ARCHIVE OF MECHANICAL ENGINEERING

Volume 67 2020 Number 3

DOI: 10.24425/ame.2020.131695

Key words: wall climbing, hand-to-handle interaction, contact dynamics, support force approximation, quasi-static solution

Grzegorz Orzechowski [1,2], Perttu Hämäläinen [3], Aki Mikkola[1]

Force approximation of the human hand in contact with a climbing wall handle

This paper presents a new algorithm that approximates the forces that develop between a human hand and the handles of a climbing wall. A hand-to-handle model was developed using this algorithm for the Open Dynamics Engine physics solver, which can be plugged into a full-body climbing simulation to improve results. The model data are based on biomechanical measurements of the average population presented in previously published research. The main objective of this work was to identify maximum forces given hand orientation and force direction with respect to the climbing wall handles. Stated as a nonlinear programming problem, solution was achieved by applying a stochastic Covariance Matrix Adaptation Evolution Strategy (CMA-ES). The algorithm for force approximation works consistently and provides reasonable results when gravity is neglected. However, including gravity results in a number of issues. Since the weight of the hand is small in relation to the hand-to-handle forces, neglecting gravity does not significantly affect the reliability and quality of the solution.

1. Introduction

Planning the motions for a climbing simulation involving multi-limbed agents is difficult because of the large number of degrees of freedom and the dynamic nature of the movement [1, 2]. One of the most challenging details is related to

✉ Grzegorz Orzechowski, email: grzegorz.orzechowski@lut.fi

[1] Department of Mechanical Engineering, LUT University, Lappeenranta, Finland. Email: aki.mikkola@lut.fi

[2] Mevea Ltd., Lappeenranta, Finland.

[3] Department of Computer Science, Aalto University, Espoo, Finland. Email: perttu.hamalainen@aalto.fi

how the fingers grasp the handholds [3]. Humans can effortlessly form a grip that is appropriate to a given task. However, in automation and robotics, and in simulation, properly defining grasp configurations is complicated by the many constraints, contacts, and measures.

As suggested by Cutkosky [4], how grasp is defined depends on the expected conditions and measurement requirements (forces, sensitivity, mobility, *etc.*) and the attributes of the interacting objects (geometry, texture, fragility, *etc.*). Cutkosky also presented a partial taxonomy of grasp configurations (used in manufacturing) that are dependent on object size, precision/power requirements, and geometry. Automatic grasp selection is difficult, because of the large number of possible grasp configurations and the large number of variables involved. This problem is usually solved by applying grasp heuristics, often based on grasp databases for given shapes [5–7]. Most grasp investigations are performed for robotic multi-finger actuators and augmented prostheses and exoskeletons [8]. Grasp quality is often evaluated in terms of stability using an analytical ϵ-metric [9] that aims at balancing arbitrary external perturbations with minimal finger forces. While it is shown that this criterion ignores force and torque limits and is not robust in real world applications [6], it can adequately satisfy the typical objectives for robotic manipulator modeling.

This study introduces a hand-to-handle model developed to solve hand-grasping problems for typical wall climbing applications. Both the human hand and climbing wall handles of the climbing wall are modeled using an Open Dynamics Engine (ODE) [10]. The resulting model is used to approximate the force on the handle that can be supported by the hand. The standard quality criterion for grasp stability cannot be used here, because the applied force and torque limitations for the biomechanical hand must be considered. In addition, the object being grasped is fixed in this case, which differs from the typical scenario in which a movable object is grasped so that it can be moved. Therefore, this study proposes an alternative set of objectives. Hand-to-handle grasping configurations are based on a hand configuration database developed by observing experienced human wall climbers. Moreover, the determinations of maximum force are accomplished via nonlinear programming (NLP).

The hand-to-handle model introduced here is intended for use in a full-body wall climber model similar to the one presented by Naderi *et al.* [2]. The wall climber model will define hand orientation and the direction of forces on the handle. The hand-to-handle model, in turn, will provide the target position of the hand, the grasp configuration, and the maximum support forces. In the original climber model, hand-to-handle interaction was modeled as spherical joint. Despite the simplicity of this approach, the climber model generates a realistic solution to the path-planning problem. Nevertheless, the inclusion of realistic hand-to-handle interaction will allow for solutions that are more robust.

2. Hand model and force computation

Wall climbers interact with their environment through the multiple contact points between their hands and feet and the climbing wall and wall handles. From the computational point of view, hand-to-handle interaction is complicated by the complexities of handle geometry and the hand model, the multi-point contacts with friction between the hand and the handle, and the need to establish an appropriate grasp configuration.

In this study, only hand-to-handle interaction is considered. Accordingly, the proposed model consists of only the wall, the handle, and the hand, which itself comprises the palm and finger phalanxes. All bodies are considered rigid and are modeled using simple geometries. The wall, handle, and palm are modeled as rectangular cuboids, while the phalanxes are modeled as capsules (cylinders with hemispheres at the ends). These simple geometries enable the realistic capture of hand-to-handle contacts and efficient contact point searching. By allowing the handle to change inclination with respect to the wall and by varying the sizes of its sides, a large variety of typical handles can be modeled.

Fig. 1 illustrates the proposed hand model. It is made up of sixteen bodies: the palm and three bodies for each of the five fingers. All finger joints are rotational

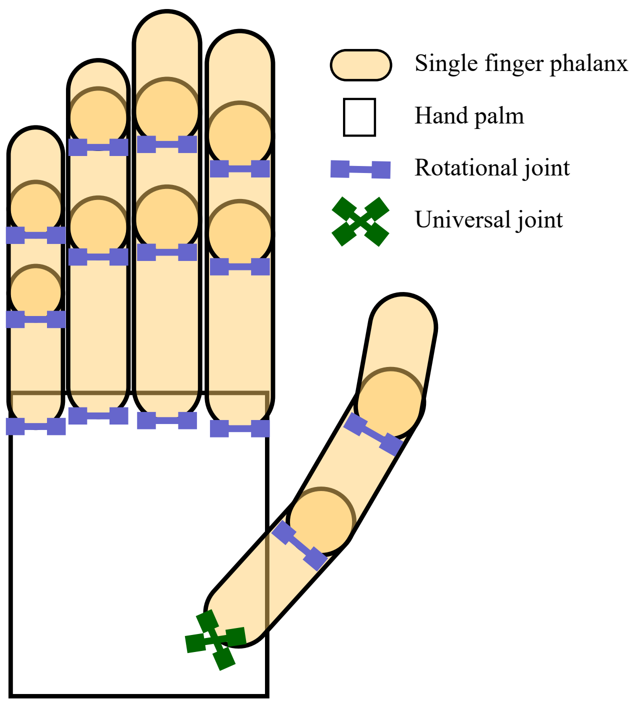


Fig. 1. Multibody hand model in its initial configuration. The phalanx capsules overlap to avoid hooking between the climbing wall handle and the knuckles of the hand

except for the one joint between the palm and the first phalanx of the thumb. This is modeled as a universal joint. The use of rotational joints between finger phalanxes means that finger abduction and adduction movements cannot be modeled. However, since climbers tend to keep their fingers together, this simplification should not adversely influence results.

The maximum forces and torques allowed in the joints, and the joint rotation limits are based on measured human biomechanical data [11–13]. Hand dimensions come from an external human model developed using the Unity3D environment (also used to run the master climber model). Hand dimensions in the model can be adjusted to accommodate each individual climber's needs. Total hand mass is set to about 450 g, which is typical for a human male [14]. The palm has been modeled based on a cuboid with dimensions of 104.6 mm × 88.7 mm × 40 mm (height × width × depth) and mass of 371.4 g. Therefore, mass moments of inertia of the palm are 293.2 kg mm^2, 388.4 kg mm^2, and 582.5 kg mm^2 for an axis of rotation at, respectively, the height, width, and depth. Capsules of the hand phalanxes have dimensions and inertia properties as presented in Table 1.

Table 1.

Hand phalanxes sizes and inertia properties

Phalanx	h_c [mm]	r_c [mm]	m_c [g]	I_h [kg mm^2]	I_r [kg mm^2]
Index 1	55.5	12	32.3	2.22	20.70
Index 2	33.9	12	22.6	1.52	7.71
Index 3	25.0	12	18.5	1.23	4.49
Middle 1	57.9	12	33.4	2.30	22.70
Middle 2	37.1	12	24.0	1.63	9.15
Middle 3	23.0	12	17.6	1.17	3.91
Ring 1	54.5	11	26.3	1.52	15.50
Ring 2	37.1	11	19.7	1.12	7.02
Ring 3	20.0	11	13.2	0.73	2.33
Pinkie 1	36.8	10	15.7	0.745	5.17
Pinkie 2	28.1	10	13.0	0.609	3.07
Pinkie 3	29.0	10	13.3	0.623	3.25
Thumb 1	45.1	12	27.7	1.89	13.50
Thumb 2	51.8	12	30.7	2.10	17.90
Thumb 3	29.6	12	20.6	1.38	6.01

Phalanxes with number 1 are connected directly to the palm, number 2 are in the middle, and the ones with the number 3 are at the fingertips. h_c is height of the cylindrical part of the capsule, r_c radius, m_c total mass, I_h moment of inertia for an axis of rotation at the height, and I_r is moment of inertia for an axis perpendicular to the height.

The Open Dynamics Engine (ODE), which has already been used successfully in various grasp simulations [6], is used for this model analysis [10]. For many applications, the ODE proves to be computationally efficient compared to state-of-the-art multibody solvers [15], and physics engines [16]. It can handle complex systems with multiple bodies and joints, collision detection, and friction.

The contact and friction model in ODE is based on multiple simplifications, that sacrifice physical accuracy over computational efficiency [10]. Collision detection is handled before each simulation step. After that, a list of contact points, surface normal vectors, and penetration depths is delivered to the ODE solver. Therefore, contact surface must be approximated with discrete points, and normal forces are computed assuming that all the contacts are frictionless. ODE allows the control of the spring and damping constants of the contact points (as well as in joints, joints limits, etc.) through two parameters w_{ERP} called joint Error Reduction Parameter (ERP) and w_{CFM} Constraint Force Mixing (CFM). More details about these parameters, are provided in the ODE Manual [10]. Stiffness k_{cp} and damping d_{cp} of the contact point can be computed as

$$\begin{aligned} k_{cp} &= \frac{w_{ERP}}{h w_{CFM}}, \\ d_{cp} &= \frac{1 - w_{ERP}}{w_{CFM}}, \end{aligned} \tag{1}$$

where h is the simulation step size. It is advised to set both w_{ERP} and w_{CFM} to small positive numbers. Both parameters must be nonnegative and w_{ERP} is at most, equal to one. For the simulations in the study, those parameters are $w_{ERP} = 9.74 \times 10^{-4}$ and $w_{CFM} = 3.19 \times 10^{-4}\,\mathrm{m\,N^{-1}\,s^{-1}}$. Time step is $h = 10^{-3}$ s. With those settings, stiffness at contact points is equal to $k_{cp} = 3.05\,\mathrm{kN\,m^{-1}}$ and damping coefficient is $d_{cp} = 3.13\,\mathrm{kN\,s\,m^{-1}}$. Parameters were chosen to get a stable and efficient simulation.

Friction model efficiency is achieved using linearization to approximate friction [17]. Friction model in ODE uses Coulomb model at contact points as $|f_T| \leqslant \mu |f_N|$ where f_N and f_T are normal and tangential forces and μ is the friction coefficient. Geometric interpretation of the Coulomb model results in the so called friction cone that is approximated in the the ODE with a pyramid. Moreover, the limit for the tangential force is computed once per simulation step as $f_T^h = \mu |f_N|$. The coefficient of friction between the handle and the hand is set to $\mu = 1.0$ [18, 19].

The maximum force that can develop between the hand and the climbing wall handle for a given hand orientation and force direction is estimated. The hand and handle forms an equilibrium system, and dynamic settling is employed to meet the equilibrium condition.

Simulation is divided into two stages: placement of the hand on the handle and application of the support force. The procedure is depicted in Algorithm 1. In the algorithm, δ represent small values greater than zero. The simulation itself is deterministic. It is implemented as a state machine and it contains no discrete events.

Algorithm 1 Computing force for a fixed palm position and a given hand target configuration

Require: hand orientation $\boldsymbol{\theta}_{\text{hand}}$, force director $\boldsymbol{e}_{\text{force}}$, initial hand position $\boldsymbol{r}_{\text{hand}}^{\text{init}}$, assumed hand fingers posture $\boldsymbol{\theta}_{\text{fingers}}^{\text{init}}$

Fix hand orientation to $\boldsymbol{\theta}_{\text{hand}}$ with constraint equations.

Apply artificial force $\boldsymbol{F}_{\text{hand}}^{\text{fix}}$ using PI controller with $\boldsymbol{err_r} = \boldsymbol{r}_{\text{hand}} - \boldsymbol{r}_{\text{hand}}^{\text{init}}$.

bool *in_equilibrium*() $\Leftarrow$ palm velocity $|\dot{\boldsymbol{r}}_{\text{hand}}| < \delta$ **and** hand in contact with handle **and** hand joint forces, and torques within a limit. {*Function to check if hand is settled.*}

repeat {*Stage 1 – place hand on the handle*}

  Apply torques to finger joints using PID controller with $\boldsymbol{err_\theta} = \boldsymbol{\theta}_{\text{fingers}} - \boldsymbol{\theta}_{\text{fingers}}^{\text{init}}$

until fingers angular velocity $\left|\dot{\boldsymbol{\theta}}_{\text{fingers}}\right| < \delta$ **and** *in_equilibrium*()

Fix hand grip by changing all finger joints to fixed. {*From now on* $\boldsymbol{\theta}_{\text{fingers}}$ *is constant.*}

Apply external support force $\boldsymbol{F}_{\text{hand}}^{\text{ext}} = F \cdot \boldsymbol{e}_{\text{force}}$ for small F.

while $\left|\boldsymbol{F}_{\text{hand}}^{\text{fix}}\right| \geq \delta$ **do** {*Stage 2 – find maximum support force*}

  while *in_equilibrium*() **and** $\left|\boldsymbol{F}_{\text{hand}}^{\text{fix}}\right| \geq \delta$ **do**

    decrease $\left|\boldsymbol{F}_{\text{hand}}^{\text{fix}}\right|$

  end while

  while *in_equilibrium*() **do** {*Will result in* $F_{\max}$ *if* $\left|\boldsymbol{F}_{\text{hand}}^{\text{fix}}\right| < \delta$}

    increase F

  end while

  if too many iterations **then**

    return FAILED

  end if

end while

return $F_{\max}$, $\boldsymbol{r}_{\text{hand}}^{\text{final}}$, $\boldsymbol{\theta}_{\text{fingers}}^{\text{final}}$

In the initialization stage, both the orientation and position of the palm are fixed by constraint equations and an artificial force. The force ensures that the hand maintains its position as the hand grip is formed. In the first stage, torques are applied to the finger joints until the fingers contact the climbing wall handle in the target position. Because of collision detection between the fingers, the wall, and the wall handle, the fingers can achieve a reasonable grip on the handle. After the grip is formed, the hand posture is fixed.

It is interesting to note that, the forming grip on the handle is an over-actuated problem [20]. In the model, the hand has sixteen actuators (one finger joint torque for each axis of rotation in the hand as shown in Fig. 1) and only three outputs (palm position). The human hand has even more actuators as up to 35 muscles are involved in the hand grasp [21]. Over-actuated systems have many advantages as they are fault tolerant, provide greater accuracy without extra mass added, enhance maneuverability, reduce time response, and others [22]. However, having more actuators than control outputs makes control complicated. For example, it is important for actuators to collaborate to achieve the control goal, as a counteraction may result in poor control quality and energy losses. There are special methods developed to control over-actuated systems, like control allocation approach [22]. However, those methods are complicated frameworks with limited applicability. Less complicated methods, e.g. rudimentary control, simply encourage actuator collaboration. The method used in the paper is similar to the master-slave rudimentary framework, where firstly "master" controls are determined (hand target pose), and then they are used on the "slave" control inputs (torques in the finger joints). Simplicity of the master-slave scheme makes the hand pose forming efficient and intuitive. However, the quality of the grip depends heavily on a selection of the "master" controls. The use of dedicated algorithm to control an over-actuated system may result in more robust solutions and allow wider applicability of the procedure without the need for customized "master" controls. But this variant has not yet been analyzed.

Maximum external support forces are computed in the second stage. In the main loop, the artificial forces are gradually decreased, while the support forces are gradually increased. Artificial forces must be decreased in stages mainly when gravity is included and when friction cannot keep the hand in place on its own. Once maximum force is determined, it is input back into the overall simulation.

The procedure described by Algorithm 1 allows for the calculation of maximum support forces when initial palm position and target finger angles are given. However, the main objective of the procedure is to compute maximum forces supported by the hand at the handle for a given hand orientation and force direction. This problem is handled as an unconstrained nonlinear programming task, with an objective function f defined as

$$f = \left|\boldsymbol{r}_{\text{hand}}^{\text{init}} - \boldsymbol{r}_{\text{hand}}\right| - F_{\max} w_{\text{fingertips}} \tag{2}$$

where $\boldsymbol{r}_{\text{hand}}^{\text{init}}$ and $\boldsymbol{r}_{\text{hand}}$ are, respectively, initial and final position of the hand, $F_{\max}$ is the maximum force supported by the hand, as output from the ODE simulation, and $w_{\text{fingertips}}$ is a coefficient that depends on the number of fingertips in contact with the handle. The function f was adjusted manually, to decrease the number of failed simulations. The first term in Equation (2) ensures that the hand does not move far from its initial position (large hand displacement indicates simulation problems). Coefficient $w_{\text{fingertips}}$ has a value of 1 when all five fingertips are in contact and lower values as the number of fingertips in contact drops. It is to encourage a firmer grip and discourage artificial solutions where no fingertips touches the handle (as can be seen for example in Fig. 5a). It is worth noting that the simulation often fails (especially in early stages of the optimization process), and does not provide any useful support force values. In such cases, a large penalty value is returned which depends on the distance that the model travels and simulation progresses. For details of the NLP procedure, see Algorithm 2. Palm orientation and force direction are provided in advance.

Algorithm 2 Procedure used for nonlinear programming

Require: hand orientation $\boldsymbol{\theta}_{\text{hand}}$, force director $\boldsymbol{e}_{\text{force}}$

1: **call** CMA-ES to minimize $f(x)$ where $x = \left[\boldsymbol{r}^{\text{init}}_{\text{hand}}, \boldsymbol{\theta}^{\text{init}}_{\text{fingers}}\right]$
2: **repeat** {*within CMA-ES call loop*}
3:     **call** Algorithm 1
4:     compute objective using Equation 2
5: **until** CMA-ES converge
6: **return** $F^{\text{best}}_{\text{max}}$, $\boldsymbol{r}^{\text{best}}_{\text{hand}}$, $\boldsymbol{\theta}^{\text{best}}_{\text{fingers}}$ {*all values for the run with best (smallest) fitness function*}

The ten optimization input variables include the three initial hand positions and seven variables that describe grasp configuration. The angles used for the target grasp configurations are based on eight postures demonstrated by an experienced climber gripping the target climbing wall handle. These 16 grasp angles are converted via principal component (PC) analysis into a set of seven PC variables. The seven PC variables are then input to the optimizer. They can be combined using normalized coefficients. This procedure reduces the dimensionality of the problem (from 16 angles to seven PCs in this case) and accounts for biomechanical dependencies between the hand angles. Having the feasible hand posture on the handle, complemented by initial hand position, the hand can be positioned on the handle as it is described in Stage 1 of Algorithm 1. PI and PID controllers parameters must be tuned accordingly, but once done, they have worked well for all scenarios.

The primary objective of the computation is to evaluate the maximum forces that can be supported. Preference is given to solutions with multiple finger-handle contact points (controlled by coefficient $w_{\text{fingertips}}$) and to solutions with small palm displacements. Adjustments are based on the observation that solutions with a small number or no finger-handle contacts and solutions with large hand displacements often result in unnatural grasp configurations. The hand postures shown in Fig. 5 are appropriate examples.

The nonlinear programming solution is the maximum force value (given in advance for hand orientation and force direction) together with the related hand position and finger angles. The NLP is solved using a Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm. CMA-ES was chosen as it has proven its efficiency and effectiveness in many practical applications with small population sizes [23], and the NLP is non-convex. In this case, the small population requirement is especially important, because the single force evaluation (in Algorithm 1) consumes several seconds of CPU time. Convexity of the problem cannot be assured due to sensitivity of the solution on the initial conditions (see

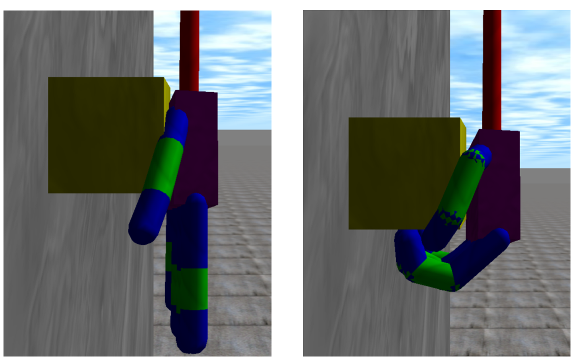
Algorithm 2). For example, two sets of initial finger angles that describe similar hand poses, often result in significantly different values of the support force or the number of fingertips that are in contact with the handle.

Analysts should recognize, however, that although the CMA-ES is a stochastic algorithm, convergence to a global optimum is not guaranteed.

3. Support force and hand postures

This section describes the procedure used to deal with support force and hand postures. For the full climber model, various hand orientations and force directions must be sampled. In this study, however, the search space is limited so that more coincident results can be presented. Force is presumed to act along the palm in a pulling direction. Moreover, the orientation of the hand with respect to the climbing wall handle has been limited to scenarios that only allow pulling down and pulling up from various angles. Pulling sideways is not allowed.

Fig. 2 illustrates the simulation stages described by lines four to seven in Algorithm 1. Fig. 2a shows the initial hand position with fully straight fingers. Here, the palm has a fixed orientation and a fixed force direction (indicated by the slender, red cylinder). It is placed at the initial hand position. In Fig. 2b, finger torques have been applied, and the hand is gripping the handle. To form a firm grip, the palm has had to move from its initial position. In the subsequent step, the finger position is fixed and external force is applied. The hand is allowed to translate in the second stage (lines eight to 16 in Algorithm 1), however, this rarely happens in good quality solutions. Moreover, despite the use of fixed joints between the fingers, the value of the torque that is used to support the hand and the applied forces is tested for each iteration.



(a) (b)

Fig. 2. Hand placement on the climbing wall handle showing: a) the hand at the start of the simulation and b) the hand after its finger configuration has been fixed to grip the handle; in this example, gravity had been neglected

The results are based on two independent test cases for 57 randomly chosen inputs (hand orientation angles within chosen limits). The first test case includes gravitational forces, while second one does not. In both cases, the PID torque controllers in the finger joints have identical settings. Neglecting gravitational forces greatly simplifies the numerical solution, and because hand mass is a fraction of total body mass, neglecting hand weight does not significantly affect full climber model results.

Table 2 gives a summary of the force value results of the analysis. From the analysis that neglects gravity, the results are more consistent than those from the analysis that includes the gravity forces. In particular, the inclusion of gravity seriously degraded results for simulation of the pulling up motion. Numerical analysis failures and extreme force values appeared only for this case. Further grasp configuration analyses will reveal more about the issues with simulations performed when gravity has been included.

Table 2.

Brief summary of the computed force values

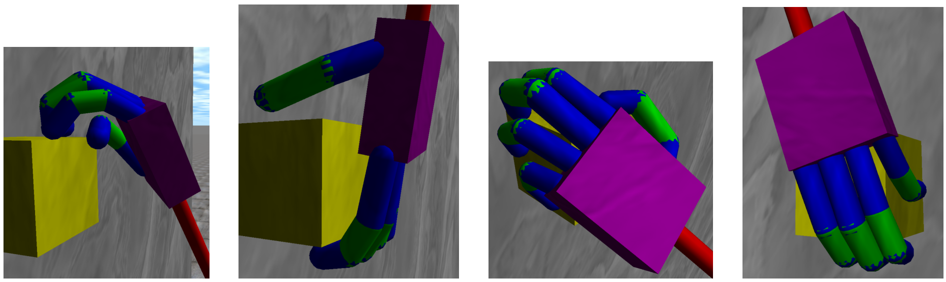
	Gravity included			Gravity not included		
	Total	Pull Up	Pull Down	Total	Pull Up	Pull Down
Count [–]	50	23	27	57	25	32
Minimum [N]	2.0	2.0	81.8	53.9	79.9	53.9
Maximum [N]	1072.4	1072.4	166.0	178.4	178.4	151.1
Average [N]	250.1	415.4	109.2	108.8	118.1	101.6
Std dev. [N]	332.7	439.6	18.4	26.6	27.1	24.1
Failures [–]	7	7	0	0	0	0

The count indicates the number of tests used to compute indicators. Failures are not included in the count, because they do not provide meaningful force values.

Interestingly, the largest forces supported by the hand in the model are too small to support a human climber with an average mass. For example, the maximum forces obtained for the gravity-neglected case correspond to a climber that weighs only 30 to 35 kg hanging on two hands. However, it is known that human climbers can easily hang on two hands even on the more challenging climbing wall handles. What is limiting generated force in the analyses are the torque limits computed for the hand joints.

The biomechanical data used in this study assumes the average population, and it is presumed that practicing climbers are able to strengthen their grip significantly making them capable of accommodating larger hand-to-handle forces. However, the authors do not have access to detailed biomechanical data for climbers.

Fig. 3 presents example configurations from a solution of the NLP problem for tests when gravity is neglected. All the results, despite being presented for extreme force values, give reasonable solutions with intuitive grasp configurations



(a) (b) (c) (d)

Fig. 3. Various grasp configurations for pulling with no gravity – The forces supported by the hand in each case are a) 53.9 N, b) 178.4 N, c) 71.8 N, and d) 167.8 N. The number of fingertips in contact with the climbing wall handle are a) five, b) four, c) five, and d) four. The solutions yielding the smallest and largest support forces are cases a) and b), respectively. Cases c) and d) result in the third smallest and third largest force

and forces. Other results from this category are of similar quality. Fig. 4, in turn, shows the support force values as a function of the angles between the horizontal axis and the force directions. As expected, force increases as angle decreases. That is to say, the hand and the forces become more aligned with the wall. The data points in this case, however, show significant dispersion.

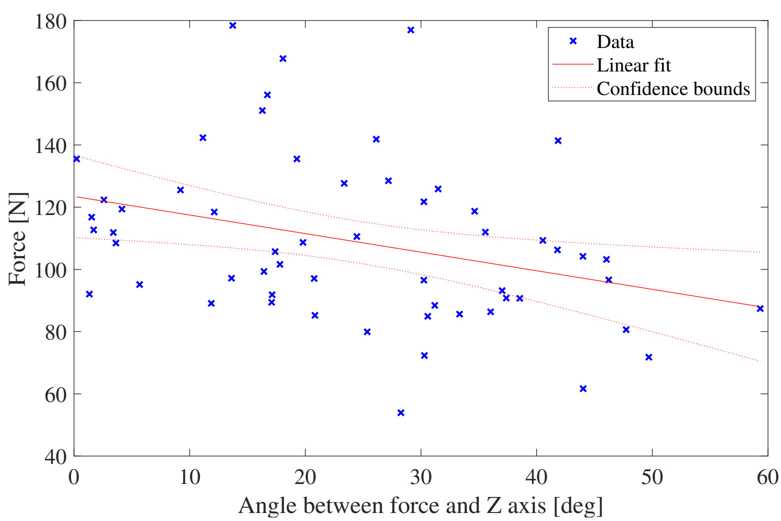
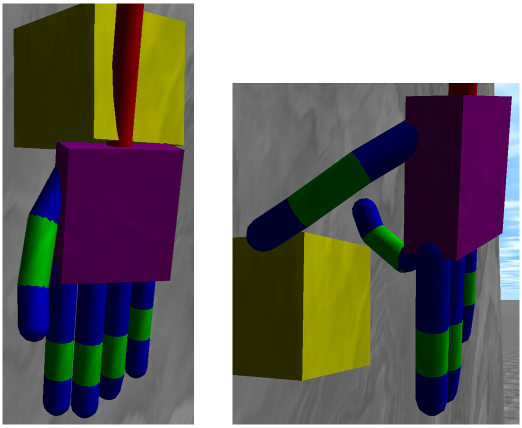


Fig. 4. Force value versus angle between the horizontal axis and force – The points represent the data from all 57 examples computed for simulation when gravity was neglected. The red line is a linear regression based on the data points

Fig. 5 presents solutions for cases that include gravity. The pulling up simulation cases have fewer contact points, small forces, and unusual hand configurations. The straight fingers in these configurations are the result of gravity forces over-



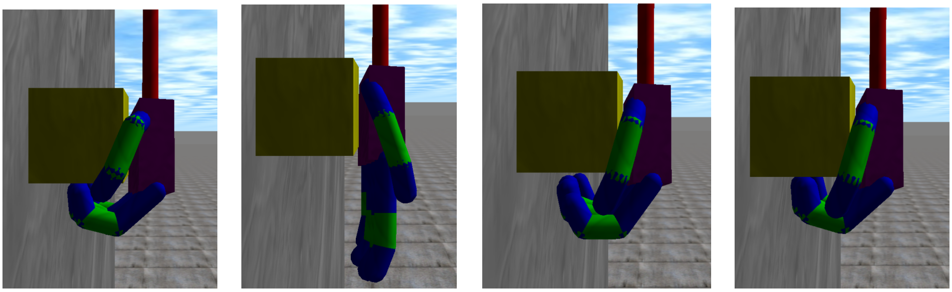
(a) (b)

Fig. 5. Various configurations for pulling with gravity included. The forces supported by the hand in each case are: a) 1072.4 N and b) 2.0 N. The number of fingertips in contact with the climbing wall handle are a) zero and b) one. The solutions yielding the smallest and largest support forces are cases a) and b), respectively

whelming the torque limits in the finger joints. This straightening of the fingers prevents the hand from adopting an appropriate grasp configuration for pulling up, which, in turn affects the quality of the solution.

Additional analysis is provided afterwards to investigate the effect on the solution of the PID settings that drive finger torques when gravity is included.

Fig. 6 shows the final configuration of the fingers for a different scenario. When gravity is not present (Fig. 6a), PID settings are sufficient to bend the fingers and allow fingertips to touch the climbing wall handle. In Fig. 6b, the PID settings are similar, but with the included gravity, the fingers are barely able to bend. When



(a) (b) (c) (d)

Fig. 6. The hand after the finger configuration is fixed with: a) gravity excluded, b) gravity included, PID parameters the same as in case (a), c) gravity included and the PID parameters increased 60 times over those of case (a) and (b), and d) gravity included, with the PID parameters increased 120 times over those of case (a) and (b)

PID parameters are increased 60 times (Fig. 6c), the fingers bend more, but they still do not touch the handle surface. A further increase in torque (120 times larger PID parameters with respect to a and b, as shown in Fig. 6d), allows the fingers to touch the handle without issue. However, as will be explained in the next paragraph, increasing the PID parameters did not improve simulation results.

The grasp configurations shown in Figs. 6a and 6b correspond to the model settings used to obtain the results presented in Table 2. In Table 3, two additional test cases for 57 randomly chosen inputs are simulated and summarized for the hand model settings that correspond to the hand configurations in Figs. 6c and 6d. For both test cases, the pull-up test simulations yield poor quality results with large differences in results and a high standard deviation. The pull-down results for the test case with PID values increased 60 times looks reasonable, but there is no clear improvement over the results in Table 2. Moreover, the pull-down test for the case with the PID parameters increased 120 times gives a larger dispersion of force values, which is not desirable.

Table 3.
A brief summary of the computed force values for test cases with increased PID finger torque parameters compared to the test cases summarized in Table 2

	Gravity 60×PID			Gravity 120×PID		
	Total	Pull Up	Pull Down	Total	Pull Up	Pull Down
Count [–]	48	16	32	41	16	25
Minimum [N]	2.9	2.9	67.5	23.6	28.7	23.6
Maximum [N]	11233.4	11233.4	157.1	11456.4	11456.4	934.4
Average [N]	446.0	1128.0	104.9	574.8	1262.5	134.7
Std dev. [N]	1617.7	2730.4	18.7	1779.7	2752.0	169.1
Failures [–]	9	9	0	16	11	5

Count number shows the number of tests used to compute the indicators. Failures were not included in the count, because they do not provide meaningful force values.

4. Conclusions

This paper presents a new procedure for computing the forces that can be supported by the hand of a climber when gripping a climbing wall handle. The problem has been stated as a nonlinear programming exercise that optimizes hand position and grasp configuration to get an estimate of maximum support forces. The results obtained for the model, when gravity is neglected, seem to be reasonable. The grasp configurations look natural for each hand-to-handle configuration, and the resultant force values are consistent with reasonably small dispersion. Since hand weight is a small fraction of total body weight, neglecting gravity in the hand simulation has a negligible effect on full climber model results.

The results of the simulations that included gravity were of worse quality. To get reasonable configurations for the pulling up (against gravity) scenario, the driving torques for the fingers had to be increased significantly as compared to the no-gravity solutions. However, despite the assumed PID settings, the results are not consistent, the resulting forces are extreme, and many of the simulations failed. Another issue with the simulations that included gravity is that the dynamic settling procedure is less robust. Therefore, due to numerous problems with the hand-to-handle model with included gravity, it is advised to rely on the results obtained based on the model without gravity.

According to the numerical results, hand-to-handle contacts can carry a force of approximately 110 N depending on grasp configuration. The calculated forces were constrained by the torque limits computed for the hand joints. The data used in this study were based on the biomechanical data from the average population. Experienced climbers have developed stronger grips. As a result, they are able to support larger forces. The introduced computational procedure can be tailored to be climber specific by utilizing measured gripping torque estimates.

The results of the presented procedure can be used to generate a data-driven model for hand-to-handle simulation, e.g., mapping arm and handle positions and rotations for optimal hand pose. Such a data-driven model could, for example, augment a full-body climber model. For more general applicability, the grip optimization problem can be solved in advance for various handle types and hand models. Furthermore, the presented procedure, after minor modification, may be used for a multitude of applications where the hand grasping problem must be solved efficiently, e.g., in computer games and animation.

Acknowledgements

The work was supported by the Academy of Finland under the decision No. 305738, and by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie project No. 845600 (RealFlex). The support is gratefully acknowledged.

Manuscript received by Editorial Board, May 20, 2020;
final version, August 31, 2020.

References

[1] T. Bretl. Motion planning of multi-limbed robots subject to equilibrium constraints: the free-climbingrobot problem. *The International Journal of Robotics Research*, 25(4):317–342, 2006. doi: 10.1177/0278364906063979.

[2] K. Naderi, J. Rajamäki, and P. Hämäläinen. Discovering and synthesizing humanoid climbing movements. *ACM Transactions on Graphics, Los Angeles*, 36(4):art.43, 2017. doi: 10.1145/3072959.3073707.

[3] A.T. Miller and P. K. Allen. Graspit! A versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, 11(4):110–122, 2004. doi: 10.1109/MRA.2004.1371616.

[4] M.R. Cutkosky. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE Transactions on Robotics and Automation*, 5(3):269–279, 1989.

[5] A. Herzog, P. Pastor, M. Kalakrishnan, L. Righetti, T. Asfour, and S. Schaal. Template-based learning of grasp selection. In *2012 IEEE International Conference onRobotics and Automation*, pages 2379–2384, Saint Paul, USA, 14–18 May 2012. doi: 10.1109/ICRA.2012.6225271.

[6] D. Kappler, J. Bohg, and S. Schaal. Leveraging big data for grasp planning. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311, Seattle, USA, 26–30 May 2015. doi: 10.1109/ICRA.2015.7139793.

[7] V. Lippiello, F. Ruggiero, B. Siciliano, and L. Villani. Visual grasp planning for unknown objects using a multifingered robotic hand. *IEEE/ASME Transactions on Mechatronics*, 18(3):1050–1059, 2013. doi: 10.1109/TMECH.2012.2195500.

[8] J. DeGol, A. Akhtar, B. Manja, and T. Bretl. Automatic grasp selection using a camera in a hand prosthesis. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 431–434, Orlando, USA, 16-20 August 2016. doi: 10.1109/EMBC.2016.7590732.

[9] C. Ferrari and J. Canny. Planning optimal grasps. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, pages 2290–2295, Nice, France, May 1992.

[10] R. Smith. *Open Dynamics Engine: User Guide*. 2006.

[11] C.J. Hasser. Force-Reflecting Antropomorphic Hand Masters. Technical Report AL/CF-TR-1995-0110, Armstrong Laboratory, Ohio, USA, 1995.

[12] F. Wang, M. Shastri, C.L. Jones, V. Gupta, C. Osswald, X. Kang, D.G. Kamper, and N. Sarkar. Design and control of an actuated thumb exoskeleton for hand rehabilitation following stroke. In *2011 IEEE International Conference on Robotics and Automation*, pages 3688–3693, Shanghai, China, 9-13 May 2011. doi: 10.1109/ICRA.2011.5980099.

[13] Y. Yoshii, H. Yuine, O. Kazuki, W-L. Tung, and T. Ishii. Measurement of wrist flexion and extension torques in different forearm positions. *BioMedical Engineering OnLine*, 14:art.115, 2015. doi: 10.1186/s12938-015-0110-9.

[14] S. Plagenhoef, F.G. Evans, and T. Abdelnour. Anatomical data for analyzing human motion. *Research Quarterly for Exercise and Sport*, 54(2):169–178, 1983. doi: 10.1080/02701367.1983.10605290.

[15] N. Niemi. *Comparison of Open Dynamics Engine, Chrono and Mevea in simple multibody applications*. PhD Thesis, LUT University, Lappeenranta, Finland, 2017.

[16] T. Erez, Y. Tassa, and E. Todorov. Simulation tools for model-based robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4397–4404, Seattle, USA, 26-30 May 2015. doi: 10.1109/ICRA.2015.7139807.

[17] E. Drumwright, J. Hsu, N. Koenig, and D. Shell. Extending open dynamics engine for robotics simulation. In: N. Ando, S. Balakirsky, T. Hemker, M. Reggiani, and O. von Stryk, editors, *Simulation, Modeling, and Programming for Autonomous Robots*, Lecture Notes in Computer Science, 6472:38–50. Springer, Berlin, Heidelberg, 2010. doi: 10.1007/978-3-642-17319-6_7.

[18] M.J. Carré, S.E. Tomlinson, J.W. Collins, and R. Lewis. An assessment of the performance of grip enhancing agents used in sports applications. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 226(7):616–625, 2012. doi: 10.1177/1350650112439647.

[19] F.K. Fuss, G. Niegl, and A.M. Tan. Friction between hand and different surfaces under different conditions and its implication for sport climbing. In: *The Engineering of Sport 5*, volume 2, pages 269–275, University of California, Davis, 2004.

[20] M.G.E. Schneiders, M.J.G. van de Molengraft, and M. Steinbuch. Benefits of over-actuation in motion systems. In *Proceedings of the 2004 American Control Conference*, volume 1, pages 505–510, Boston, USA, June 2004. doi: 10.23919/ACC.2004.1383653.

[21] A.E. Flatt. Grasp. *Baylor University Medical Center Proceedings*, 13(4):343–348, 2000. doi: 10.1080/08998280.2000.11927702.

[22] M. Duan. *Energy-Optimal Control of Over-Actuated Systems – with Application to a Hybrid Feed Drive*. Ph.D. Thesis, University of Michigan, Ann Arbor, Michigan, USA, 2018.

[23] N. Hansen. The CMA Evolution Strategy: A Comparing Review. In: J.A. Lozano, P. Larrañaga, I. Inza, and E. Bengoetxea, editors, *Towards a New Evolutionary Computation*, Studies in Fuzziness and Soft Computing, vol. 192, pages 75–102. Springer, Berlin, Heidelberg, 2006. doi: 10.1007/3-540-32494-1_4.