**AUTHORS RESPONSE TO REVIEWER COMMENTS**

Comments from reviewers have been presented in **bold** **text**; and comments from reviewers #1 and #2 have been listed as C1.X and C2.X, respectively, with ascending numbers used to label these comments. Responses to comments are in plain text and have been labelled using a similar format (e.g. R1.1 indicates the response to comment 1.1). Where revisions have been made they are referred to using line numbers of the revised manuscript, and these changes have been highlighted in the revised manuscript using **bold red text**.

**Reviewer #1**

**C1.1: Fox presents a comparison of 4 different algorithms that are used to solve for the kinematics and ground force trajectories of a model follow experimental measurement and yet are dynamically consistent. Specifically the analysis examines the solution times, residual forces and moments, joint kinematics, and joint kinetics that result from applying RRA, RRA3, AddBiomechanics, and Moco Track to experimental measurements of Hamner et al.'s 5 m/s treadmill running data. The analysis shows that there are meaninful differences between these different methods across each of the categories analysed.**

**In addition to the paper Fox has made his scripts publicly available and performed his analysis on publicly available data. This extra contribution is of great value and will serve as an excellent basis to fairly evaluate improvements made to both the settings of the algorithms he has tested, and to evaluate the performance of new algorithms. There has been a need for a paper exactly like the one Fox has written, and I'm glad to see that he has written it.**

R1.1: Thank you for this and your other comments on the paper. I believe that these (alongside reviewer #2’s comments) have improved the clarity of the paper, discussion of the results, and usefulness of the associated analysis code.

**C1.2: Expectations: It is easy for me to ask for something that takes an unreasonable amount of effort to address. It is not my intention to force you to do an unreasonable amount of work. I expect that you give each of my comments consideration and respond thoughtfully. If it's really not possible to meet some of my requests in the time you have, please say so.**

R1.2: Throughout my responses I have tried to implement all suggestions. Where any aspect has not been fully addressed, I have attempted to provide some discussion on the reasoning behind this.

**Paper**

**C1.3: It would be helpful to have a couple of paragraphs that place the OpenSim tools to improve dynamic consistency in the context of the wider literature. With some searching I've found some relevant literature that could be mentioned:  
  
André Venne, François Bailly, Eve Charbonneau, Jennifer Dowling-Medley & Mickaël Begon (2023) Optimal estimation of complex aerial movements using dynamic optimisation, Sports Biomechanics, 22:2, 300-315, DOI: 10.1080/14763141.2022.2066015**

**Lin YC, Pandy MG. Three-dimensional data-tracking dynamic optimization simulations of human locomotion generated by direct collocation. Journal of biomechanics. 2017 Jul 5;59:1-8.**

**Alonso FJ, Cuadrado J, Lugrís U, Pintado P. A compact smoothing-differentiation and projection approach for the kinematic data consistency of biomechanical systems. Multibody System Dynamics. 2010 Jun;24:67-80.**

**Remy, C. D., and Thelen, D. G. (January 6, 2009). "Optimal Estimation of Dynamically Consistent Kinematics and Kinetics for Forward Dynamic Simulation of Gait." ASME. J Biomech Eng. March 2009; 131(3): 031005.** [**https://doi.org/10.1115/1.3005148**](https://doi.org/10.1115/1.3005148)

**Jackson, J. N., Hass, C. J., and Fregly, B. J. (September 16, 2015). "Residual Elimination Algorithm Enhancements to Improve Foot Motion Tracking During Forward Dynamic Simulations of Gait." ASME. J Biomech Eng. November 2015; 137(11): 111002.** [**https://doi.org/10.1115/1.4031418**](https://doi.org/10.1115/1.4031418)

R1.3: The introduction has been updated to include a brief section that acknowledges previous efforts of researchers (some of the suggested studies above, plus some other additions from the literature) to develop approaches to reduce dynamic inconsistency in biomechanical data – alongside the approach that OpenSim offers. I have held back from reviewing this literature in detail, given the focus of this study is on the tools that OpenSim offers – with the reason being that they are widely used, and I believe much of the other work in this body of literature would be out of the reach of a number of biomechanical researchers (i.e. due to the complexity and manual implementation of modelling the multibody system). I have attempted to communicate this in the introduction and why the direction of the paper is geared towards the OpenSim tools (see …).

*Add excerpt…*

These additional past studies (e.g. Remy & Thelen, Jackson et al.) do, however, provide good benchmarks to compare the results of the present study against – please refer to R1.4 below for a more detailed discussion on how this has been implemented.

**C1.4: It would also be helpful to, in the discussion, place the results of your benchmark simulations in the context of the wider literature.**

R1.4: …review discussion – should be scope for this…

**C1.5: It would be helpful to begin the methods sub sections RRA, iterative RRA, MocoTrack, and AddBiomechanics with a brief summary (1-3 sentences) of how the methods work specifically covering: what quantities the algorithm manipulates, and briefly how the manipulation is done. This will help the readers to interpret the differences that are observed in the results more easily.**

R1.5: …scope to add this at the beginning of each tool section in data analysis…complete details from Dembia et al. and Werling et al. (e.g. equations etc.) are probably outside the scope of a few sentences, hence readers are referred to these papers for added details…

**C1.6: Using these additional algorithmic details (of the previous point) you should be able to explain some of the kinematic differences that appear in the solution of AddBiomechanics, and the joint torque differences that appear in the solution of Moco. It's possible that AddBiomechanics allows this deviation to develop due to the regularization term (the third term) in Eqn. 2 of Werling et al. For Moco, its possible that these joint torques deviations could be reduced by updating some of the costs and/or switching to the foot-ground contact model.**

**Obviously one can't be sure of this without doing a sensitivity analysis, which is beyond the scope of this paper. If you can, with a few hours of work, determine which parameters in AddBiomechanics and in Moco Track affect these differences it would be a useful addition to the discussion.**

**Werling K, Bianco NA, Raitor M, Stingel J, Hicks JL, Collins SH, Delp SL, Liu CK. AddBiomechanics: Automating model scaling, inverse kinematics, and inverse dynamics from human motion data through sequential optimization. bioRxiv. 2023 Jun 15:2023-06.**

R1.6: …AddBiomechanics focused on maker fitting, is marker drive – different to other tools, hence potential reason for differences…it is also an entirely different approach to others really, so not surprising that it perhaps has the largest kinematic deviations…

…unlikely to do full dataset…perhaps some score for case studies of changing Moco parameters to identify effects…

**C1.7: The long list of numerical results reported directly in the paper would be more easily read as a table. The figures are really useful, so its great that you've included this data graphically.**

R1.7: In line with similar comments made by reviewer #2 (see C2.3), the mean and standard deviation values originally reported in text have been removed and included in tables in the revised manuscript (see Tables X-X, line …).

**C1.8: It is mentioned on page 8 of Dembia et al. that ground forces can be applied directly to the model (which requires the addition of non-physical residual actuors) or simulated using a foot-ground contact model. On page 7 lines 111-112 it is mentioned that the experimental external GRFs are applied directly to the model. Given that:**

**a. Please report the range of the forces generated by the residual actuators during the course of Moco Track's solution. This might be helpful to explain one puzzling feature of Figure 5: Moco Track's ankle torques track the data very well, but the rest of the joint torques are quite noisy.**

**b. If time permits, it would be very helpful to see how the solutions change when the foot-ground contact model is used in combination with an additional tracking term on the ground force trajectory.**

**Dembia CL, Bianco NA, Falisse A, Hicks JL, Delp SL. Opensim moco: Musculoskeletal optimal control. PLOS Computational Biology. 2020 Dec 28;16(12):e1008493.**

R1.8: …could do a case study on foot ground contact tracking…unsure though if this is testing the same thing – kinematic deviations are expected with ground contact model, and the parameters used in these contact models are highly sensitive…

**C1.9: It would be helpful to see the ground forces generated by the different methods in comparison to the recorded ground forces and CoP trajectories. While RRA applies the ground forces directly, both AddBiomechanics and Moco Track can make adjustments: AddBiomechanics makes small adjustments to compensate for force plates that are not perfectly flat (see page 27 of Werling et al.), while the ground forces from Moco Track can be solved for using a foot contact model (see page 8 of Dembia et al.).**

**I suspect that some of the noisy joint torques that you're seeing in Moco's solution in Figure 5 will also be visible in the simulated ground reaction forces and CoP trajectories.**

**Werling K, Bianco NA, Raitor M, Stingel J, Hicks JL, Collins SH, Delp SL, Liu CK. AddBiomechanics: Automating model scaling, inverse kinematics, and inverse dynamics from human motion data through sequential optimization. bioRxiv. 2023 Jun 15:2023-06.**

R1.9: …need to check on this…unsure how to get the altered forces?

**C1.10: This is more of a comment, but the oscillatory joint torques produced by Moco in Figure 5 reminds me of the kind of output that is produced when the squared position differences are minimized on an integrated system. The classic way to deal with this problem is to minimize the squared differences of both position and velocity. The work that you are doing is quite valuable so that the defaults in Moco Track (maybe) can be improved.**

R1.10: …could do a case study on this too, where velocity is also tracked/minimised…?

…this actually could be somewhat of a fix as the case study test proved to help with the smoothness of some joint moments – will it fix with group average? In hindsight, it was probably fair to do this as an appropriate comparator to RRA given that it tracks accelerations and not positions…

**C1.11: While it is nice to see some of the author's personality in the writing, you may reconsider referring to the paper as a quest. On page 25 line 332 you are not doing yourself any favours by adding 'For the sake of complete honesty, very little effort went into ...'. The work you have done is valuable but these statements may lead some readers to underappreciate your work.**

R1.11: …consider casual language…

**C1.12: The references need some attention:**

**a. None of the entries has doi numbers listed.**

**b. If possible, replace references 4 and 10 with a referreed source. I realize this is probably not possible in the case of RRA, which to my knowledge has never (oddly) been the sole topic of a published paper. RRA is described in Chand John's PhD thesis and in Delp et al. 2007. As for reference 10, the url can be included as a note with reference 2 or as a foot note. If you decide to keep references 4 and 10 as web references you'll need to add some extra information such as the date you accessed it.**

**c. Reference 5: inconsisent capitialization of the journal title**

**John CT. Stabilization of human walking by muscles revealed using three-dimensional muscle-driven simulations. Stanford University; 2012.**

**Delp SL, Anderson FC, Arnold AS, Loan P, Habib A, John CT, Guendelman E, Thelen DG. OpenSim: open-source software to create and analyze dynamic simulations of movement. IEEE transactions on biomedical engineering. 2007 Oct 22;54(11):1940-50.**

R1.12: …review and update references…

**Code**

**I have cloned and run (some) of the python scripts that you've put on https://github.com/aaronsfox/dynamic-consistency-quest on both a Windows machine and a Linux machine. I have a few comments:**

**C1.13: On Windows I've run 01\_runSimulation.py with (lines 154-157)**

**runRRA=True**

**runRRA3=True**

**runMoco=True**

**runAddBiomech=True**

**and the script fails on line 532 when os.rename throws a FileExistsError.**

**a. If you instead use os.replace you'll have the desired effect without the FileExistsError. I've just updated os.rename with os.replace and the script runs**

**b. until it hits the runAddBiomech code block at line 1117 and then it complains that os.chdir(f'..\\..\\..\\data\\HamnerDelp2013\\{subject}\\addBiomechanics\\{runLabel}') cannot be found. Yet if I start the script in this block by setting runRRA, runRRA3, and runMoco to False, the script appears to terminate without running addBiomechanics. Looking at the code, I can see that this is intentional from the message "This section of the script does not run the entire ...". It would be nice to see this message printed to the screen, and maybe with some short instructions on how to proceed to use AddBiomechanics: for example, is the dataset you've loaded already available at AddBiomechanics? If not, what should I do to process your files so that I can reproduce your results?**

**c. The scripts 02\_collateSimulations.py and 03\_analyseSimulations.py ran smoothly. When I looked at results/HamnerDelpDataset/outputs I was happy to see that the results had been updated. However, when I looked at results/HamnerDelpDataset/figures the figures had not been updated. It would be best to update the script so that the figures are updated automatically.**

**If I manually deleted the existing figures then new ones were generated, and these new plots matched the published versions with slight differences in timing (due to the machine). Some of the line colours in residualForces.png, and residualMoments.png looked odd (for example the AddBiomechanics box and whisker plots had blue whiskers) but this migth be a problem with my python installation: it was giving a 'FutureWarning' that passing 'palette' without assigning 'hue' is deprecated ... along with a few other warnings.**

**d. If possible, please make sure that someone can run all of your scripts from scratch (with new simulations) without error.**

R1.13: …rename function updated to replace…added some printed out instructions for addBiomechanics (i.e. upload the copied results to server to reprocess…)…altered boxplot colouring approach to use Matplotlib instead of Seaborn, as version changes seem to have an effect on how patches, colouring etc. are used with Seaborn…

**C1.14: As it turns out, may main work computer is a Linux machine, and so, I've also tested 01\_runSimulations.py on my Linux machine. There are a number of path related problems that should be fixed if you want to be cross-platform compatible:**

**a. Every path that appears in your python scripts needs to be updated using the os.path.join command. For example:**

**Near line 178 of 01\_run Simulations.py appears a path that is Windows specific:**

**With open(f'..\\..\\data\\HamnerDelp2013\\{subject}\\expData\\gaitTimes.pkl', 'rb') as openFile:**

**Replacing this line with the following allows it to run on any machine:**

**with open(os.path.join('..','..','data','HamnerDelp2013',subject,'expData','gaitTimes.pkl'), 'rb') as openFile:**

**b. The XML files that OpenSim generates for RRA, for example, have windows-specific slashes. For example in subject01\_run5\_cycle1\_setupRRA.xml the model file is set as**

**..\..\model\subject01\_adjusted\_scaled.osim**

**which is windows-specific. Changing these path slashes**

**../../model/subject01\_adjusted\_scaled.osim**

**allows the model to be loaded. I think that this can be done by using the os.path.join command in, for example, the rraTool.setModelFilename(...) command (and all similar commands).**

R1.14: Thank you for this recommendation — I believe this will be useful to me for this project plus any in the future where I share analysis code. I have replaced the paths in all scripts to be constructed using *os.path.joint()* which should hopefully address cross-platform issues. Unfortunately, I am not adept to or have access to a Linux environment and hence was unable to test this. However, I have added details to the datasets main README (see …add zenodo link…) about code being tested only on a Windows machine and the associated Python dependencies.

**C1.15: The script assumes that OpenSim is installed at 'C:\\OpenSim 4.3\\Geometry'. It would be very helpful if, perhaps in the README.md file, the details of the installation and dependencies on your machine where listed. In addition, it would be useful if some hint was provided about which paths need to be updated in the code if the installation layout and/or operating system differs from the one that you used.**

R1.15: In response to this comment, details have been added to the datasets main README (see …add zenodo link…) specifying the assumption that the OpenSim geometry install path is at *C:\OpenSim 4.3 Geometry\*. The analysis script (see *runSimulations.py*) also now prints out a message about where the geometry path has been set and prompts users to change this if incorrect.

**C1.16: There are likely many other small things to correct to get the code you've written functioning cross-platform. Getting code to run flawlessly cross platform is tedious work, but it will allow a lot of other people to use your code. I see two ways to proceed:**

**a. Get access to a Linux machine (or a colleague with a Linux machine) and do the work to get everything functioning on Linux. If you have never used Linux, it would be best to get help with this - there are a lot of small things that will be time consuming for you to learn on your own.**

**b. Explicitly mention that the code is developed for Windows, but in principle, can be made to work cross-platform with the edits that I've mentioned. The code is publicly available, and so, in the best case someone who'd like to use your work will contribute the updates needed to make the code run cross-platform.**

R1.16: As noted in R1.14, I am unable to confirm proper functioning of code on a non-Windows system, and hence added details to the datasets main README (see …add zenodo link…) about code being tested only on a Windows machine and the associated Python dependencies.

**C1.17: While it is convenient to put binary data in a github repository, this can cause problems: if the data ever changes git will store just store, basically, the old file and the new file. In the case of simulation data this will make the repository size balloon. Usually data is stored separately from git, and not under revision control. You could for, example, store the non-python code contents of data/HamnerDelp2013, paper, preprint, and results folders in Zenodo.**

R1.17: I noted earlier that reviewer #1 accessed the associated code and data directly from the GitHub page – however the desired approach for eventual readers is to use the SimTK link (i.e. <https://simtk.org/projects/dynamic-quest>) specified in the *Data Analysis* section of the paper. The SimTK page directs to a Zenodo link (originally <https://zenodo.org/records/8307565>; updated version now …add Zenodo link…) which will archive the code and data in the current state at paper acceptance. It is possible that some updates may occur on the GitHub page, but these will be unaffected by readers visiting the SimTK page and linked Zenodo archive. Theoretically this should provide a stable, consistent and working analysis code for any readers of the paper.

**Reviewer #2**

**C2.1: While the framing of the study as a “Quest” makes the paper more interesting for the reader, the language at times crosses the line into an overly casual tone. Specifically, it is suggested to revise the following lines:**

**- Line 55: “nay, quest”**

**- Line 332: “for the sake of complete honesty, very little effort…”**

R2.1: …reconsider casual language…

**C2.2: Lines 43 and 44 are somewhat misleading. Specific recommendations do exist for RRA. The OpenSim documentation provides guidelines for acceptable kinematic error and residual forces/moments, Hicks et al 2015 provide an extensive discussion on the topic, and references [5] (Samaan et al) and [6] (Sturdy et al) in the present manuscript all provide suggestions on acceptable thresholds and how many iterations should be required. For purposes of the present manuscript, it would be more accurate to state that there are not yet guidelines for selecting between RRA and other new approaches, such as Moco and AddBiomechanics.**

R2.2: I believe there was a lack of clarity around the language originally used in this statement which I have attempted to clarify in the revised submission. It’s important to note that this is referring to repeat iterations of the same RRA process (i.e. using the same task weights on an updated model and kinematics) – which differs to the iterative process of the studies by Samaan et al. and Sturdy et al. (i.e. iteratively adjusting task weights to optimise and achieve minimised residuals). The statement in the revised submission has been edited to include the fact that benchmarks for residuals (i.e. from Hicks et al.) exist – however it is difficult to determine whether a singular or iterative RRA approach can achieve these benchmarks. Further, a statement has been added to highlight the lack of research benchmarking the RRA to both existing and new approaches for residual reduction (see …).

*Add excerpt…*

**C2.3: The Results section would benefit greatly from a table that compares the mean±SD values which are currently included in the text. As written, it is difficult for the reader to quickly compare the various methods. It would particularly help highlight the tradeoff in computational time and residual forces/moments and kinematic error.**

R2.3: In line with similar comments made by reviewer #1 (see C1.7), the mean and standard deviation values originally reported in text have been removed and included in tables in the revised manuscript (see Tables X-X, line …). I have, however, left the in-text reporting for computational time — as this would be reported in a single-line table and potentially not offer great benefit over the figure combined with the in-text reporting.

**C2.4: The plots (Figures 1-5, 7) are impossible to read in grayscale (and difficult to read even in color). Distinct symbols and different line styles should be used to ensure readability without color. Legends should be added to the figures.**

R2.4: …to get nice line styles visible, altered some line styles and added markers, to make these visuals distinguishable with different line-styles, the SD around data has been removed…to be fair some of these are still difficult to distinguish given the overlap of data, particularly with joint moment data and the different scale the solvers other than Moco are on…marker shapes changed…legends added…royal society open science a predominantly online journal viewed web-based?

**C2.5: Line 267 states that MocoTrack substantially outperforms previous efforts in [5] (Samaan et al) and [6] (Sturdy et al). The optimizations tested by Samaan et al required 64 and 98 hours to complete, which is indeed a substantial difference. Sturdy et al tested 200 iterations which required ~2 hrs to complete. However, most of the improvements for the 5 m/s case occurred between 30-50 iterations, which would correspond to approximately 20-30 minutes. This is comparable to the MocoTrack results. It seems that a comprehensive investigation of the available options should consider this approach, as it is unclear how the results from 30-50 “optimized” RRA iterations would compare to MocoTrack.**

R2.5: Thank you for flagging these additional findings, particularly from the work of Sturdy et al. I believe testing of the algorithm employed by Sturdy et al. is outside the scope of this study (or more so would simply be repeating their original analyses) — however a more balanced discussion of the computational time for *MocoTrack* in the present study compared to this existing literature has been included in the revised submission (see line …).

*Add excerpt…*

**C2.6: Figure 6 is a nice visual comparison of the various methods. However, RRA commonly allows substantial pelvis translational drift (particularly in the vertical and anterior/posterior directions) in order to reduce the residual forces. It appears that in Figure 6 pelvis translational error is not represented. This is ok, as it allows visual comparison of other kinematic differences, but no mention is made about pelvis translational error, and it is omitted Figure 4. Results for pelvis translational error should be reported.**

R2.6: Not including pelvis translations in the original kinematic figure was an oversight – so thank you for identifying this error. These data have now been included with the in the figure displaying mean kinematics from each tool (see Figure …, line …). Figure 6 in the original submission (and maintained in the revised submission) did, however, already consider pelvis translations in displaying the average joint motions from each tool. The updated mean kinematics in Figure … demonstrate the similarity in pelvic translation data across the tools – and hence why it seemed like they had been excluded. The additional pelvis translation data now included in Figure … will help readers identify the subtle differences in Figure … (e.g. the slightly more posteriorly translated pelvis from *AddBiomechanics* results).

**C2.7: The noise in joint kinetics from MocoTrack warrants further discussion. A common use of RRA is as a preprocessing step before a muscle-driven simulation. The oscillations in joint moments in the MocoTrack results would lead to extremely poor muscle-driven simulation performance. The deeper question here is – are the noisy joint moments more “true”? The author mentions soft tissue motion as a potential source of the oscillation. However, does that mean that the joint moments are in fact oscillatory? Biomechanists tend to prefer smooth joint moments as input to a muscle-driven simulation. However, if the joint moments are in fact oscillatory due to soft tissue motion, that represents a significant claim.**

R2.7: …don’t think there is any way to confirm or refute this given the potential sources of error across data collection/analytical process…could probably get rid of these by altering the parameters of Moco (i.e. minimise accelerations, or the suggestion about velocities earlier) – but who’s to say that these are more or less true…as pointed out here, it is likely necessary to avoid these oscillations when running muscle driven simulations…so even if the oscillations were real, they would need to be removed…ADD SOME DISCUSSION ON THIS IN PAPER…

**C2.8: Line 123: Typo, “participants” should be “participant’s”**

R2.8: This typo, alongside a number of similar typos related to “participant’s” (e.g. in figure captions) have been corrected in the revised manuscript.

**C2.9: Line 396: The last sentence in the acknowledgments appears to have been cut off**

R2.9: …acknowledgements has been cut-off – fix…