012). *Theory of Parallel Mechanisms,* Springer Science & Business Media.

Hwang, B., Thomas, S. R., Haas, C. T., Caldas, C. H. (2009). "Measuring the Impact of Rework on Construction Cost Performance." *Journal of Construction Engineering and Management*, 135(3), 187-198.

Jahanshahi, M. R., Masri, S. F., Sukhatme, G. S. (2011). "Multi-Image Stitching and Scene Reconstruction for Evaluating Defect Evolution in Structures." *Structural Health Monitoring,* 10(6), 643-657.

Jaw, J. J., and Chuang, T. Y. (2008). "Registration of Ground-Based LiDAR Point Clouds by Means of 3D Line Features." *Journal of the Chinese Institute of Engineers, Transactions of the Chinese Institute of Engineers,Series A/Chung-Kuo Kung Ch'Eng Hsuch K'An,* 31(6), 1031-1045.

Johnson, A. E., and Hebert, M. (1998). "Surface Matching for Object Recognition in Complex Three-Dimensional Scenes." *Image and Vision Computing*, 16(9–10), 635-651.

Johnson, A. E. (1997). "Spin-Images: A Representation for 3D Surface Matcing".

Josephson, P., and Hammarlund, Y. (1999). "The Causes and Costs of Defects in Construction: A Study of Seven Building Projects." *Automation in Construction*, 8(6), 681-687.

Kamat, V. R., Martinez, J. C., Fischer, M., Golparvar-Fard, M., Pea-Mora, F., Savarese, S. (2011). "Research in Visualization Techniques for Field Construction." *Journal of Construction Engineering and Management,* 137(10), 853-862.

Kamat, V. R., and Martinez, J. C. (2003). "Validating Complex Construction Simulation Models using 3D Visualization." *Systems Analysis Modelling Simulation,* 43(4), 455-467.

Kamat, V. R., and Martinez, J. C. (2001). "Visualizing Simulated Construction Operations in 3D." *Journal of Computing in Civil Engineering,* 15(4), 329-327.

Kang, Y., O'Brien, W. J., Thomas, S., Chapman, R. E. (2008). "Impact of Information Technologies on Performance: Cross Study Comparison." *Journal of Construction Engineering and Management*, 134(11), 852-863.

Kassimali, A. (1999). "Member Fixed-End Forces due to Fabrication Errors." *Matrix Analysis of Structures,* Pacific Grove, CA: Brooks/Cole, Pacific Grove, CA, 394-410.

Kim, C., Kim, C., Son, H. (2013a). "Automated Construction Progress Measurement using a 4D Building Information Model and 3D Data." *Automation in Construction*, 31, 75-82.

Kim, C., Son, H., Kim, C. (2013b). "Fully Automated Registration of 3D Data to a 3D CAD Model for Project Progress Monitoring." *Automation in Construction*, 35, 587-594.

Kim, C., Son, H., Kim, C. (2013c). "Knowledge-based approach for 3D reconstruction of as-built industrial plant models from laser-scan data." *Proc., ISARC 2013 - 30th International Symposium on Automation and Robotics in Construction and Mining, Held in Conjunction with the 23rd World Mining Congress*, 885-893.

Kim, Y. S., Haas, C. T., Greer, R. (1998a). "Path Planning for Machine Vision Assisted, Teleoperated Pavement Crack Sealer." *Journal of Transportation Engineering*, 124(2), 137-143.

Kim, Y., Haas, C. T., Greer, R. (1998b). "Man-machine balanced crack sealing process for UT automated road maintenance machine." *Proc., Proceedings of the International Conference on Applications of Advanced Technologies in Transportation Engineering*, 114-123.

Kjer, H., and Wilm, J. (2010). "Evaluation of Surface Registration Algorithms for PET Motion Correction." Technical University of Denmark, Bachelor Thesis.

Koch, C., Jog, G., Brilakis, I. (2013). "Automated Pothole Distress Assessment using Asphalt Pavement Video Data." *Journal of Computing in Civil Engineering*, 27(4), 370-378.

Koch, C., and Brilakis, I. (2011). "Pothole Detection in Asphalt Pavement Images." *Advanced Engineering Informatics*, 25(3), 507-515.

Kwon, S., Bosche, F., Kim, C., Haas, C. T., Liapi, K. A. (2004). "Fitting Range Data to Primitives for Rapid Local 3D Modeling using Sparse Range Point Clouds." *Automation in Construction*, 13(1), 67-81.

Lee, J., Son, H., Kim, C., Kim, C. (2013). "Skeleton-Based 3D Reconstruction of as-Built Pipelines from Laser-Scan Data." *Automation in Construction,* 35, 199-207.

Lei, Z., Taghaddos, H., Hermann, U., Al-Hussein, M. (2013). "A Methodology for Mobile Crane Lift Path Checking in Heavy Industrial Projects." *Automation in Construction,* 31, 41-53.

Li, T. (1997). "Numerical Solution of Multivariate Polynomial Systems by Homotopy Continuation Methods." *Acta Numerica,* 6, 399-436.

Liu, X., and Wang, J. (2013). *Parallel Kinematics: Type, Kinematics, and Optimal Design,* Springer Science & Business Media.

Liu, Y., Cho, S., Spencer Jr, B., Fan, J. (2014). "Concrete Crack Assessment using Digital Image Processing and 3D Scene Reconstruction." *Journal of Computing in Civil Engineering.*

Love, P. (2002). "Influence of Project Type and Procurement Method on Rework Costs in Building Construction Projects." *Journal of Construction Engineering and Management,* 128(1), 18-29.

Love, P. E. D., Irani, Z., Edwards, D. J. (2004). "A Rework Reduction Model for Construction Projects." *IEEE Transactions on Engineering Management,* 51(4), 426-440.

Lowe, D. G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints." *International Journal of Computer Vision,* 60(2), 91-110.

Masuda, T. (2001). "Generation of geometric model by registration and integration of multiple range images." *Proc., 3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference On,* 254-261.

McPhee, J. J. (1996). "On the use of Linear Graph Theory in Multibody System Dynamics." *Nonlinear Dynamics,* 9(1-2), 73-90.

Memarzadeh, M., Golparvar-Fard, M., Niebles, J. C. (2013). "Automated 2D Detection of Construction Equipment and Workers from Site Video Streams using Histograms of Oriented Gradients and Colors." *Automation in Construction.*

Memon, Z. A., Abd.Majid, M. Z., Mustaffar, M. (2005). "An automatic project progress monintoring model by integrating auto CAD and digital photos." *Proc., Proceedings of the 2005 ASCE International Conference on Computing in Civil Engineering,* 1605-1617.

Milberg, C., and Tommelein, I. D. (2005). "Application of tolerance mapping in AEC systems." *Proc., Proceedings Construction Research Congress.*

Milberg, C., and Tommelein, I. D. (2003). "Application of tolerance analysis and allocation in work structuring: Partition wall case." *Proc., Proceedings of the 11th Annual Conference of the International Group for Lean Construction.*

Moghadam, M., Al-Hussein, M., Al-Jibouri, S., Telyas, A. (2012). "Post Simulation Visualization Model for Effective Scheduling of Modular Building Construction." *Canadian Journal of Civil Engineering,* 39(9), 1053-1061.

Nahangi, M., Haas, C. T., West, J., Walbridge, S. (2015a). "Automatic Realignment of Defective Assemblies using an Inverse Kinematics Analogy." *ASCE Journal of Computing in Civil Engineering.*

Nahangi, M., Yeung, J., Haas, C. T., Walbridge, S., West, J. (2015b). "Automated Assembly Discrepancy Feedback using 3D Imaging and Forward Kinematics." *Automation in Construction,* 56, 36-46.

Nahangi, M., Safa, M., Shahi, A., Haas, C. (2014). "Automated registration of 3D point clouds with 3D CAD models for remote assessment of staged fabrication." *Proc., ASCE Construction Research Congress,* American Society of Civil Engineers, Atlanta, GA, 1004-1013.

Nahangi, M., and Haas, C. T. (2014). "Automated 3D Compliance Checking in Pipe Spool Fabrication." *Advanced Engineering Informatics,* 28(4), 360-369.

O'Brien, W. J., Hurley, M. J., Mondragon Solis, F. A., Nguyen, T. (2011). "Cognitive Task Analysis of Superintendent's Work: Case Study and Critique of Supporting Information Technologies." *Electronic Journal of Information Technology in Construction,* 16, 529-556.

Park, H. S., Lee, H. M., Adeli, H., Lee, I. (2007). "A New Approach for Health Monitoring of Structures: Terrestrial Laser Scanning." *Computer-Aided Civil and Infrastructure Engineering,* 22(1), 19-30.

Patel, Y., and George, P. (2012). "Parallel Manipulators Applications—a Survey." *Modern Mechanical Engineering,* 2(03), 57.

Patterson, L., and Ledbetter, W. B. (1989). "Cost of quality. A management tool". 100-105.

Penrose, R. (1955). "A generalized inverse for matrices." *Proc., Mathematical Proceedings of the Cambridge Philosophical Society,* Cambridge University Press, 406-413.

Rabbani, T., Dijkman, S., van den Heuvel, F., Vosselman, G. (2007). "An Integrated Approach for Modelling and Global Registration of Point Clouds." *ISPRS Journal of Photogrammetry and Remote Sensing,* 61(6), 355-370.

Ray, S. J., and Teizer, J. (2012). "Real-Time Construction Worker Posture Analysis for Ergonomics Training." *Advanced Engineering Informatics,* 26(2), 439-455.

Rusinkiewicz, S., and Levoy, M. (2001). "Efficient variants of the ICP algorithm." *Proc., 3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference On,* 145-152.

Salvi, J., Matabosch, C., Fofi, D., Forest, J. (2007). "A Review of Recent Range Image Registration Methods with Accuracy Evaluation." *Image and Vision Computing,* 25(5), 578-596.

Sarah Slaughter, E., and Eraso, M. (1997). "Simulation of Structural Steel Erection to Assess Innovations." *IEEE Transactions on Engineering Management,* 44(2), 196-207.

Sharma, V., Al-Hussein, M., Abourizk, S. M. (2006). "Residential Construction Lot Grading Approval Process Optimization: Case Study of City of Edmonton." *Journal of Construction Engineering and Management,* 132(12), 1225-1233.

Sharp, G. C., Lee, S. W., Wehe, D. K. (2004). "Multiview Registration of 3D Scenes by Minimizing Error between Coordinate Frames." *Pattern Analysis and Machine Intelligence, IEEE Transactions On,* 26(8), 1037-1050.

Simpeh, E. K., Ndihokubwayo, R., Love, P. E. (2012). "Evaluating the direct and indirect costs of rework." *Proc., 8th ICEC World Congress,* Durban.

Singh, S. (1997). "State of the Art in Automation of Earthmoving." *Journal of Aerospace Engineering,* 10(4), 179-188.

Sommese, A. J., Verschelde, J., Wampler, C. W. (2004). "Advances in Polynomial Continuation for Solving Problems in Kinematics." *Journal of Mechanical Design,* 126(2), 262-268.

Son, H., Kim, C., Kim, C. (2015). "3D Reconstruction of as-Built Industrial Instrumentation Models from Laser-Scan Data and a 3D CAD Database Based on Prior Knowledge." *Automation in Construction,* 49, 193-200.

Son, H., Kim, C., Kim, C. (2014a). "Fully Automated as-Built 3D Pipeline Extraction Method from Laser-Scanned Data Based on Curvature Computation." *Journal of Computing in Civil Engineering*, B4014003.

Son, H., Kim, C., Kim, C. (2013). "Fully automated as-built 3D pipeline segmentation based on curvature computation from laser-scanned data." *Proc., Computing in Civil Engineering - Proceedings of the 2013 ASCE International Workshop on Computing in Civil Engineering*, 765-772.

Son, H., Kim, C., Kim, C. (2014b). "3D Reconstruction of as-Built Industrial Instrumentation Models from Laser-Scan Data and a 3D CAD Database Based on Prior Knowledge." *Automation in Construction*, (0).

Song, J., Fagerlund, W. R., Haas, C. T., Tatum, C. B., Vanegas, J. A. (2005). "Considering Prework on Industrial Projects." *Journal of Construction Engineering and Management*, 131(6), 723-733.

Spencer, B., and Nagarajaiah, S. (2003). "State of the Art of Structural Control." *Journal of Structural Engineering*, 129(7), 845-856.

Talmaki, S., Kamat, V. R., Saidi, K. (2015). "Feasibility of Real-Time Graphical Simulation for Active Monitoring of Visibility-Constrained Construction Processes." *Engineering with Computers*, 31(1), 29-49.

Tang, P., and Akinci, B. (2012). "Formalization of Workflows for Extracting Bridge Surveying Goals from Laser-Scanned Data." *Automation in Construction*, 22, 306-319.

Tang, P., Huber, D., Akinci, B. (2011). "Characterization of Laser Scanners and Algorithms for Detecting Flatness Defects on Concrete Surfaces." *Journal of Computing in Civil Engineering*, 25(1), 31-42.

Tang, P., Huber, D., Akinci, B., Lipman, R., Lytle, A. (2010). "Automatic Reconstruction of as-Built Building Information Models from Laser-Scanned Point Clouds: A Review of Related Techniques." *Automation in Construction*, 19(7), 829-843.

Taylor, S. (2009). "Offsite production in the UK Construction industry – prepared by HSE." <http://www.buildoffsite.com/downloads/off-site_production_june09.pdf> (06/26, 2013).

Teizer, J., Allread, B. S., Mantripragada, U. (2010). "Automating the Blind Spot Measurement of Construction Equipment." *Automation in Construction*, 19(4), 491-501.

Teizer, J. (2008). "3D Range Imaging Camera Sensing for Active Safety in Construction." *Electronic Journal of Information Technology in Construction*, 13, 103-117.

Teizer, J., Bosche, F., Caldas, C. H., Haas, C. T., Liapi, K. A. (2005a). "Real-time, three-dimensional object detection and modeling in construction." *Proc., 22nd International Symposium on Automation and Robotics in Construction, ISARC 2005.*

Teizer, J., Kim, C., Haas, C. T., Liapi, K. A., Caldas, C. H. (2005b). "Framework for Real-Time Three-Dimensional Modeling of Infrastructure." *Transportation Research Record*, (1913), 177-186.

Teizer, J., Liapi, K. A., Caldas, C. H., Haas, C. T. (2005c). "Experiments in real-time spatial data acquisition for obstacle detection." *Proc., Construction Research Congress 2005: Broadening Perspectives - Proceedings of the Congress*, 1099-1108.

Tractel. (2015). "Portable wire rope winch for material handling." <http://www.tractel.com/en/series.php?id_serie=47> (March 2015, 2015).

Trimble. (2006). "Trimble RealWorks Survey User Guide Version 6.0".

Tsai, L. (1999). *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*, John Wiley & Sons.

Turkan, Y., Bosche, F., Haas, C. T., Haas, R. (2012). "Automated Progress Tracking using 4D Schedule and 3D Sensing Technologies." *Automation in Construction*, 22(0), 414-421.

Uchida, T., and McPhee, J. (2011). "Triangularizing Kinematic Constraint Equations using Gröbner Bases for Real-Time Dynamic Simulation." *Multibody System Dynamics*, 25(3), 335-356.

Vanden Wyngaerd, J., and Van Gool, L. (2002). "Automatic Crude Patch Registration: Toward Automatic 3D Model Building." *Computer Vision and Image Understanding*, 87(1-3), 8-26.

Velamati, S. (2012). *Feasibility, Benefits and Challenges of Modular Construction in High Rise Development in the United States: A Developer's Perspective*, Massachusetts Institute of Technology, Boston, US.

Wampler, C., Morgan, A., Sommese, A. (1990). "Numerical Continuation Methods for Solving Polynomial Systems Arising in Kinematics." *Journal of Mechanical Design*, 112(1), 59-68.

Wang, P., and Abourizk, S. M. (2009). "Large-Scale Simulation Modeling System for Industrial Construction." *Canadian Journal of Civil Engineering*, 36(9), 1517-1529.

Wang, X., Truijens, M., Hou, L., Wang, Y., Zhou, Y. (2014). "Integrating Augmented Reality with Building Information Modeling: Onsite Construction Process Controlling for Liquefied Natural Gas Industry." *Automation in Construction*, 40, 96-105.

Weisstein, E. W. (2014). "Moore-penrose matrix inverse." <http://mathworld.wolfram.com/Moore-PenroseMatrixInverse.html> (July 10, 2014).

Xiong, X., Adan, A., Akinci, B., Huber, D. (2013). "Automatic Creation of Semantically Rich 3D Building Models from Laser Scanner Data." *Automation in Construction*, 31, 325-337.

Yang, J., Arif, O., Vela, P. A., Teizer, J., Shi, Z. (2010). "Tracking Multiple Workers on Construction Sites using Video Cameras." *Advanced Engineering Informatics*, 24(4), 428-434.

Yates, J. K., and Lockley, E. E. (2002). "Documenting and Analyzing Construction Failures." *Journal of Construction Engineering and Management*, 128(1), 8-17.

Zhu, Z., and Brilakis, I. (2010). "Machine Vision-Based Concrete Surface Quality Assessment." *Journal of Construction Engineering and Management*, 136(2), 210-218.

Zitová, B., and Flusser, J. (2003). "Image Registration Methods: A Survey." *Image and Vision Computing*, 21(11), 977-1000.