Dynamic Analysis of Continuous Cascaded Generalized Inverse Resolution of Kinematically Redundant Manipulators with Flexible Joints

Travis Baratcart, Valerio Salvucci, and Takafumi Koseki
Department of Electrical Engineering and Information Systems, The University of Tokyo, Japan
b_travis@koseki.t.u-tokyo.ac.jp

***Abstract*— In the resolution of redundancy in systems, by far the most popular approach involves optimization with respect to the 2-norm, due to it's ease of use, uniqueness, and continuity of resolution. Due to the 2-norm's inability to consider input bounds, however, such methods fail to exploit systems' full feasible output range. The Cascaded Generalized Inverse (CGI) was introduced to extend the realizable output range of the pseudo-inverse, but in doing so, introduces discontinuity in the resolution scheme. This paper analyzes the effects of such discontinuity on the dynamic performance of one popular redundant application: kinematically redundant manipulators. Joint flexibility is simulated and it is shown that discontinuity arising in CGI imparts joint velocity error and oscillation in the system dynamics. The Continuous Cascaded Generalized Inverse (cCGI), which we previously proposed, is a variation of CGI which guarantees continuity in resolution. cCGI was also simulated successfully extending the resolution range of the pseudo-inverse, but without the same dynamic consequences seen in CGI resolution.**

I. Introduction

A system is said to be redundant when the number of inputs available, n, exceeds the degrees of freedom of the desired output, m. Such redundancy is valued for its ability to add dexterity and robustness, but its implementation necessarily complicates the system's control structure.

The problem of redundancy is considered as follows: The matrix $\boldsymbol{B} \in \Re^{mxn}$ translates actuation values $\boldsymbol{u} \in \Re^n$ to task space values $\boldsymbol{v} \in \Re^m$ as follows:

$$\boldsymbol{v} = \boldsymbol{B}\boldsymbol{u} \tag{1}$$

If the system is redundant ($n > m$), there exists an infinite number of potential solutions for $\boldsymbol{u}$, through utilization of the null space of $\boldsymbol{B}$. This allows for the benefit of realization of subtasks not directly related to the input-output relationship of the main system. As a result of this greater allotted freedom, redundancy also complicates the translation of a desired output to the input space.

Many methods have been introduced to resolve this problem, but the vast majority of implementations make use of the Moore-Penrose pseudo-inverse, which optimizes the 2-norm of the resolved inputs [1]. Pseudo-inverse resolution is highly valued due to its closed-form, easily generalized structure, uniqueness, and continuity in resolution.

However, a significant drawback exists in utilization of pseudo-inverse resolution, in its failure to exploit a system's full output range — a 2-norm optimal solution does not necessarily fall within inputs' physical limits.

Various other resolution methods have been developed to overcome this difficulty, including optimization using the infinity-norm [2], [3] and neural-network-based methods [4], [5]. However, the pseudo-inverse's generality and ease of implementation have led most system designers to continue to utilize it over such alternatives.

The Cascaded Generalized Inverse (CGI) was introduced to extend the feasible output range of the pseudo-inverse by iteratively reallocating excess input contributions to under-utilized inputs. CGI has been applied in research to many problems, including aircraft control applications [6], [7], VTOL control [8], ship berthing [9], and ship positioning systems [10].

In [11] we demonstrated that despite successfully extending the 2-norm, CGI resolution loses one of the main benefits of 2-norm resolution — continuity. It was also shown that through a restriction on the resolution domain, continuity can be assured in CGI resolution. This domain restriction comprises the Continuous Cascaded Generalized Inverse (cCGI). In [11], the issue of discontinuity in CGI and its mitigation through cCGI were introduced in general terms. In this work, we demonstrate the practical advantage of cCGI resolution with respect to CGI and 2-norm, through consideration of a potential application.

Chosen for its accessibility, the considered application is the kinematically redundant manipulator. Kinematically redundant manipulators are manipulators which have more joint degrees of freedom than the task space. The freedom allowed by kinematic redundancy allows for such behaviors as obstacle [12] and singularity avoidance [13].

When modeling the dynamics of such manipulators, one typically assumes the joints to have infinite stiffness. While this assumption may radically simplify system modeling, such unmodeled terms have the potential to cause unexpected and harmful behavior in the dynamic performance of the manipulator. It will be shown that the discontinuity introduced by CGI resolution combined with the unideal property of finite joint stiffness will lead to undesired joint oscillation and joint velocity error, and the merits of cCGI resolution in extending 2-norm resolution without the downsides of such discontinuity will be explored.

II. Extending the Resolution Range of 2-norm

A. Pseudo-Inverse Resolution

In resolution of redundant systems, by far the most popular method involves application of the Moore-Penrose pseudo-inverse which resolves (1) while optimizing the 2-norm of the resultant vector, or

$$\min(\sqrt{u_1^2+u_2^2+...+u_n^2}) \tag{2}$$

where u_i is the i'th element of the input vector, $\boldsymbol{u}$.

Pseudo-inverse resolution defines the resolved input as

$$\boldsymbol{u} = \boldsymbol{B}^{\dagger}\boldsymbol{v} \tag{3}$$

where if $\boldsymbol{B}$ is full row rank, the pseudo-inverse, $\boldsymbol{B}^{\dagger}$, of $\boldsymbol{B}$ is defined as follows:

$$\boldsymbol{B}^{\dagger} := \boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{B}^T)^{-1} \tag{4}$$

Pseudo-inverse resolution is popularly utilized due to its ease of implementation (closed-form solution for general systems), uniqueness, and continuity of resolution. However, 2-norm resolution fails to make use of a system's full attainable output space. A 2-norm optimal resolution does not necessarily fall within physical input constraints, leading 2-norm to frequently yield unfeasible resolutions despite feasible alternatives.

Applications in which the lost output space of 2-norm are a serious factor typically make use of methods to extend the resolvable range of the pseudo-inverse, rather than switching resolution schemes entirely.

B. The Redistributed Pseudo-Inverse

A simple attempt at extending the resolution range of 2-norm is the Redistributed Pseudo-Inverse [14]. This method is a two-step approach which can be described as follows:

1) Find the pseudo-inverse resolution of the system. If all resolved inputs lie within their output bounds, the process ends.
2) If any resolved inputs lie outside their output bounds, the corresponding inputs are truncated to the respective maximums or minimums.
   The new system created, corresponding to the un-maximized inputs and the desired output minus the contributions of the maximized inputs, is then resolved using the pseudo-inverse.

As the Redistributed Pseudo-Inverse re-resolves inputs using 2-norm, however, the full output space of the re-resolved inputs will not be fully exploited.

C. The Cascaded Generalized Inverse (CGI)

The 2-step process of the Redistributed Pseudo-Inverse is readily generalized into the potentially $(n-m+1)$-step process of the Cascaded Generalized Inverse (CGI) [15]. Similar to the Redistributed Pseudo-Inverse, CGI computes the 2-norm resolution of the system, truncates saturated inputs, and re-resolves remaining inputs through pseudo-inverse resolution. CGI however, repeats the process if re-resolved inputs saturate until either a realizable solution occurs or all inputs saturate. CGI represents the largest extension of 2-norm resolution.

D. The Continuous Cascaded Generalized Inverse (cCGI)

In [11] we demonstrated that utilization of CGI, and by proxy the Redistributed Pseudo-Inverse, introduces the risk of discontinuity in resolution. A domain restriction was imposed, comprising the Continuous Cascaded Generalized Inverse (cCGI) that states that only one variable may be permitted to saturate per pseudo-inverse iteration. Resolution using Continuous CGI comprises the greatest extension of pseudo-inverse resolution which ensures continuity.

Although most easily applied recursively, as introduced, a closed-form representation for cCGI was developed in [11] as follows:

Define $\boldsymbol{B}_i$ as the i'th column of $\boldsymbol{B}$ and $\boldsymbol{B}'_{i_1.i_2...i_z}$ as the matrix $\boldsymbol{B}$ with the i_1, i_2, ... , and i_z'th column removed. Likewise define u_i as the i'th element of $\boldsymbol{u}$, and $\boldsymbol{u}'_{i_1.i_2...i_z}$ as the vector $\boldsymbol{u}$ with the i_1, i_2, ... , and i_z'th element removed.

Parentheticals are used following input values to denote the resolution of the inputs in the corresponding case. For example, $u_3(1(\mathrm{a}))$ and $\boldsymbol{u}'_3(1(\mathrm{a}))$ refer respectively to the values of the third element of $\boldsymbol{u}$ and the vector $\boldsymbol{u}$ with the third element removed attained in the case 1(a). If inputs are referred to without parentheticals, it will refer to the value attained from the currently discussed case.

Case 0 :

$$\boldsymbol{u} = \boldsymbol{B}^{\dagger}\boldsymbol{v} \tag{5}$$

Case $i_1(\mathrm{a})$:

$$u_{i_1} = u_{i_1}^{\max} \tag{6}$$

$$\boldsymbol{u}'_{i_1} = (\boldsymbol{B}'_{i_1})^{\dagger}(\boldsymbol{v} - \boldsymbol{B}_{i_1}u_{i_1}^{\max}) \tag{7}$$

Case $i_1(\mathrm{b})$

$$u_{i_1} = -u_{i_1}^{\max} \tag{8}$$

$$\boldsymbol{u}'_{i_1} = (\boldsymbol{B}'_{i_1})^{\dagger}(\boldsymbol{v} + \boldsymbol{B}_{i_1}u_{i_1}^{\max}) \tag{9}$$

Case $i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1}).i_z(\mathrm{a})$:

$$u_{i_1} = \begin{cases} u_{i_1}^{\max} & \text{if } x_1 = \mathrm{a} \\ -u_{i_1}^{\max} & \text{if } x_1 = \mathrm{b} \end{cases} \tag{10}$$

$$\vdots$$

$$u_{i_{z-1}} = \begin{cases} u_{i_{z-1}}^{\max} & \text{if } x_{z-1} = \mathrm{a} \\ -u_{i_{z-1}}^{\max} & \text{if } x_{z-1} = \mathrm{b} \end{cases} \tag{11}$$

$$u_{i_z} = u_{i_z}^{\max} \tag{12}$$

$$\boldsymbol{u}'_{i_1...i_z} = (\boldsymbol{B}'_{i_1...i_z})^{\dagger}(\boldsymbol{v} - \sum_{\gamma=1}^{z}\boldsymbol{B}_{i_\gamma}u_{i_\gamma}) \tag{13}$$

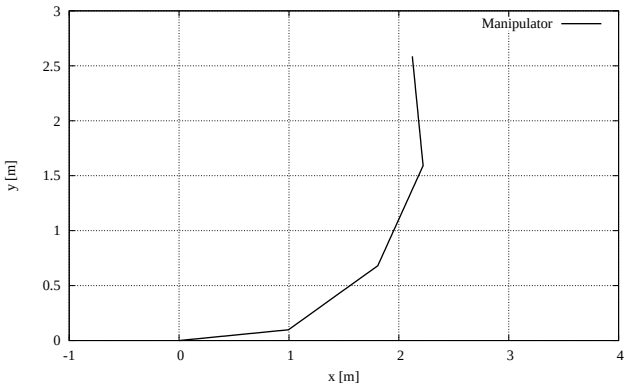

Fig. 1. 4-link planar manipulator used for demonstration of discontinuity in CGI Resolution

Case $i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1}).i_z(\mathrm{b})$:

$$u_{i_1} = \begin{cases} u_{i_1}^{\max} & \text{if } x_1 = \mathrm{a} \\ -u_{i_1}^{\max} & \text{if } x_1 = \mathrm{b} \end{cases} \tag{14}$$

$$\vdots$$

$$u_{i_{z-1}} = \begin{cases} u_{i_{z-1}}^{\max} & \text{if } x_{z-1} = \mathrm{a} \\ -u_{i_{z-1}}^{\max} & \text{if } x_{z-1} = \mathrm{b} \end{cases} \tag{15}$$

$$u_{i_z} = -u_{i_z}^{\max} \tag{16}$$

$$\boldsymbol{u}'_{i_1\ldots i_z} = (\boldsymbol{B}'_{i_1\ldots i_z})^{\dagger}(\boldsymbol{v} - \sum_{\gamma=1}^{z} \boldsymbol{B}_{i_\gamma} u_{i_\gamma}) \tag{17}$$

and define

$$\text{case } 0 := |u_i(0)| \leq u_i^{\max}, \forall i \in [1,n] \tag{18}$$

$$\text{case i(a)} := u_i(0) > u_i^{\max}, |u_j(0)| \leq u_j^{\max}, \forall j \neq i \tag{19}$$

$$\text{case i(b)} := u_i(0) < -u_i^{\max}, |u_j(0)| \leq u_j^{\max}, \forall j \neq i \tag{20}$$

$$\begin{aligned} &\text{Case } i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1}).i_z(\mathrm{a}) := \\ &\quad (\text{Case } i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1})), \\ &\quad (u_{i_z}(i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1})) > u_{i_z}^{\max}), \\ &\quad \text{and } (|u_{i_j}(i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1}))| \leq u_{i_j}^{\max}, \forall j \neq z) \end{aligned} \tag{21}$$

$$\begin{aligned} &\text{Case } i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1}).i_z(\mathrm{b}) := \\ &\quad (\text{Case } i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1})), \\ &\quad (u_{i_z}(i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1})) < -u_{i_z}^{\max}), \\ &\quad \text{and } (|u_{i_j}(i_1(x_1).i_2(x_2)...i_{z-1}(x_{z-1}))| \leq u_{i_j}^{\max}, \forall j \neq z) \end{aligned} \tag{22}$$

III. Cascaded Generalized Inverse and Discontinuity

The problem of discontinuity in CGI resolution can be considered through the following problem. Consider the four-link, kinematically redundant manipulator with 1 m links, in Fig. 1.

The simplest way to formulate this problem of kinematic-redundancy is through consideration of system velocities,

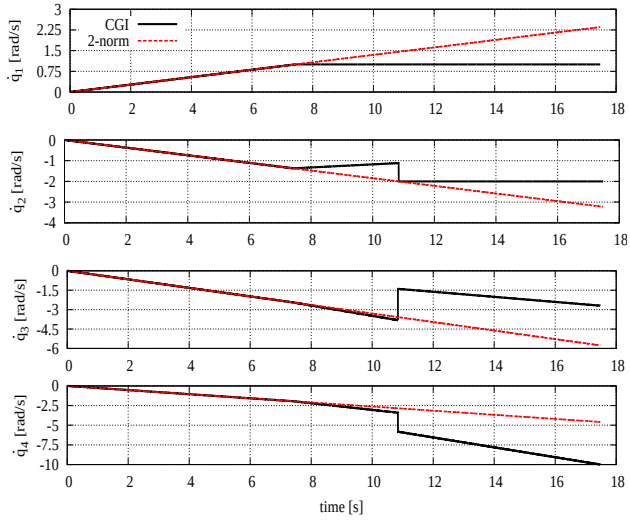

Fig. 2. Static resolution of kinematic redundancy using CGI. CGI continues resolution by discontinuously reassigning velocity amongst q_2, q_3, and q_4.

forming a variant of (1. The Jacobian matrix, $\boldsymbol{J}$, relates joint velocities, $\dot{\boldsymbol{q}}$, to end-effector velocity, $\boldsymbol{v}_{\text{eff}}$ as follows:

$$\boldsymbol{v}_{\text{eff}} = \boldsymbol{J}\dot{\boldsymbol{q}} \tag{23}$$

where

$$[\boldsymbol{J}]_{(i,j)} = \frac{\delta[\boldsymbol{x}_{\text{eff}}]_i}{\delta \boldsymbol{q}_j} \tag{24}$$

and $\boldsymbol{x}_{\text{eff}} \in \Re^m$ is the vector of end-effector position coordinates.

In this case, there are four joint values to actuate the arm in 2-d space, so there are 2 degrees of redundancy which must be resolved.

To simulate the statics of this arm, the arm is placed in the position $\boldsymbol{q} = [\frac{\pi}{32}, \frac{\pi}{6}, \frac{\pi}{6}, \frac{\pi}{6}]$ (rad) and is tasked with increasing end-effector velocity in the direction $\theta = 0$. Joints closer to the base experience higher arm inertia, and therefore would naturally have lower limited velocity than higher joints, so maximum joint velocities are set as $\dot{\boldsymbol{q}}^{\max} = [1, 2, 10, 10]$ (rad/sec).

Figure 2 illustrates the results of static resolution of the arm using CGI and unconstrained 2-norm resolution. Both 2-norm and CGI resolve the system equivalently until $\|v\| \approx 7.5$ (m/sec), at which point 2-norm resolution begins to demand unrealizable joint velocity from q_1. CGI, on the other hand, is able to continue resolving the system by re-tasking to other joints. It is seen, however, that when $\dot{q}_2$ saturates in the first level 2-norm resolution, at $\|v\| \approx 11$ (m/sec), CGI continues to resolve the system by discontinuously reassigning velocity amongst $\dot{q}_2$, $\dot{q}_3$, and $\dot{q}_4$.

When cCGI is utilized to resolve this system, resolution follows CGI resolution precisely until the saturation of $\dot{q}_2$ in the first level (which causes the discontinuity in CGI resolution). At this point Continuous CGI stops resolution, effectively avoiding the discontinuity seen in CGI.

Such discontinuity (and its elimination) can have profound impact when controlling physical systems. In this paper, we

endeavor to show the effects of such discontinuity affecting the dynamics of kinematically-redundant manipulators with flexible joints.

IV. Dynamic Analysis Setup

In this section the motivation and theory behind the simulations to follow will be discussed.

In Sec. III the problem of discontinuity in CGI was introduced. In the following simulations, the physical impact of this discontinuity will be explored through analysis of the dynamic performance of such an arm, implemented using 2-norm, CGI, and cCGI resolution. Although typically neglected in dynamic compensation (as it will be in the following simulation), joint flexibility will be included in the arm dynamic model.

First let us consider how to model such finite joint stiffness. Robotic manipulators with infinite joint stiffness can be dynamically modeled as

$$(\boldsymbol{M}+\boldsymbol{M}_j)\ddot{\boldsymbol{q}}+\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})+\boldsymbol{g}(\boldsymbol{q})=\boldsymbol{\tau} \tag{25}$$

where $\boldsymbol{M}\in\Re^{nxn}$ is the manipulator inertia matrix, $\boldsymbol{M}_j\in\Re^{nxn}$ is the diagonal motor inertia matrix, $\boldsymbol{C}\in\Re^{nx1}$ is the vector of Coriolis and centrifugal force terms, $\boldsymbol{g}\in\Re^{nx1}$ is the vector of gravity terms, and $\boldsymbol{\tau}\in\Re^{nx1}$ is the applied joint torque [16].

Finite joint stiffness implies a level of freedom between the joint coordinates, $\boldsymbol{q}\in\Re^{nx1}$, and the motor coordinates, $\boldsymbol{\theta}\in\Re^{nx1}$. The dynamics of such a manipulator are formed by the simultaneous solution of two condition equations, describing the motor's motion and the link's motion (and their interaction), as follows:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}+\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})+\boldsymbol{g}(\boldsymbol{q})+\mathbf{D}(\dot{\boldsymbol{q}}-\dot{\boldsymbol{\theta}})+\mathbf{K}(\boldsymbol{q}-\boldsymbol{\theta})=0 \tag{26}$$

$$\boldsymbol{M}_j\ddot{\boldsymbol{\theta}}+\mathbf{D}(\dot{\boldsymbol{\theta}}-\dot{\boldsymbol{q}})+\mathbf{K}(\boldsymbol{\theta}-\boldsymbol{q})=\boldsymbol{\tau} \tag{27}$$

where $\mathbf{K}\in\Re^{nxn}$ and $\mathbf{D}\in\Re^{nxn}$ are the diagonal matrices of joint stiffness and joint damping terms, respectively.

To simulate the effect of discontinuity in flexible joint manipulators, a 4-link kinematically redundant manipulator was chosen, similar to the one utilized in Sec. III, but with considered finite joint stiffness. For ease of understanding, the manipulator was selected as having 1 kg, 1 m links and link inertia values perpendicular to the direction of rotation, I_{z_i}, were all set at unit value. Maximum joint velocities were set as $\dot{\boldsymbol{q}}^{\max}=[0.5,0.7,5.0,5.0]$ (rad/sec). Joint stiffness values are chosen as $\mathbf{K}=100000\boldsymbol{I}_4$ (N/rad), which is chosen as approximately 1 order of magnitude above common high stiffness threshold of variable series elastic motors (motors with intended compliance). Joint damping should be considered nonzero, but much less than joint stiffness, so values of $10\boldsymbol{I}_4$ (N·s/rad) were chosen. The manipulator is assumed to be planar, so gravity terms can be neglected.

A normal implementation would make use of position and velocity feedback in order to correct for the drift occuring from velocity-space resolution. However, in order to effectively isolate the effect of discontinuity on the arm's dynamic performance and to allow for accurate numerical comparison between cases, open loop control of the arm will be utilized. In the region of interest (when oscillation occurs), the physical effect of a controller can still be equivalently felt as a result of the system damping simulated. A block diagram of the utilized open-loop control scheme can be found in Fig. 3. The control structure makes the assumption of infinite stiffness, i.e. that the system block dynamics can be described by (25). In reality however, the system dynamics are described by the equations for finite stiffness found in (26-27). This allows for observation of the unexpected results of finite stiffness while using a standard compensation model which neglects these terms.

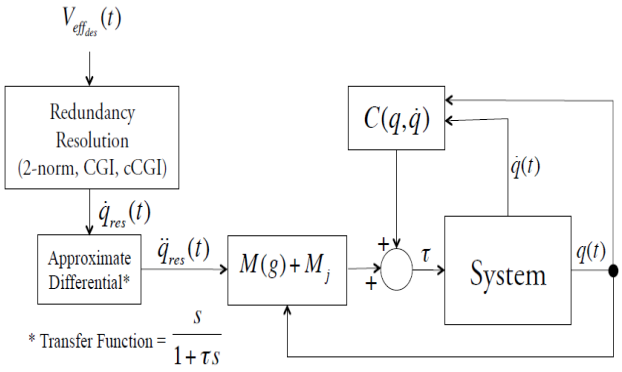

Fig. 3. Block diagram describing open loop control of kinematically redundant planar arm

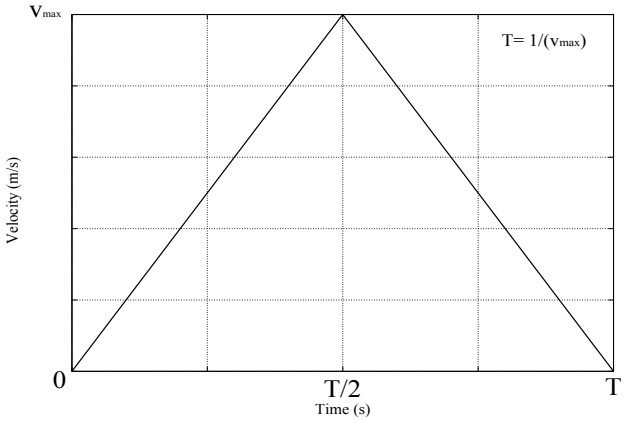

Fig. 4. Velocity profile used in demonstrating effects of discontinuity. The arm is commanded to travel 1 m with a given maximum velocity.

In the dynamic analysis we neglect friction, as the effect is equivalent to damping in the region of interest. Backlash is neglected as both motors and gears without backlash exist. Time delay is neglected as open loop control is utilized and the physical system is time-invariant.

The described arm is commanded from rest at an initial position $\boldsymbol{q}=\boldsymbol{\theta}=[0,\frac{\pi}{4},\frac{\pi}{4},\frac{\pi}{3}]$ (rad) to move with a straight-line end-effector trajectory 1 m in the direction $\phi=\frac{\pi}{4}$ (rad) with a velocity trajectory as described in Fig. 4. The system updates with a frequency of $\frac{1}{\Delta T}=10$ kHz, and τ for the differential calculation is selected as 10 ms.

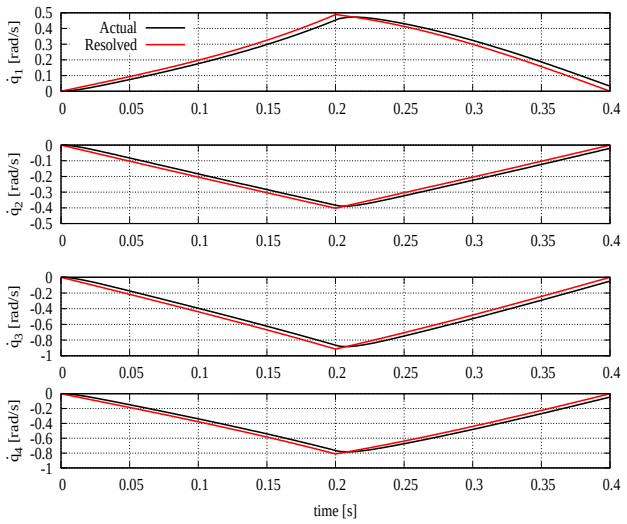

Fig. 5. Joint velocities of trajectory realizable with pseudo-inverse Redundancy Resolution

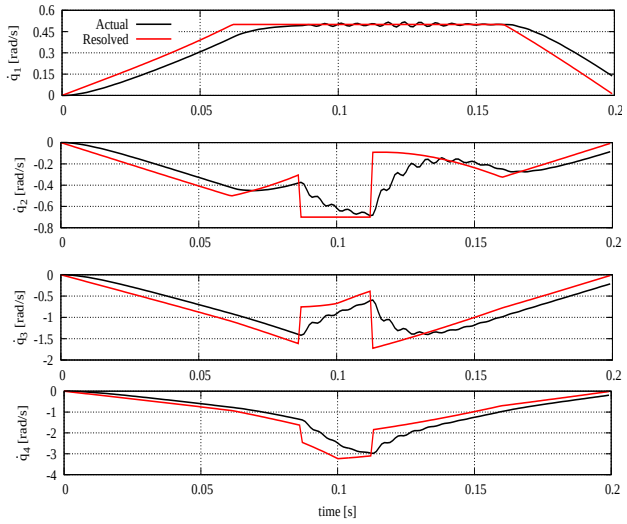

Fig. 6. Joint velocities of trajectory realizable only with CGI resolution. Discontinuously resolved velocities generate velocity oscillation and error.

V. Simulations and Results

Figure 5 is intended as a comparison case and simulates the trajectory with v_{max} sufficiently low, such that the entire trajectory is realizable using only 2-norm resolution. The maximum velocity is chosen as $v_{max} = 2.5$ (m/sec), which is very near to the failure region of 2-norm. The actual arm trajectory is seen to follow the resolved trajectory nearly precisely. Very small oscillation is observed whenever discontinuous torque is applied, but oscillation magnitude is negligible and quickly suppressed by system damping.

Figure 6 illustrates an example trajectory realizable only with CGI resolution (2-norm yields unrealizable results, and the CGI resultant violates the conditions of cCGI resolution). Simulation is conducted with a maximum velocity of $v_{max} = 5$ (m/sec). At $t \approx 0.085$ s, a discrepancy occurs between the saturated value of $\dot{q}_2$ in the 2-norm resolution and the re-optimization against saturated $\dot{q}_1$. This causes CGI to discontinuously redistribute velocity contributions amongst $\dot{q}_2$, $\dot{q}_3$, and $\dot{q}_4$. A discontinuous velocity trajectory yields very high acceleration and jerk values, leading to large oscillation and tracking error. As the arm decelerates, the opposite happens as the 2-norm resolution desaturates, causing another large jerk, exacerbating the oscillation.

Figure 7 illustrates the simulation repeated within the resolvable range of Continuous CGI resolution. Maximum velocity, v_{max} is set as 4.2 (m/sec). As the velocity profile lies within the resolvable region of cCGI resolution, the corresponding resolved joint velocity trajectories are continuous, implying well bounded acceleration and jerk values. As a result, the cCGI example traces the same path as the CGI example with only a 19 percent increase in time (still a large improvement over 2-norm), while eliminating nearly all of the experienced oscillation.

Figure 8 illustrates the maximum velocity error observed with CGI resolution for the given maximum velocity, or

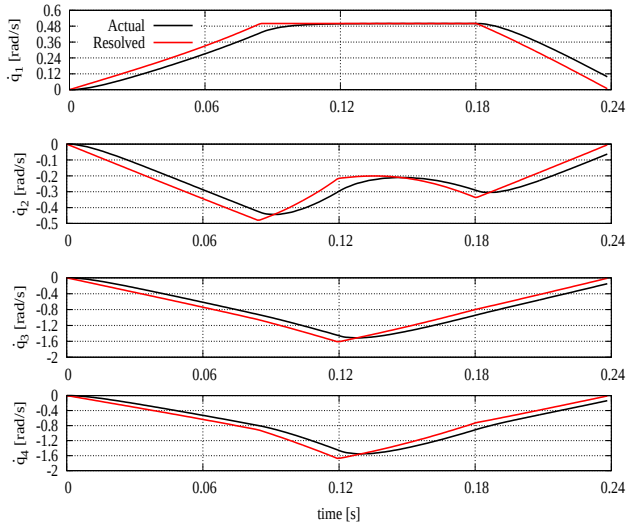

Fig. 7. Joint velocities of trajectory unrealizable with 2-norm, but realizable with Continuous CGI resolution. Resolution is guaranteed continuous, so the large oscillation observed in standard CGI resolution is not observed.

$$\max(|\dot{q}_i(t) - \dot{q}_i^{\text{res}}(t)|) \tag{28}$$

where $\dot{q}_i^{\text{res}}$ is the resolved solution for $\dot{q}_i$.

The CGI resolution remains equivalent to both the cCGI and pseudo-inverse resolution until the pseudo-inverse fails to produce realizable joint velocities at $v_{max} \approx 2.5$ (m/sec). Beyond this, the CGI resolution remains equivalent to cCGI resolution until $v_{max} \approx 4.3$ (m/sec) at which point two inputs saturate in one pseudo-inverse iteration, violating the assumption made in cCGI resolution. After this, only CGI realizes the desired trajectory, which as we saw before is accomplished by discontinuously changing joint tasks.

It can be seen that the maximum error remains low and increases at a fairly constant rate throughout the entire

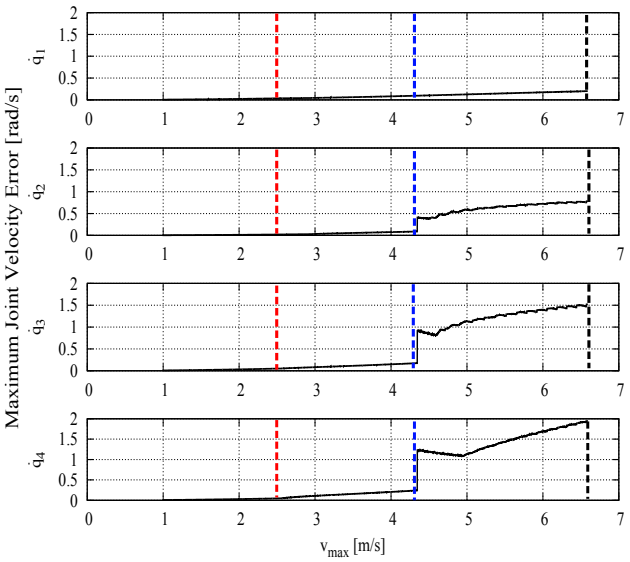

Fig. 8. Maximum joint velocity error given maximum end-effector velocity. 2-norm, cCGI, and CGI stop resolving the system at the red, blue, and black dotted-lines, respectively. Magnitude remains low through 2-norm and cCGI resolution, but rapidly increases once cCGI stops resolution.

domain of cCGI resolution, even after 2-norm fails at $v_{max} = 2.5$ (m/sec). Throughout this region, any error is attributable to the Tustin Transform, which allows for noise protection in our differential calculation. At $v_{max} \approx 4.3$ (m/sec), precisely when cCGI fails to resolve the system, the event which causes cCGI's failure triggers a large increase in joint velocity error in CGI resolution.

CGI is seen to continue resolution until $v_{max} \approx 6.6$ (m/sec) or $\approx 53\%$ over cCGI resolution, at the expense of oscillation and joint velocity error in the physical system. On the other hand, cCGI resolution is seen to improve the maximum realizable velocity $\approx 70\%$ over 2-norm without significant effect on the dynamic performance.

VI. Conclusion

The Cascaded Generalized Inverse is a method introduced to extend the resolution range of 2-norm redundancy resolution, but introduces the possibility of discontinuity in resolution.

This paper analyzes the effect of this discontinuity when CGI is applied in resolving kinematic redundancy in manipulators with finite joint stiffness, and is contrasted with the performance seen in 2-norm and Continuous CGI (cCGI) resolution, a variation on CGI resolution ensuring continuity of resolution.

The following results were found:

1) Both cCGI and CGI effectively extend the realizable output range of pseudo-inverse resolution, though CGI extends the system further.
2) Discontinuity observed in CGI resolution causes large torque and jerk, effecting undesired oscillation and velocity error in the system.
3) cCGI was able to increase maximum velocity significantly ($\approx 70\%$ over 2-norm) without the possibility of discontinuity (and the negative effects caused by it).

Although in general systems, the choice between cCGI and CGI resolution will be highly dependent on the considered system and performance requirements (weighing the dynamic effects of CGI against its larger output space), we have demonstrated that so long as computational power is not a significant concern, there is no reason not to use cCGI resolution over 2-norm. cCGI offers a larger realizable output range without any of the risks of CGI resolution.

Future works include experimental verification of simulation results.

References

[1] C. A. Klein and C. H. Huang, "Review of pseudoinverse control for use with kinematically redundant manipulators," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-13, no. 2, pp. 245-250, Mar. 1983.

[2] V. Salvucci, Y. Kimura, O. Sehoon, and Y. Hori, "Force maximization of bi-articularly actuated manipulators using infinity norm," *IEEE Trans. Mechatron*, vol. 18, no. 3, pp. 1080-1089, Jun. 2013.

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

[4] Y. Zhang, J. Wang, and Y. S. Xia, "A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits," *IEEE Trans. Neural Netw.*, vol.14, no. 3, pp. 658-667, May 2003.

[5] H. Ding and J. Wang, "Recurrent neural networks for minimum infinity-norm kinematic control of redundant manipulators," *IEEE Trans. Syst. Man Cybern. A., Syst. Humans*, vol. 29, no. 3, pp. 269-276, May 1999.

[6] Y. Zhang, C. A. Rabbath, and C. Y. Su, "Reconfigurable control allocation applied to an aircraft benchmark model," in *American Control Conf.*, Seattle, WA., 2008, pp. 1052-1057.

[7] Y. Zhang, V.S. Suresh, B. Jiang, and D. Theilliol, "Reconfigurable control allocation against aircraft control effector failures," in *IEEE Int. Conf. on Control Applications*, Singapore, 2007, pp. 1997-1202.

[8] A. Marks, J.F. Whidborne, and I. Yamamoto, "Control allocation for fault tolerant control of a VTOL octorotor," in *UKACC Int. Conf. on Control*, Cardiff, Wales, 2012, pp. 357-362.

[9] V. P. Bui, H. Kawai, Y. B. Kim, and K. S. Lee, "A ship berthing system design with four tug boats," *J. of Mechanical Science and Technology*, vol. 25, no. 5, pp. 1257-1264. May 2011.

[10] X. Shi, Y. Wei, J. Ning, and M. Fu, "Constrained control allocation using Cascading Generalized Inverse for dynamic positioning of ships," in *Int. Conf. on Mechatronics and Automation*, Beijing, China, 2008, pp. 1636-1640.

[11] T. Baratcart, V. Salvucci, and T. Koseki, "On the continuity of cascaded generalized inverse redundancy resolution, with application to kinematically redundant manipulators," in *IEEE Industrial Electronics Conf.*, Vienna, Austria, 2013.

[12] A. Maciejewski, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments," *Int. J. of Robotics Research*, vol. 4, no. 3 pp. 109-117, Sep. 1985.

[13] S. Chiaverini , "Singularity-Robust task-priority redundancy resolution for real- time kinematic control of robot manipulators," *IEEE Trans. Robot. Autom.*, Vol. 13, No. 3, pp. 398 - 410, Jun. 1997.

[14] J. C. Virnig and David S. Bodden. "Multivariable control allocation and control law conditioning when control effectors limit (STOVL aircraft)," in *AIAA Guidance, Navigation, and Control Conf.*, Scottsdale, AZ., 1994, pp. 572-582.

[15] K. A. Bordignon, "Constrained control allocation for systems with redundant control effectors," Ph.D. Dissertation, Aero. Eng., Virginia Poly. Inst. and State. Univ., Blacksburg, 1996.

[16] A. De Luca, R. Farina, P. Lucibello, "On the Control of Robots with Visco-Elastic Joints," in *Int. Conf. on Mechatronics and Automation*, Barcelona, Spain, 2005, pp. 4297 - 4302.