Human-Performed Assembly Task Evaluation through a Virtual Operator

Ashvin Sobhee, Sébastien Druon, André Crosnier

Human-Performed Assembly Task Evaluation through a Virtual Operator

Ashvin Sobhee, Sébastien Druon, André Crosnier
LIRMM, UMR 5506 Université Montpellier II - CNRS
161 rue Ada, 34000 Montpellier - France
sobhee,druon,crosnier@lirmm.fr

***Abstract*— The study presented in this paper aims at achieving optimal paths for a human-performed assembly task. Optimality, from an operator's point of view, means that minimal fatigue occurs. Trajectories for assembly parts, generated by potential fields, are optimized in terms of the work the operator has to do. Optimization is carried out by genetic algorithm. The trajectories have to satisfy the constraints imposed by physical human limitations. We therefore introduce a virtual human operator in the working environment, and determine by means of inverse kinematics if the trajectories are feasible. Our method is tested on the assembly of a mechanical part in 3D space.**

I. Introduction

Path planning, whose aim is to find a path lying completely in free space from an initial to a final configuration, is an integral part of industrial assembly. Although extensive work exists on the subject, most of it is focused on robot-performed tasks. While robots have undeniably overcome human limitations in many ways, many tasks still need to be done by human workforce, for economic or technical reasons. Such is the case when limited series of mechanical parts need to be manufactured. When assembly is to be performed by humans, the path planning scheme has to integrate ergonomic issues. A trajectory which is optimal for a robot is not necessarily optimal for the worker, and sometimes it might just not be humanly possible to realize. In this paper we propose a 3D path planning algorithm where the physical fatigue of the human operator is the center of interest.

The paper is organized as follows. In the next section, we present the main principles of our approach as well as some existing work in related areas. Section III gives the expressions of the potential fields producing the forces giving the trajectories and describes how the path is computed. Section IV deals with the task to be performed, the computation of the work done during the task as well as the constraints the virtual human operator's presence brings in the loop. Section V explains hows optimization is carried out by our genetic algorithm. Simulations results from the 3D assembly of a pump are presented and commented on in section VI and we conclude in section VII.

II. Problem Statement

A. Description of the approach

Figure 1 gives an overview of our approach. Trajectories for assembly parts are first are generated by attractive and repulsive potential fields. The parameters of the repulsive potential fields are initially generated by genetic algorithm (GA). Our choice for the potential field method lies on the fact that it is goal-oriented. Assembly path planning assumes the existence of paths, which means that the whole operational space doesn't need to be searched to prove the existence of one. The trajectories are obtained as a series of positions in operational space. A virtual manikin is introduced in the environment to simulate a human operator. At each simulation step, path feasibility checking is done with respect to constraints due to the environment and constraints brought in by the manikin using inverse kinematics.

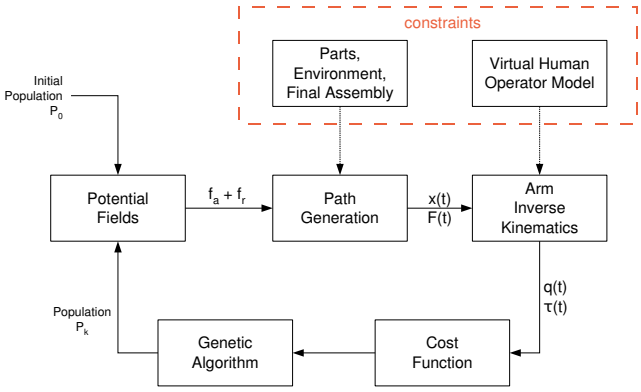

Fig. 1. Block diagram of the proposed approach

Although fatigue may have various causes, our study is focused on the amount of work done by the operator along the trajectories taken during handling of parts. Using the arm configuration and the torque acting on it, the work done is computed and used as a cost function for the GA. By applying selection, cross-over and mutation operators, the GA then produces a new population. The process is repeated for a chosen number of generations.

B. Related Work

In order to study the constraints of the human body in the context of manual assembly tasks, it has become common to introduce a human dummy in virtual environments for 3D simulations. Ferré et al. [4] proposed 2 approaches to add a manikin in an assembly process. The first one is to compute a path for the part alone by using a probabilistic diffusion

algorithm. The feasibility of the path for the manikin is then checked by inverse kinematics. When feasibility fails, the second approach is then considered. It consists in grouping the manikin altogether with the part being handled into a single system. This obviously increases the dimensionality of the problem, but allows to solve more constrained cases the first approach doesn't. Both methods are valid for one-hand and two-hand manipulation. Lämkull et al. [7] compared simulation results from digital human modelling with real working conditions in a case study concerning manual automobile assembly. The aim of the study was to examine how far the ergonomics simulations could correctly predict real outcomes. Evaluation of ergonomics conditions in simulations were based on 3 criteria: joint configuration, work distance/hand position and field of vision. Conclusions were that digital human modelling tools could correctly predict ergonomic issues for standing and uncontrained working postures. The study also identified areas where improvements could de made.

The amount of work that has to be done and the number of decisions that have to be taken have a direct impact on the operator's productivity. For the latter to work with maximum efficiency, physical fatigue and mental load have to be minimized. Rodríguez et al. [11] estimated fatigue apparition caused by a body segment's own mass and carrying of an external load. The half-joint concept was introduced to calculate fatigue at joint-level, i.e at muscle group level. The idea is to break each dof at each joint into a half-joint pair so as to isolate the activity of antagonistic muscle groups. The parameters of the model are normalized torque, joint strength and the maximum holding time that a posture can be maintained. The model also includes a term for static recovery produced by a period of rest. Simulation results from the model were compared with results obtained from experimental subjects.

GAs have been widely used in path planning due to their simplicity, and have been adapted in various ways. Nishimura et al. [10] used GAs to optimize the joint configuration to achieve collision-free motion for the body of a hyper multi-joint manipulator, while the end-effector was driven to the goal position by forces generated by potential fields. The chromosomes had a 2D structure in which the variation of the angle for each joint was stored at a time step. The chromosomes were also variable in length such that they could represent a movement of the manipulator of any duration. A technique that combined the closed-loop pseudoinverse method with GAs for trajectory planning of redundant manipulators was presented by da Graça Marcos et al. [3]. The method uses an extended square Jacobian and an extended position vector in operational space, both whose additional values were generated by GA. Several fitness functions such as largest joint displacement, total joint torque or joint power consumption, all to be minimized, were proposed. Christiand & Yoon [2] proposed an assembly algorithm that considers both path planning and the assembly task itself. 2D trajectories were generated by potential fields and GA was used to optimize a cost function based on the overall path distance, the number of assembly orientation changes and the number of gripper changes. The chromosomes consisted of the assembly sequence and a repulsive radius for each part. The repulsive potential field associated to each part was to ensure that no collision among parts occurred at the moment of assembly. However, optimization of the repulsive radii of obstacles was not considered in the algorithm.

III. Path Generation

This section describes how attractive and repulsive forces are obtained from potential fields. The potential fields we implemented are based on the formulation used by Khatib [6]. Path computation from the resulting displacement of the parts is also explained.

A. Potential Fields

Objects in the virtual environment can be either a part to be assembled or an obstacle. Each object has a repulsive potential field attached to it. When a part is being handled, it is continuously attracted to its final position, and also subjected to repulsion from obstacles and other parts which might lie on its way. Note that its own repulsive potential field has no effect on its surrounding during this process.

The repulsive potential fields we implemented are spherical in shape, and are based on the bounding spheres (BS) of the objects. They are defined by a repulsion constant C_r and a radius ρ_r. Consider a static a moving part A and a static obstacle B, having a_c and b_c as centers of their respective BS. (Figure 2). As said earlier, it is the obstacle's repulsive field which will apply forces on the part. The distance ρ between these 2 BS is $\rho = \|a-b\|$, a and b being points on the BS surfaces. The force applied on part A will depend on how far a is from b, and is derived from the repulsive potential U_r. The repulsive potential at point a is given by

$$U_r = \begin{cases} \frac{1}{2}C_r\left(\frac{1}{\rho}-\frac{1}{\rho_r}\right)^2 & \text{if } \rho \le \rho_r \\ 0 & \text{if } \rho > \rho_r \end{cases} \quad (1)$$

The repulsive force at a in the workspace is obtained by

$$\mathbf{f}_r(a) = -grad|U_r(a)|, \quad (2)$$

thus giving

$$\mathbf{f}_r(a) = \begin{cases} C_r\left(\frac{1}{\rho}-\frac{1}{\rho_r}\right)\frac{1}{\rho^2}\frac{\mathrm{d}\rho}{\partial a} & \text{if } \rho \le \rho_r \\ 0 & \text{if } \rho > \rho_r, \end{cases} \quad (3)$$

The attractive potential at point a_c is given by

$$U_{att} = \frac{1}{2}C_{att}(a_c - a_f)^2, \quad (4)$$

and the attractive force is proportional to the distance away from the goal position. Its expression is similarly derived from its potential, giving

$$\mathbf{f}_{att} = -C_{att}(a_c - a_f), \quad (5)$$

where a_f is the goal position to be reached and C_{att} is the attraction constant. The proportional nature of the attractive

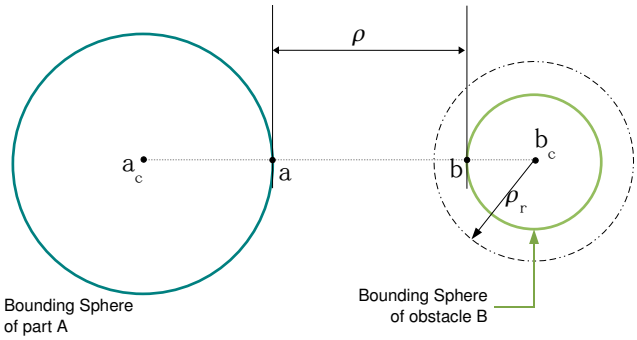

Fig. 2. ρ is the distance between bounding spheres. ρ_r is the repulsive radius of obstacle B

force can make it very strong when a part is initially far from its final position. Indeed, the attractive force can sometimes be so strong that it can overcome the repulsive forces of nearby objects, and this results into collisions. To avoid this, we have saturated the attraction when parts are beyond a certain limit from their final position.

B. Path Computation

Paths are obtained from the displacement of parts caused by attractive and repulsive forces generated by potential fields. Following Newton's 2^{nd} Law of Motion, these forces induce an acceleration which, at an instant t, can be expressed as

$$\mathbf{I}\ddot{\mathbf{x}}(t) = \mathbf{f}_a(t) + \mathbf{f}_r(t), \tag{6}$$

where $\mathbf{I}$ is a [6×6] mass matrix. This acceleration can in turn be expressed as

$$\ddot{\mathbf{x}}(t) \approx \frac{\dot{\mathbf{x}}(t+\delta t) - \dot{\mathbf{x}}(t)}{\delta t}, \tag{7}$$

δt being the integration step. Combining equations (6) and (7) gives

$$\dot{\mathbf{x}}(t+\delta t) = \dot{\mathbf{x}}(t) + \mathbf{I}^{-1}\big(\mathbf{f}_a(t) + \mathbf{f}_r(t)\big)\delta t. \tag{8}$$

Integrating further gives the configuration at instant $t+\delta t$

$$\mathbf{x}(t+\delta t) = \mathbf{x}(t) + \dot{\mathbf{x}}(t)\delta t. \tag{9}$$

The path is thus formed by a series of configurations obtained by the same calculation at each step.

IV. Virtual Human Operator

In this section we specify the type of task the operator has to accomplish, and derive expressions to compute the amount of work that is done during the task. Constraints arising from physical human limitations are also specified and adapted to the virtual manikin.

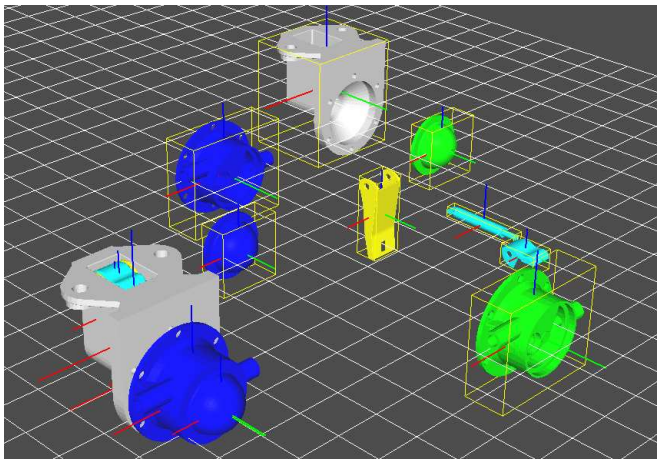

Fig. 3. Differents parts for the assembly of a pump: body, bottom, push rod, axis, lever and top cover.

A. Task Specification

The first step in assembly is movement towards the task vicinity [1]. We do not deal with the mounting operation. The chosen task requires the operator to take the parts from their initial configuration and to bring them in the vicinity of the operation. The final configuration of the parts have been chosen in such a way that when they are all in the mounting vicinity, they form an exploded view of the assembly.

The assembly of a pump has been chosen for our experimentations. The model was taken from OpenCASCADE [5] and consists of 6 parts to be mounted. Figure 3 shows the different parts and the mounted assembly. 2 additional parts (coloured in green) were introduced in the setting to act as obstacles. The grid on which the assembly rests can be considered as a table.

B. Virtual Arm Model

We have placed a virtual human operator in a standing posture in the environment as shown in Figure 4. The upper arm and the forearm are represented as linear segments. The hand is not shown. Like [4], we assume that the operator's position enables him to reach all the parts. Although it is generally accepted that the human arm is mechanically a 7 DOF redundant serial kinematic chain ([9], [12], [13]), we have modelled a 6 DOF arm for the manikin. The DOF that has been dropped is that of medial rotation of the shoulder. We have seen that the absence of this DOF doesn't considerably penalize the workspace of the operator. This also makes inverse kinematics easier since the resulting mechanism is non-redundant.

In operational space, the hand is described in the shoulder frame R_s by a [6×1] vector $\mathbf{x}$. $\mathbf{x}$ corresponds to a [6×1] vector $\mathbf{q}$ which is the description of the hand in joint space, or configuration space. The forward and inverse kinematic models of the arm allow the transition from one space to another.

The total torque to which the arm is subjected during the

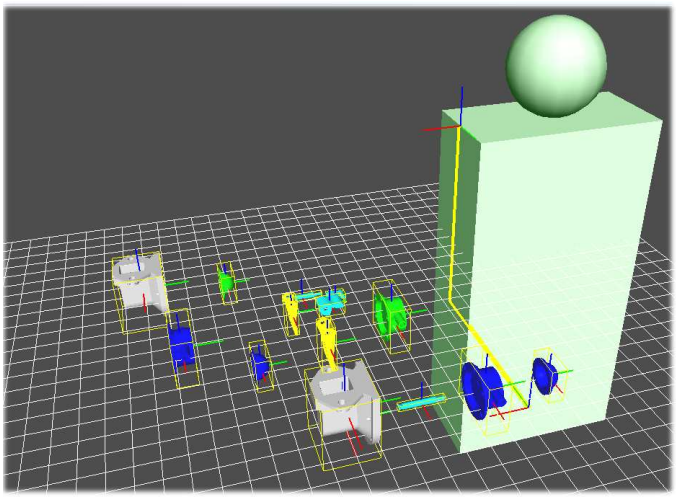

Fig. 4. Virtual human operator in the workspace.

handling operation is given by

$$\tau = \mathbf{M}(q)\ddot{q} + \mathbf{v}(q,\dot{q}) + \mathbf{g}(q) + \mathbf{J}^T(q)\mathbf{w}(q), \tag{10}$$

where **M**(q) is the kinetic energy matrix, **v**(q,$\dot{q}$) the centrifugal and Coriolis force vector, **g**(q) the gravity force vector, **J**(q) the Jacobian matrix, and **w**(q) the weight of the part. The path being a series of configurations as said earlier, let us consider the total torque at one particular configuration. This is given by

$$\tau = \mathbf{g}(q) + \mathbf{J}^T(q)\mathbf{w}(q), \tag{11}$$

The power developed by the operator in terms of the forces acting on its arm is given by

$$P = \tau^T \dot{q} \tag{12}$$

The work done along the path Π is the integration of the power with respect to time

$$W = \int_{\Pi} \tau^T \dot{q}\ \mathrm{d}t. \tag{13}$$

C. Operator-Related Constraints

Once the configuration **x** of a part is obtained, as described on section III.B, the operator's hand has to follow it at each step to simulate the task of bringing the part to the assembly vicinity. Our study does not investigate into object grasping. We assume that an object is held by making its center of gravity coincide with that of the operator's hand. The virtual arm's behaviour must exhibit characteristics which are as close as possible to that of a real operator. Hence it becomes necessary to impose some constraints.

Firstly, joint limits have been imposed so that the workspace volume of our virtual operator corresponds to that of a human arm. Maximal joint values were obtained from a kinesiological source [8]. When the virtual hand is following a part's trajectory, it's joint configuration **q** can be computed from its operational configuration via the inverse kinematic model. In our algorithm **q** is calculated from **x** at each step. If any joint q_i of the vector **q** is outside its admissible limits, the path is considered infeasible for a human operator, and has to be rejected.

Another issue is the limited power that humans have; they cannot handle infinite weights. It is important that this limitation is reflected on our virtual operator as a matter of safety. One of the ergonomics requirements for an automobile company stated in [7] is that assembly of parts must not exceed an assembly force of 15N forn one finger, 30N for two/three fingers and 50N for hand at average position of the wrist. Using this data, we have determined a maximum force/torque value, $\tau_{max}(q)$, dependent on the joint configuration **q**, which can be exerted on the arm model.

V. Assembly Task Evaluation

The assembly task is evaluated with respect to the constraints imposed by the virtual manikin and the environment. This section describes outlines the steps of our genetic algorithm and how these constraints are accounted for in the optimization process.

A. Genetic Algorithm

Since the trajectory of a part is influenced by the repulsive forces acting on it, it is the parameters which govern these forces that have to be optimized. These are the repulsive constant C_r, and the radius of the repulsive boundary ρ_r. The chromosomes for the GA thus consist of 2 arrays: the first one is an array of the repulsive radii ρ_{r_i}, and the second one is for the repulsive constants C_{r_i}.

The optimal path is searched for each part one by one. For example, let p_k be the k^{th} part to be assembled for a given sequence consisting of n parts. The algorithm takes into account that parts p_1 to p_{k-1} have been assembled while parts p_{k+1} to p_n are still in their initial configurations.

For each generation g_i, each chromosome c_j is used to determine the path of p_k only. The cost of each chromosome is evaluated, and a selection rate is applied. By using cross-over and mutation operators over the selected portion of the actual population, new chromosomes are produced for the next generation g_{i+1}. This process is carried on for the chosen number of generations N_{gen}, at the end of which a chromosome giving the optimal path for part p_k is obtained.

Our cross-over operator was designed such that the off-spring is composed of the even-indexed ρ_{r_i} and C_{r_i} of the first parent and the odd-indexed ρ_{r_i} and C_{r_i} of the second parent. The mutation operator generates a new chromosome by changing a randomly chosen $\left[\rho_{r_i}, C_{r_i}\right]$ pair in a randomly chosen chromosome. This comes to changing the potential field of a randomly selected object in the environment.

The genetic algorithm is therefore run for N_{gen} generations for each part, thus giving a set of n chromosomes, each one giving the optimal path for 1 part alone.

B. Cost Evaluation

The cost associated with each chromosome is the total work done by the operator to move the part along the path it - the chromosome - gives. If the path does not satisfy the constraints of the virtual operator, this must appear in the

cost. In addition to the two previously mentioned constraints originating from the manikin, a third constraint comes as a consequence of the geometry of the repulsive potential fields. Their spherical shape makes them generate forces in all directions. This can cause objects to be pushed downwards, through the table. This is obviously not realistic and can be detected in the virtual environment by keeping record of the object's center of gravity. Any trajectory making an object pass through the table has to be rejected.

If at any moment one of these constraints is not met, a penalty is applied to the cost of the chromosome. This will cause it to be eliminated from the next population in the GA.

VI. Simulation and Results

In this section we show the results obtained by our approach. We show the behaviour of the cost function, and the joint values also, for the manipulation of the pump bottom since it is the heaviest part and is therefore likely to show information in a more evident manner than the lighter parts.

In our experiments the selection, cross-over and mutation rates were respectively 25%, 75% and 5%. The radius ρ_r of the repulsive potential field of each object was randomly chosen by the GA in the interval $[\rho_{r_{min}}, \rho_{r_{max}}]$. $\rho_{r_{min}}$ was chosen to be equal to the radius of the object's BS while $\rho_{r_{max}}$ was twice $\rho_{r_{min}}$.

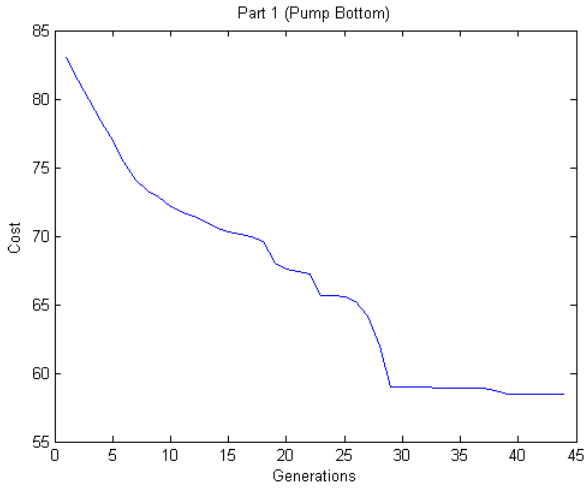

Fig. 5. Evolution of cost for pump bottom

Figure 5 shows the variation of cost for assembling the pump bottom over 45 generations. It can be observed that at around the 30^{th} generation, the cost varies only slightly and reaches a plateau at around the 40^{th} generation.

The values of the joint angles for the manipulation of the pump bottom are shown in Figure 6 at 3 generations - 5^{th}, 20^{th} and 40^{th} - during the optimization process. Lines of the same colour correspond to the same joint. It can be seen that as the number of generations increases, the joints reach their final values in a shorter time. This implies that the optimization process finds paths which are shorter in length. Figure 7 shows the paths obtained for the pump bottom.

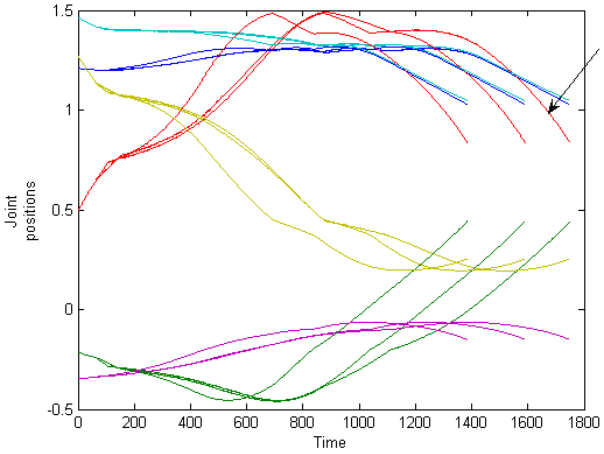

Fig. 6. Joint values for manipulation of the pump bottom at the 5^{th}, 20^{th} and 40^{th} generations(from right to left). The joints reach the final values in shorter time.

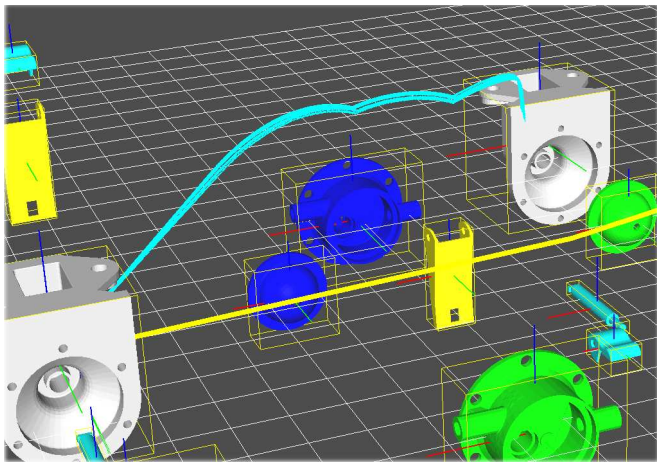

Fig. 7. Paths generated by potential fields for the pump bottom. The yellow line across the image is the virtual operator's arm.

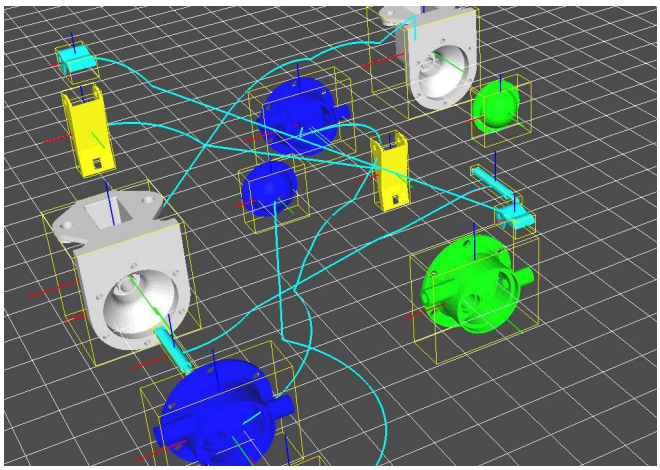

Fig. 8. Optimal paths obtained for each part at the end of the simulation

The optimal paths obtained at the end of the simulation are displayed in Figure 8.

VII. Conclusions and Future Work

The aim of the work presented in this paper is to provide optimal paths to a human operator for an assembly task. Trajectories are generated by potential fields. By introducing a virtual manikin in the environment, the feasibillity of the path is verified through imposed constraints. We have chosen work as a criterion to optimize the paths and results show that the approach achieves paths which the manikin completes in lesser time as optimization proceeds (Figure 6).

Collision detection has not been treated in the example we have chosen. The repulsive potential fields naturally pushed the objects away from one another, and there was enough clearance in the working environment to omit collision detection between the virtual operator's arm and other objects. The problem however becomes crucial when the operational space is confined.

References

[1] P. Akella, M. A. Peshkin, J. E. Colgate, W. Wannasuphoprasit, N. Nagesh, J. Wells, S. Holland, T. Pearson, and B. Peacock, "Cobots for the automobile assembly line," in *ICRA*, 1999, pp. 728–733.

[2] N. Christiand and J. Yoon, "A novel optimal assembly algorithm for the haptic interface application of a virtual maintenance system," in *IEEE International Conference on Robotics and Automation*, 2008.

[3] M. da Graça Marcos, J. T. Machado, and T.-P. Azevedo-Perdicolis, "Trajectory planning of redundant manipulators using genetic algorithms," in *Communications in Nonlinear Science and Numerical Simulation*, vol. 14, no. 7, July 2009, pp. 2858–2869.

[4] E. Ferré, J.-P. Laumond, G. Arechavaleta, and C. Estevès, "Progresses in assembly path planning," in *Proceedings of the International Conference on Product Life Cycle Management*, July 2005.

[5] http://www.opencascade.org/.

[6] O. Khatib, "The potential field approach and operational space formulation in robot control," in *Proc. Fourth Yale Workshop on Applications of Adaptive Systems Theory, Yale University, New Haven, Connecticut*, May 1985, pp. 208–214.

[7] D. Lämkull, L. Hanson, and R. Örtengren, "A comparative study of digital human modelling simulation results and their outcomes in reality: A case study within manual assembly of automobiles," *International Journal of Industrial Ergonomics*, vol. 39, no. 2, pp. 428–441, 2009.

[8] K. Luttgens and N. Hamilton, *Kinesiology: Scientific Basis of Human Motion (9th ed.),Dubuque, IA: Brown & Benchmark.*, 1997.

[9] M. Mihelj, "Human arm kinematics for robot based rehabilitation," in *Robotica*, vol. 24, no. 03, 2006, pp. 377–383.

[10] T. Nishimura, K. Sugawara, I. Yoshihara, and K. Abe, "A motion planning method for a hyper multi-joint manipulator using genetic algorithm," *IEEE International Conference on Systems, Man, and Cybernetics*, 1999.

[11] I. Rodríguez, R. Boulic, and D. Meziat, "A Model to Assess Fatigue at Joint-Level Using the Half-Joint Concept," in *IEEE-RAS International Conference on Humanoid Robots*, 2003, pp. 97–105.

[12] D. Tolani, A. Goswami, and N. I. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graphical Models*, vol. 62, pp. 353–388, 2000.

[13] B. Tondu, "A theorem on the manipulability of redundant serial kinematic chains," in *Engineering Letters*, vol. 15, no. 2, 2007.