Chapter 4

LAYOUT DESIGN OF ROBOTIC WORKCELL

This chapter presents the first application of the point cloud simulation approach described in the previous chapter. The layout design of a workcell is the first step for starting the cellular manufacturing operation in the industry. The point cloud approach optimizes the layout of the existing industrial robotic workcell and can also generate layout design for new robotic workcell. The layout optimization of a single robotic workcell involves several parameters which interact with each other sophisticatedly during operation. The arrangement of machines and robots directly or indirectly affects the efficiency of the workcell. Various factors such as cost, energy, time and smooth manipulator motion can be optimized by the efficient arrangement of machines in the workcell. Thus, the design of robotic workcell layout influences the overall productivity of the industry.

The procedure of generating the point cloud models of the workcell objects has been described in the previous chapter, and the utilization of point clouds to optimize the layout design of the robotic workcell has been demonstrated in this chapter. Section 4.1 gives an introduction to the layout design and brief overview of the approaches regarding the topic. Section 4.2 formulate the layout design problem for the industrial robotic workcell took in this study. Section 4.3 presents the results generated by solving the formulated problem and discusses some salient point from the results. Finally, Section 4.4 concludes the chapter and summarizes the layout design process.

4.1. Introduction

To sustain in the global market, the current situation demands cellular manufacturing system in industries to become more flexible and simultaneously improve the product design to enhance its value. Consequently, industries are in need to adopt flexible and accurate planning methods which can accept the design and manufacturing changes readily (Izui *et al.*, 2013; Mahr, 2000). Layout designing of a robotic workcell is a significant part of cellular manufacturing system, which has been investigated in several ways over the past few decades.

Several robotic workcell layout designing algorithms and software are currently available which comprise of realistic simulation in the virtual reality environment (Dawande *et al.*, 2007; Lee and Elmaraghy, 1990; Pires and Costa, 2000). However, the commercial layout designing packages are non-generic and sophisticated; therefore, layout design of robotic workcell becomes difficult for small and medium scale enterprises.

Also, the graphical simulation by computer-aided robotic software such as RobotStudio and several others need highly skilled operators, e.g., an integrated technique for simulating a manufacturing system involves various sophisticated devices and skilled persons (Dangelmaiera *et al.*, 2005; Connolly, 2009). In consequence, industrial robotic workcell has been designed by utilizing conventional layout designing method (Tay and Ngoi, 1996) and unable to meet the market competition. Thus, a generic approach for designing the layout of an industrial robotic workcell is indeed required, providing optimal solution comparable with commercial packages.

Further, the mathematical complexity involved in modeling the workcell objects compels the designer to adopt several assumptions and in turn compromises with the accuracy of the generated solutions. For instance, the series of identical spheres placed adjacently to each other models the entire shape of the machine, while a line segment passing through the center of each link resembles the robot model. Moreover, the collision between robot and machine has been presented as the intersection between a line and a circle.

Similarly, the intersection between two circles describes the overlapping between two machines (Mata and Tubaileh, 1998). The collision detection problem in a workcell has also been identified by Bosscher and Hedman (2011). The above mentioned mathematical complexity issue in the workcell layout design has been taken in the indigenously developed point cloud simulation approach and a suitable solution has been presented.

The CAD based planning methods have been utilized to solve several industrial problems such as path planning (Fang *et al.*, 2016), assembly (Michalos *et al.,* 2015) and painting (Chen *et al.,* 2009). CAD based approaches has also been used for the path planning of the robotic manipulator and the task-based motion planning of workcell. The CAD-based layout designing and optimization require an extended understanding of CAD packages. The two readily used CAD packages in industries: AutoCAD and SolidWorks, have an Application Programming Interface (API). The API provides an environment, by which the geometrical data available in CAD model can be processed as per programming done using computer languages such as Matlab, Visual Basic, Visual C++, and Visual C# (Neto and Mendes, 2013; Baizid *et al.*, 2016). This data extraction process is a part of the digital data acquisition techniques, which has also been used in planning and simulation of the assembly operation (Ming *et al.*, 2013).

The point cloud simulation approach works on point cloud models as an input that are developed by using their CAD models. Then, all the point cloud models have been used in an algorithm to optimize their location to produce the layout of a robotic workcell. The point cloud model can be developed in several ways and this approach can be developed in higher level programming languages. In this thesis work, Matlab is used which is a higher-level programming language in which meshing, simulation and optimization is possible in one package.

Regarding optimization, usually, the stochastic algorithms such as simulated annealing and genetic algorithm has been used for solving most of the layout optimization problems. The non-linear optimization problem requires large search domain for which exhaustive search techniques should be used. Barral *et al.* (2001) developed a modified simulated annealing algorithm to design the layout of a robotic workcell and Kazuhiro *et al.* (2013) solved the multi-objective layout optimization problem for the cellular manufacturing system using genetic algorithm. However, the point cloud simulation approach can be clubbed with any optimization algorithm according to the formulated problem.

In this chapter, the point cloud simulation approach is used to optimize the layout of a single robot workcell involving modeling, simulation and optimization in realistic environment. The data used in this chapter is collected from the robotic workcell in foundry shop having four machines and one robot. This data is utilized to formulate a minimization problem subjected to several constraints as explained further.

4.2. Problem Formulation

The layout design problem depends on several factors such as objective function, point cloud data of machines and robot, kinematic limits of robot joints and imposed constraints. The simulation of complex collision situations requires the topological information of the robot and the surrounding machines. By arranging point clouds in workcell, the simulation of realistic working situations can be possible. The aim of the layout design problem is to search the optimal location and orientation of machines in the configuration space of the robot. In this search, job point of the point cloud machines acts as variable. The location of machines is defined by the location of its job point and their point cloud model transforms according to it. The orientation is defined as the angle between x-axis of the Cartesian coordinate frame and the x-axis of job point.

Considering, a robotic manipulator having *n* degree-of-freedom fixed at the origin of Cartesian coordinate frame. The robot passes through a sequence of *N* machines, located by *N* job points denoted by $x^j \in R^p$ and their orientation is denoted by α^j, where p is the dimension of the robot's workspace and $j = 1$ *to N*. The problem has been solved in joint configuration space and the joint values corresponding to each job point acts as variable. The optimal location of the *j*th machine in world coordinate frame which corresponds to *j*th robot configuration has been calculated by using forward kinematics in RVC toolbox having robot joint coordinate vectors *q(j)*. In general, the optimization criteria for the layout design concerning the motion of the robotic manipulator are minimum time, cost and distance. In the present problem formulation, the objective function is based on the minimization of the total distance traveled by the robot end-effector as a function of its joint coordinates.

While solving the optimal layout design problem large point cloud data of machines and robot have been involved during iteration. These data matrices contain the information of machine's geometry in the form of points in coordinate space which has been utilized in checking the collision. The point to point collision checking increases the iteration time because the optimizer first checks the overlapping between two machines and then examine the collision between each robot link with every machine in the workcell. In a single point cloud there are thousands of points and checking collision between each and every combinations of point to point collision between two clouds is undoubtedly a time taking process.

This problem was resolved by developing conditional equations using the Axis Align Bounding Box method (Ericson, 2004). The collision between two machines in Cartesian coordinate space was checked by calculating the intercepts between their minimum bounding boxes on each coordinate axis. The minimum bounding box of an object is a rectangular parallelepiped-shaped volume in coordinate space within which all points of the point cloud of an object lie.

Objective function:

$$\mathrm{Min}\left\{\sum_{\boldsymbol{i=1}}^{\boldsymbol{N-1}} \sum_{\boldsymbol{j=i+1}}^{\boldsymbol{N}} c_{ij} \cdot f_{ij} \cdot \left|\boldsymbol{q^i} - \boldsymbol{q^j}\right|\right\} \quad \ldots (4.1)$$

Subject to:

$$\vec{\boldsymbol{q}}^{\boldsymbol{j}} = \mathbb{IK}\left(\mathrm{R}, \mathrm{HT}(\vec{\boldsymbol{X}}^{\boldsymbol{j}})\right) \quad j = 1, \ldots\ldots, N \quad \ldots (4.2)$$

$$\boldsymbol{N_{jj+1}} = [\ \{\ ({}_{x}^{j}M_{min} \leq {}_{x}^{j+1}M_{max}) \oplus ({}_{x}^{j}M_{max} \geq {}_{x}^{j+1}M_{min})\ \} \ldots$$
$$\vee \{\ ({}_{y}^{j}M_{min} \leq {}_{y}^{j+1}M_{max}) \oplus ({}_{y}^{j}M_{max} \geq {}_{y}^{j+1}M_{min})\ \} \ldots$$
$$\vee \{\ ({}_{z}^{j}M_{min} \leq {}_{z}^{j+1}M_{max}) \oplus ({}_{z}^{j}M_{max} \geq {}_{z}^{j+1}M_{min})\ \}\ ] \quad j = 1, \ldots, N\text{-}1 \ldots (4.3)$$

$$\boldsymbol{T_{jk}} = [\{\ ({}_{x}^{j}M_{min} \leq {}_{x}^{k}R_{max}) \oplus ({}_{x}^{j}M_{max} \geq {}_{x}^{k}R_{min})\ \} \ldots$$
$$\vee \{\ ({}_{y}^{j}M_{min} \leq {}_{y}^{k}R_{max}) \oplus ({}_{y}^{j}M_{max} \geq {}_{y}^{k}R_{min})\ \} \ldots$$
$$\vee \{\ ({}_{z}^{j}M_{min} \leq {}_{z}^{k}R_{max}) \oplus ({}_{z}^{j}M_{max} \geq {}_{z}^{k}R_{min})\ \}\ ] \quad j = 1,..,N \quad k = 1,..,n \ldots (4.4)$$

Where f_{ij} and c_{ij} correspond to material flow and transportation cost respectively. $\boldsymbol{q^i}$ and $\boldsymbol{q^j}$, are the joint angle values of *i*th and *j*th machine in the workcell. IK is the inverse kinematics of the given robot model evaluated by using the RVC toolbox. HT is the homogeneous transformation of the *j*th machine's job point, $\vec{\boldsymbol{X}}_{\boldsymbol{j}}$. $\boldsymbol{N_{jj+1}}$ and $\boldsymbol{T_{jk}}$ are the intercepts produced due to collision between two point clouds. These intercepts set the clearance between the machines and robot. ${}_{x}^{j}M_{min}$ and ${}_{x}^{j}M_{max}$ are the minimum and the maximum value of *j*th machine minimum bounding box edge along the x-axis, respectively. ${}_{x}^{k}R_{min}$ and ${}_{x}^{k}R_{max}$ are the minimum and maximum value of the minimum bounding box along the x-axis of *k*th link of robot *R,* respectively. Similarly, other values along y and z-axis for *j*th, *j+1*th machine and robot links can be obtained. $\oplus$ is the logical symbol for the *XOR* gate and $\vee$ is the logical symbol for *OR* gate.

The equation 4.1 shows the objective function which takes the difference between the joint configuration of robot at two consecutive machines, along with the transportation and material flow cost between two machines. Equation 4.2 checks whether the allocated point lies within the robot configuration space and limits the optimal solutions within the robot reachability. Equation 4.3 shows the mathematical expression for checking overlapping between two machines; similarly, equation 4.4 represents the collision detection between robot and machines.

The point cloud simulation approach accurately simulates the collision situation during motion by the help of equations 4.3 & 4.4. These equations are the governing equations of the approach. The basic principle behind these equations is that the two point clouds collide with each other only when their bounding boxes are in contact with each other in any direction of the coordinate axis. It means that the range of intercepts of the bounding volume of the two point clouds must be separated by a definite value in order to avoid collision.

The equations 4.3 & 4.4 state that for the collision to occur either between machine and robot or machine and machine, then, simultaneously on each coordinate axis the minimum value of *j*th machine minimum bounding box should be lesser than the maximum value of $j+1$th machine or robot's kth link minimum bounding box. Along with above, the maximum value of jth machine minimum bounding box should be greater than the minimum value of $j+1$th machine or robot kth link minimum bounding box on the same coordinate axis. Thus, both conditions must be checked together to avoid the collision between the point clouds.

The formulated problem is a nonlinear minimization subjected to point cloud bounded constraints. It deals with large point cloud matrix data for which an exhaustive search algorithm is certainly required. For workcell layout design, simulated annealing algorithm is efficient in obtaining the global optimum solution (Cagan *et al.,* 2002). The *simulannealbnd* function of the Global Optimization Toolbox in Matlab is based on simulated annealing algorithm and being used to solve the formulated problem.

4.3. Results & Discussion

The task is to optimally arrange the machines of an industrial robotic workcell around a robot in an arc type robot centric layout and to demonstrate the working methodology of the layout designing process. The layout design problem formulated above has been solved by using the point cloud simulation approach which simulates the realistic collision situation for most efficient workcell design. The formulated problem is then coded in Matlab by developing several functions and scripts using personal computer having i3 processor (2.20 GHz) with 4 GB Ram. Several trials of the optimization algorithm (Fig. 3.8) have been run to obtain the desired optimal solution and discussed further in this section.

Table 4.1 presents the optimal solutions produced by solving the minimization problem and compares the location and orientation of an existing workcell named as Industry Layout (IL) with the optimal solution named as Proposed Layout (PL). The end-effector moves from one machine to another as per the positions mentioned in Table 4.1, and the orientation helps machines to adjust on the available floor area. The table presents the location of job point of respective point cloud model of the machines in Cartesian coordinate space.

Table 4.1. Position and orientation of industry and proposed layout: units [m] & [degree]

Object	***Position***		**Orientation**	
	Industry Layout	Proposed Layout	Industry layout	Proposed layout
1	J1 (-0.25, -2.08)	J1´(1.66, -1.83)	90	88.6
2	J2 (1.19, -2.06)	J2´(2.35, 0.30)	216	142.5
3	J3 (1.19, -0.29)	J3´(1.57, 1.86)	180	186
4	J4 (0.00, 1.33)	J4´(0.39, 2.50)	-90	-98

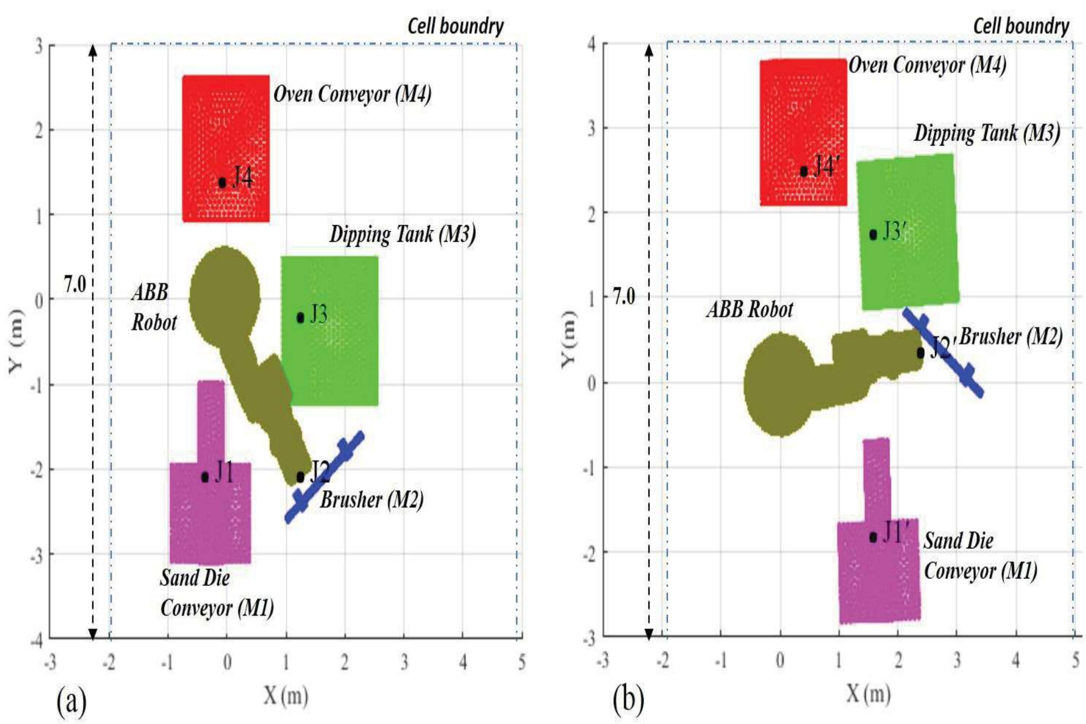

Figure 4.1. Top view of the robotic workcell (a) the industry layout (b) the proposed layout

These job points are also located on the machine in Fig. 4.1. This figure shows the 2D point cloud map of the industry and the proposed layout containing the point cloud models of a robot at the center, surrounded by an arc type arrangement of machines. These point cloud maps are generated by plotting the point cloud data matrices of the machines and robot using the location and orientation values from Table 4.1.

In Fig. 4.1, the boundary of the workcell is also mentioned such that the machines should remain in limits while searching for their optimal location. The boundary limitation adds extra constraint to the problem and restricts several feasible solutions. The point cloud maps in this figure compares the layout of the workcell before and after optimization. Fig. 4.1 (a) is the layout of the robotic workcell obtained from the industrial data which acts as the initial feasible solution to the simulated annealing algorithm.

After performing iteration using point cloud simulation approach the results obtained are shown in Fig. 4.1 (b), in which machines are displaced to new positions and their orientation are also changed. The substantial change in the position and orientation of machine M2 can be noticed, however other machines have been displaced for short distance. Such huge displacement was possibly due to the small size of the machine M2 which can be adjusted in the volume between two adjacent machines. The change in machines' location, following the minimum configuration of robot joints has been calculated from equation 4.1.

Table 4.2. Net joint angular displacement values of motors

Parameter (radian)	***Industry Layout***	***Proposed Layout***
Net Joint Displacement	40.72	16.31

The minimum net angular movement of the robot joints is the optimization criteria, and the result shows that the proposed layout has 59% lesser net joint motion than the industry layout. Table 4.2 presents the net robot joint angular displacement for both layouts. The possible reason for the improvement is evident by comparing the joint angle profile. Fig. 4.2 shows the joint angular profile for the industry layout from (0-24) and the proposed layout from (24-48). It shows the absolute joint angle values plotted v/s joint motor sequence. Each curve contains twenty-four point which are the values of joint angles, six for each machine in the workcell.

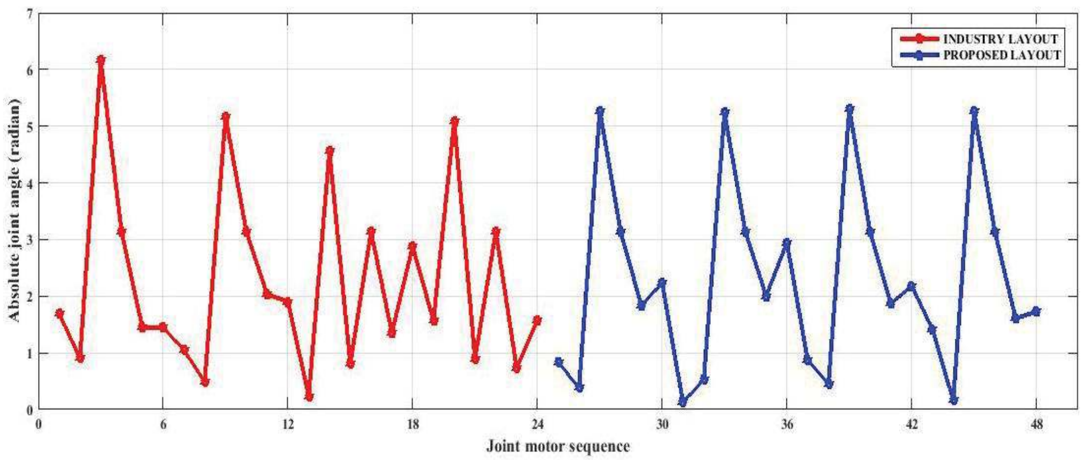

Figure 4.2. Angular values of the joint configuration of industry and proposed layout for single cycle

From Fig. 4.2, it has been observed that the proposed layout has smooth angular profile and uniform joint angle distribution while traversing from one machine to another. However, the industry layout has irregular angular profile and has more fluctuations for overall robot movement. Abrupt changes in angular profile of industry layout shows that most of the joint motors have been engaged between machines M2 to M4. While in the proposed layout from 36 to 48 (on the x-axis), the angular profile is much lower and uniform that ensures lower net motor run for performing the same operation.

Further, the location of the machines in the proposed layout results in the least angular displacement of the joint motors. Further, in industry layout, the machines are placed close to the robot base which leads to high joint movements as evident from Fig. 4.2, as the maximum joint angle of industry layout is higher in comparison with the proposed layout. The machines have been placed very close to robot base, to save the floor space in the industry layout. However, the same floor area in the proposed layout has been optimized in such a manner that there is no congestion near the robot base.

Indeed, the proposed method successfully optimizes the net joint displacement of the robot by rearrangement of machines. Although, a marginal increase in the net Euclidean distance between the machines from 5.22 m to 5.24 m has been observed but it does not change the floor area of the workcell as evident from Fig. 4.1. The simulated annealing algorithm is a stochastic method, and the results produced by iteration may vary each time according to the problem formulation. Therefore, the statistical analysis was performed, and optimization program was executed 30 times in succession. Table 4.3 shows the mean and standard deviation of computation time required to iterate for an optimal solution.

Table 4.3. Computation time

Layout	***Mean (sec)***	***Standard deviation (sec)***
Proposed layout	938.13	10.82

An advantage to develop the proposed layout designing methodology in Matlab is the isometric view of the optimal results as shown in Fig. 4.3. The three-dimensional view of the point cloud map clearly shows the layout design of the workcell in which the orientation, location, geometry and the order of the machines has been observed thoroughly. Any unwanted error occurred in the resulted layout due to CAD and point cloud modeling can be pointed out, and the prerequisite measures can be taken before establishing the workcell. Thus, the point cloud simulation approach presents a generic procedure to design the layout of an industrial robotic workcell with graphical presentation of the optimal results.

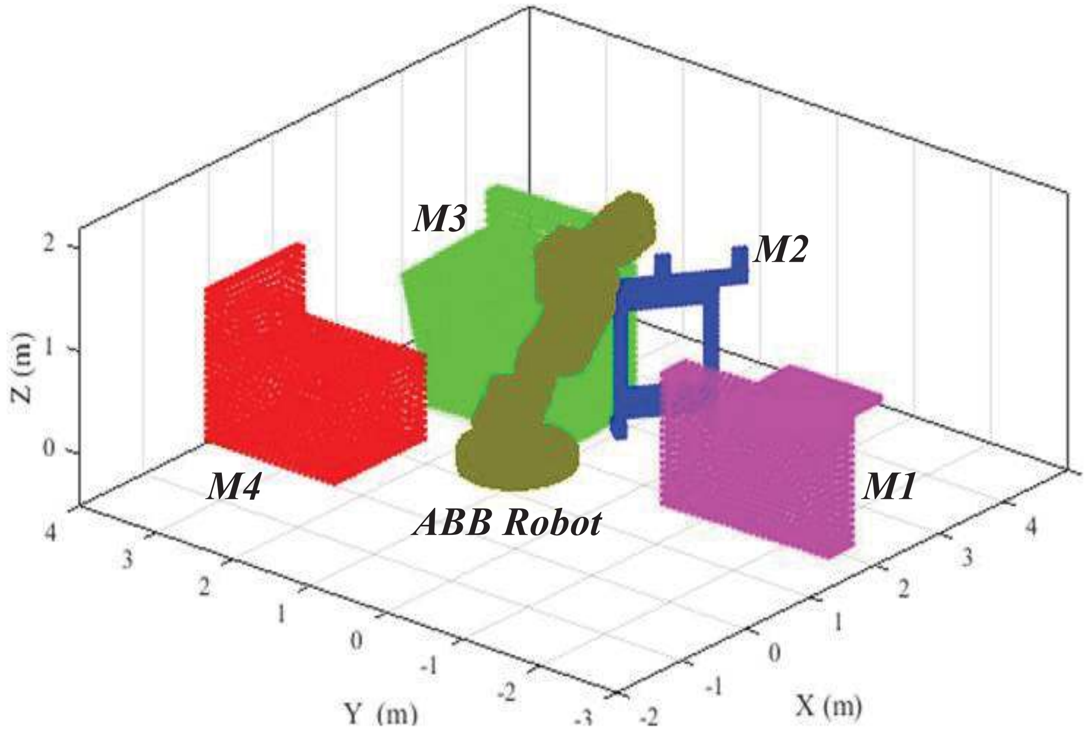

Figure 4.3. Isometric view of the optimal layout of industrial robotic workcell

4.4. Concluding Remarks

The present chapter describes the method to develop the optimal layout design of a robotic workcell, using the point cloud simulation approach. This approach is specifically developed to simulate the realistic working environment in a workcell. In present study, layout design problem in a single robotic workcell has been solved by this approach. The method takes minimum joint displacement as the optimization criterion. About 59 % reduction in the net joint displacement of the manipulator has been achieved, which is a remarkable improvement. However, the negligible increment has been noticed in the total distance travelled by the end-effector. The 3D view of the workcell layout provides a better understanding of the generated results.

Further, the disadvantage of this approach is the generation of the point cloud models of large size machines which are computationally heavy and requires high end processor during optimization. The developed approach provides accurate and calibration less results for the layout design problem and thus ease the industrial workcell establishment and promotes automation in the small-scale industries.

Finally, it is expected that the robotic workcell layout design developed by point cloud simulation approach may serve as a better alternative to other sophisticated methods in this area. This approach has been successfully able to determine the optimal layout under realistic constrained environment.

The next chapter discusses in details the second application of the point cloud simulation approach, development of layout design of multi-robot workcell by transforming single robotic workcells.