6th CIRP Conference on Assembly Technologies and Systems (CATS)

Enhancing digital human motion planning of assembly tasks through dynamics and optimal control

Staffan Björkenstam[a,*], Niclas Delfs[a], Johan S. Carlson[a], Robert Bohlin[a], Bengt Lennartson[b]

[a]Fraunhofer-Chalmers Centre, Chalmers Science Park, SE-412 88 Göteborg, Sweden
[b]Automation Research Group, Department of Signals and Systems, Chalmers University of Technology, SE-412 96 Göteborg, Sweden

* Corresponding author. Tel.: +46-31-772-4284; fax: +46-31-772-4260. *E-mail address:* staffan@fcc.chalmers.se

Abstract

Better operator ergonomics in assembly plants reduce work related injuries, improve quality, productivity and reduce cost. In this paper we investigate the importance of modeling dynamics when planning for manual assembly operations. We propose modeling the dynamical human motion planning problem using the Discrete Mechanics and Optimal Control (DMOC) method, which makes it possible to optimize with respect to very general objectives. First, two industrial cases are simulated using a quasi-static inverse kinematics solver, demonstrating problems where this approach is sufficient. Then, the DMOC-method is used to solve for optimal trajectories of a lifting operation with dynamics. The resulting trajectories are compared to a steady state solution along the same path, indicating the importance of using dynamics.

© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/).
Peer-review under responsibility of the organizing committee of the 6th CIRP Conference on Assembly Technologies and Systems (CATS)

Keywords: Assembly; Digital human modeling; Ergonomy; Dynamics; Optimal control

1. Introduction

Although the degree of automation is increasing in manufacturing industries, many assembly operations are performed manually. To avoid injuries and to reach sustainable production of high quality, comfortable environments for the operators are vital, see [1] and [2]. Poor station layouts, poor product designs or badly chosen assembly sequences are common sources leading to unfavorable poses and motions. To keep costs low, preventive actions should be taken early in a project, raising the need for feasibility and ergonomics studies in virtual environments long before physical prototypes are available.

Today, in the automotive industries, such studies are conducted to some extent. The full potential, however, is far from reached due to limited software support in terms of capability for realistic pose prediction, motion generation and collision avoidance. As a consequence, ergonomics studies are time consuming and are mostly done for static poses, not for full assembly motions. Furthermore, these ergonomic studies, even though performed by a small group of highly specialized simulation engineers, show low reproducibility within the group [3].

To describe operations and facilitate motion generation, it is common to equip the manikin with coordinate frames attached to end-effectors like hands and feet. The inverse kinematic problem is to find joint values such that the position and orientation of hands and feet matches certain target frames. For the quasi-static inverse kinematics this leads to an underdetermined system of equations since the number of joints exceeds the end-effectors constraints. Due to this redundancy there exist a set of solutions, allowing us to consider ergonomics aspects, collision avoidance, and maximizing comfort when choosing one solution.

The dynamic motion planning problem is stated as an optimal control problem, which we discretize using discrete mechanics. This results in an optimization problem, which can be solved using standard nonlinear programming solvers. Furthermore, this general problem formulation makes it fairly easy to include very general constraints and objectives.

In this paper we show, using a couple of case studies, where the quasi-static solver is sufficient, and where the DMOC solver could improve the solution. The paper extends the work presented in [4] and [5], and is a part of Cromm (Creation of Muscle Manikins) project [6].

2. Background

2.1. Manikin Model

In this section we present the manikin model and the inverse kinematic problems, both quasi-static and with dynamics.

2.2. Kinematics

The manikin model is a tree of rigid bodies connected by joints. Each body has a fixed reference frame and we describe its position relative to its parent body by a rigid transformation $T(\boldsymbol{q})$, where $\boldsymbol{q}$ is the coordinate of the joint. To position the manikin in space, i.e. with respect to some global coordinate system, it has an exterior root as the origin and a prismatic joint and a rotation joint as exterior joints as opposed to the interior links representing the manikin itself, see [4]. Together, the exterior links mimic a rigid transformation that completely specifies the position of the lower lumbar. In turn, the lower lumbar represents an interior root, i.e. it is the ancestor of all interior joints. Note that the choice of the lower lumbar is not critical. In principal, any link could be the interior root, and the point is that the same root can be used though a complete simulation. No re-rooting or change of tree hierarchy will be needed. Now, for a given configuration of each joint, collected in the joint vector $\boldsymbol{q} = [\boldsymbol{q}_1^T, \ldots, \boldsymbol{q}_n^T]^T$, we can calculate all the relative transformations $T_1, , T_n$, traverse the tree beginning at the root and propagate the transformations to get the global position of each body. We say that the manikin is placed in a pose, and the mapping from a joint vector into a pose is called forward kinematics. Furthermore, a continuous mapping $\boldsymbol{q}(t)$, where $t \in \mathbb{R}$, is called a motion, or a trajectory of the system.

2.3. Quasi Static Inverse Kinematics

In order to facilitate the generation of realistic poses that also fulfill some desired rules we add a number of constraints on the joint vector. These kinematic constraints can for example restrict the position of certain links, either relative to other links or with respect to the global coordinate system or ensure the manikin is kept in balance, see section 2.3.2. All the kinematic constraints can be defined by a vector valued function $\boldsymbol{g}$ such that

$$\boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{0} \tag{1}$$

must be satisfied at any pose. Finding a solution to equation 1 is generally referred to as inverse kinematics. Often, in practice, the number of constraints is far less than the number of joints of the manikin. Due to this redundancy there exist many solutions, allowing us to consider ergonomics aspects and maximizing comfort when choosing solution. To do so, we introduce a scalar comfort function

$$h(\boldsymbol{q}) \tag{2}$$

capturing as many ergonomic aspects as desired. The purpose is to be able to compare different poses in order to find solutions that maximize comfort. The comfort function is a generic way to give preference to certain poses while avoiding others. Typically h considers joint limits, distance to surrounding geometry in order to avoid collision, magnitude of contact forces, forces and torques on joints, see section 2.3.3. Furthermore, by combining equation 1 and 2 we can formulate the inverse kinematic problem as

$$\max_{\boldsymbol{q}} h(\boldsymbol{q}) \quad \text{subject to} \quad \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{0}. \tag{3}$$

2.3.1. Collision Avoidance

While some contact with the environment may be intended, e.g. grasping of objects and leaning, and contribute to the force and moment balance. Other contacts, for example, collisions, are undesired. The comfort function offers a convenient way to include a simple, yet powerful, method penalizing poses close to collision. In robotics this method is generally known as Repulsive Potential [7][8]. The underlying idea is to define a barrier, say, around the obstacles increasing the discomfort towards infinity near collision. This method does not address the problem of escaping an already occurring collision. The idea is merely that if the manikin starts in a collision-free pose, then the repulsive potential prevents the manikin from entering a colliding pose.

Note: It is common to think of the repulsive potential or rather its gradient field as a force field pushing an object away from obstacles. In this work, we do not want such artificial forces to contribute to the force balance. To avoid confusion with real contact forces we will not use that analogy.

2.3.2. Balance and Contact Forces

One important part of $\boldsymbol{g}$ is ensuring that the manikin is kept in balance. For this, the weight of links and objects being carried, as well as external forces and torques due to contact with the floor or other objects, must be considered. The sum of all forces and torques are

$$\boldsymbol{g}_{force}(\boldsymbol{q}) = m\boldsymbol{g} + \sum_{j \in J} \boldsymbol{f}_i,$$

$$\boldsymbol{g}_{torque}(\boldsymbol{q}) = \boldsymbol{m}_c \times m\boldsymbol{g} + \sum_{j \in J} \boldsymbol{p}_j \times \boldsymbol{f}_j + \boldsymbol{\tau}_j,$$

where m is the total body mass, $\boldsymbol{g}$ is the gravity vector, $\boldsymbol{m}_c$ is the center of mass, $\boldsymbol{f}_j$ and $\boldsymbol{\tau}_j$ are external force and torque vectors at point $\boldsymbol{p}_j$ and J is the index set. Note that the quantities may depend on the pose, but this has been omitted for clarity. In general, external forces and torques due to contacts are unknown. For example, when standing with both feet on the floor it is not obvious how the contact forces are distributed between the feet. In what follows we let $\boldsymbol{f}$ and $\boldsymbol{t}$ denote the unknown forces and torques, and we stack them into the vector $x = [\boldsymbol{q}^T \ \boldsymbol{f}^T \ \boldsymbol{\tau}^T]^T$. Then we can rephrase (3) as follows:

$$\max_{x} h(\boldsymbol{x}) \quad \text{subject to} \quad \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}. \tag{4}$$

2.3.3. Joint Torque

The joint loads are key ingredients when evaluating poses from an ergonomic perspective [9]. Furthermore, research shows that real humans tend to minimize the muscle strain, i.e. minimize the proportion of load compared to the maximum possible load [10], so by normalizing the load on each joint by the muscle strength good results can be achieved. In this article we choose the function

$$h_t = \sum_{i=1}^{n} w_i^2 \tau_i^2$$

where τ_i is the torque in joint i, and w_i is the reciprocal of the joint strength. Note that it is straightforward to propagate the external forces and torques and the accumulated link masses trough the manikin in order to calculate the load on each joint.

2.4. Discrete mechanics and optimal control

2.4.1. The constrained discrete Euler-Lagrange equations

Consider the mechanical system specified by a configuration manifold $Q \subseteq \mathbb{R}^{n_q}$ and Lagrangian $L : TQ \to \mathbb{R}$, where TQ is the tangent bundle of the configuration manifold. Furthermore, suppose the motion of the system is constrained by the equation $\boldsymbol{\phi}(\boldsymbol{q}) = \boldsymbol{0} \in \mathbb{R}^m$ to lie in the constraint manifold $C = \boldsymbol{\phi}^{-1}(\boldsymbol{0}) \subset Q$. Let $U \in \mathbb{R}^{n_u}$ be the set of admissible controls and $\boldsymbol{F} : TQ \times U \to T^*Q$ the external force acting on the system, where T^*Q is the cotangent bundle of the configuration manifold.

Introducing the multiplier $\lambda(t) \in \mathbb{R}^m$ the Lagrange-d'Alembert principle states that trajectories of the system satisfy

$$\delta \int_{t_1}^{t_2} L(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) + \boldsymbol{\phi}^T(\boldsymbol{q}(t))\lambda(t)dt$$
$$+ \int_{t_1}^{t_2} \boldsymbol{F}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t)) \cdot \delta \boldsymbol{q} dt = 0, \quad (5)$$

where variations are taken with respect to $\boldsymbol{q}$, fixed at the endpoints, and with respect to λ.

Integration by parts and the fundamental lemma of calculus of variations give the following differential algebraic equations, known as the constrained Euler Lagrange equations of motion:

$$\frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) - \frac{d}{dt}\frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))$$
$$+ \boldsymbol{F}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t)) + \boldsymbol{\Phi}^T(\boldsymbol{q}(t))\lambda(t) = \boldsymbol{0}, \quad (6a)$$
$$\boldsymbol{\phi}(\boldsymbol{q}(t)) = \boldsymbol{0}, \quad (6b)$$

where $\boldsymbol{\Phi}$ denotes the Jacobian of the constraint function.

The key idea of variational integrators is to directly approximate the variational principle (5) rather than the equations of motion (6).

We now discretize $\boldsymbol{q}(t)$ in $[t_1, t_2]$ using a fixed time step $h = (t_2 - t_1)/N$ so that $\boldsymbol{q}^{(k)}$ is an approximation of $\boldsymbol{q}(t_1 + kh)$ for $k = 0, \ldots, N$. Furthermore, we discretize the control such that $\boldsymbol{u}^{(k)}$ is an approximation of $\boldsymbol{u}(t_1 + (k + \frac{1}{2})h)$ for $k = 0, \ldots, N-1$. We are now ready to replace the continuous state space, TQ, with the discrete state space, $Q \times Q$, and construct a discrete Lagrangian $L_d : Q \times Q \times \mathbb{R} \to \mathbb{R}$ such that

$$L_d(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, h) \approx \int_{t_1+kh}^{t_1+(k+1)h} L(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))dt.$$

Introducing left and right discrete forces, $\boldsymbol{F}_d^+$ and $\boldsymbol{F}_d^-$, and discrete multipliers, $\lambda_d^{(k)}$ for $k = 0, \ldots, N$, a discrete variational principle corresponding to (5) can be formulated as

$$\delta \sum_{k=0}^{N-1} (L_d(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, h)$$
$$+ \frac{1}{2}\boldsymbol{\phi}^T(\boldsymbol{q}^{(k)})\lambda_d^{(k)} + \frac{1}{2}\boldsymbol{\phi}^T(\boldsymbol{q}^{(k+1)})\lambda_d^{(k+1)})$$
$$+ \sum_{k=0}^{N-1} (\boldsymbol{F}_d^-(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, \boldsymbol{u}^{(k)}, h) \cdot \delta \boldsymbol{q}^{(k)}$$
$$+ \boldsymbol{F}_d^+(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, \boldsymbol{u}^{(k)}, h) \cdot \delta \boldsymbol{q}^{(k+1)}) = 0 \quad (7)$$

for all variations $\delta\lambda_d^{(k)}$ and $\delta \boldsymbol{q}^{(k)}$ with $\delta \boldsymbol{q}^{(0)} = \delta \boldsymbol{q}^{(N)} = \boldsymbol{0}$.

This principle is equivalent to the discrete Euler-Lagrange equations:

$$D_2 L_d(\boldsymbol{q}^{(k-1)}, \boldsymbol{q}^{(k)}, h) + D_1 L_d(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, h)$$
$$+ \boldsymbol{F}_d^+(\boldsymbol{q}^{(k-1)}, \boldsymbol{q}^{(k)}, \boldsymbol{u}^{(k-1)}, h) + \boldsymbol{F}_d^-(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, \boldsymbol{u}^{(k)}, h)$$
$$+ \boldsymbol{\Phi}^T(\boldsymbol{q}^{(k)})\lambda_d^{(k)} = \boldsymbol{0}, \quad (8a)$$
$$\boldsymbol{\phi}(\boldsymbol{q}^{(k+1)}) = \boldsymbol{0}, \quad (8b)$$

where $D_1 L_d$ and $D_2 L_d$ are the slot derivatives with respect to the first and second argument. These equations define the variational integrator by implicitly mapping $(\boldsymbol{q}^{(k-1)}, \boldsymbol{q}^{(k)}, \boldsymbol{u}^{(k-1)}, \boldsymbol{u}^{(k)})$ to $(\boldsymbol{q}^{(k+1)}, \lambda_d^{(k)})$. Please refer to [11] for a thorough introduction to discrete mechanics and [12,13] for more on discrete mechanics and optimal control of multibody systems.

A reasonable trade-off between accuracy and performance, is to use the the midpoint rule to approximate the relevant integrals. The discrete Lagrangian then becomes

$$L_d(\boldsymbol{q}_0, \boldsymbol{q}_1, h) = h L\left(\frac{\boldsymbol{q}_0 + \boldsymbol{q}_1}{2}, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right). \quad (9)$$

Thus

$$D_1 L_d(\boldsymbol{q}_0, \boldsymbol{q}_1, h) = \frac{h}{2}\frac{\partial L}{\partial \boldsymbol{q}}\left(\frac{\boldsymbol{q}_0 + \boldsymbol{q}_1}{2}, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right)$$
$$- \frac{\partial L}{\partial \dot{\boldsymbol{q}}}\left(\frac{\boldsymbol{q}_0 + \boldsymbol{q}_1}{2}, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right)$$

and

$$D_2 L_d(\boldsymbol{q}_0, \boldsymbol{q}_1, h) = \frac{h}{2}\frac{\partial L}{\partial \boldsymbol{q}}\left(\frac{\boldsymbol{q}_0 + \boldsymbol{q}_1}{2}, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right)$$
$$+ \frac{\partial L}{\partial \dot{\boldsymbol{q}}}\left(\frac{\boldsymbol{q}_0 + \boldsymbol{q}_1}{2}, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right).$$

Furthermore, it is then natural to use the following discrete forces:

$$\boldsymbol{F}_d^+(\boldsymbol{q}_0, \boldsymbol{q}_1, \boldsymbol{u}_0, h) = \boldsymbol{F}_d^-(\boldsymbol{q}_0, \boldsymbol{q}_1, \boldsymbol{u}_0, h) =$$
$$= \frac{h}{2}\boldsymbol{F}\left(\frac{\boldsymbol{q}_0 + \boldsymbol{q}_1}{2}, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}, \boldsymbol{u}_0\right). \quad (10)$$

This discretization scheme results in a second order accurate integrator.

2.5. Optimal control problem

We consider the following optimal control problem: Minimize

$$J = \chi(\boldsymbol{q}(t_f), \dot{\boldsymbol{q}}(t_f)) + \int_{t_0}^{t_f} \mathcal{L}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t))dt \quad (11a)$$

subject to

$$\frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) - \frac{d}{dt}\frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))$$
$$+ \boldsymbol{F}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t)) + \boldsymbol{\Phi}^T(\boldsymbol{q}(t))\lambda(t) = \boldsymbol{0}, \quad (11b)$$
$$\boldsymbol{\phi}(\boldsymbol{q}(t)) = \boldsymbol{0}, \quad (11c)$$
$$\boldsymbol{g}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \boldsymbol{u}(t)) \geq \boldsymbol{0}, \quad (11d)$$
$$\boldsymbol{\psi}_0(\boldsymbol{q}(t_0), \dot{\boldsymbol{q}}(t_0)) = \boldsymbol{0}, \quad (11e)$$
$$\boldsymbol{\psi}_f(\boldsymbol{q}(t_f), \dot{\boldsymbol{q}}(t_f)) = \boldsymbol{0} \quad (11f)$$

for $t \in [t_0, t_f]$.

Thus, we want to minimize a performance index (11a), consisting of the terminal cost, χ, and the integral of the control Lagrangian, $\mathcal{L}$, along the trajectory, while satisfying the dynamics (11b)-(11c), path constraints (11d), and boundary conditions (11e)-(11f).

It is well known that the discrete mechanics formulation of the equations of motion show excellent conservation of quantities, such as momenta and energy, conserved by the continuous system. This will enable us to take larger time steps and still get physically meaningful results [14]. There is, however, yet another computational advantage when used in optimal control. Namely, since there are no explicit references to velocities in the discrete equations of motion, the resulting optimization problem can be formulated using fewer variables, compared to standard discretizations of trajectories on TQ.

Approximating the objective using the midpoint rule and enforcing the path constraints at the midpoints we get the following discrete optimal control problem: Minimize

$$J_d = \chi(\boldsymbol{q}^{(N)}, \dot{\boldsymbol{q}}^{(N)})$$
$$+ \sum_{i=0}^{N-1} h\mathcal{L}\left(\frac{\boldsymbol{q}^{(i)} + \boldsymbol{q}^{(i+1)}}{2}, \frac{\boldsymbol{q}^{(i+1)} - \boldsymbol{q}^{(i)}}{h}, \boldsymbol{u}^{(i)}\right) \quad (12a)$$

subject to

$$D_2 L(\boldsymbol{q}^{(0)}, \dot{\boldsymbol{q}}^{(0)}) + D_1 L_d(\boldsymbol{q}^{(0)}, \boldsymbol{q}^{(1)}, h)$$
$$+ F_d^-(\boldsymbol{q}^{(0)}, \boldsymbol{q}^{(1)}, \boldsymbol{u}^{(0)}, h) + \frac{1}{2}\boldsymbol{\Phi}^T(\boldsymbol{q}^{(0)})\lambda_d^{(0)} = \boldsymbol{0}, \quad (12b)$$

$$D_2 L_d(\boldsymbol{q}^{(k-1)}, \boldsymbol{q}^{(k)}, h) + D_1 L_d(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, h)$$
$$+ F_d^+(\boldsymbol{q}^{(k-1)}, \boldsymbol{q}^{(k)}, \boldsymbol{u}^{(k-1)}, h)$$
$$+ F_d^-(\boldsymbol{q}^{(k)}, \boldsymbol{q}^{(k+1)}, \boldsymbol{u}^{(k)}, h) + \boldsymbol{\Phi}^T(\boldsymbol{q}^{(k)})\lambda_d^{(k)} = \boldsymbol{0}, \quad (12c)$$

$$-D_2 L(\boldsymbol{q}^{(N)}, \dot{\boldsymbol{q}}^{(N)}) + D_2 L_d(\boldsymbol{q}^{(N-1)}, \boldsymbol{q}^{(N)}, h)$$
$$+ F_d^+(\boldsymbol{q}^{(N-1)}, \boldsymbol{q}^{(N)}, \boldsymbol{u}^{(N-1)}, h) + \frac{1}{2}\boldsymbol{\Phi}^T(\boldsymbol{q}^{(N)})\lambda_d^{(N)} = \boldsymbol{0},$$
$$(12d)$$

$$\boldsymbol{\phi}(\boldsymbol{q}^{(k)}) = \boldsymbol{0}, \quad (12e)$$

$$\boldsymbol{g}\left(\frac{\boldsymbol{q}^{(k)} + \boldsymbol{q}^{(k+1)}}{2}, \frac{\boldsymbol{q}^{(k+1)} - \boldsymbol{q}^{(k)}}{h}, \boldsymbol{u}^{(k)}\right) \geq \boldsymbol{0}, \quad (12f)$$

$$\boldsymbol{\psi}_0(\boldsymbol{q}^{(0)}, \dot{\boldsymbol{q}}^{(0)}) = \boldsymbol{0}, \quad (12g)$$

$$\boldsymbol{\psi}_f(\boldsymbol{q}^{(N)}, \dot{\boldsymbol{q}}^{(N)}) = \boldsymbol{0}, \quad (12h)$$

$$h = \frac{t_f - t_0}{N}, \quad (12i)$$

$$h \geq 0, \quad (12j)$$

where $\dot{\boldsymbol{q}}^{(0)}$, $\dot{\boldsymbol{q}}^{(N)}$ are the initial and terminal velocities. The continuous optimal control problem (11) has now been transcribed into a nonlinear programming (NLP) problem of the form: Find the vector $\boldsymbol{x}$ minimizing the scalar objective function

$$f(\boldsymbol{x}) \quad (13a)$$

such that the constraints

$$\boldsymbol{c}_l \leq \boldsymbol{c}(\boldsymbol{x}) \leq \boldsymbol{c}_u \quad (13b)$$

and simple bounds

$$\boldsymbol{x}_l \leq \boldsymbol{x} \leq \boldsymbol{x}_u \quad (13c)$$

(a) Start   (b) Enter
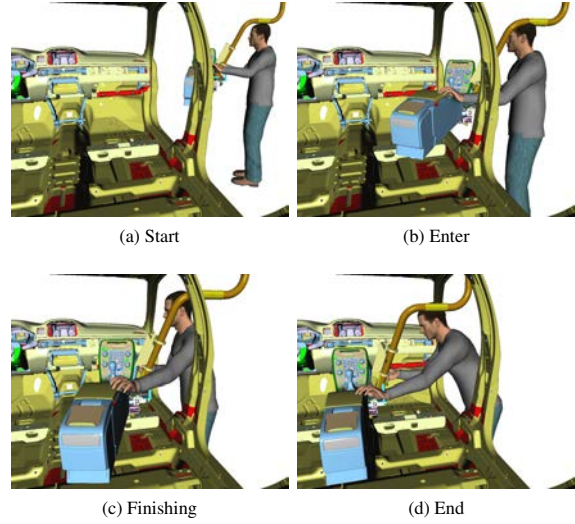
(c) Finishing   (d) End

Fig. 1: Automatic tunnel bracket assembly

are fulfilled.

An optimization problem of this form can be solved using nonlinear programming. Here we use the interior point solver IPOPT[15].

3. Quasi-static case studies

3.1. Tunnel bracket assembly

The first case is to install a tunnel bracket with the help of an auxiliary tool. The tunnel bracket and the auxiliary tool is connected by a rotation joint. The case is provided by Volvo Cars. The manikin starts outside the car with the tool and tunnel bracket already connected. The manikin grasps the tool with the left hand on a bar where the direction of the grasp is free and the right hand is connected with the fingertips to the tunnel bracket. After the setup, the assembly is completely automatic and guarantees that the motion is collision-free, except for the grasping hands, and that the manikin is in balance for each time step. The motion can be seen in figure 1. The simulation take 5.1 seconds to compute on a Intel i7 2600 computer. The forces required to move the tunnel bracket is quite low and the precision required for the final assembly step and thereby slow motion makes this a good case for the quasi-static solver.

3.2. Washer placing

The second case is to place washers inside the trunk of a car, this case is also provided by Volvo Cars. The case can be divided into two steps: first place the washers, and then to mount the bolts. Since both steps require the same reachability and force we choose to simulate only the washer placing. The manikin uses the left hand as support on the trunk floor to extend the reach, and the hand is free to rotate on that surface. The case is tried with 8 different manikins to cover the anthropometric variables length and weight, and also both sexes.
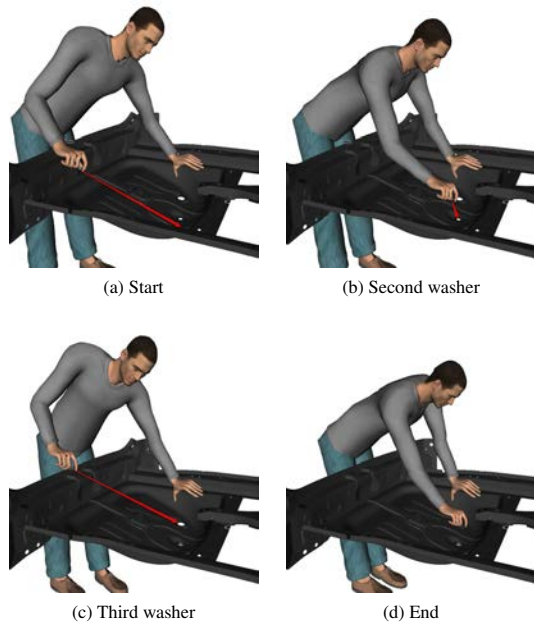
(a) Start   (b) Second washer

(c) Third washer   (d) End

Fig. 2: Placing of multiple washers for the 90 percentile male manikin

Fig. 3: Top: The reachability for the shortest manikin is insufficient
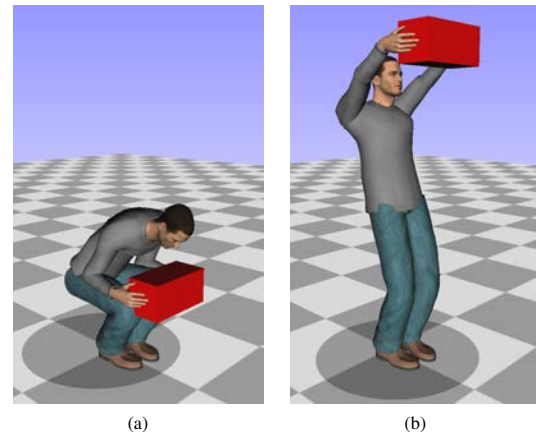Bottom: The end is in reach for the manikin

(a)   (b)

Fig. 4: Weight positions: (a) start, (b) finish

In figure 2, we see the 90 percentile male manikin perform the placing without any reachability issues. In figure 3, we see the difference between 5 percentile lady versus 50 percentile male where the lady can not reach all the way. This simulation takes an average of 14.5 seconds for all eight manikins on a Intel i7 2600 computer. The washers only weigh a few grams each and the precision in which they need to be assembled, and thereby the slow motion, makes this a good case for the quasi-static solver.

4. Dynamic case study

Here we compute trajectories for the manikin using the optimal control approach described Section 2.4. We then compute quasi-static solutions along the optimal paths, and compare the results. To make the problem more computationally attractive, we reduce the manikin model to a mechanical model of 40 degrees of freedom. This is done by removing joints, primarily in the spine and hands. The example we study is a lifting operation using both hands, moving a weight from one predefined position to another, starting and ending at rest. We chose the height of the initial position of the weight to be 0.5 m above the ground plane and place the finish position at 1.8 m, while orientation and horizontal positions are identical, the positions can be seen in Figure 4. The weight is modeled as a rigid body, adding another six degrees of freedom to the system. To model contact, rigid constraints are added between the weight and the two hands, and also between the feet and ground. The reaction forces from the ground are, however, only allowed to push on the manikin, and must also fulfill Coulomb friction conditions. The resulting discrete optimal control problem has the structure of (12) with:

$$\chi(\boldsymbol{q}, \dot{\boldsymbol{q}}) = 0$$
$$\mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{u}) = \boldsymbol{u}^T \boldsymbol{u}$$
$$\boldsymbol{\psi}_0(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \dot{\boldsymbol{q}}$$
$$\boldsymbol{\psi}_f(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \dot{\boldsymbol{q}}$$

where the control signal, $\boldsymbol{u}$, is chosen to be the normalized actuator torque. The problem is then solved for both a 10 kg and
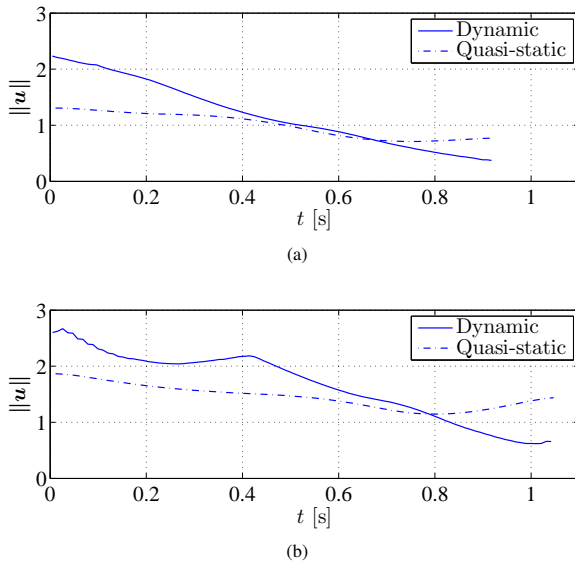
Fig. 5: Control effort in weight lifting example: (a) 10 kg, (b) 20 kg

a 20 kg weight using techniques from [16]. This results in two optimal trajectories for the system: the trajectory for the 10 kg weight with a duration of 0.92 s, and the trajectory for the 20 kg weight, which has a duration of 1.05 s. The quasi-static control signal, $\{\boldsymbol{u}_s^{(i)}\}_{i=1}^N$, is then computed as the steady state solution with minimum norm along the discrete trajectory, $\{\boldsymbol{q}^{(i)}\}_{i=1}^N$, i.e. for each $i = 1, \ldots, N$: Minimize

$$(\boldsymbol{u}_s^{(i)})^T \boldsymbol{u}_s^{(i)}$$

subject to

$$\frac{\partial L}{\partial \boldsymbol{q}}(\boldsymbol{q}^{(i)}, \boldsymbol{0}) + \boldsymbol{F}(\boldsymbol{q}^{(i)}, \boldsymbol{0}, \boldsymbol{u}_s^{(i)}) + \Phi^T(\boldsymbol{q}^{(i)})\lambda_s^{(i)} = \boldsymbol{0},$$

where $\boldsymbol{u}_s^{(i)}$ and $\lambda_s^{(i)}$ are decision variables.

In Figure 5, we compare the control signal magnitudes for the dynamic and quasi-static solutions. As expected the dynamic solutions, on average, require more control effort than the quasi-static solutions. In particular in the beginning of the lift, where a considerable effort is needed to accelerate both the weight and the manikin itself. It is interesting to note that in the end of the lift the dynamic solutions actually require less torque. This is explained by the fact that the direction of the lift is upward, hence the gravitational pull helps the deceleration.

5. Conclusions

In this paper we showed the importance of modeling dynamics when planning for manual assembly operations. Two case studies where performed on industrial cases, giving examples of where the quasi-static solution is sufficient. To demonstrate the dynamic effects, a third test case was studied, which indicates the importance of modeling dynamics in lifting operations. There is still work to be done before the dynamic solver reaches the maturity of the quasi-static solver. In particular, the solver needs to be equipped with collision avoidance and a comfort function.

References

[1] A.-C. Falck, R. Örtengren, D. Högberg, The impact of poor assembly ergonomics on product quality: A costbenefit analysis in car manufacturing, Human Factors and Ergonomics in Manufacturing & Service Industries 20 (1) (2010) 24–41.

[2] A.-C. Falck, M. Rosenqvist, A model for calculation of the costs of poor assembly ergonomics (part 1), International Journal of Industrial Ergonomics 44 (1) (2014) 140–147.

[3] D. Lämkull, L. Hanson, R. Örtengren, Uniformity in manikin posturing: A comparison between posture prediction and manual joint manipulation, International Journal of Human Factors Modelling and Simulation 1 (2008) 225–243.

[4] R. Bohlin, N. Delfs, L. Hanson, D. Högberg, J. Carlson, Unified solution of manikin physics and positioning - exterior root by introduction of extra parameters, Proceedings of DHM, First International Symposium on Digital Human Modeling.

[5] N. Delfs, R. Bohlin, S. Gustafsson, P. Mårdberg, J. S. Carlson, Automatic creation of manikin motions affected by cable forces, Procedia CIRP 23 (2014) 35–40.

[6] H. L. B. R. Högberg, D., J. Carlson, Creating and shaping the dhm tool imma for user-centred product and production design, International Journal of the Digital Human (IJDH).

[7] J.-C. Latombe, Robot motion planning, Vol. 124, Springer Science & Business Media, 2012.

[8] S. M. LaValle, Planning algorithms, Cambridge university press, 2006.

[9] R. Westgaard, A. Aarås, The effect of improved workplace design on the development of work-related musculo-skeletal illnesses, Applied Ergonomics 16 (2) (1985) 91–97.

[10] J. Rasmussen, M. Damsgaard, E. Surma, S. T. Christensen, M. de Zee, V. Vondrak, Anybody-a software system for ergonomic optimization 4.

[11] J. E. Marsden, M. West, Discrete mechanics and variational integrators, Acta Numerica 2001 10 (2001) 357–514.

[12] S. Leyendecker, J. Marsden, M. Ortiz, Variational integrators for constrained dynamical systems, ZAMM 88 (9) (2008) 677–708.

[13] S. Leyendecker, S. Ober-Blöbaum, J. E. Marsden, M. Ortiz, Discrete mechanics and optimal control for constrained systems, Optimal Control Applications and Methods 31 (6) (2010) 505–528.

[14] A. Lew, J. E. Marsden, M. Ortiz, M. West, An overview of variational integrators.

[15] A. Wächter, L. T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Mathematical Programming 106 (1) (2006) 25–57.

[16] S. Björkenstam, J. S. Carlson, B. Lennartson, Exploiting sparsity in the discrete mechanics and optimal control method with application to human motion planning, in: Automation Science and Engineering (CASE), 2015 IEEE International Conference on, IEEE, 2015, pp. 769–774.