A Survey on Inverse Dynamics Solvers for Cable-Driven Parallel Robots

Yin Pok Chan[1], Jonathan Eden[2], Darwin Lau[1] and Denny Oetomo[2]
[1]The Chinese University of Hong Kong, Hong Kong {ypchan,darwinlau}@mae.cuhk.edu.hk
[2]The University of Melbourne, Melbourne, Australia {jpeden@student.,doetomo@}unimelb.edu.au

Abstract

In the force control of cable-driven parallel robots (CDPRs), inverse dynamics (ID) algorithms are required that can consider the actuation constraints imposed by cables. As a result, a variety of different ID solution methods have been developed. This paper uses the open-source Cable-robot Analysis and Simulation Platform for Research (CASPR) to compare the properties exhibited by different ID solvers over a wide range of different CDPR types with different mechanism parameters. The results are analysed to provide better insight into determining the appropriate ID solver for different applications and a test set of CDPRs is developed to be used as a benchmarking set for future ID research.

1 Introduction

Cable-driven parallel robots (CDPRs) are parallel robots that use cables for the robot actuation instead of rigid links. Due to properties like low inertia, CDPRs have a number of advantages including a large operating workspace and the capability to produce highly dynamic movements, which leads to a range of potential industrial applications [1; 2; 3; 4]. Nonetheless, the inability for cables to produce pushing forces induces a constraint on the number of cables. To achieve full manipulation of a CDPR, it is necessary that the number of cables, m, is larger than the number of degree of freedoms (DoFs), n, i.e., $m > n$, which results in actuation redundancy [5].

CDPRs can be controlled through either end-effector position or force control, where the cable lengths or forces need to be resolved for, respectively. For position control, the cable lengths required to achieve a certain configuration of the mechanism are solved through inverse kinematics (IK). As cable forces are not directly controlled in IK, its use can result in cable slack or undesirable high tensions [6].

In contrast to position control, force control uses the system's dynamics and equations of motion, to generate a desired motion for the mechanism through the determination of a set of feasible cable forces. This process is also known as inverse dynamics (ID). When $m > n$, ID can have potentially infinite different solutions due to actuation redundancy. This redundancy gives flexibility in achieving different objectives, such as the minimization of cable forces. At the same time, the redundancy raises the problem of how to determine the best set of cable forces among the infinite number of solutions.

To solve the redundant ID problem, a number of different methods have been developed. These methods can generally categorised into: 1) optimisation based; and 2) heuristic based methods. Optimisation based methods aim to find a set of cable forces that optimise a physically meaningful objective function. A number of different objectives have been considered including the minimization of the 1-norm [7; 8], 2-norm [7; 8] and ∞-norm [9] of the cable forces as well as the calculation of "optimally safe" force distributions [6]. These methods typically provide physically desirable cable force selections. However, the optimisation based methods possess high worst-case computational time, limiting their applicability to real time systems. In light of this, other methodologies focus more on the computational efficiency and the boundedness of the number of computations [10; 11; 12]. These methods make use of heuristics and therefore do not explicitly consider optimality. Different ID methodologies produce solutions with different properties such as continuity and real-time capability.

Given a collection of ID solvers, it is an important and difficult task for CDPR researchers to determine the right solver for their robot and application. For instance, with an operating frequency requirement, a particular ID method may not be able to produce solutions. Furthermore, a change in the CDPR, such as the number of cables, can result in different ID methods having different changes in their computational time. A clear comparison of how different solvers perform in various situations can therefore help researchers determine a suitable ID method.

To evaluate the differences between solution methods, Pott formulated a table that made a general comparison of properties exhibited by different solvers, such as real-time capability and continuity [11]. The computational time of the different solvers was recorded and compared by doing simulations on a single sample trajectory for a single CDPR. Despite the general insight offered by these tests, the issue of how these solvers perform for CDPRs with different parameters including the number of links, number of DoFs and number of cables, remains unexplored. Due to the lack of variations in the CDPR parameters tested, the comparison results can hardly be extended to other CDPRs.

Traditionally, the evaluation of different ID methods can be a challenging task. This is because the evaluation should consider variations in CDPR parameters such as the number of links, number of cables and number of DoFs, where for each variation it is necessary to re-derive the system kinematics and dynamics. Recently, an open-source platform CASPR [13], which uses a generalised model formulation [7] was developed. CASPR is capable of modelling different types of CDPRs, possesses a library of different ID solvers and allows for sets of CDPR trajectories to be simulated. Such a software platform represent an ideal tool for the fair comparison of ID solvers.

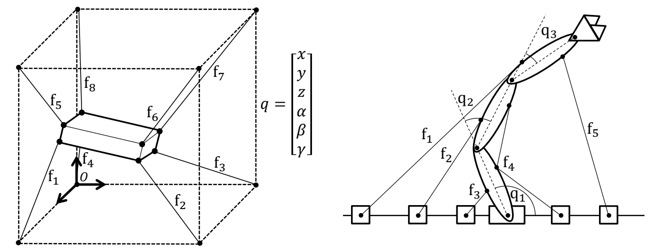
This paper uses the CASPR software framework to both model different CDPR parameters and provide a platform for the fair comparison of the performance of different ID solvers. A range of different desirable ID properties are proposed as tools for the analysis of ID solvers. Through the generation and analysis of a set of simulation results, the paper offers clearer insights for the selection of the appropriate solver for different applications. Additionally, the set of test robots developed represent a test set that can be used for the benchmarking of future ID algorithms.

2 Background on Inverse Dynamics

Figures 1a and 1b show the configurations of a spatial cable-driven robot (SCDR) and a multi-link cable-driven robot (MCDR), respectively. For a CDPR with n DoFs actuated by m cables, the system dynamics are given by the equation of motion

$$M(\mathbf{q})\ddot{\mathbf{q}} + \boldsymbol{\eta}(\dot{\mathbf{q}}, \mathbf{q}) = -L(\mathbf{q})^T\mathbf{f}, \tag{1}$$

where $\mathbf{q} \in \mathbb{R}^n$ is the mechanism configuration (*pose*), $M(\mathbf{q}) \in \mathbb{R}^{n\times n}$ refers to the mass-inertia matrix and $\boldsymbol{\eta}(\dot{\mathbf{q}}, \mathbf{q}) \in \mathbb{R}^n$ is a vector formed by the centrifugal, Coriolis and gravitational terms. The terms M and $\boldsymbol{\eta}$ form the system wrench which is related to the cable force



(a) Example spatial SCDR (b) Example MCDR

Figure 1: Example CDPRs

vector $\mathbf{f} \in \mathbb{R}^{m\times 1}$ by the transpose of the Jacobian matrix $L(\mathbf{q}) \in \mathbb{R}^{m\times n}$. The cable forces are constrained to be positive and are bounded by the minimum and maximum feasible cable forces, f_{min} and f_{max}, such that

$$0 \le f_{min} \le f_i \le f_{max}, \quad i \in \{1, \ldots, m\}. \tag{2}$$

In the process of solving inverse dynamics (ID), a set of cable forces, which satisfies the bound (2), should be determined to achieve the desired motion specified by the joint position $\mathbf{q}$, velocity $\dot{\mathbf{q}}$ and acceleration $\ddot{\mathbf{q}}$. Due to the actuation redundancy and cable force constraints, there exists either no feasible force set that can achieve the desired motion or infinite feasible combinations of cable forces that can produce the same desired motion.

2.1 Inverse Dynamics Solvers

Two main categories of solution methods have been developed for the redundant ID problem:

1. optimisation-based,
2. heuristic-based.

Optimisation-based methods

The main principle behind the optimisation approaches is to minimise a physically meaningful objective function while satisfying the constraints (1) and (2). The general formulation of optimisation methods is as follows

$$\begin{aligned} \mathbf{f}^* &= \arg\min_{\mathbf{f}} g(\mathbf{f}), \\ \text{subject to } & M(\mathbf{q})\ddot{\mathbf{q}} + \boldsymbol{\eta}(\dot{\mathbf{q}}, \mathbf{q}) = -L(\mathbf{q})^T\mathbf{f}, \\ & 0 \le f_{min} \le f_i \le f_{max}, \quad i \in [1, m], \end{aligned} \tag{3}$$

where the optimal force $\mathbf{f}^*$ is determined by minimising the objective $g(\mathbf{f})$ while satisfying the constraints.

A common class of objective function corresponds to the p-norm of the cable force vector $g(\mathbf{f}) = \|\mathbf{f}\|_p$, where $p \in \{1, 2, ..., \infty\}$. When (3) is solved with $p = 1$ and $p = 2$, the problem becomes a *linear program (LP)* [7], and a *quadratic program (QP)* [7], respectively. As the objective functions in both cases are convex, optimal force distributions can be found by well-established convex optimisation techniques. When $p = \infty$ [9], $g(\mathbf{f}) = ||\mathbf{f}||_\infty = \max|f_i|, i \in [1, m]$, (3) can be solved by the *minimum infinity-norm* method developed by Cadzow [14] and Shim [15], and further improved by Ha and Lee [16], which formulates the problem as a LP.

Instead of using a cable force norm minimising objective function, Borgstrom *et al.* [6] determine the '*optimally safe*' cable force distribution which represents the cable force that maximises the distance to the constraint boundaries. This approach is suggested as a safe way to determine cable forces such that there is no slack or cable breakage and can be formulated as the optimisation

$$S^* = \max_{\mathbf{f}, S} S,$$

$$\text{subject to } M(\mathbf{q})\ddot{\mathbf{q}} + \boldsymbol{\eta}(\dot{\mathbf{q}}, \mathbf{q}) = -L(\mathbf{q})^T \mathbf{f},$$

$$(f_{min} + S)\mathbf{1}_m^T \leq \mathbf{f} \leq (f_{max} - \alpha S)\mathbf{1}_m^T, \tag{4}$$

for some $\alpha \geq 0$, where S is the slack variable and $\mathbf{1}_m \in \mathbb{R}^m$ is a vector of ones.

Another method known as the *feasible polygon* method has been developed by Gouttefarde *et al.* [8] to perform ID on CDPRs with 2 degrees of redundancy ($r = m - n = 2$). This method exploits the unique properties of the 2 degree of redundancy system in which it is known that the feasible set of solutions form a convex polygon, referred to as the *feasible polygon*. By geometric reasoning, the vertices of the feasible polygon can be found in a clockwise or anti-clockwise order, and the feasible polygon can then be used to obtain the solutions for different objective functions.

Heuristic-based methods

Optimisation-based methods can lead to challenges in real-time control, because their worst-case computational time is often unbounded or very large [11]. This motivates the development of heuristic approaches, which emphasise computational efficiency at the cost of solution optimality and/or flexibility.

Pott *et al.* [10] proposed the closed-form method (*CFM*) ID solver. In this method, the candidate cable force solution $\tilde{\mathbf{f}}$ is given by the closed-form expression

$$\tilde{\mathbf{f}} = \mathbf{f}_m - L(\mathbf{q})^{+T} \left(\mathbf{w} + L(\mathbf{q})^T \mathbf{f}_m \right), \tag{5}$$

where the mean feasible force vector $\mathbf{f}_m$ is defined such that it has components $f_{m_i} = \frac{f_{min} + f_{max}}{2}$, and $\mathbf{w}$ refers to the system wrench. If $\tilde{\mathbf{f}}$ satisfies the actuation constraints (2), then the ID problem is declared feasible and the solution is returned, otherwise the problem is identified as infeasible.

When the closed form method identifies a feasible solution, its solution corresponds to the solution of the optimisation (3) with objective function $g(\mathbf{f}) = \|\mathbf{f} - \mathbf{f}_m\|_2$. As a result of its fixed solution form, the CFM possesses a well-defined and bounded computational time, as only matrix multiplications, transposes and an inversion is involved in its computation [10]. The fixed solution form, however, also results in cases where the CFM fails to identify a feasible solution although one exists.

An improved version of CFM (*ICFM*) has hence been proposed to increase the set of problems where a feasible solution can be found [11]. This is achieved by iteratively identifying infeasible cable force terms and then fixing the terms with the value of the nearest cable force bound (either the minimum or maximum force). The CFM is then performed again with the redundancy r reduced by one. The recursive process continues until a set of feasible solution is calculated or the remaining redundancy is negative, suggesting no feasible solution. The ICFM can therefore identify feasible solutions where the CFM may not without a significant increase in computational time. However, there are still instances in which ICFM results in no feasible solution despite solutions existing.

Based on the CFM, Müller *et al.* [12] developed the puncture method (PM) as a mean to reduce the solution cable force norm of a CFM solution. This is achieved by choosing the solution that lies at the intersection of the force constraint boundary and the line that connects the CFM solution with the solution to the unconstrained minimisation of the quadratic cable force norm. The puncture method has also been applied to the ICFM resulting in the improved puncture method (*IPM*) [11]. Both methods generate low force distributions that are more similar to those obtained using the QP methods, without a significant increase in the CFM computational time.

2.2 Properties of ID Solvers

The solutions obtained from the algorithms presented in Section 2.1 possess a number of different properties based upon their methodologies. Table 1 summarises a number of the properties for each of the solvers. A more detailed description of the properties is provided below.

Continuity

An ID solver is considered continuous if there is continuous variation of the solution in response to continuous changes in the kinematic variable $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$. Continuity is crucial in controlling CDPRs as sudden and large change in reference cable forces may result in undesired actuation dynamics such as oscillations and force error.

As indicated in Table 1, continuity is guaranteed for the QP, CFM and PM solvers. In contrast, the solvers that make use of linear programming (including the minimum-infinity norm and optimally-safe methods) do not necessarily guarantee solution continuity. This is because LP is not strictly convex. As a result, the optimal solution is not necessarily unique and discontinuity can

Properties	Inverse Dynamics Solvers						
	Linear Programming	Quadratic Programming	Minimum Infinity Norm	Feasible Polygon	Optimally Safe	Closed Form	Puncture Method
Continuity	No	Yes	No	Depends[1]	No	Yes	Yes
Force Magnitudes	Low	Low	Low	Low	Controllable	Mid	Low

Table 1: Comparison of ID solvers regarding continuity and force magnitudes

occur when there is switching between the different solutions.

Force Magnitudes

Cable forces generated by different solvers have tendencies to be distributed around areas with a certain force magnitude within the feasible force region. Table 1 shows that the LP, QP, minimum-infinity norm, PM and the feasible polygon methods produce force sets with lower tensions. These optimisation-based methods use the force norm as their objective functions, such that their solution tends towards the lower force bound.

In contrast, the CFM looks to produce force with magnitudes in the vicinity of the mean feasible force while the optimally safe method does the same in the instance in which $\alpha = 1$. This is chosen to avoid cable forces near the cable force bounds which can result in cable slack and low robot stiffness [6]. In the case of the optimally safe method, the use of the tuning parameter α allows for the resulting force distribution to be tuned towards a desired magnitude in which $\alpha = 0$ results in maximum cable force and the force decreases as α is increased.

Real-Time Capability

In force control of CDPRs, it is crucial to ensure the operating frequency is high enough, so that the force controllers can react to changes of system promptly and control the CDPR accurately. As ID is performed every control time step, an ID solver is considered as real-time capable if the solution is available within the time defined by the frequency requirement. The real-time capability of the different solvers will be evaluated through the use of CASPR in Section 3 of this paper.

Scalability

A solver is considered to have a high scalability, if a significant change in the *number of DoFs, number of cables* and/or *number of links* does not result in a substantial increase in computational time [17]. In this paper, besides comparing the absolute computational time used by each solver, the relative increase/decrease in computational time for a particular solver with respect to changes in the variables concerned, is also analysed.

[1]This changes based on whether the 1 or 2 norm is used.

3 Simulation and Comparison Results

The scalability of different ID solvers is analysed in this section using CASPR. First, Section 3.1 investigates the effect of adding additional cables (and hence additional degrees of redundancy) onto a spatial CDPR. Then, Section 3.2 considers the effect of adding new links, where two different joint types are considered in order to investigate the effect of different Jacobian forms. The set of test robots developed for these simulations represent a means of evaluating the effect of different numbers of cables, DoFs and links on the performance of an ID solver. As such the test can be used as a means of benchmarking different solvers in future studies.

For each variation of robot in the test set, ID is performed with each solver along joint trajectories which require movements in all DoFs. The mean and worst-case computational time of the 3 different runs are recorded, and the average time among the 3 runs is used. The average of the mean computational time over the whole range of parameters concerned, is considered as the mean computational time μ_{time} of the solver.

Scalability is quantified by comparing the relative percentage change with respect to each increase in the number of cables and links, in the SCDR and MCDR simulations, respectively. The average of the relative percentage changes over all parameter increase, is considered as the mean relative percentage change μ_{change}.

All simulations were conducted using MATLAB R2017a (64-bit), on the same hardware with the Intel Core i5-6500 CPU @ 3.20GHz and 8.00GB of RAM. For the optimisation solvers, the analysis was performed in CASPR using the optimisation toolbox solvers of MATLAB in which the *dual-simplex* and *interior-point-convex* algorithms were used for the LP and QP, respectively. The mean computational time μ_{time}, the minimum and maximum of mean computational time over the range of parameters μ_{min} and μ_{max}, and the mean relative percentage change μ_{change} of the simulations are summarised in Table 2.

3.1 Simulations on SCDRs

This set of tests investigates the scalability of the different ID solvers for the 6-DoF IPAnema1 spatial SCDR with 7 to 12 cables [4]. The joint trajectory used in the

Simulations	Inverse Dynamics Solvers						
	Linear Programming	Quadratic Programming	Minimum Infinity Norm	Feasible Polygon	Optimally Safe	Closed Form	Puncture Method
SCDRs							
- μ_{time} [ms]	3.429	2.384	10.663	1.071	1.348	0.540	0.569
- μ_{min} [ms]	3.425	2.343	10.600	1.071	1.238	0.532	0.559
- μ_{max} [ms]	3.438	2.433	10.786	1.071	1.531	0.547	0.576
- μ_{change} [%]	0.0699	0.0561	0.0699	N/A	3.3765	0.5523	0.0444
MCDRs (revolute)							
- μ_{time} [ms]	3.6247	2.5416	10.7150	0.8392	9.7589	0.6992	0.6338
- μ_{min} [ms]	3.482	2.346	10.053	0.8392	0.776	0.609	0.547
- μ_{max} [ms]	3.781	2.789	11.329	0.8392	11.159	0.814	0.725
- μ_{change} [%]	0.4776	1.6703	0.5865	N/A	138.5537	3.2748	3.2188
MCDRs (spherical)							
- μ_{time} [ms]	3.788	2.738	10.707	1.280	8.915	0.660	0.690
- μ_{min} [ms]	3.416	2.324	9.970	1.280	0.880	0.515	0.542
- μ_{max} [ms]	4.325	3.391	11.235	1.280	11.675	0.824	0.845
- μ_{change} [%]	2.6654	4.3194	1.1966	N/A	53.7076	5.3953	5.0633

Table 2: Comparison of ID solvers regarding real-time capability and scalability[2]

tests is shown in Figure 2, which involves movements of all DoFs. Figures 3 and 4 show the mean and worst-case computational time for different solvers, respectively.

It is found from Figures 3 and 4 that heuristic methods require the least computational time to perform ID, with a mean time of approximately 0.55 and 0.60 ms for CFM and PM, respectively. All these heuristic methods also showed high scalability, where the mean percentage changes of computational time with respect to the increasing number of cables is less than 0.5%.

The optimally safe method displayed a mean computational time of about 1.35 ms which outperformed ordinary LP and QP solvers, which required approximately 3 ms. However, the method also showed a significantly higher worst-case computational time than both the LP and QP. When compared to the LP and QP, which can be observed to scale well in both the mean and worst-case computational time, the worst-case computational time of the optimally safe method showed a sharp increase, in

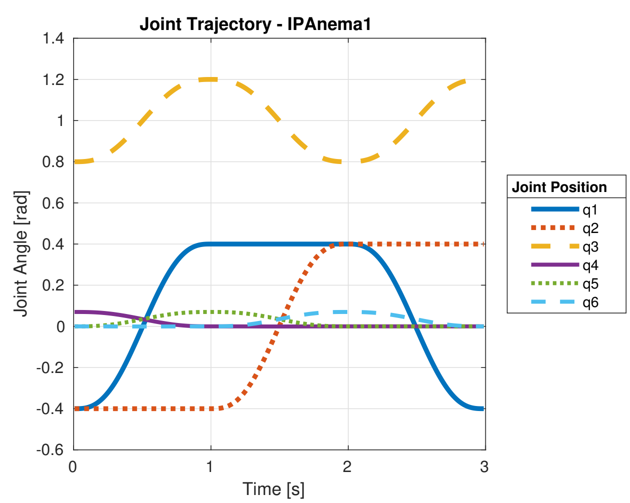


Figure 2: Joint trajectory used in tests on IPAnema1

[2]Results of ICFM and IPM are shown in the columns of closed form and puncture method.

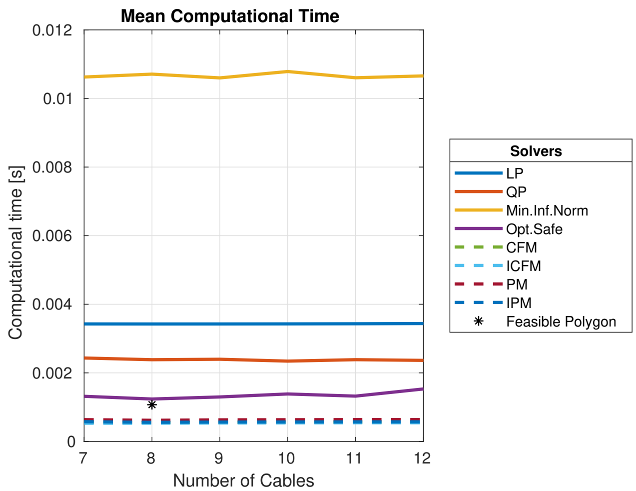


Figure 3: Mean computational time of different solvers with increasing number of cables

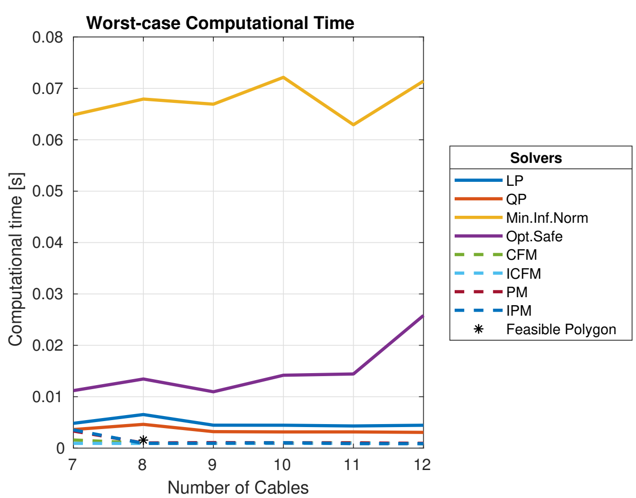


Figure 4: Worst-case computational time of different solvers with increasing number of cables

particular when the number of cables increases from 11 to 12, displaying an percentage increase of about 78%.

From the tests conducted it can be observed that the minimum infinity-norm method is not real-time capable, with a mean computational time of approximately 10 ms, which is the highest among all solvers. This solver makes use of an LP in its implementation, however the implementation time is in general significantly higher. The high computational time is therefore likely to be caused by the auxiliary process of transferring the problem into an LP form rather than the solving of the LP problem.

As the feasible polygon method is only applicable to CDPRs with 2 degrees of redundancy, it was only tested in the case with 8 cables. The mean computational time required by the method is approximately 1.07 ms, which is observed to be higher than the heuristic methods, but lower than the optimally safe method, showing real-time potential when it can be implemented. Scalability of this method is not considered in the current test set.

3.2 Simulations on MCDRs

To evaluate the scalability of the ID solvers as the number of links increases, a range of MCDRs with an increasing number of links (and cables) are tested. To analyse the effect of different joint types on the scaling, two sets of MCDRs were constructed, one with only revolute joints and the other with only spherical joints. The range of the number of links is from 1 to 10, with 2 to 20 cables for the revolute joint MCDRs, and 4 to 40 cables for the spherical joint MCDRs. Figures 5 and 6 show the trajectories used in the tests for revolute joint and spherical joint MCDRs, respectively.

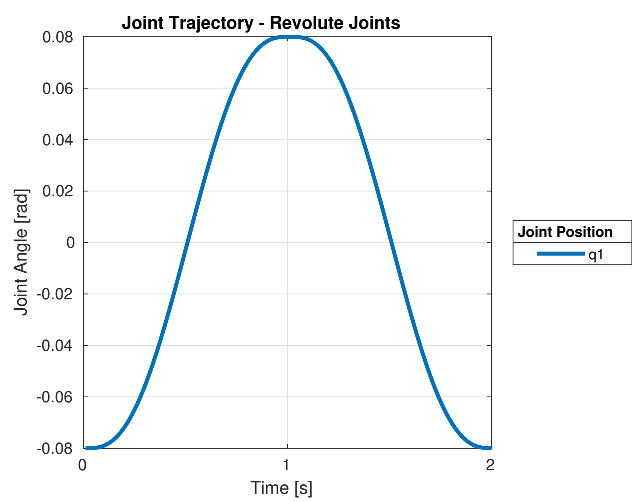


Figure 5: Joint trajectory used in tests on MCDRs with 1 revolute joint

The mean and worst-case computational time used by different solvers in the revolute and spherical joint MC-

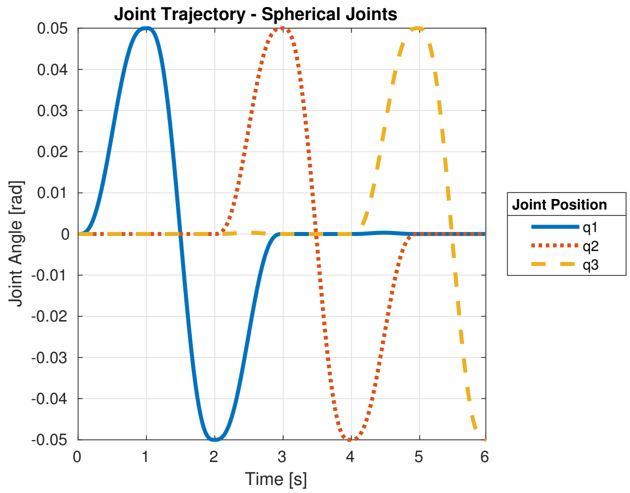


Figure 6: Joint trajectory used in tests on MCDRs with 1 spherical joint

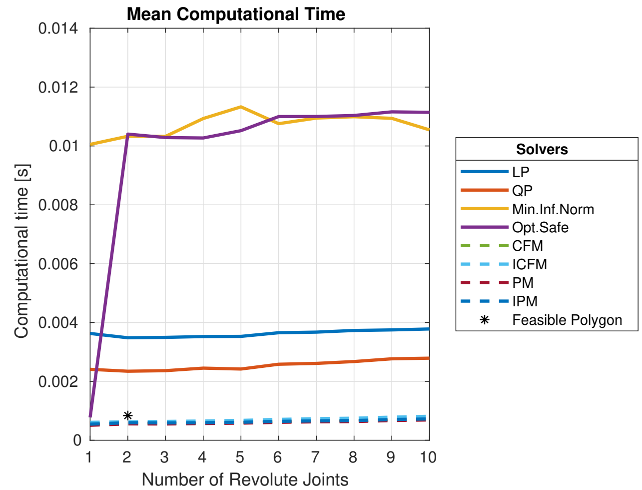


Figure 7: Mean computational time of different solvers with increasing number of revolute joints

DRs are shown in figures 7-10. Similar to the SCDR test set, it can be observed that the heuristic methods, CFM and PM, showed good performance with approximate mean computational times of 0.6 ms and 0.7 ms in the revolute and spherical cases, respectively. These methods showed a relative percentage change of about 6% in the spherical cases, which is larger than that in the revolute MCDR as well as the SCDR tests. This is likely due to the higher increase in the number of DoFs in the spherical MCDR tests than in the revolute cases and the SCDR tests. Nonetheless, the absolute change in computational time is insignificant and hence maintains the high real-time capability of the methods.

In addition to the computational time, it was observed that from the 6 spherical joint case onwards, the CFM and PM could not always generate feasible results. This was not the case for the ICFM and IPM which always found a solution with only a limited increase in the required computational time. This result is consistent with the properties of the ICFM and IPM [11].

In contrast to the heuristic methods, the LP and QP methods required a greater mean computational time in general, with about 3.62 and 2.54 ms on revolute joint MCDRs, and about 3.79 and 2.74 ms in the spherical cases, respectively. In the revolute cases, the LP and QP methods showed a mean relative increase of about 0.5% and 1.7% in computational time, respectively. However, the increase was more substantial in the case of the spherical links, with an average of about 2.7% and 4.3%. Since in both test sets the degree of redundancy increases at the same rate, this increase in computational time indicates that the number of DoFs of the problem has a substantial effect.

Both the minimum infinity-norm and the optimally

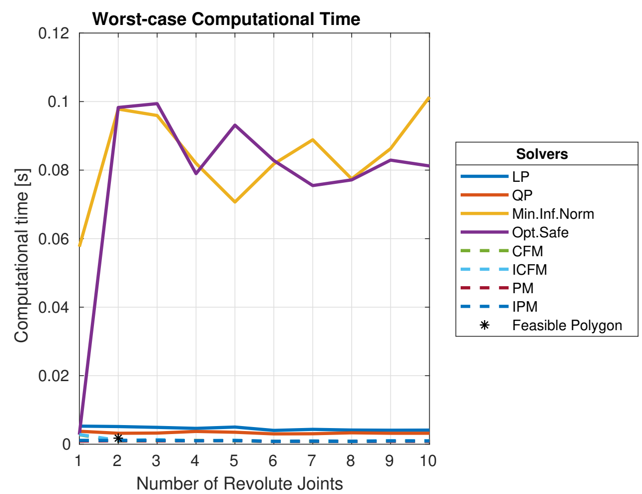


Figure 8: Worst-case computational time of different solvers with increasing number of revolute joints

safe methods took noticeably longer time as the number of links increased, with a mean computational time of around 10 ms and 9 ms, respectively. In both cases an LP is solved, however, both cases are substantially slower to implement as the number of links increases. The large difference is believed to be related to the additional calculations involved in the algorithms, which require much longer computational time when the problem dimension scales up.

While both the minimum infinity-norm and the optimally safe methods methods are not real-time capable for a high number of links, the optimally safe method performed well in cases of single-link robots, which only required about 0.78 ms in the revolute case, and 0.88 ms in the spherical case. The efficient computation method

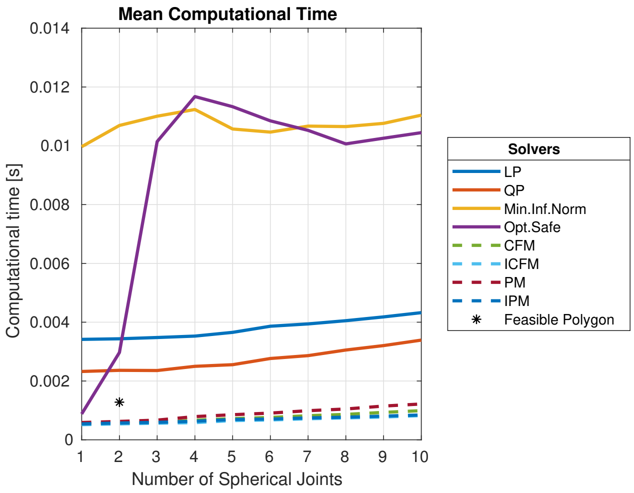


Figure 9: Mean computational time of different solvers with increasing number of spherical joints

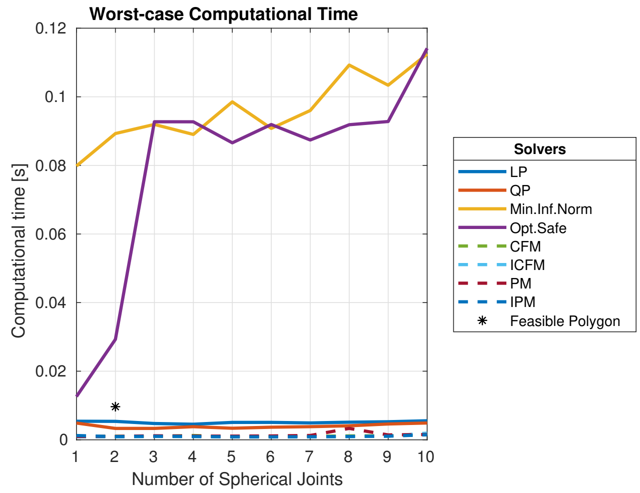


Figure 10: Worst-case computational time of different solvers with increasing number of spherical joints

proposed by Borgstrom *et al.* [6] has its most substantial effect on computational time in the case in which warm starting can be used to obtain the solution of the next time instant. As the number of links increased, it was observed that the warm starting method was infeasible for a increasing percentage of the trajectories. As a result, only the single-link robots show real-time potential.

Only the 2-link cases of both the revolute and spherical joint MCDRs, with 4 and 8 cables respectively, are tested using the feasible polygon method due to its restriction in redundancy. Scalability of the method is hence not considered. Similar to the results from SCDR tests, the performance of the method lies between heuristic methods and ordinary LP, QP solvers, with a mean computational time of approximately 0.84 ms in the revolute cases, and 1.28 ms in the spherical joint cases, showing potential for real-time applications.

4 Conclusion

In this paper, properties exhibited by different ID solvers are compared. Continuity and the magnitudes of the force set solutions are discussed by analysing the implementations of solvers, while real-time capability and scalability are investigated through simulations on various SCDRs and MCDRs through the use of CASPR. It was shown that compared to optimisation-based methods, heuristic methods generally present higher real-time capability and scalability with respect to increasing number of links, DoFs and cables. Both the minimum-infinity norm and optimally-safe methods were found to not be real-time applicable on multi-link robots, however, the optimally-safe method outperformed ordinary LP and QP methods in single-link cases. Simulation results provide better insights into choosing the right solver for CDPR researches. Based on the current results, future work will look to expand the test set and explore other properties such as the effect of different poses in the workspace on the computational time of the solvers.

Acknowledgements

This work was supported by the grants from the Early Career Scheme sponsored by the Research Grants Council (Reference No. 24200516).

References

[1] J. S. Albus, R. V. Bostelman, and N. Dagalakis, "The NIST robocrane," *J. Robot. Syst.*, vol. 10, no. 5, pp. 709–724, 1993.

[2] S. Kawamura, W. Choe, S. Tanaka, and S. R. Pandian, "Development of an ultrahigh speed robot FALCON using wire drive system," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1995, pp. 215–220.

[3] J. Lamaury and M. Gouttefarde, "Control of a large redundantly actuated cable-suspended parallel robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 4659–4664.

[4] A. Pott, H. Mütherich, W. Kraus, V. Schmidt, P. Miermeister, and A. Verl, "Ipanema: A family of cable-driven parallel robots for industrial applications," in *Cable-Driven Parallel Robots*, ser. Mech. and Mach. Sci., vol. 12, pp. 119–134.

[5] A. Ming and T. Higuchi, "Study on multiple degrees-of-freedom positioning mechanism using wires (part 1) - concept, design and control," *Int. J. Japan Social Eng.*, vol. 28, no. 2, pp. 131–138, 1994.

[6] P. H. Borgstrom, B. L. Jordan, G. S. Sukhatme, M. A. Batalin, and W. J. Kaiser, "Rapid computation of optimally safe tension distributions for parallel cable-driven robots," *IEEE Trans. Robot.*, vol. 25, no. 6, pp. 1271–1281, 2009.

[7] D. Lau, D. Oetomo, and S. K. Halgamuge, "Generalized modeling of multilink cable-driven manipulators with arbitrary routing using the cable-routing matrix," *IEEE Trans. Robot.*, vol. 29, no. 5, pp. 1102–1113, 2013.

[8] M. Gouttefarde, J. Lamaury, C. Reichert, and T. Bruckmann, “A versatile tension distribution algorithm for n-DOF parallel robots driven by n+2 cables,” *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1444–1457, 2015.

[9] C. Gosselin and M. Grenier, “On the determination of the force distribution in overconstrained cable-driven parallel mechanisms,” *Meccanica*, vol. 46, no. 1, pp. 3–15, 2011.

[10] A. Pott, T. Bruckmann, and L. Mikelsons, “Closed-form force distribution for parallel wire robots,” in *Computational Kinematics*. Springer, 2009, pp. 25–34.

[11] A. Pott, “An improved force distribution algorithm for over-constrained cable-driven parallel robots,” in *Computational Kinematics*, 2014, pp. 139–146.

[12] K. Müller, C. Reichert, and T. Bruckmann, “Analysis of a real-time capable cable force computation method,” in *Cable-Driven Parallel Robots*, ser. Mech. and Mach. Sci., 2015, vol. 32, pp. 227–238.

[13] D. Lau, J. Eden, Y. Tan, and D. Oetomo, “CASPR: A comprehensive cable-robot analysis and simulation platform for the research of cable-driven parallel robots,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2016, pp. 3004–3011.

[14] J. A. Cadzow, “A finite algorithm for the minimum l_∞ solution to a system of consistent linear equations,” *SIAM Journal on Numerical Analysis*, vol. 10, no. 4, pp. 607–617, 1973.

[15] I. C. Shim and Y. S. Yoon, “Stabilized minimum infinity-norm torque solution for redundant manipulators,” *Robotica*, vol. 16, no. 2, pp. 193–205, 1998.

[16] I. Ha and J. Lee, “Analysis on a minimum infinity-norm solution for kinematically redundant manipulators,” *Transaction on Control, Automation and Systems Engineering*, vol. 4, no. 2, pp. 130–139, 2002.

[17] D. Lau, D. Oetomo, and S. K. Halgamuge, “Inverse dynamics of multilink cable-driven manipulators with the consideration of joint interaction forces and moments,” *IEEE Trans. Robot.*, vol. 31, no. 2, pp. 479–488, 2015.