Pak. J. Biotechnol. Vol. 15 (Special Issue ICRAME 17) Pp. 75-78 (2018) Ameet Singh et al.,
www.pjbt.org PISSN: 1812-1837, EISSN 2312-7791

OPTIMIZATION OF CYCLE TIME OF PALLETIZATION USING ABB ROBOT THROUGH SIMULATED ANNEALING ALGORITHM AND TRANSLATIONAL APPROACH

Ameet Singh[1], V. Sugumaran[2], G. Balamuruga Mohan Raj[3]

[1,2] *School of Mechanical and Building sciences, VIT University, Chennai, Tamil Nadu, India*
[3]*Dept. of Mechanical Engineering, Sri Manakula Vinayagar Engineering College, Madhagadipet, Pudhucherry, India.* [1]*Ameet.singh2015@vit.ac.in*, [2]*v_sugu@yahoo.com*, [3]*gbmraj@gmail.com*

ABSTRACT

The paper is related to pick and place robotic system (IRB - 1600). The main objective is to minimize the optimum cycle time of the robotic system. To achieve the following objective simulated annealing algorithm (SA) is carried out. Industrial constraint regarding the functioning of real industrial robotic system has been taken in concern during the research practice. Numerical results show that the different algorithm perform optimally for the tested instances in a reasonable computational time. Complete installation and cycle time of the system has been reduced and optimized with proper design and simulation using offline programming.

Keywords: *Cycle time, Metaheuristics, Pick and place robotic system, Simulated annealing algorithm, Offline programming.*

1. INTRODUCTION

Robotic system plays a vital role in automated manufacturing industries. Out of various types of robots, pick and place robot is chosen for the research work. It is defined as the system in which product is collected from conveyor and deposited to its specific location. Recently, robotic pick and place operations have been widely applied on automated equipments and press fabrication industries. The objective to implement robots in automation planet is reducing the task time as well as increasing the productivity rate (Tewolde, et al., 2002).

Two ways of palletization are introduced in the paper *i.e.* online programming and offline programming (OLP). Robotstudio is an interface, which is used here to program the robot, as it uses exact replica of the real robot controller. Virtual controller uses same software which is used to control the robot. Overall safety and risk of collision are the main constraints during offline prog ramming. However, it is advantageous to use robotstudio software tool. One requires 3-D cad models of both parts and apparatuses which will be utilized as a part of cell. These 3-D cad models of parts can be made in any solid modeling software like solidworks or cattia (Andrew swary, 2012). Additionally, by utilization or the created OLP, the administrator can produce robot programs effortlessly and rapidly (Larkin, et al., 2012). Mirlind Bruqi et al., (2013) introduced the power of simulation tools for solving industrial problems. The shift from conventional mass production has accelerated in recent years. Taking into consideration the complexity of the manufacturing systems, the performance of the industrial manipulators can be improved by using optimization techniques (Bobrov, et al., 1985 and Shin, et al., 1986) and offline programming processes (Baizid, et al., 2013) that drives the end product to have high quality and low cost.

Rana and zalzala (1997) applied EA to the collision free path planning of robotic arm. Hiten patel et al., (2014) presented a review study on cycle time reduction in manufacturing industries. Udaykiran et al., (2015) discussed a study on reduction and optimization of installation and cycle time of MIG welding robotic system in automation factory.

The research in this project would result in more efficient and generalized solution. At the end, this leads to less execution time for process and increases the productivity. The method depicts the promising results and an abundant future research in the field of optimization of cycle time.

2. Design Work

Palletizing path planning in manufacturing industries is processed in two ways:

2.1 Offline and online Programming: The offline programming is processed using robotstudio software where the complete installation of the cell is desired in software. The set of coordinates x, y and z- axis for pallets, sheets and boxes respectively are calculated by using offline simulation in robotstudio. The online programming is implemented after the completion of offline programming which is taken as a reference for installation. The robot coordinates *i.e.* (x = 0, y = 0 and z = 0) with reference to other work objects namely, pallet, sheet and infeeder. The electrical connections for powering up the PLC controller and robot controller are to be connected within the reach of operator to operate in critical solutions. The complete station is enclosed in a cell, the offline RAPID program is dumped into flex pendant which is used to operate robot in order to pick and place work objects. The things that have to be considered while installing the station are:

- Height and position of robot
- Wire feeder
- Vacuum gripper
- Reachability to work object

Figure 2: IRB 1600

2.2 Palletizing Powerpac: The is palletizing powerpac (PzPP) is a robotstudio solution for offline programming and simulation of palletizing application, minimum system requirements needed for palletizing.

3. Procedure: Offline program is simulated in robotstudio, the complete station layout is designed and installed, the installation is done with proper positioning of robot and required equipment in positions as mentioned in layout. The main work objects are:

- Infeeder
- Palletfeeder
- Sheetfeeder

4. Optimization: The objective of optimization is to define the optimal relative position between a robotic manipulator and work objects. Here, for this robotic problem optimization is done using simulated annealing algorithm (SA) tool.

4.1 Optimization using SA

4.1.1 Theory of simulated annealing: The annealing process can be defined as follows. We consider a succession of decreasing temperatures, starting from some maximum T_0. We may implement a general simulated annealing algorithm by the following steps.

Step 1: beginning at an initial temperature T_0, we pick an initial set of parameter values with function value *E.*

Step 2: select another point in the parameter space, within a neighborhood of the original, and calculate the corresponding function value.

Step 3: distinguish with the comparison of the two points in terms of their function value, using the metropolis criterion.

Step 4: Repeat steps 2-3. At every phase compare the function value of new points with the function value of the present point until the sequence of accepted points is judged, by some criterion, to have reached an equilibrium state

Step 5: once an equilibrium state has been achieved for a given temperature, the temperature is lowered to a new temperature as defined by the annealing schedule.

4.1.2. Model parameters

C.T. - Cycle time of palletizing process
IA1 - Cycle time of an in feeder
PA2 - Cycle time of a pallet feeder
SA3 - Cycle time of a sheet feeder
Ix1, Iy1, Iz1 - x, y and z - coordinates of in feeder respectively
Px2, Py2, Pz2 - x, y and z - coordinates of pallet feeder respectively
Sx3, Sy3, Sz3 - x, y and z - axis of sheet feeder respectively
N - Population size in genetic algorithm
Pc - Crossover rate of genetic algorithm method
Pm - Mutation rate of genetic algorithm method

5. Mathematical Formulation: An objective function is developed to minimize the cycle time of palletization process. The objective function has a non - linear form, because the feeder positions in workspace affects the cycle time, which is also known as workobject coordinates.

Minimize $C.T. = IA_1 + PA_2 + SA_3.$ (1)

Subject to following bounds

$280\ mm \le Ix_1 \ge 1188\ mm,\ 463\ mm \le Iy_1 \ge 1150\ mm,\ 50\ mm \le Iz_1 \ge 630\ mm.$ (2)

Bound (2) represents the range of values for x, y and z - coordinates of infeeder respectively.

$400\ mm \le Px_2 \ge 1210\ mm,\ -580\ mm \le Py_2 \ge -1220\ mm,\ 0\ mm \le Pz_2 \ge 470\ mm.$ (3)

Bound (3) represents the range of values for x, y and z - coordinates of pallet feeder respectively.

$-350\ mm \le Sx_3 \ge -750\ mm,\ -450\ mm \le Sy_3 \ge -1050\ mm,\ 0\ mm \le Sz_3 \ge 510\ mm$ (4)

Bound (4) represents the range of values for x, y and z - coordinates of sheet feeder respectively.

Velocity of robot = *10000mm/s.*

6. RESULTS AND DISCUSSION

Cycle time reduction involves several steps and it can be achieved by following methods:

- Speed control
- Reducing the air position
- Path position planning
- Fine tuning

- Configuration settings

The values of x, y and z axes for all four feeders workobjects after optimization are:
In feeder: *(900, 463, 150) mm*
Pallet feeder: *(800, -600, 100) mm*
Sheet feeder: *(-400, -650, 100) mm*

6.1 Optimization using simulated annealing algorithm tool: A method for solving both un-constrained and bound-constrained optimization problems is principle of simulated annealing algorithm. The method models the physical process of heating a material and then slowly stepping down the temperature to decrease defects, thus reducing the department of energy. The distance of the new point from the current point, is dependent on probability distribution with a scale proportional to the temperature. The algorithm accepts all new points that tend to minimize the objective with an optimum sequence of results.

Graph plots for results in simulated annealing algorithm:

- *Stopping Criteria (Maximum iterations):* Maximum iterations bound the number of iterations the algorithm takes corresponding to the simulation time.

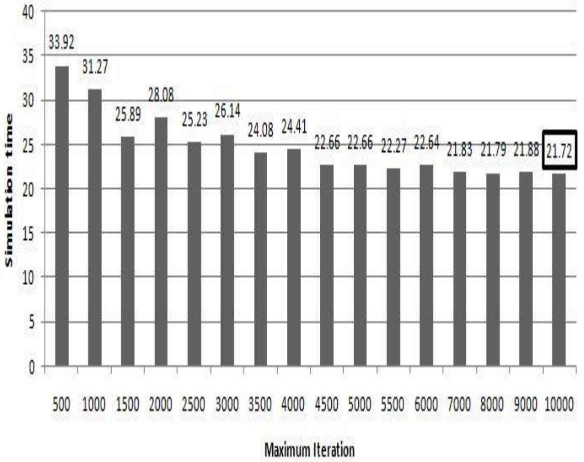

Figure 3: Maximum Iteration vs. Simulation time.

- *Time Limit:* Time limit bounds the number of seconds the algorithm runs.

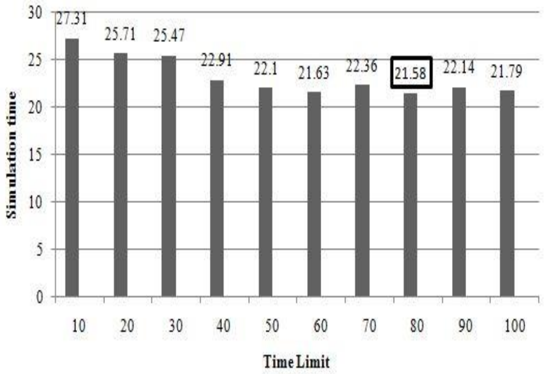

Figure 4: Time limit vs. Simulation time

- *Reannealing interval:* It is described as the number of points to accept before reannealing.

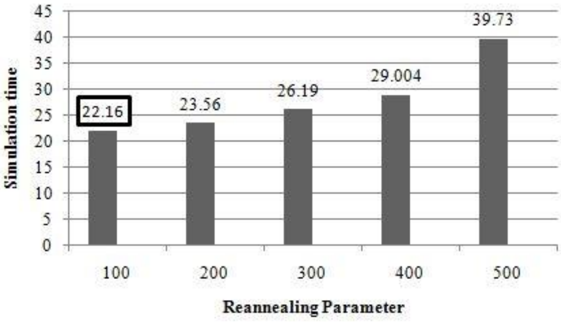

Figure 5: Reannealing Parameter vs. Simulation time

- *Initial Temperature:* It is defined as the temperature before the run.

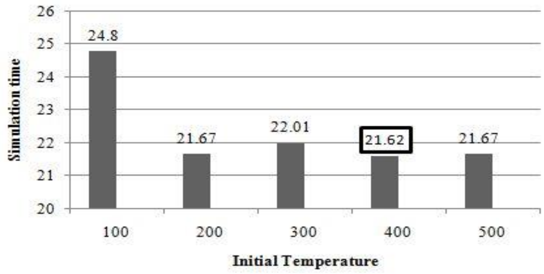

Figure 6: Initial temperature vs. Simulation time.

- *Annealing Parameters:* It specifies the function used to generate new points for the next iteration. Two parameters are included *i.e.* Fast annealing parameter and Boltzmann constant parameter. Fast annealing takes random steps, with size proportional to temperature and hence the result obtained with exponential temperature update is *22.308 milliseconds,* whereas the simulation time with Boltzmann constant is *22.66 milliseconds.* Hence, fast annealing parameters proved to give better results. Moreover, fast annealing parameters with logarithmic temperature update results in *21.87 milliseconds.* At last, same existing parameter with linear temperature update is *26.76 milliseconds.* The best result is chosen, and the results are discussed.
- *Best function*: It plots the lowest objective function to date as it is shown in Figure 7.

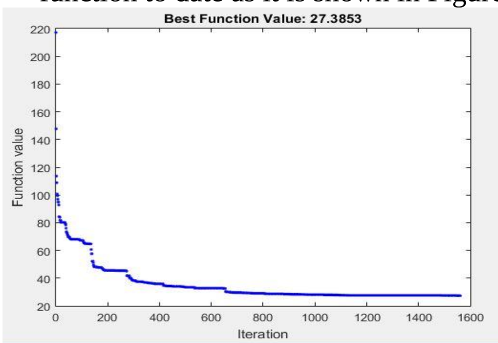

Figure 7: Iteration vs. Function value

- *Best point:* The cycle time of *21.12 milliseconds* is the nearest optimal solution provided by the algorithm with corresponding model parameters and hybrid function parameters as it works as an alternate solver that runs at specified times and fm in search is selected for this unconstrained problem.

7. Conclusion

In food packaging industries and other manufacturing industries the palletizing process is mostly online. Time taken for installation and teaching the path for palletizing directly over a station results in the reduction of cycle time and increase in productivity. Online process increases the cycle time as the palletizing path to be performed will be analyzed only after the product is available online. Hence, path generation of any product using offline programming, would increase the productivity and reduce the cycle time as discussed earlier in results and discussion section i.e. *27.4milliseconds* reduced to *23.4milliseconds.* In conclusion, SA approach is found to be an appropriate method for determining optimal solution for palletizing process. SA approach has a least cycle time of *21.12 milliseconds.* Considering the future research work, obstacle avoidance can be taken into account by extending the proposed algorithm.

REFERENCES

Tewolde, G.G. and W. Sheng, Robot path integration in manufacturing process; Genetic algorithm versus Ant colony optimization. Ieee transaction on systems, mass and cybernetics 38 (2): 278-287 (2002).

Andrew swary, Offline programming and simulation in robotic welding application speeds up programming time, reduces robot downtime, Practical welding today, March 9 (2012).

Larkin N. Milojevic, A. Pan, Z. Polden and J. Norrish, Offline programming for short batch robotic welding, 16th joining of materials (JOM) conference (2012).

Mirlind Bruqi, Optimizing of flexible palletizing lines using simulation tools. Trends in the Development of Machinery and Associated Technology, TMT 24(9): (2013).

Bobrov, J.E., S. Dubowsky and J.S. Gibson, Time optimal control of robotic manipulators along specified paths. Int. J. Robot Res. 4(3): 3-17 (1985).

Shin, K. and N. Mckay, Selection of near minimum time geometric paths for robotic manipulators, IEEE Trans. Automated control 31 (6): 501-511 (1986).

Baizid, K., R. chellali, T. Bentaleb, A. yousnaj and A. meddahi, Virtual reality based tool for optimal robot placement in robotized site based on CAD's application programming interface, Proceeding of virtual reality international 9(4): (2013).

Rana, A. and S. Zalzala, Collision free motion planning of multi-arm robots using evolutionary algorithms, Preceding of the institution of mechanical Engineers part 1, 211: 373-384 (1997).

Hiten Patel and Sanjay C. Shah, Review on cycle time reduction in manufacturing industries. Journal of emerging technologies and innovative research (JETIR) 10(14): (2014).

UdayKiran, G. and P. Vishnu Chithamacharyulu, Study of reduction and optimization of installation and cycle time of MIG welding robotic system in automation factory. International Journal of App-lied Engineering Research (2015).