Efficiently downsampling inelastic cyclic stress-strain data while preserving its curvature

A Hartloper

August 5, 2021

Abstract

A downsampling method for stress-strain data that uses few points to preserve curvature is proposed in this document. This method uses existing algorithms that satisfy a local perpendicular distance criterion within a pre-defined tolerance. The proposed method is then extended to preserve the global difference in energy between the original and downsampled data within a specified tolerance. The global energy downsampling method is particularly useful because it is directly related to the metric used in material calibration. Results from two datasets indicate that, with around 0.5 % difference in energy, the downsampled data has 70–100 times less points. The proposed method leads to 2–3 times less points than the previous integer reduction downsampling method and also appears to give a better representation of the stress-strain data.

1 Introduction

The raw data (i.e., extensometer displacement, load cell force) in experiments used for material model calibration are typically sampled at 20 Hz in the EPFL test setup. As experiments range from around a few minutes (e.g., LP1) to hours (LP5), this results in thousands to hundreds of thousands of data points for each experiment. However, the sampled data is often denser than required to reasonably describe the stress-strain behavior of materials under the applied strain rates. When it comes to calibrating material models, the time required is proportional to the number of data points. Therefore, time can be saved by reducing the number of data points prior to running the calibration while maintaining the fidelity of the data.

Experience has shown that the fidelity of an experiment can be kept with around 10–100 times less points depending on the load protocol. Calibrating the UVC model using ten load protocols *without* reducing the number of points could, therefore, take weeks to a year. A "good" method of reducing the number of data points, or *downsampling* the data, is essential to keep the calibration time reasonable (say, on the order of hours to days per material).

The properties that define a good downsampling method in this context are now defined.

A good downsampler for inelastic, cyclic, stress-strain data of structural steels:

1. samples enough points in the initial elastic region to obtain a good estimate of the initial elastic modulus,

2. samples the upper yield point,

3. samples all the maxima and minima in each loading cycle (i.e., the *peaks* of the stress-strain graph),

4. samples enough points to retain the fidelity of curved regions in the stress-strain graph,

5. samples as few points as necessary in straight regions,

6. can handle noise in the measurements,

7. can handle different strain rates,

8. has interpretable heuristic parameters,

9. is easy to use.

A method that combines different techniques to satisfy these desirable properties is developed in this document.

My initial work to downsample stress-strain data combined two techniques: (i) find the peaks of the stress-strain data, then (ii) downsample by an integer factor[1]. This method preserves the peaks, however, it either removes too many points in the curved regions, or is inefficient in the straight regions. Moreover, it is a matter of "art" to pick a reasonable integer factor for each case, and, this has to be done twice for each experiment because of the different strain rates in most load protocols! There are several areas to improve this method that lead to this study.

The initial inspiration for improvement was this blog[2]. Then I found a wealth of literature, particularly in the domain of cartography. Two subproblems are defined in the literature related to this problem [Imai and Iri, 1988]: the min-# problem and the min-ϵ problem. The objective of the first problem is to determined the minimum number of sample points to provide a reasonably good approximation of the curve, the second is to find the minimum error between the original and sampled curves for a given number of points. In the first, an error critieria is specified (ϵ), and in the

---

[1] https://en.wikipedia.org/wiki/Downsampling_(signal_processing)

[2] https://kaushikghose.wordpress.com/2017/11/25/adaptively-downsampling-a-curve/.

second the number of points is specified (N). Many approximate solutions exist [Ramer, 1972, Douglas and Peucker, 1973, Heckbert and Garland, 1997] as well as optimal ones [Imai and Iri, 1988, Perez and Vidal, 1994, Chan and Chin, 1996]. The optimal solution to the first problem is easier (takes less computational effort) than the second one by a factor of $\log N$ [Chan and Chin, 1996]. I have focused on applications of approximate solutions to the min-# problem.

The most popular algorithm to determine an approximate solution to the min-# problem appears to be the Ramer–Douglas–Peucker (RDP) algorithm [Ramer, 1972, Douglas and Peucker, 1973, Heckbert and Garland, 1997]. The RDP algorithm is not optimal and only finds a number of points N close to the minimum. A number of additional points are certain to be added due to the criteria of selecting sample points in the algorithm. This point will be explained later in this document. However, the RDP algorithm is still quite good, comparatively fast, and has been widely implemented. Excellent visualizations of the RDP algorithm can also be found[3][4].

The aim in this document is to use these established downsampling techniques to improve upon my existing integer reduction method. Ideally, the proposed method will be able to satisfy the desirable criteria set forth above. The following sections outline the development process and the evaluation of the proposed method.

2 Proposed method

The proposed method is composed of two main parts and two auxiliary parts. The first main part is to find the peaks in the stress-strain data. The second main part is to downsample the entire stress-strain history based on its curvature. The auxiliary parts are to keep additional points in the initial elastic region and remove cycles after saturation in constant amplitude tests. The final downsampled data is the unique set of sample points from all the parts.

Mosts of the heavy lifting is done in the second main part. The first main part is essential to overcome *aliasing* in the sampled data and determine properties such as the number of cycles. The auxiliary parts ensure fidelity of the initial elastic region and remove data that does not contribute much to the material behavior. Details are given for each part and a summary of the entire procedure is given afterwards.

---

[3]https://karthaus.nl/rdp/
[4]https://en.wikipedia.org/wiki/Ramer-Douglas-Peucker_algorithm

2.1 Identifying peaks in the stress-strain graph

The peaks in the stress-strain graph are identified by the minima or maxima in each cycle. Cycles can be fairly reliably identified by the switching of signs in the stress signal. Therefore, the peaks are the minima and maxima between each change in sign of the stress signal. This method works well if the stress crosses zero in each cycle and there is no substantial noise in the stress signal around zero.

An attempt at identifying upper yield point, if it exists, is made in this first part. First, the 0.2 % offset yield stress is computed using the initial elastic modulus. The initial elastic modulus itself is computed using linear regression up to $0.65 f_{y,n}$, where $f_{y,n}$ is the nominal yield stress of the material. Then, the upper yield point is selected as the point of maximum stress up to the 0.2 % offset yield stress. If there is no upper yield point (due to the material or imperfections in the specimen), then the maximum will typically be at the 0.2 % offset yield stress, and this point is sampled instead. The upper yield point is the first "peak" in the stress-strain graph.

Another important point in stress-strain data tested according to the RESSLab protocols is the first instance 2 % strain amplitude is crossed. This point is important because the strain rate changes by a factor of around 30 after 2 % strain. Therefore, this point is sampled as well. A final point is also sampled. Unless otherwise specified (i.e., due to buckling), the final point is selected at 12.5 % strain due to limitations in the small extensometer used for M8 experiments, or the last point in the history if 12.5 % strain is never obtained. Any data past the final point are neglected.

To summarize, the sampled points from the first part are:

- the upper yield point (or 0.2 % offset yield stress point),

- the point where 2 % strain amplitude is crossed (not applicable for LPs 4 and 5),

- the peaks in the stress-strain graph, and

- the final point.

2.2 Curvature-preserving downsampling

We are concerned with preserving the "large-scale" curvature in a stress-strain graph, not localized curvature due to noisy measurements. The idea in the second part of the overall procedure is to remove as many points as possible in the "straight" regions of the stress-strain graph and keep as many as necessary in the "curved" regions. The key to these methods is to define what is "straight", and to sample a point every time that this condition is violated.

2.2.1   First attempt: Maximum deivation downsampler

The method, named the "maximum deviation downsampler", is outlined in Algorithm 1 based on a blog by Kaushik Ghose in footnote 2. After further reading I realized the link with other methods outlined in the introduction. This method requires a set of points and a tolerance, ϵ. The algorithm starts at the first point and steps forward through each point in the set. A line is computed between the starting point and the current point. The perpendicular distance is computed between the line and each point in the range of the starting and current point. Therefore, the perpendicular distance acts as a proxy for the radius of curvature. If the perpendicular distance is greater than the tolerance, the point before the current point is sampled; the sampled point is then set as the starting point and the process continues.

The keys to making Algorithm 1 in the current context are to: compute an appropriate perpendicular distance, to use an appropriate tolerance, and to remove noise from the data. One challenge in stress-strain data is that the "distances" in stress and strain have different units. This is handled by normalizing the stress data by the range in stress ($\sigma_i^* = \sigma_i/(\max\sigma - \min\sigma)$), and the strain by the range in strain ($\varepsilon_i^* = \varepsilon_i/(\max\varepsilon - \min\varepsilon)$). Therefore, the normalized values have a range of 1.0 in both axes.

Given this normalization, a tolerance of $\epsilon = 0.001$ is usually appropriate. This tolerance can be interpreted that "straight" is defined as a deviation less than 1/1000'th of the maximum "distance" spanned by the graph. Finally, noise in the data can cause spurious sampling of points. Therefore, a moving average filter (Savitzky-Golay, `scipy.signal.savgol_filter`) is applied to the stress-strain data prior to normalization. The moving average filter requires a window length, w and an interpolation order, p. Filtering the stress-strain data removes the noise, but leads to aliasing. Recall that the peaks of the stress-strain signal are already included in part 1 of the overall procedure, therefore, minor aliasing is not an issue in part 2. Therefore, the set of 2-D points $\{x_i\}$ input to Algorithm 1 are the filtered, normalized stress-strain data.

2.2.2   Second attempt: The RDP algorithm

The RDP algorithm is similar to the maximum deviation downsampler in that it selects sample points based on local perpendicular distances. What differs is the choice of starting sample points and how sample points are progressively chosen. The process is illustrated in Figure 1. The RDP algorithm starts with $i = 0$ and $j = N - 1$, then the first sample point is that with maximum $d_{ij}^{(k)}$. This bisects the overall curve, and each section is sampled recursively until the maximum deviation is lower than the tolerance in all sections. Notice that several points are "skipped" in the regions of low curvature, while the density of points sampled increases in the curved

---

Algorithm 1 Maximum deviation downsampler.

1: **input:** Set of points, $\{x_i\}_{i=0}^{N-1}$ with $x_i \in \mathbb{R}^n$ and $N \in \mathbb{Z}$; and tolerance, $\epsilon \in \mathbb{R}$.
2: **output:** Set of indices between $[0, N)$ to keep, k_{sample}.
3: $k_{sample} \leftarrow \emptyset$
4: $i \leftarrow 0$
5: **for** $j \in (0, \ldots, N)$ **do**
6:      $l_{ij} \leftarrow x_j - x_i$ $\triangleright$ Line between i and j
7:      **for** $k \in (i, \ldots, j)$ **do**
8:          $d_{ij}^{(k)} \leftarrow x_k \perp l_{ij}$ $\triangleright$ Perpendicular line between x_k and l_{ij}
9:      **end for**
10:     **if** $\max \left\| d_{ij}^{(k)} \right\|_2 > \epsilon$ **then**
11:          $k_{sample} \leftarrow k_{sample} \cup \{j - 1\}$
12:          $i \leftarrow j - 1$
13:     **end if**
14: **end for**

---

regions. The same need for normalization and sensitivity to noise exist in the RDP algorithm as in the maximum deviation downsampler. Therefore, the stress signal is filtered, then the stress-strain is normalized prior to applying the RDP algorithm.

2.2.3   A related approach: Limiting global error

Both the max deviation downsampler and the RDP algorithm operate using a local tolerance, ϵ. However, it may be more appropriate to use a global tolerance on the difference between sampled and original data. The downsampled data can be compared with the original data using our "standard" error metric:

$$\tilde{\varphi} = \sqrt{\frac{\int (\sigma_{test} - \sigma_{ds})^2 \mathrm{d}\varepsilon^*}{E_{orig}}}, \tag{1}$$

where E_{orig} is the total accumulated squared strain energy in the original data, $\sigma_{test}, \sigma_{ds}$ are the stress signals of the original and downsampled data, and ε^* is the accumulated strain. The definition of $\tilde{\varphi}$ is similar to that of $\bar{\varphi}$ from de Castro e Sousa et al. [2020] but simplified to a single test. Linear interpolation is used to compute the downsampled data at each point between the sampled points, and the integrals are computed using the trapezoidal rule.

     An algorithm to keep the error metric below a global tolerance, ϵ_g is provided in Algorithm 2. The idea is to select an initial ("large") local tolerance, ϵ_0. Then reduce the local tolerance and re-run the downsampling until $\tilde{\varphi} < \epsilon_g$. This always converges because the error is zero at the limit of
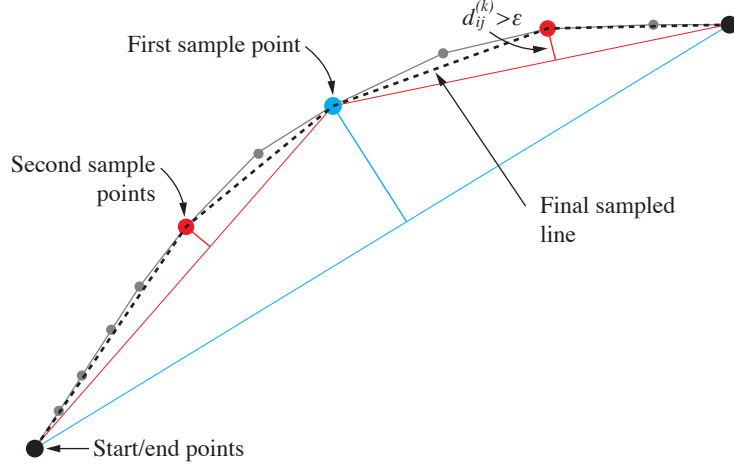
Figure 1: RDP algorithm process. Starts by sampling the end points, then recursively sampling the point with maximum perpendicular distance between each set of sample points. Stops when the maximum perpendicular distance is less than the tolerance.

sampling all the points. This procedure is made accessible by the highly efficient implementation of the RDP algorithm in the Python package `polyprox`, however, the same process can be applied using other downsampling methods with a bit of time. Note that my implementation uses the filtered/scaled "σ_{test}" to compute $\tilde{\varphi}$, thus, the true error between the sampled data tends to be slightly higher.

2.3 Additional sampling in the initial elastic region

The initial elastic region is assumed to be bounded by the first point in the data and the upper yield point obtained in part 1 of the overall procedure. It is important to have a few points in this region to obtain a good estimate of the initial elastic properties from the downsampled data. Therefore, $n_{elastic}$, additional points are sampled in the initial elastic region at approximately evenly spaced strain intervals. Each additional point is sampled using

$$j = \arg\min |x_i - x_t|, \tag{2}$$

where x_t are the target evenly spaced strain points.

2.4 Saturation in constant amplitude tests

Cyclic hardening saturates logarithmically under constant amplitude cyclic loading in structural steel materials. Therefore, many cycles may be associated with a small increase in stress and a large portion of data is associated with a small amount of information. One option to reduce this data is to cut

7

Algorithm 2 Downsampling based on global energy error.

1: **input:** Set of points, $\{x_i\}_{i=0}^{N-1}$ with $x_i \in \mathbb{R}^n$ and $N \in \mathbb{Z}$; initial local tolerance, $\epsilon_0 \in \mathbb{R}$; and global tolerance $\epsilon_g \in \mathbb{R}$.

2: **output:** Set of indices between $[0, N)$ to keep, k_{sample}.

3:   $\chi \leftarrow 1.05$ ▷ Factor to accelerate convergence

4:   $n_g \leftarrow 10$ ▷ Maximum number of iterations

5:   $i \leftarrow 0$

6:   $\tilde{\varphi} \leftarrow 10 \cdot \epsilon_g$ ▷ Just to be larger than ϵ_g initially

7:   $\epsilon \leftarrow \epsilon_0$

8:   **while** $\tilde{\varphi} > \epsilon_g$ and $i < n_g$ **do**

9:       $k_{sample} \leftarrow$ indices from downsampling with local tolerance ϵ

10:      $\tilde{\varphi} \leftarrow$ error metric computed using $\{x_i\}$ and k_{sample}

11:      **if** $\tilde{\varphi} > \epsilon_g$ **then**

12:          $\epsilon \leftarrow \epsilon \cdot (\epsilon_g/(\chi\,\tilde{\varphi}))$

13:          $i \leftarrow i + 1$

14:      **end if**

15: **end while**

---

the cycles after saturation has been reached. The saturation point is defined herein as the index of the first peak i that satisfies $\sigma_i > \eta \cdot \max \sigma$ in the positive loading direction and $\sigma_i < \eta \cdot \min \sigma$ in the negative loading direction, with $0 < \eta \leq 1$. The data after the saturation point is not sampled.

Cutting cycles after saturation in constant amplitude tests requires the definition of the saturation tolerance, η. The minimum number of cycles to include, n_{cyc}, is also specified.

2.5   Overall procedure

The overall procedure is summarized in Algorithm 3. The procedure consists of the two main and auxillary parts outlined earlier. The original stress-strain data input to the procedure is $X = \{x_i\}_{i=0}^{N-1}$. Algorithm 3 produces a set of indices, k_{sample}. The downsampled stress-strain data is $X_{ds} = \{x_i \mid i \in k_{sample}\}$. I emphasize that the downsampled data is sampled directly from the original data, therefore, $X_{ds} \subset X$. (Just to emphasize that the filtered, normalized data is only used to determine the indices and the added elastic points are the closest indices to the targets.) The global error method can be used instead by swapping line 16 with the results from Algorithm 2.

I recommend using the following input parameters in Algorithm 3:

- f_{yn}: based on the material

- $\eta = 0.99$

- $n_{cyc} = 20$

- if local method: $\epsilon = 0.001$

- if global method: $\epsilon_0 = 0.1$, $\epsilon_g = 0.005$

- $w = 5$

- $\alpha = 1$

- $p = 1$

- $n_{elastic} = 7$

The parameter $\eta = 0.99$ was selected as being close to 1.0 and $n_{cyc} = 20$ was selected from de Castro e Sousa et al. [2020] because at least 20 cycles are indicated. Two window lengths are required for pre- and post-2 % strain due to the different strain rates. A value of $w = 5$ provides a weighted average of ± 2 points, and $\alpha = 11$ was selected because, even though the strain rate is around 26 times lower, the window length lead to too much aliasing in the pre-2 % region with larger values. Linear interpolation ($p = 1$) is recommended because it is most effective at reducing the noise. The trade-off is that linear interpolation increase aliasing around peaks, however, the peaks are already included in part 1. Seven additional points are added to the initial elastic region to give nine points total (seven + start + upper yield). This should give 4–5 points in the $0.65 f_{yn}$ region used to compute the initial elastic modulus. The ϵ, ϵ_g recommendations were selected through trial and error to see when the stress-strain graphs became visually similar.

3 Results

### 3.1 Different filtering parameters

Three options are evaluated: (1) $w = 5, \alpha = 11$, (2) no filtering, (3) $w = 5, \alpha = 1$. Across all 16 datasets, the average total number of points in each dataset with downsampling are: 5340, 8830, 5574. Relative to Option 1, this is a 1.65 and 1.04 times increase in the number of points for Options 2 and 3, respectively. Option 1 is leads to lower number of points for all the load protocols compared to Option 3, except for LPs 4 and 5 (constant amplitude). Detailed results are shown for LP6 from the IPE360 web dataset in Figures 2 and 3. For this particular case, the number of points in the original data is 17259, the number of points for Option 1 is 358, for Option 2 is 2465, and for Option 3 is 633.

3.2 Detailed results

Detailed results from the HEM320 web and S235/275 15 mm plate datasets are provided in this section to comment on certain aspects of downsampling

**Algorithm 3** Overall summary of the proposed downsampling method.

1: **input data:** set of stress-strain data, $\{x_i\}_{i=0}^{N-1}$
2: **input parameters:** nominal yield stress, f_{yn}; saturation tolerance in constant amplitude loading, η; minimum cycles in constant amplitude loading, n_{cyc}; downsampler tolerance(s), ϵ or ϵ_0, ϵ_g; filter window length post-2 % strain, w; filter window length pre-2 % factor, α; filter interpolation order, p; number of additional points in the initial elastic region, $n_{elastic}$.
3: **output:** k_{sample} the indices of the sampled points.

4: Compute the initial elastic modulus and initial 0.2 % offset yield stress using the nominal yield stress, f_{yn}
5: $k_1 \leftarrow$ indices of the peaks in the stress-strain graph
6: **if** constant amplitude loading **and** want to cut cycles **then**
7:     Determine n_{sat} cycles to reach $\sigma_i > \eta \cdot \max \sigma$ and $\sigma_i < \eta \cdot \min \sigma$
8:     **if** $n_{sat} < n_{cyc}$ **then**
9:         $n_{sat} \leftarrow n_{cyc}$
10:     **end if**
11:     Truncate data past n_{sat} cycles
12: **end if**
13: Filter stress data pre-2 % strain with window length $\alpha \cdot w$ and order p
14: Filter stress data post-2 % strain with window length w and order p
15: Normalize the stress-strain data
16: $k_2 \leftarrow$ indices from downsampler with local (ϵ) or global method (ϵ_g, ϵ_0)
17: $k_3 \leftarrow$ indices of $n_{elastic}$ evenly spaced points in the initial elastic region
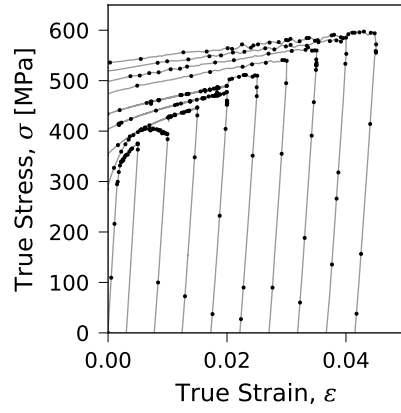18: $k_{sample} \leftarrow k_1 \cup k_2 \cup k_3$
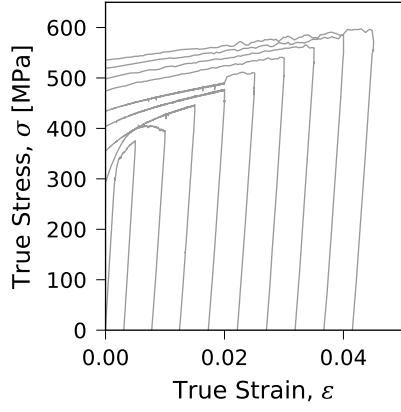
(a) Original
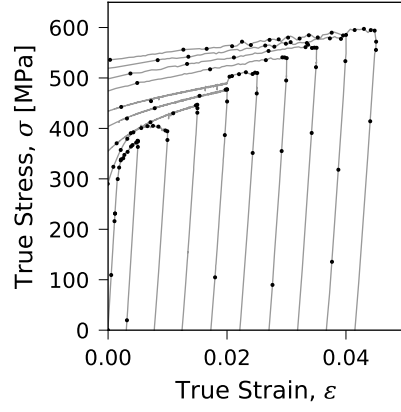
(b) Option 1
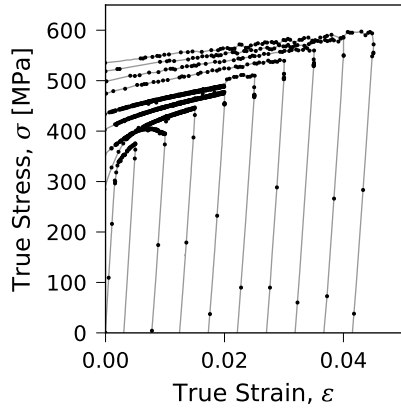
(c) Option 2

(d) Option 3

Figure 2: Zoom on the positive quadrant of LP6 from the IPE360 web dataset. Line is the original data, dots are the sampled points using the different filtering options.
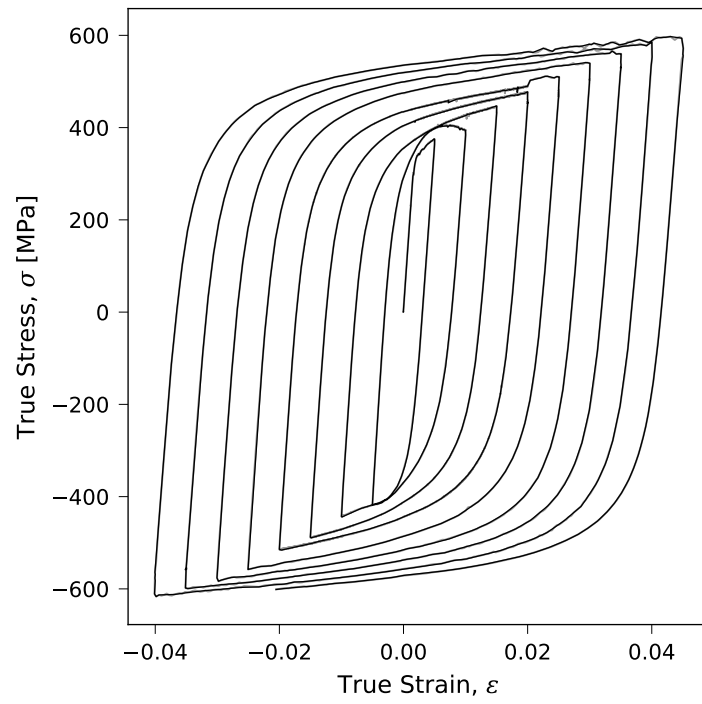
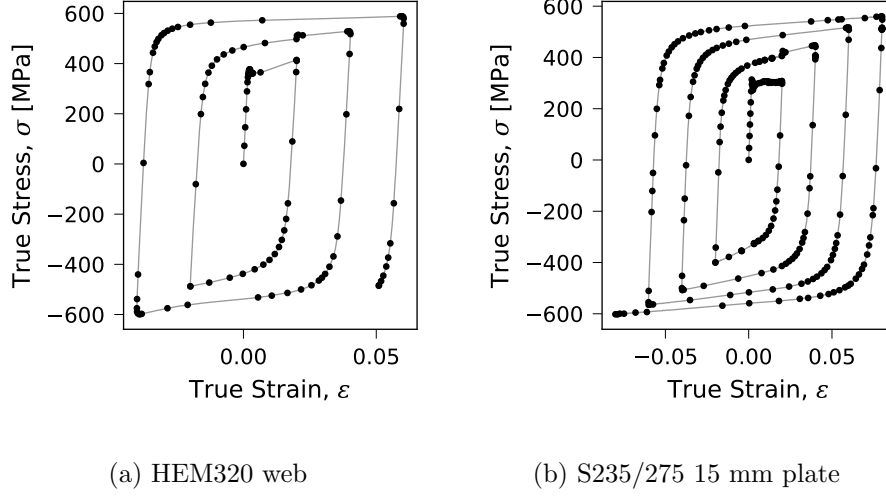Figure 3: LP6 from the IPE360 web dataset downsampled (black line) with filtering from Option 3.

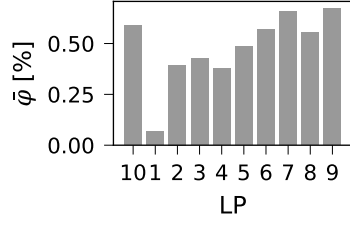(a) HEM320 web   (b) S235/275 15 mm plate

Figure 4: Sampled points in LP8 from datasets. Line is the original data, dots are the sampled points using the proposed method.

stress-strain data. The results shown in this section were computed using the recommended parameters for the global-error downsampling method using the RDP algorithm. Typical downsampled results are compared with original data in Figure 4. Note that cycle cutting is *not* employed for the constant amplitude tests.

The error metrics for the ten load protocols and reduction factors are provided in Figure 5. The error metric, $\tilde{\varphi}$, is computed using k_{sample} from the proposed global downsampling method. The $\tilde{\varphi}$ values are computed using the "true" original data (i.e., not filtered and not normalized), this is why they may be larger than 0.5 %. The average value for the HEM320 web dataset is 0.480 % and for the S235/275 dataset is 0.469 %. The error metric is not computed for the integer reduction method because I did not save the indices. A more detailed comparison of the different methods in terms of resulting stress-strain is provided for one example in Figure 6.

The reduction factors are the number of points in the original divided by the number of samples in the downsampled data. Reduction factors are provided for the proposed (New) and integer reduction (Old) downsampling methods. The total number of data points for all LPs are reduced by 105 times for the HEM320 web and 77 times for the S235/275. The new method reduces the total number of data points by 3.7 times compared to the old method for the HEM320 web and 2.3 times for the S235/275.
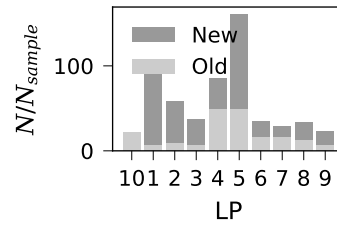
13
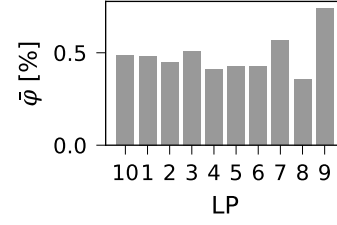
(a) Error, HEM320 web (b) Error, S235/275 15 mm plate
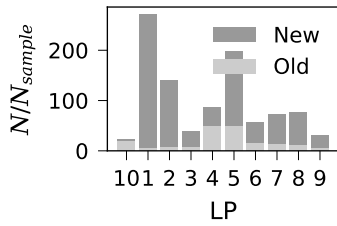
(c) Reduction, HEM320 web (d) Reduction, S235/275 15 mm plate

Figure 5: Error and Reduction factors between original and downsampled data. New is the proposed method, Old is the integer factor downsampling.

4 Discussion

4.1 Evaluation of results

Figure 4 shows that the sampled points tend to be concentrated in the regions of high curvature and the initial elastic region. Additionally, the peaks of the stress-strain graph are all sampled. Using the global method, the procedure is straightforward and interpretable. Figures 5a and 5b show that the error with the global method is around 0.5 %. At this level of error, the data is reduced by between 50–90 times compared to the original.

First, the oscillations in the stress signal are sampled when no filtering is applied, resulting in 1.65 times as many sample points compared to $w = 5, \alpha = 11$ option. This result indicates that filtering the stress signal is important to reducing the number of sample points, although some sampling of the oscillations still occurs as shown in Figures 2 and 3. Second, there is a 5 % increase in the number of points when the window length is always equal to 5 throughout. Moreover, for LPs 4 and 5, $w = 5, \alpha = 11$ is too large of a window length and leads to additional sample points. Therefore, I recommend to use the simpler option which is to use $w = 5, \alpha = 1$. Recall that the filtering only affects the shape of the signal used for the selection of sample points via the RDP algorithm.
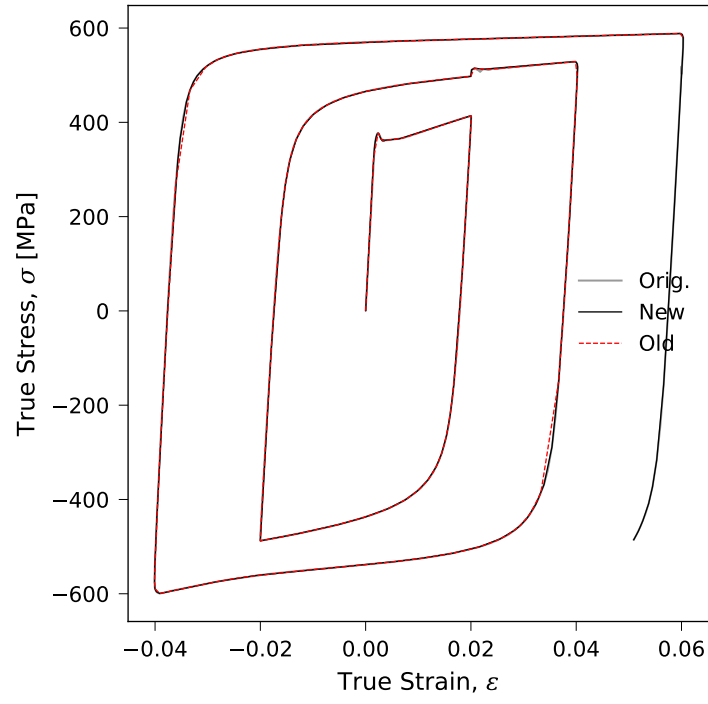
14

Figure 6: Comparison of different methods for downsampling LP8 in the HEM320 web dataset. New is the proposed method, Old is the integer factor downsampling.

The main strengths of the global error method are its reliability and its relation to the inverse problem in de Castro e Sousa et al. [2020]. For example, the old method does not sample the curved parts of the graph in Figure 6 as well as the new method. This is despite having multiple times more points for the same graph (see Figure 5c). The global method allows an analyst to decide on an acceptable downsampling tolerance with relation to the inverse problem of material calibration. For instance, $\tilde{\varphi} = 0.5$ % is approximately one order of magnitude lower than the values of $\bar{\varphi}$ expected in VC/UVC calibration [de Castro e Sousa et al., 2020]. Therefore, the downsampling process is expected to have a fairly limited impact on the minimization of φ defined in de Castro e Sousa et al. [2020]. A stricter tolerance can be specified depending on the application and requirements of the analyst.

Another strength of the RDP method is that it is strain rate independent because it is based on the curvature of the data and will sample points appropriately. However, the strain rate leads to the main difficultly in the proposed downsampling method, which is the choice of filtering parameters. The recommended w and α parameters appear to work well as long as the standard load protocols are followed, however, different values are certainly required for different strain and sampling rates. My implementation of Algorithm 3 in `rlmtp` (RESSLab Material Test Processing) uses the default values set forth above, and also allows users to modify all of these parameters. The input parameters for tests that do not follow the standard load protocols can be saved in a file and read by the algorithm when the downsampling occurs.

4.2   Choice of algorithm

The choice between the maximum deviation downsampler and RDP algorithms is discussed in this section. The max deviation downsampler is not recommended for two reasons: (i) it has higher errors because it does not guarantee the maximum perpendicular distance is always less than the tolerance, and (ii) my implementation in Python is slow compared to the RDP algorithm in `polyprox`. The first issue being more critical, of course. The first issue is rooted in the nature of our data having many inflection points. With multiple inflection points, although $d_{ij}^{(k)} < \epsilon$, there may be some $d_{mj}^{(l)} > \epsilon$ with $m > i$ and $m < l < j$. Therefore, "stepping forward", as in the maximum deviation downsampler, is not as good of a strategy as used in the RDP algorithm. The max deviation downsampler tends to produce higher errors in terms of $\tilde{\varphi}$ for the same ϵ (although it results in less points). This makes the second issue not worth investigating, although it could likely be fixed by implementing the max deviation downsampler in C as is done with the RDP algorithm in `polyprox`.

With regards to the RDP algorithm, the cyclic nature of our stress-strain

data implies that selecting the start and end points, then sampling the maximum perpendicular distance between these, is not a great heuristic. See Figure 1. This is because the maximum perpendicular distance from the line linking the start and end points is not necessarily at a region of high curvature. However, the RDP algorithm is not optimal in producing the minimum number of points, but it will be on the "safe side" of adding a few additional points to get the job done. The results could be more efficient (i.e., satisfy the local ϵ criteria with less points) by using an optimal solution from Chan and Chin [1996]. The RDP implementation be made more time efficient (particularly for the global method) considering Hershberger and Snoeyink [1992]. However, Algorithm 2 can already be run on the most extensive load protocols in a few seconds using `polyprox`.

4.3 Different distance measures

A challenge in the max deviation downsampler and RDP methods is the difference in units between stress and strain. I currently overcome this challenge by re-scaling the stress-strain data, then using the 2-norm when computing the perpendicular distance. Options that could be used rather than rescaling the data are:

1. use the angle between the line and each point,

2. use the norm induced by an alternative inner product: $\|d_k\|_S = \sqrt{d_k \cdot S \cdot d_k}$, and

3. use an energy criterion.

The first option is similar to the maximum perpendicular distance, but more sensitive to noise. This is because of the small angle rule, $\sin \theta \approx \theta$, and the perpendicular distance is $\|v\|_2 \sin \theta$, where $v = x_k - x_i$. If $\|v\|_2$ is small, the angle can have a large variation (i.e., in the case of high frequency noise), however, the perpendicular distance remains small.

In the second option, Algorithm 1 remains essential the same except for how $d_{ij}^{(k)}$ is computed. To use a matrix-vector notation:

$$d_{ij}^{(k)} = \begin{bmatrix} (d_{ij}^{(k)})_\varepsilon \\ (d_{ij}^{(k)})_\sigma \end{bmatrix}, \quad S = \begin{bmatrix} \frac{1}{(\max \varepsilon - \min \varepsilon)^2} & 0 \\ 0 & \frac{1}{(\max \sigma - \min \sigma)^2} \end{bmatrix},$$

where the values of S would in this case give a similar effect to the scaling that I already use. Other options for S could be selected. Essentially, we compute a scaled distance rather than scale the data itself. This method would still require a selection of tolerance ϵ, but it may be different depending on the matrix S.

In the third option, the perpendicular distance criterion is replaced by the energy bounded by the polygon. In this case, instead of d_k's we compute

$$e = \int_{\varepsilon_i}^{\varepsilon_j} (\sigma - l_\sigma)\mathrm{d}\varepsilon,$$

where l_σ represents the stress signal of the line between x_i and x_j. If the stress-strain graph is nearly straight, then $(\sigma - l_\sigma) \approx 0$. The point $j - 1$ would be sampled if $e > \epsilon$.

The benefits of the second method are that it does not require any scaling of the data and the error is closely related to the calibration problem. However, the link between the energy, e, and the curvature in the stress-strain graph is less clear, making it more difficult to reason about an appropriate tolerance. Everything considered, a tolerance still needs to be selected regardless of the method. I went with the data scaling option because, in my opinion, the link between the algorithm and the geometric interpretation of curvature is most clear. Furthermore, this makes the overall procedure more easily adapted to existing algorithms (e.g., `polyprox`).

4.4 More than two dimensions

This document assumes two-dimensional data parametrized as a function of time, i.e., $x(t) \in \mathcal{D} \times \mathcal{T}$, where $\mathcal{D} \subset \mathbb{R}^n$ and n is the dimension of the data ($n = 2$ in our case—uniaxial stress/strain), and $t \in \mathcal{T}$ is the time. Algorithm 3 can easily be extended to $n > 2$ by preserving curvature in each considered data pair. The same procedure can then be used with an appropriate tolerance, ϵ, and an appropriate scaling procedure by sampling a point anytime the tolerance is exceeded by one of the pairs. Considering higher dimensions may be useful, for instance:

- when downsampling temperature and stress-strain data together (can preserve curvature in both temperature-strain and stress-stress), and

- when downsampling data considering multiple components of the stress-strain tensor (e.g., multiaxial material tests).

5 Conclusion

Integer reduction downsampling is not efficient for the stress-strain data that we collect for material tests. A better downsampling procedure was proposed in Algorithm 3 consisting of two main steps and two auxiliary steps. The main steps are (i) to determine the peaks of the stress-strain graph and (ii) to then use a downsampling algorithm that preserves the curvature in the stress-strain graph. The maximum deviation downsampler and Ramer–Douglas–Peucker (RDP) algorithms were considered in (ii). A

global-error preserving method was proposed because these algorithms operate on a local criterion. The auxillary steps are to add points to the initial elastic region and to remove cycles after saturation of cyclic hardening in constant amplitude tests. Heuristic parameters applicable to tests carried out according to the standard load protocols were provided. Results from two datasets were provided to indicate the performance of the proposed method. The main conclusions are:

- Re-scaling the stress and strain signals prior to downsampling is suggested to mimic the geometric notion of curvature in stress-strain graphs. A scaled distance could be computed instead, however, this seems to be more complicated than just scaling the data. An acceptable local error, ϵ, has to be selected regardless of this choice.

- The RDP algorithm is better than the maximum deviation downsampler because it is more strict in enforcing the local tolerance and it is faster.

- The RDP algorithm leads to satisfactory results as a curvature-preserving downsampler. It is not the optimal solution to the min-# problem and results may be slightly improved with an optimal algorithm.

- Filtering the stress signal is necessary to get the best performance out of downsamplers, however, this adds a complication regarding the selection of filter parameters.

- The global error method (Algorithm 2) is recommended because of its link with material calibration problems.

- Using the proposed method, the number of data points is reduced by one to two orders of magnitude based on the examined datasets. This reduction was around 2–3 times better than the integer reduction method. The proposed method appears to provide to a better representation of stress-strain data with fewer points.

References

Hiroshi Imai and Masao Iri. Polygonal Approximations of a Curve — Formulations and Algorithms. In Godfried T. Toussaint, editor, *Machine Intelligence and Pattern Recognition*, volume 6 of *Computational Morphology*, pages 71–86. North-Holland, January 1988. doi: 10.1016/B978-0-444-70467-2.50011-4.

Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244–256, November 1972. ISSN 0146-664X. doi: 10.1016/S0146-664X(72)80017-0.

David H Douglas and Thomas K Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, December 1973. ISSN 0317-7173. doi: 10.3138/FM57-6770-U75U-7727.

Paul S Heckbert and Michael Garland. Survey of Polygonal Surface Simplification Algorithms. Technical report, Carnegie Mellon University, School of Computer Science, Pittsburgh, Pennsylvania, USA, 1997.

Juan-Carlos Perez and Enrique Vidal. Optimum polygonal approximation of digitized curves. *Pattern Recognition Letters*, 15(8):743–750, August 1994. ISSN 0167-8655. doi: 10.1016/0167-8655(94)90002-7.

W.S. Chan and F. Chin. Approximation of Polygonal Curves with Minimum Number of Line Segments or Minimum Error. *International Journal of Computational Geometry & Applications*, 06(01):59–77, March 1996. ISSN 0218-1959, 1793-6357. doi: 10.1142/S0218195996000058.

Albano de Castro e Sousa, Yusuke Suzuki, and Dimitrios Lignos. Consistency in Solving the Inverse Problem of the Voce-Chaboche Constitutive Model for Plastic Straining. *Journal of Engineering Mechanics*, 146(9):04020097, 2020. ISSN 0733-9399, 1943-7889. doi: 10.1061/(ASCE)EM.1943-7889.0001839.

John Hershberger and Jack Snoeyink. Speeding Up the Douglas-Peucker Line-Simpli cation Algorithm. Technical Report TR-92-07, University of British Columbia, Department of Computer Science, Vancouver, BC, Canada, 1992.

Ronald W. Schafer. What Is a Savitzky-Golay Filter? [Lecture Notes]. *IEEE Signal Processing Magazine*, 28(4):111–117, July 2011. ISSN 1558-0792. doi: 10.1109/MSP.2011.941097.

A   Filtering the stress signal

The oscillations in the stress signal arises due to the closed-loop PID control when the tangent modulus approaches zero. The oscillating signal is a combination of the intended "static" (low frequency) response plus the unintended "dynamic" (high frequency) oscillations. We do not want to calibrate to the oscillated signal because the load cell force measurement (and consequently the material-level stress) in these regions also contains the inertia of the system. Therefore, elastic unloading does not take place, and the VC/UVC models cannot replicate this behavior. The high frequency content can be removed through filtering. It is desirable to remove the

oscillations because the local downsampler tolerance, ϵ, will often be within the magnitude of oscillations observed in our tests. This results in a sampling of the oscillations and leads to excess points that cannot be simulated by the material model.

Two types of frequency impulse response (FIR) filters have been investigated for filtering the stress signal. The first type are derived using a windowing method, specifically the Kaiser method, and the second are derived using a moving average. For the Kaiser method, a shape factor, β, and order, N, are determined through empirical equations[5] for a given attenuation. The attenuation I used is

$$\alpha = \max\left[-20\log_{10}[\delta], \alpha_{sb}\right], \tag{3}$$

where δ is the maximum allowable gain (ripple) in the pass band, and α_{sb} is the attenuation in the stop band. The maximum in Equation 3 is because only one parameter is used in the Kaiser method so either the specified ripple in the pass band or the stop band attenuation will control.

The definition of the Kaiser method is completed by selecting a transition width, Δw, and cutoff frequency, f_c. The pass band (f_p) and stop band edges are $f_c \pm \Delta w$. Therefore, in total $\delta, \alpha_{sb}, \Delta w, f_c$ need to be specified. The pass band edge frequency should be related to the underlying signal, fast Fourier transforms (FFTs) are applied to all the LPs in the HEM320 web, IPE360 web, and S235/275 datasets. The resulting normalized power is shown in Figure 7. This figure implies that the power decays by four orders of magnitude by 1 Hz. A back of the envelope calculation for LP9 steps 14–16 gives a 1.5 % strain peak-to-peak cycle is covered with a strain rate of 0.008 strain/s. This gives a frequency of 1.875 Hz. Therefore, a pass band edge frequency of 2 Hz seems appropriate.

The final consideration for FIR filters is the need to account for the group delay. FIR filters typically have a constant group delay equal to $N/2$, where N is the order of the filter. I accounted for this in the FIR filter designed with the Kaiser method by ensuring that the order of the filter is an even integer, then shifting the filtered signal by $N/2$ samples. This process deletes the last $N/2$ samples.

The moving average filter, a special case of the Savitzky-Golay (S-G) filter, can be viewed as an FIR filter as well [Schafer, 2011]. The moving average is equivalent to the S-G filter with polynomial order $p = 0$ or $p = 1$ (because the odd terms of the S-G filter are equal to zero, i.e., $p = 3$ is the same as $p = 2$). The general S-G filter fits a polynomial of order p centered at each sample using data from $\pm(w-1)/2$ samples (w must be an odd integer). In terms of an FIR filter, the window length, w, and polynomial order define the cutoff frequency and attenuation. The group delay for S-G

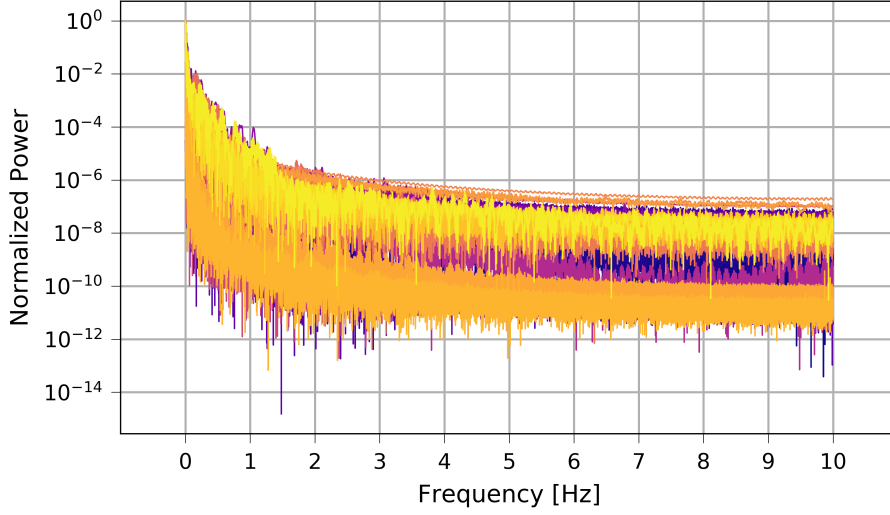[5]https://ch.mathworks.com/help/signal/ref/kaiserord.html

Figure 7: Normalized power computed using FFTs for all load protocols in the HEM320 web, IPE360 web, and S235/275 datasets.

filters is $(w - 1)/2$, different options can be used to offset the group delay[6], the default is just to ignore the first/last $(w - 1)/2$ values.

The frequency domain response of the selected FIR filters can be seen in Figure 8. The frequency is plotted between 0 and 10 Hz because the Nyquist frequency is 10 Hz for a sample rate of 20 Hz ($f_N = 0.5 f_s$). For the filter designed using the Kaiser method I selected: $f_p = 2$ Hz, $\Delta w = 0.15(f_N - f_p)$ (based on default MATLAB lowpass method[7]), $\delta = 0.01$, and $\alpha_{sb} = 10$ dB. The δ controls the α value, as the drop is more than 10 dB in the stop band.

From Figure 8, the specified S-G filters lead to an attenuation of around 10 dB in the stop band and have a similar cutoff frequency to the Kaiser method. In the pass band (Figure 8b), the S-G filters begin to attenuate at a lower frequency and have a more gradual response. The cutoff frequency in S-G filters decreases as w increases.

The Kaiser filter and the S-G filter with $w = 5, p = 0$ are used to filter LP6 from the IPE360 web dataset in Figure 9. Both options are effective are effective at removing the oscillations in the stress signal prior to around 2 % strain amplitude. However, both methods only moderately smooth the lower frequency oscillations in the cycle to +4 % strain. The "Gibbs phenomenon" in the Kaiser filtered signal is notable in the peaks at negative excursions at -2.5–3.5 %. I speculate that this is because the very quick change in the

---

[6]https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.savgol_filter.html#scipy.signal.savgol_filter

[7]https://ch.mathworks.com/help/signal/ref/lowpass.html
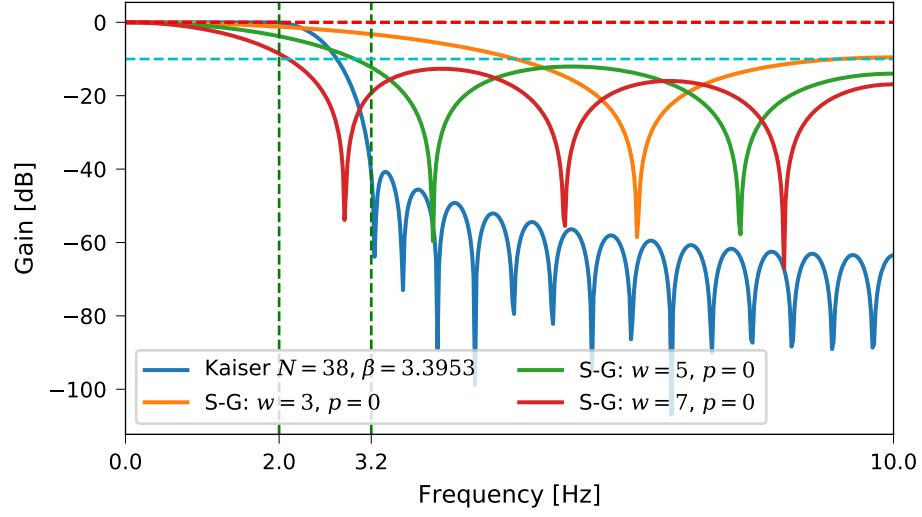
(a) Entire response
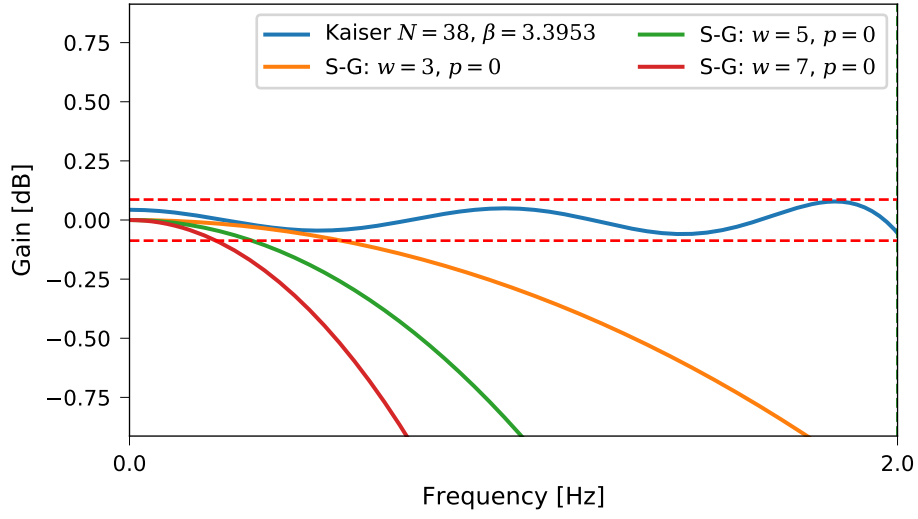
(b) Zoom on the pass band

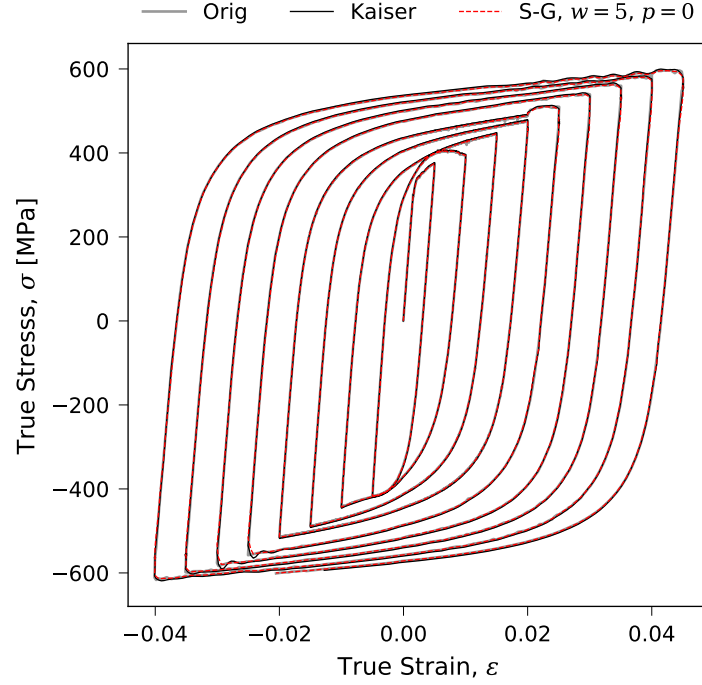Figure 8: Frequency response of FIR filters.

Figure 9: Filtering options applied to LP6 of the IPE360 web dataset.

stress signal and the great attenuation of the high frequency components, as shown in Figure 8. This does not happen with the S-G filter because there is only modest attenuation of the high frequency signals. The missing signals at the end of the Kaiser filtered signal are also apparent in Figure 9.

Overall, the S-G filter performs better than the Kaiser method filter in this case. The Remez exchange algorithm (or Parks-McClellan)[8], rather than the Kaiser method, could be used to design a filter with similar characteristics to the S-G filter [Schafer, 2011]. However, one may as well use the underlying target in this case. Therefore, the S-G filter with $w = 5, p = 0$ is used for filtering the stress signal.

---

[8]`https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.remez.html`