Technical Report: Measuring fidelity of DTs Elevator

Paula Muñoz, Javier Troya, Antonio Vallecillo
ITIS Software
University of Málaga (Spain)

Manuel Wimmer
CDL-Mint
Johannes Kepler University (Austria)

Contents

I Elevator in the University of Mondragon

A. System description

At Mondragon University, they have an elevator, as shown in Figure 1, which transports students between the ground floor and the fourth floor. This elevator provides accessibility to classrooms on the upper floors for students who cannot use the stairs and allows for the transportation of heavy loads. Recognizing the frequent usage of the elevator and its critical role, the University has decided to optimize its operation and maintenance to save energy and prevent breakdowns.

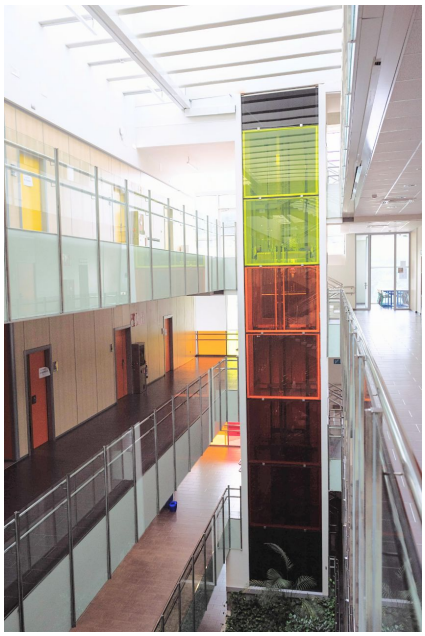


Fig. 1: Elevator in the University of Mondragón (Spain)

To achieve this, they decided to implement a Digital Twin System (DTS) using the commercial simulator *Elevate* [1] as the digital replica. This simulator is capable of emulating elevator movement, including the acceleration reached during floor transitions. Based on this acceleration, the speed and descent times can be obtained to estimate the degradation of the equipment and verify if the configuration is optimal.

The objective of our tool will be to assess whether the simulator's accelerations sequences are sufficiently faithful to take on the role of a DT, or if, on the contrary, it requires better calibration. The simulator's accelerations will be compared with those of the real system, which have been measured by traveling in the elevator using a *WITMOTION WT901BLECL* accelerometer.

B. Scenarios

The selected scenarios to measure the fidelity of the simulator involve the various possible floor transitions for the elevator. Firstly, from the highest floor (the fourth) to the ground floor and back up (Figure 2). Secondly, stopping at even-numbered floors, starting from the fourth floor, descending to the second floor, reaching the ground floor, and going back up to the highest floor, with another stop at the second floor (Figure 3). Finally, from the fourth floor, we descend to the ground floor, stopping at every floor along the way (Figure 4).

In each of the figures, we can observe pairs of symmetric curves with respect to the x-axis, indicating a floor transition. Looking at the simplest scenario (Figure 2), we see four curves divided into two groups. The first group, composed of the first two curves, represents the acceleration during the descent of the elevator from the fourth floor to the ground floor. The first curve shows the negative acceleration during the descent, while the second curve represents the positive acceleration during the elevator's braking upon reaching the floor. Similarly, the next two curves represent the acceleration during the ascent. The figures display the acceleration data obtained from the real system (top figure) and from the Elevate simulator (bottom figure).

These acceleration and deceleration curves for floor transitions are remarkably similar between the simulation and the real system, disregarding the accelerometer noise near zero when the elevator is stationary. However, there is one aspect of the real system that is absent in the simulation, which is that every time the elevator brakes, in either direction, an additional and much smaller acceleration is added to smooth out the elevator's stop and improve the user experience. This braking pattern is characteristic to this specific elevator model and can be seen as a small curve following the acceleration change in the same direction. It indicates a braking that occurs upon reaching the destination floor.

The simulator is deterministic, meaning that all executions of the same scenario will result in the same trace. However, due to the nature of the physical system, the operational traces exhibit some variability. Therefore, we have conducted multiple measurements for each of the scenarios:

- **Scenario (4-0-4).** 10 measurements.
- **Scenario (4-2-0-2-4).** 3 measurements.
- **Scenario (4-3-2-1-0-1-2-3-4).** 3 measurements.

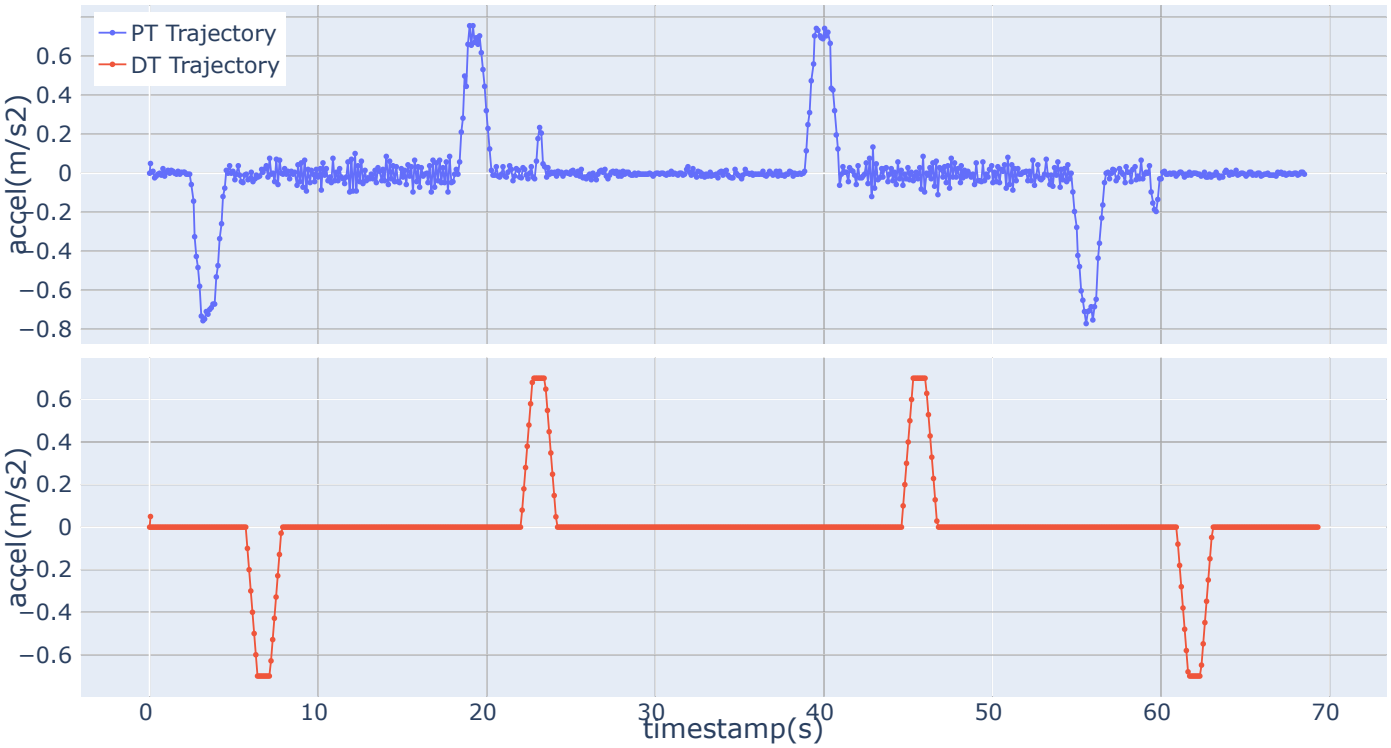


Fig. 2: Traces for scenario (4-0-4) (Real system above, simulator below)

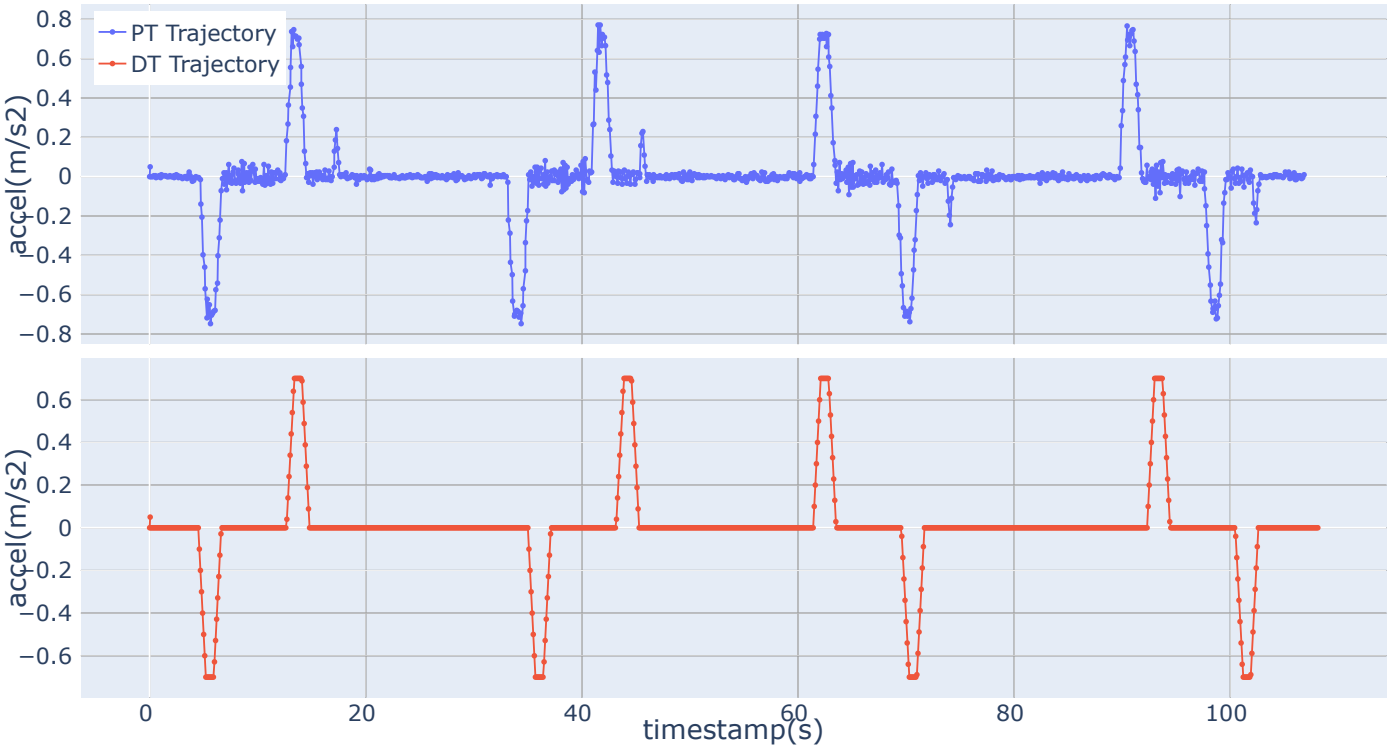


Fig. 3: Traces for scenario (4-2-0-2-4) (Real system above, simulator below)

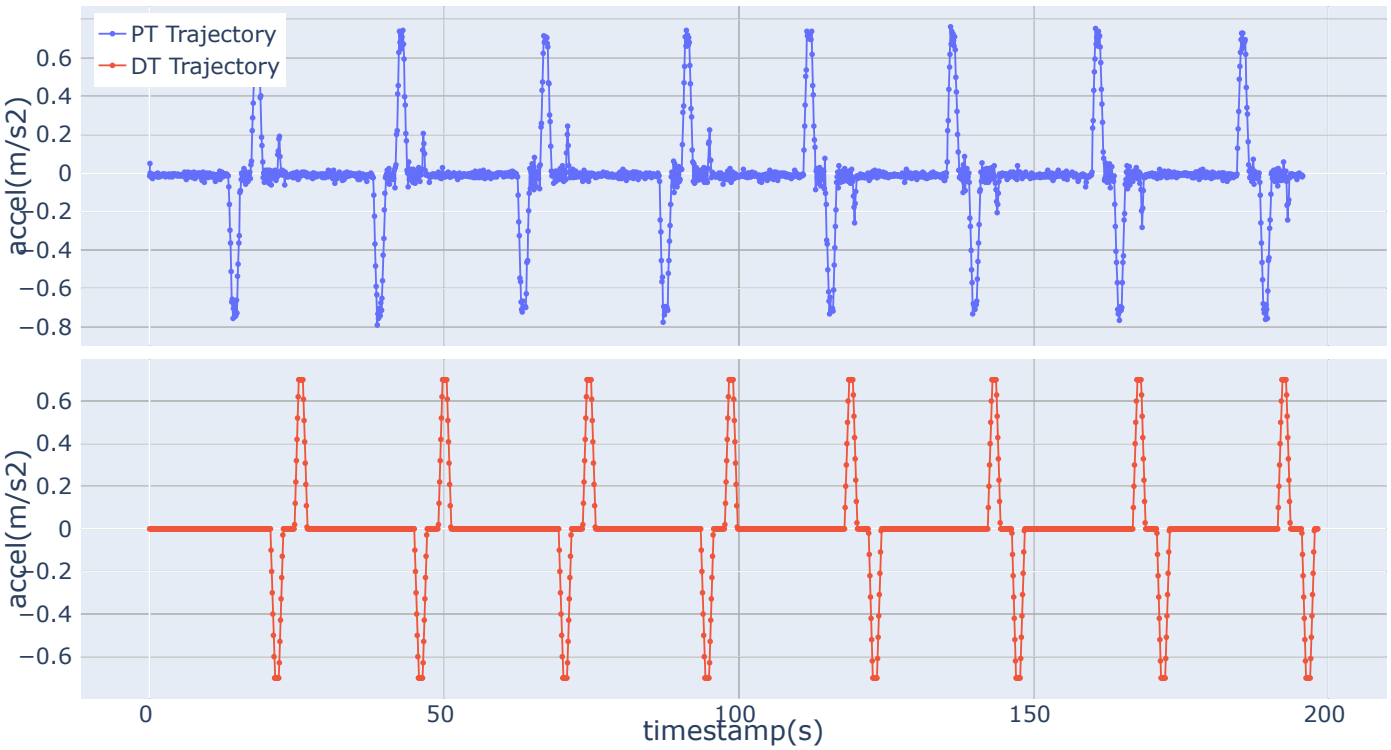


Fig. 4: Traces for scenario (4-3-2-1-0-1-2-3-4) (Real system above, simulator below)

C. Gap Tunning

Verifiability: The analysis performed in this section is available at https://github.com/atenearesearchgroup/fidelity-measure-for-dts/blob/main/src/evaluation/incubator/incubator_gap_tuning.ipynb

One of the main configuration parameters for alignment algorithms is the scoring system. In order to discern among the many possible alignments between two sequences, it is necessary to specify to the algorithm which decisions to prioritize when aligning the sequences. The classical Needleman-Wunsch algorithm, on which our proposal is based, has two configurable penalties (mismatch and gap) and one reward (match). These values can be assigned based on an input matrix that prioritizes certain characters over others (in the original approach for protein sequence alignment) or through fixed values.

One of the most common configurations when using fixed values is to assign +1 for a match, 0 for a mismatch, and -1 for a gap. This type of configuration prioritizes mismatches over gaps, favoring solutions with fewer gaps. In our case, we adapted this scoring configuration:

- **Match:** Value in the range (0, 1]. The more similar the snapshots are, the closer the value is to 1, based on the comparison function.
- **Mismatch:** Neutral penalty, 0. This occurs when the two snapshots fall outside the range of the Maximum Acceptable Difference (MAD). However, the algorithm considers that these snapshots should have matched for the optimal alignment.
- **Gap:** Negative penalty, aimed at prioritizing mismatches over gaps. This represents a state that is absent in the other trace.

This negative penalty can be configured in two ways:

- **Simple gap:** A fixed penalty that is added to the score each time a gap is included in the alignment.
- **Affine gap:** A configuration with two fixed penalties: one penalty for initiating a gap (P_{op}) and another usually smaller penalty for extending a previously initiated gap (P_{ex}).

The first approach produces alignments in which single-position gaps and matches alternate in the sequences. However, this alignment scheme may be less effective when the objective is to identify periods of anomalous behavior, as it tends to result in alignments with intermittent gaps in the trace. Conversely, in the second approach, we introduce penalties for such alignments and instead prioritize alignments where gaps are grouped together. Longer gaps facilitate the identification of anomalies, resulting in more meaningful alignments.

However, the latter approach demands more processing space and computational capacity. It not only requires one matrix to align the sequences using Dynamic Programming but also necessitates two additional matrices to evaluate whether to insert a gap or not in each of the sequences. Hence, in our algorithm, we incorporated the flexibility to configure alignments using both of these techniques. Depending on the specific scenario and the importance given to resource optimization, the user can select either approach. To assess the optimal configurations for penalties and their impact on alignment, we prepared experimental datasets for which we analyze the fidelity metrics introduced in Section III of the General Concepts Technical Report [2].

The configurations for the experiments conducted with the elevator are as follows:

Parameter	**Range**	**Increments**
Maximum Acceptable Distance (MAD)	[0.10, 0.22]	0.04
Penalty opening a gap (P_{op})	[-3.0, 0.0]	0.50
Penalty extending a gap (P_{ex})	[-2.0, 0.0]	0.10

This resulted in an analysis of 588 alignments applied to Scenario (4-0-4). The chosen range of MAD values depends on the system's domain. In the following two sections, we will examine the effects of these configurations on various metrics using figures such as Figure 5a and 5b. In these figures, all subfigures share the x-axis, where each unit represents an alignment applied to the scenario. Depending on the input values (MAD, Init_gap, Cont_gap), we obtain alignments with different statistics. The shading in this and subsequent figures represents a breakpoint, dividing the values into two groups based on their characteristics.

1) Simple Gap

The statistics for the percentage of matched snapshots, Frèchet distance, and average Euclidean distance (in the relevant areas) between aligned points are depicted in Figures 5a and 5b. These figures specifically focus on the 85 input configurations where the P_{op} value is set to 0, implying that the cost of opening and extending a gap is the same. The breakpoint, which signifies a shift in the values, is observed at the peak of the percentage of aligned snapshots. This peak represents a significant increase of 10% in the number of aligned snapshots compared to the remaining samples.

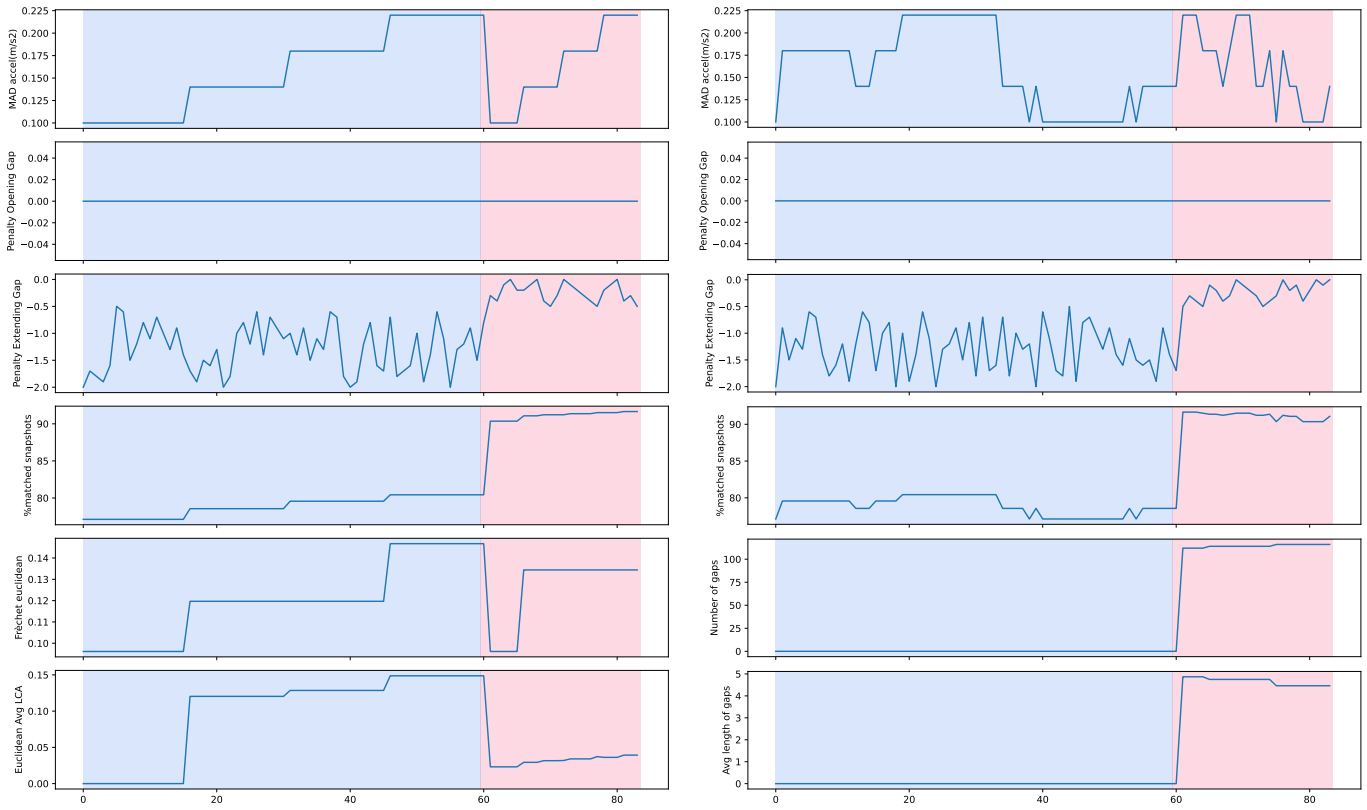
Figure 5a presents the samples sorted along the x-axis based on the increasing percentage of matched snapshots. The shaded red area highlights the alignments that achieve the best snapshot percentage and distances (Frèchet and Euclidean). These shaded alignments include MAD values that are distributed across the entire range, indicating that the increase in the MAD value makes the alignment constraint more flexible, thereby increasing the percentage matched snapshots. On the other hand, when considering P_{ex}, it becomes evident that satisfactory results are only obtained with values greater than -0.5. Values below this threshold imply that the algorithm struggles to provide enough flexibility to incorporate an adequate number of gaps to align snapshots, leading to unsatisfactory outcomes.

We can understand this more easily by looking at Figure 5b. Instead of arranging the samples based on the percentage of matched snapshots, they are now sorted in increasing order of the number of gaps in the alignment. In this case, we can observe a similar breakpoint as in Figure 5a, a 10% increase of matches snapshots. Once a certain number of gaps is reached, the alignments become satisfactory. This happens again for values of P_{ex} greater than -0.5, which means that the penalty for introducing a gap is low enough for the algorithm to prioritize an alignment that includes an adequate number of gaps. These gaps help to characterize the behavior of our system in comparison to the simulation, enabling the inclusion of delays, for example.

To verify the statistical relevance of the input values in relation to the output values, we performed linear regressions that relate the input parameters (MAD, P_{op}, P_{ex}) to the percentage of aligned snapshots. The results of this analysis are available in Table I. In this table, we have the values for the analysis of all samples (Simple-All) and specifically for the red-colored segment with the most optimal values (Simple-Segment). The results for the three input parameters are the following:

- **MAD** has a significant influence on explaining the variability of the data across all samples, as indicated by the statistical relevance with p-values below 0.05 and the coefficients. However, by examining Figure 5a, we can observe that the values are distributed throughout the entire range in an increasing fashion. This implies that increasing the MAD loosens the alignment restrictions, resulting in a higher percentage of matched snapshots.
- $\mathbf{P_{op}}$ has no statistical relevance, since it doesn't play any role in this analysis, we kept its value at zero. The coefficients are really close to zero, and the p-values are above 0.05.
- $\mathbf{P_{ex}}$ coefficients and p-values indicate that it is statistically relevant when considering all samples but irrelevant within the range of optimal values in the segment. This means that modifying the value of P_{ex} within the appropriate range of values [-0.5, 0) has little impact on the percentage of aligned snapshots.

The conclusion is that we can achieve satisfactory results for this example with $\mathbf{P_{ex}}$ **values between [-0.5, 0)**, regardless of the specific variations within that range.



(a) Ordered by % of matched snapshots.

(b) Ordered by the number of gaps.

Fig. 5: Analysis of alignment statistics for simple gap.

TABLE I: Analysis of the influence of gap penalty on the percentage of aligned snapshots.

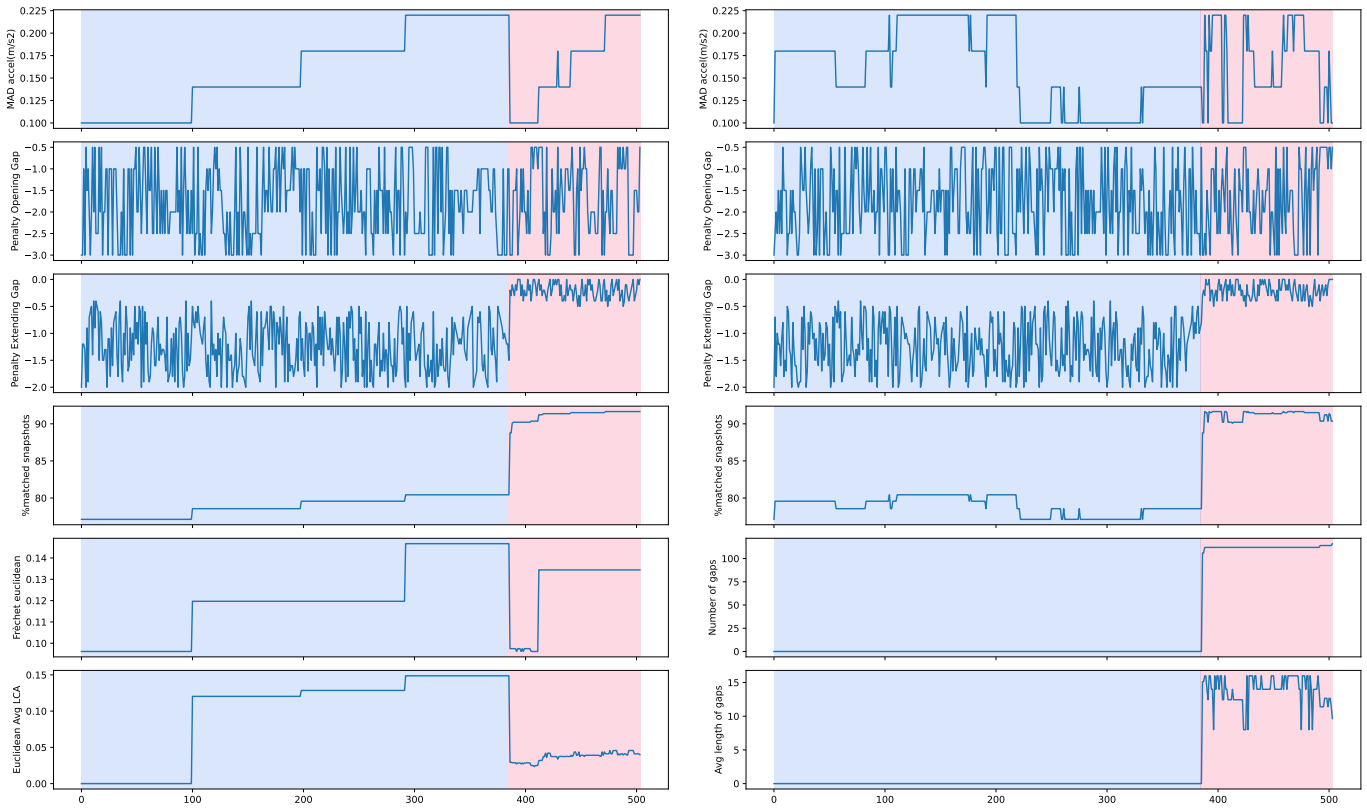
Model	R-squared	F-statistic	Coef. MAD	p-value MAD	Coef. P_{op}	p-value P_{op}	Coef. P_{ex}	p-value P_{ex}
Simple-All	0.562	213.606	29.048 ± 3.532	0.000	0.278 ± 0.185	0.134	6.233 ± 0.261	0.000
Simple-Segment	0.694	86.233	11.468 ± 0.720	0.000	0.050 ± 0.037	0.179	0.091 ± 0.220	0.681
Affine-All	0.610	63.394	27.030 ± 8.643	0.002	-0.000 ± 0.000	0.003	6.905 ± 0.638	0.000
Affine-Segment	0.878	71.774	9.327 ± 0.804	0.000	0.000 ± 0.000	0.000	-0.398 ± 0.212	0.075

2) Affine Gap

The analysis for the Affine Gap approach is similar to that performed for the simple approach. In Figures 6a and 6b, we have the same statistical analysis for the Affine Gap approach. The first plot displays the statistics sorted by % of matched snapshots, while the second plot sorts them by the number of gaps.

If we look at Figure 6a, the results are similar to those of the Simple Gap. MAD values in the segment with the highest percentage of alignments are evenly distributed throughout the range. Same happens for P_{op}, which shows that we can get satisfactory alignments for any value within the range of [-3, -0.5]. As for P_{ex}, the optimal values are obtained, just like in the previous case, within the range [-0.5, 0). The algorithm requires the gap costs not to be too high in order to include them and obtain relevant alignments.

Similarly, in Figure 6b, as we reduce the cost of P_{ex}, we increase the number of gaps that the algorithm adds to the alignment, allowing flexibility in the alignment choices and improving the number of aligned snapshots. We can also observe that as we reduce the cost of the gaps, the average length of the gaps decreases, prioritizing shorter gaps over longer ones.



(a) Ordered by % of matched snapshots.

(b) Ordered by the number of gaps.

Fig. 6: Analysis of alignment statistics for the affine gap.

Regarding the analysis of the statistical significance of the data, the results are in Table I (Affine-All, Affine-Segment), and are similar to the previous ones.

- **MAD** has an influence on explaining data variability across all samples from its statistical significance, indicated by p-values below 0.05 as it happened in the previous method. Also, as in the previous example, when examining Figure 5a, it becomes clear that the values are progressively distributed across the entire range. This suggests that increasing the MAD leads to a relaxation of alignment restrictions, ultimately resulting in a higher proportion of matched snapshots.
- $\mathbf{P_{op}}$ is statistically relevant and has influence for all samples since the p-values below 0.05. However, the coefficient is really low (< 0.001). This means that a one-unit change in the P_{op} produces a change smaller than 0.001 in the % of matched snapshots.
- $\mathbf{P_{ex}}$ is statistically relevant for the complete sample but not for the segment, as happened in the Simple Gap. This means that the variation within the range of [-0.5, 0) does not produce remarkable statistical changes.

Therefore, the appropriate configurations for the algorithm would include a $\mathbf{P_{ex}}$ **value between [-0.5, 0) and an** $\mathbf{P_{op}}$ **value between (-3, 0).** In our approach, we typically use the combination of **-1 as** $\mathbf{P_{op}}$ **and -0.1 as** $\mathbf{P_{ex}}$, which is one of the recommendations from BLAST.

BLAST suggests that the penalty for initiating a gap should be 10 to 15 times higher than the penalty for extending it. The values for the penalties of opening and extending a gap for BLAST are obtained empirically and usually depend on the frequency scoring matrix used for the alignment [3]. However, generally, as a default value, the penalty for opening the score is approximately ten times higher than the cost for continuing a gap.

D. Low Complexity Areas Weight

We have also recognized the significance of identifying and filtering out snapshots that represent states of the system that are not relevant to the behavior of interest. An example of such a state is when the elevator is stationary on a floor. These states can be easily identified as their values are lower than the precision of the accelerator ($0.05\ m/s^2$), i.e., they are indistinguishable from 0. To mitigate their impact and produce consistent alignments, we introduce the concept of the *Low-complexity area weight* (LCAW). The LCAW represents the weight assigned to alignments of values within the low-complexity areas to reduce their influence. Its value depends on two factors: the percentage of relevant snapshots to the behavior of interest (r) and the desired weight of influence (s) to assign to them.

In the case study of the elevator, the analysis of its traces revealed that it was in motion only 10% of the time, hence $r = 0.1$. Furthermore, we decided to assign a weight of $s = 1/20$ to the non-relevant snapshots, which means that their weight is 1/20th of the weight assigned to the relevant snapshots. Consequently, we set the *LCAW* to 0.005.

An example of the influence of adding these weights is shown in Figures 7 and 8. In this scenario, we would anticipate that all four peaks of acceleration should be aligned, given that they represent the same behaviors. However, this is not the case. The last peak aligns with a slight anomaly present in the PT's curve, which is part of the normal behavior of the elevator during operation. Specifically, when the elevator reaches its destination floor, it accelerates once more to smooth the stopping motion. This particular anomaly is not observed in the simulator, which assumes a constant speed until the destination floor is reached.

The reason behind the misalignment of this anomaly is that in most of the snapshots taken from the simulator, the elevator remains stationary, resulting in acceleration values close to zero. Consequently, the snapshot comparison function gives these snapshots high rewards due to their very small differences in acceleration values. Consequently, the algorithm prioritizes aligning these numerous stationary snapshots rather than the (few) ones depicting the curves that characterize the floor change behavior of the elevator.

To address this issue, Figure 8 demonstrates the utilization of *LCAW*. This approach forces the algorithm to concentrate on the regions that portray the relevant behavior, leading to more meaningful alignments.

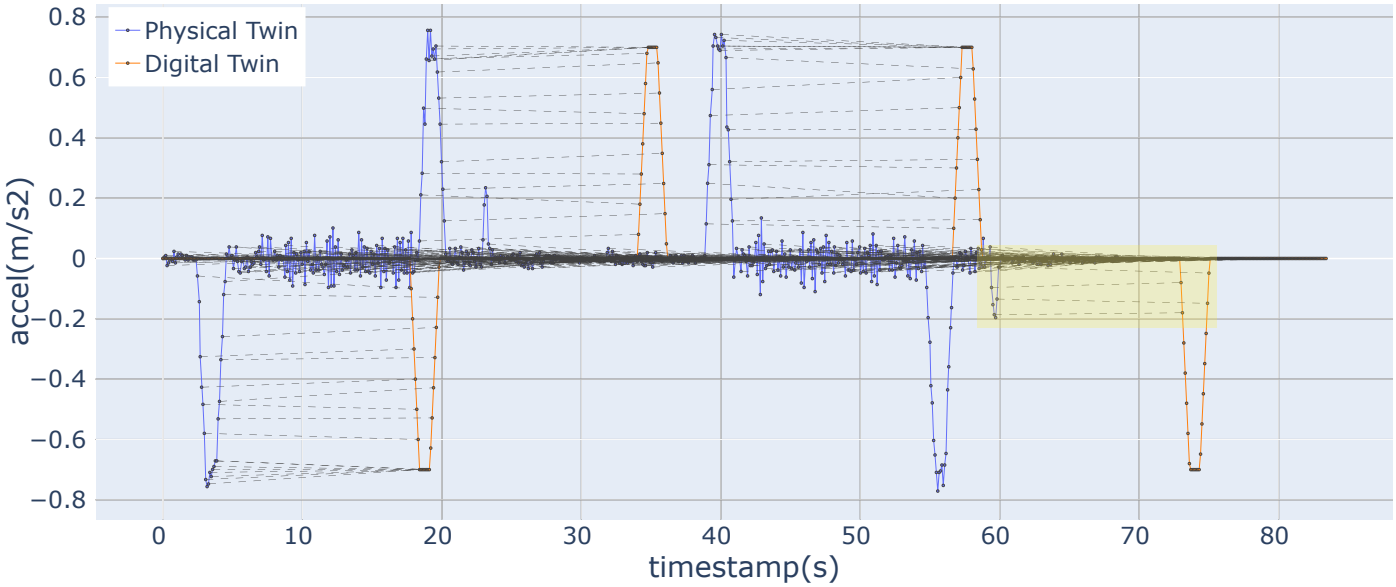


Fig. 7: Alignment of the scenario (4-0-4) without applying LCAW. The aligned anomaly is shaded yellow.

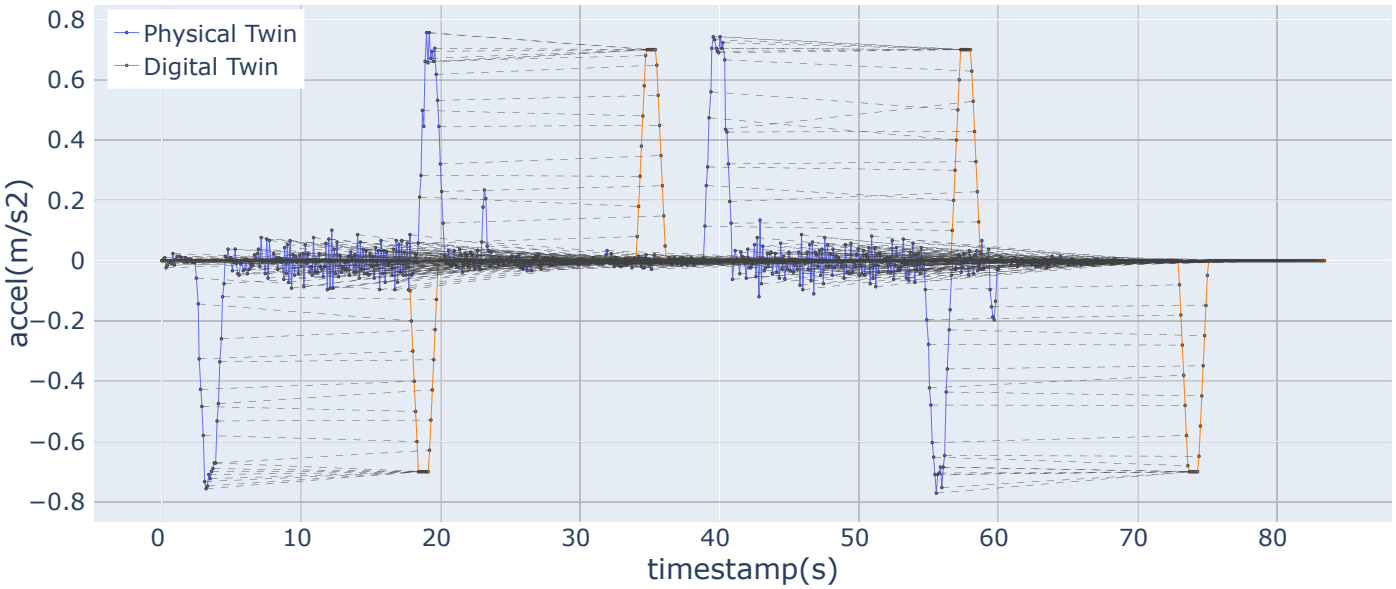


Fig. 8: Alignment of the scenario (4-0-4) applying LCAW.

E. Fidelity assessment

Verifiability: The analysis performed in this section is available at https://github.com/atenearesearchgroup/fidelity-measure-for-dts/blob/main/src/evaluation/lift/lift_variability_analysis.ipynb

Next, we analyze the level of fidelity we achieve when aligning the simulator trace with the trace of the same behavior in the real system. For this analysis, we use some of the fidelity metrics we have defined in Section III of the General Concepts Technical Report [2]: the percentage of aligned snapshots, the Frèchet distance, and the average Euclidean distance (in the relevant area) between aligned snapshots.

To interpret these metrics, let us consider the values for the alignment of two identical traces. In such a scenario, regardless of the value of MAD:

- the **percentage of matched snapshots (%MS)** would be 100%
- the **Frèchet distance** would be zero
- the **average Euclidean distance** would also be zero

These would be the results we would obtain for a model that had the maximum level of fidelity and was capable of accurately emulating the system. Anything that deviates from this model indicates a lower level of fidelity. We can compare different models and assess their fidelity level based on the metrics, using this perfect alignment as a reference. In our work, we proposed a set of fidelity indicators to decide the degree of fidelity of a DT with respect to a PT depending on the values of the three metrics.

- **Alignment with *%MS* above 95%** ($\pm 2\%$)[1] is considered good enough, and the degree of fidelity depends on the distance between the traces.
- If ***%MS* is between 90% and 95%** ($\pm 2\%$)**,** alignment is low, but the distance metrics can be considered. The acceptable distance between the traces is application-dependent, and whether it is the Fréchet or the Euclidean distance that really matters.
- **If *%MS* below 90%** ($\pm 2\%$): traces could not be properly aligned, and therefore no faithful behavior can be expected.

The alignment algorithm applies the following configuration for all scenarios:

Parameter	**Range**	**Increments**
Maximum Acceptable Distance (MAD)	[0.02, 0.30]	0.02
Penalty opening a gap (P_{op})	-1	-
Penalty extending a gap (P_{ex})	-0.1	-
Low Complexity Areas Weight (LCAW)	0.005	-

The specific and detailed guidelines on how to set the configuration values for LCAW (see I-D) and Affine Gap (see I-C) are available in the previous section. The MAD value was determined empirically by identifying the point where the fidelity metrics reached a plateau. To establish a specific value in a practical example, we also need to consider the maximum distance we are willing to allow for aligning two snapshots. In the following sections, we will go deeper into this concept and provide a more detailed analysis based on the available data.

[1]Note that we are assuming a maximum permissible error (MPE) of 2% [4] for the assessment of *%MS*, since most times thresholds are not completely accurate.

1) Scenario (4-0-4)

To analyze the fidelity of the simulation compared to the physical system for the scenario (4-0-4), we aligned the available traces within the MAD range of [0.02, 0.30]. Since the simulation is deterministic, we only have one trace. However, to capture the variability in the elevator's movement, we took ten samples of its operation in this scenario. Consequently, we obtained ten alignments with fifteen variations for all the different MAD values within the specified range.

Figure 9 and Table II present the aggregated results of the 150 alignments taking into account only the relevant areas. The statistics of the Low Complexity Areas are not taken into account. On the x-axis, we have the fifteen MAD values. The figure includes ten gray lines for each metric, each representing the results for a sample. The thick-colored lines represent the mean for all the samples, and the shaded areas indicate the standard deviation.

The results of the three fidelity metrics for this initial scenario indicate that the values of these metrics increase as MAD does. Eventually, they reach a plateau between 0.10 and 0.14 m/s^2, with approximately 90% of the snapshots aligned. The Frèchet distance for these values is approximately 0.12, and the average Euclidean distance in the relevant areas is 0.05. Additionally, all the sample metrics follow a similar trend, as indicated by the clustering of the gray lines around the mean with a consistent trajectory. This means that even with variability in the traces, the fidelity level is consistent through all the samples.

The minimum values for the three metrics can be observed at the minimum MAD value (0.02 m/s^2). At this point, there is the smallest number of snapshots aligned (55%), and the distances between them are very close to zero. This occurs because it represents the strictest alignment criteria, where a distance of 0.02 m/s^2 between snapshots is required to align them. This value is very small considering our case study and is even smaller than the sensitivity of the accelerometer, which is 0.05 m/s^2. This limitation restricts the number of snapshot pairs that the algorithm can align, so we could consider this alignment unsatisfactory.

The alignment with the minimum MAD (0.02 m/s^2), for one of the ten samples is shown in Figure 10. In this figure, we can observe the aligned snapshots connected by dashed lines. These lines are nearly horizontal because we are aligning snapshots with minimal differences. However, we can see that some snapshots defining the acceleration curve do not align even though they may represent similar behavior. This occurs because the alignment condition is too strict, and since the data collection is periodic and subject to some variability in the sampling rate, the discrete information we have is not identical. As a result, such a stringent condition informs us about which points are at this minimum distance, but it does not fully help us identify equivalent behavior among traces.

To achieve a more satisfactory alignment that includes a greater number of points in the relevant areas, we can consider using a higher MAD value. For example, at 0.1 m/s^2, we can see that the Frèchet distance reaches 0.1 m/s^2, the Euclidean mean is 0.2 m/s^2, and the percentage of aligned points has already plateaued at around 91%. This alignment is also shown in Figure 10, where we can observe how the algorithm is capable of determining the equivalence in the elevator's acceleration pattern. The lines of equivalence are slightly less horizontal because the allowed distance difference is slightly larger.

Assuming a MAD value of approximately 0.14 m/s^2, the point at which the values reach a plateau in Figure 9, the %MS is approximately 94% with a standard deviation below 2%. Under these conditions, the maximum distance between snapshots is 0.1 m/s^2, and the average distance is 0.04 m/s^2. The remaining 10% of snapshots are divided into 4% of mismatches belonging to the additional acceleration pattern and 6% of gaps due to the delay of the digital twin, which starts its movement after the real system.

If we continue increasing the MAD values, we can observe that the distance values continue to grow slowly because we allow the inclusion of snapshot pairs that are further apart. This alignment is in Figure 10 for a MAD of 0.2 m/s^2. We can notice that some of the isolated points that were visible in the acceleration curves of Figure 10 are now aligned. However, some of the points in the braking pattern are now aligned with points of the stationary phase, which do not represent the same behavior. This means that the MAD value is unsuitable for this example, and the alignment results are unfit to evaluate the fidelity.

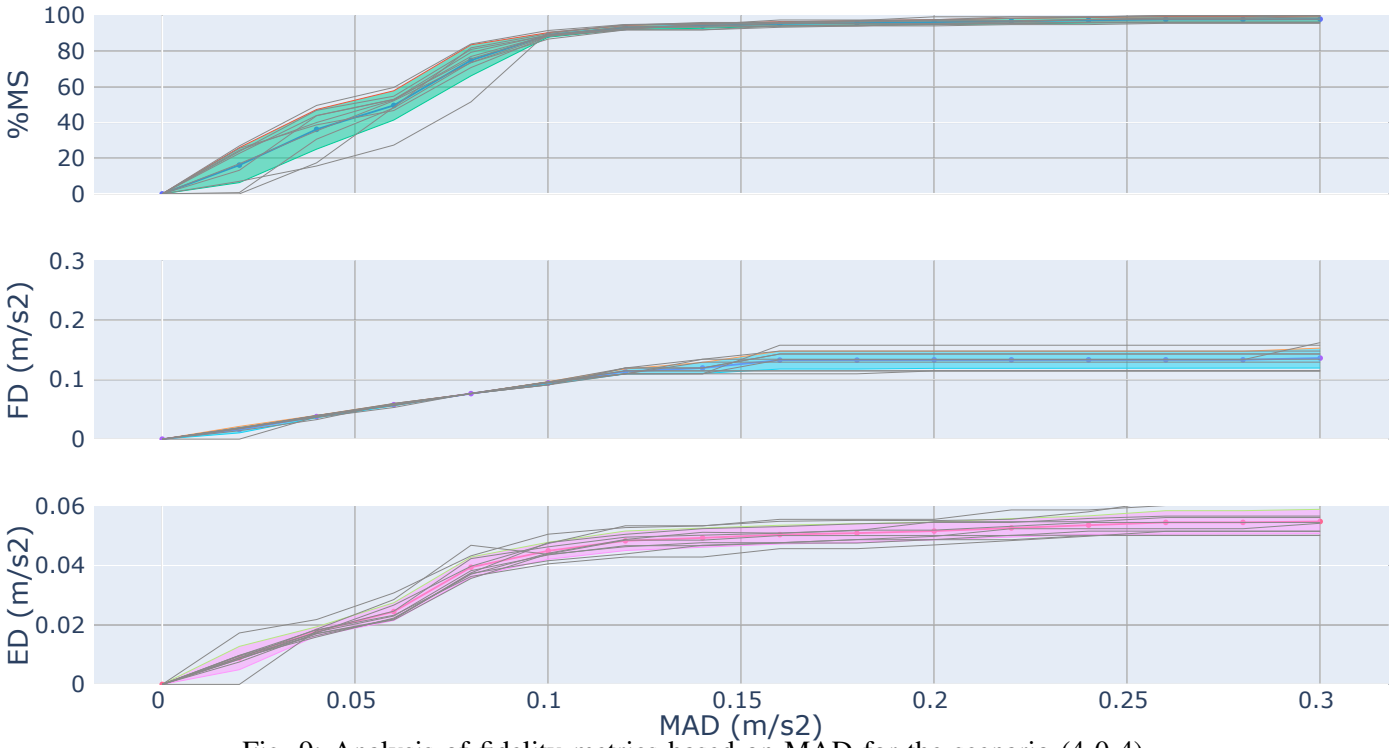


Fig. 9: Analysis of fidelity metrics based on MAD for the scenario (4-0-4).

TABLE II: Results of the fidelity metrics for the scenario (4-0-4).

MAD accel(m/s2)	% matched	Frèchet	Avg. Euclidean
0.02	16.1316 ± 9.8250	0.0160 ± 0.0055	0.0089 ± 0.0039
0.04	36.1800 ± 11.1304	0.0379 ± 0.0020	0.0178 ± 0.0015
0.06	49.5932 ± 8.2041	0.0579 ± 0.0019	0.0245 ± 0.0030
0.08	74.8621 ± 8.7093	0.0766 ± 0.0002	0.0394 ± 0.0034
0.10	89.1154 ± 1.2697	0.0945 ± 0.0022	0.0449 ± 0.0029
0.12	93.2984 ± 1.1581	0.1138 ± 0.0043	0.0483 ± 0.0033
0.14	94.2117 ± 1.3701	0.1197 ± 0.0093	0.0493 ± 0.0032
0.16	95.2888 ± 1.2188	0.1329 ± 0.0151	0.0504 ± 0.0032
0.18	95.7394 ± 1.2691	0.1329 ± 0.0151	0.0509 ± 0.0032
0.20	96.3388 ± 1.4999	0.1334 ± 0.0144	0.0516 ± 0.0030
0.22	96.8848 ± 1.4946	0.1334 ± 0.0144	0.0526 ± 0.0032
0.24	97.4169 ± 1.4676	0.1334 ± 0.0144	0.0536 ± 0.0031
0.26	97.7929 ± 1.6132	0.1334 ± 0.0144	0.0545 ± 0.0039
0.28	97.7929 ± 1.6132	0.1334 ± 0.0144	0.0545 ± 0.0039
0.30	97.9458 ± 1.7403	0.1363 ± 0.0168	0.0548 ± 0.0041

With this information, we can draw two conclusions:

- The plateau in the metrics is reached at a **MAD value that is approximately 2 or 3 times the accuracy of the measuring instrument**: our accelerometer has an accuracy of 0.05 m/s^2. With this value, the alignments are satisfactory, as presented in Figure 10.
- **The percentage of aligned snapshots is between 90 and 95%** ($\pm 2\%$), which is enough to consider the distance metrics. Since the Euclidean distance is below the accuracy of the accelerometer, we could say that the Elevate simulator is faithful enough for this scenario.

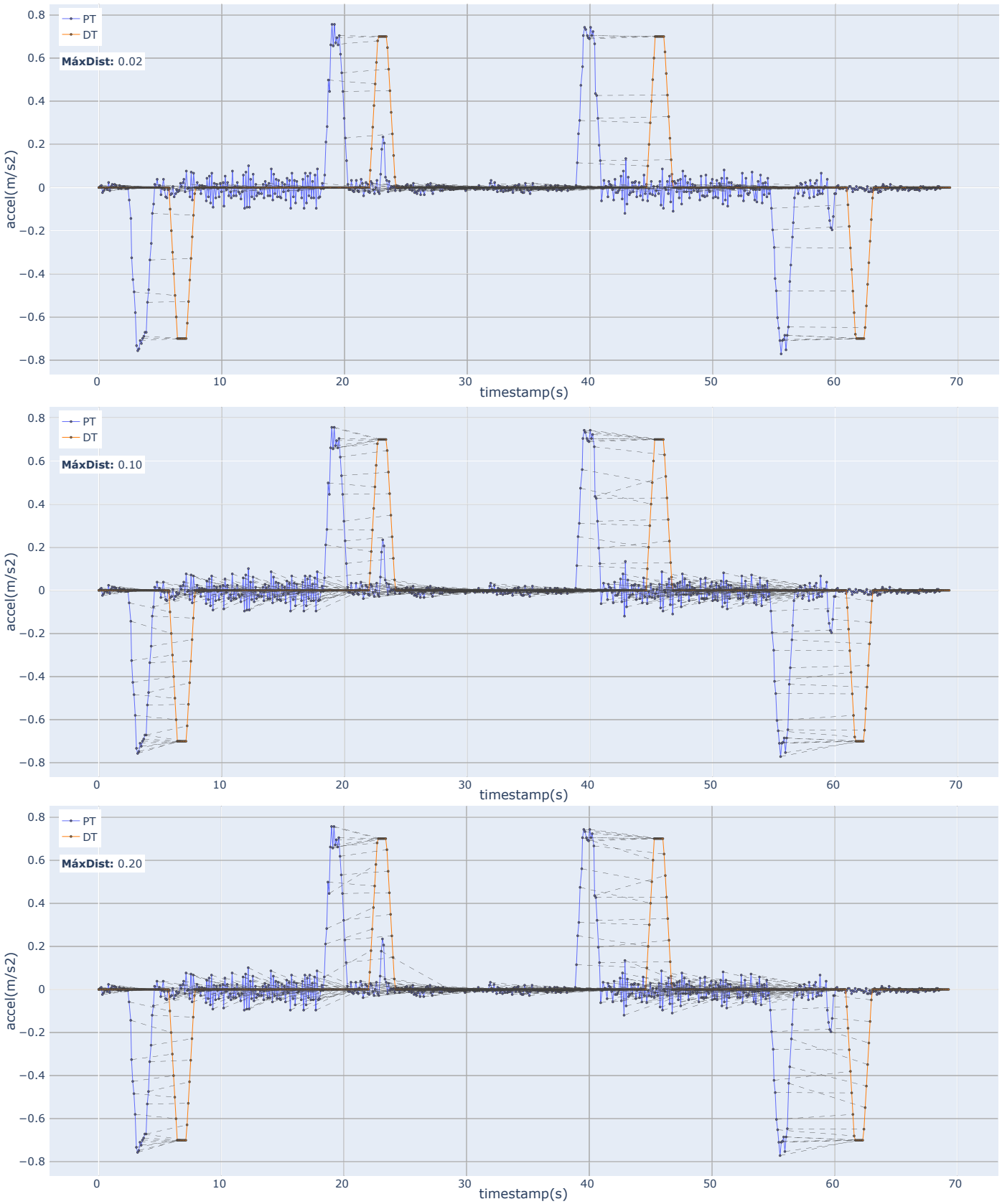


Fig. 10: Alignments of the scenario (4-0-4) with a MAD of 0.02, 0.10, and 0.20 m/s^2, from top to bottom.

2) Scenario (4-2-0-2-4)

To analyze the scenario (4-2-0-2-4), we follow the same procedure as in the previous scenario: for a MAD range of [0.02, 0.30], we align a single trace from a deterministic simulator with the samples of the real system's behavior for this scenario. In this case, we have three samples, and given the fifteen distinct MAD values, we will analyze the results of 45 alignments taking into account only the relevant areas. The statistics of the Low Complexity Areas are not taken into account.

Figure 11 and Table III present the aggregated results of all the alignments. Again, the figure includes three gray lines representing each of the samples. The thick-colored lines represent the mean of the samples, and the shaded region represents the standard deviation.

The fidelity metric results indicate that the plateau is reached in this case for a MAD of approximately $0.14m/s^2$, where we have 93.8% of the aligned snapshots, a Frèchet distance of 0.1 m/s^2, and an average Euclidean distance in the relevant areas of 0.04 m/s^2. These values are approximately equal to those obtained in the previous scenario, demonstrating consistency in the fidelity level of the simulator across different scenarios.

If we analyze the specific alignments for different MADs for one of the samples, we obtain the following results:

- **MAD $\mathbf{0.02m/s^2}$.** (Figure 12, top). The MAD is too strict, and only 24% of the snapshots can be aligned. This value is even smaller than the sensitivity of the accelerometer (0.05 m/s^2). This alignment is considered unsatisfactory.
- **MAD $\mathbf{0.10m/s^2}$.** (Figure 12, middle). We observe that the snapshots in the acceleration curves are aligned, and we achieve 90.4% of matched snapshots, with a Frèchet distance of 0.1 and an average Euclidean of 0.04.
- **MAD $\mathbf{0.20m/s^2}$.** (Figure 12, bottom). The snapshots of the acceleration curves are aligned. However, we see that some points in the braking pattern start to align with points where the elevator is stopped: This MAD is unsuitable for evaluating fidelity for this scenario.

The results are similar to those obtained in the previous scenario. Based on this, we can draw two conclusions:

- The plateau in the metrics is reached at a **MAD value that is approximately 2 or 3 times the accuracy of the measuring instrument**: our accelerometer has an accuracy of 0.05 m/s^2. With this value, the alignments are satisfactory, as presented in Figure 12.
- **The percentage of aligned snapshots is between 90 and 95%** ($\pm 2\%$), which is enough to consider the distance metrics. Since the Euclidean distance is below the accuracy of the accelerometer, we could say that the Elevate simulator is faithful enough for this scenario.

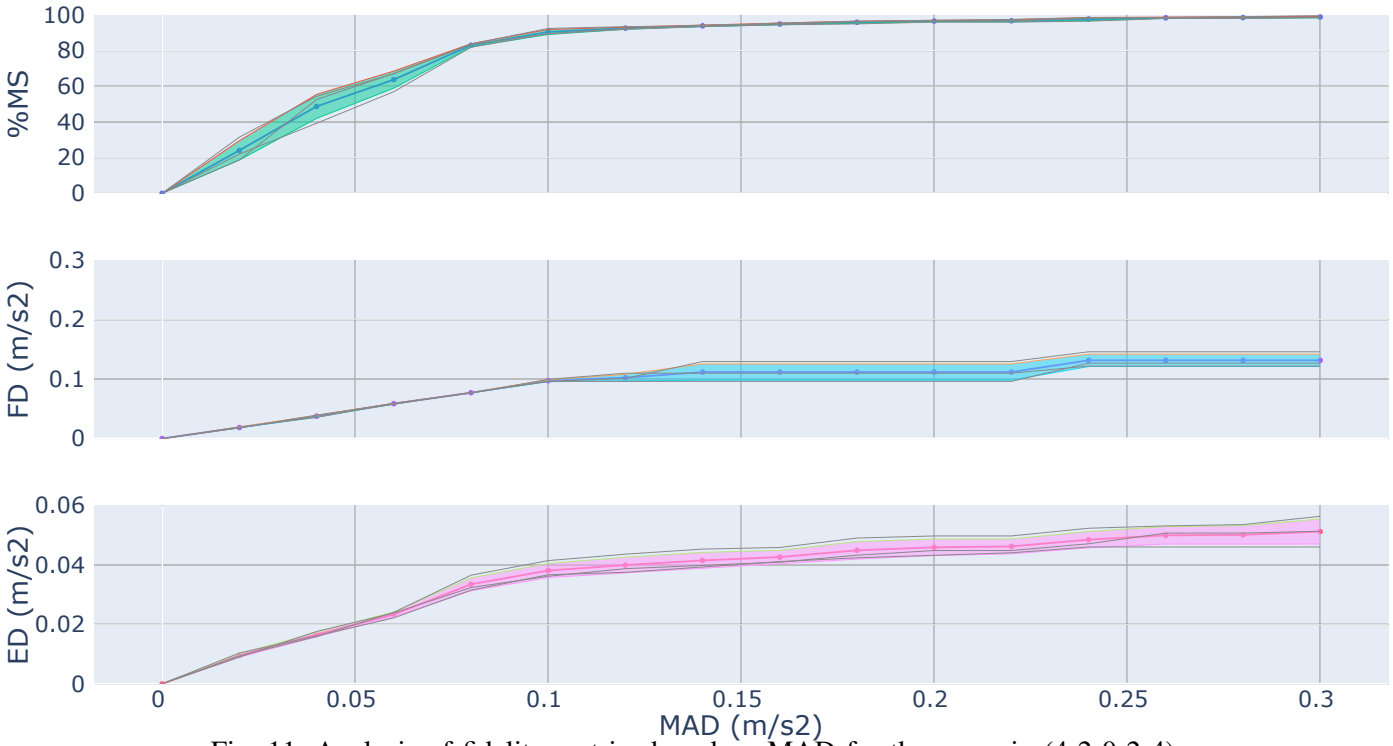


Fig. 11: Analysis of fidelity metrics based on MAD for the scenario (4-2-0-2-4).

TABLE III: Results of the fidelity metrics for the scenario (4-2-0-2-4).

MAD accel(m/s2)	% matched	Frèchet	Avg. Euclidean
0.02	24.0562 ± 5.4164	0.0185 ± 0.0006	0.0096 ± 0.0006
0.04	48.6915 ± 6.7269	0.0375 ± 0.0016	0.0166 ± 0.0007
0.06	63.7049 ± 4.7855	0.0587 ± 0.0005	0.0233 ± 0.0008
0.08	82.9693 ± 0.8321	0.0770 ± 0.0004	0.0335 ± 0.0022
0.10	90.4178 ± 1.3920	0.0973 ± 0.0016	0.0381 ± 0.0024
0.12	92.5549 ± 0.6230	0.1024 ± 0.0057	0.0399 ± 0.0026
0.14	93.8671 ± 0.3377	0.1118 ± 0.0137	0.0415 ± 0.0027
0.16	94.8709 ± 0.4140	0.1118 ± 0.0137	0.0426 ± 0.0023
0.18	95.8561 ± 0.6508	0.1118 ± 0.0137	0.0449 ± 0.0029
0.20	96.5215 ± 0.4162	0.1118 ± 0.0137	0.0459 ± 0.0028
0.22	96.6907 ± 0.6140	0.1118 ± 0.0137	0.0462 ± 0.0025
0.24	97.6760 ± 0.8713	0.1315 ± 0.0104	0.0485 ± 0.0027
0.26	98.3405 ± 0.2576	0.1315 ± 0.0104	0.0500 ± 0.0029
0.28	98.5000 ± 0.4344	0.1315 ± 0.0104	0.0501 ± 0.0031
0.30	98.8278 ± 0.4905	0.1315 ± 0.0104	0.0512 ± 0.0042

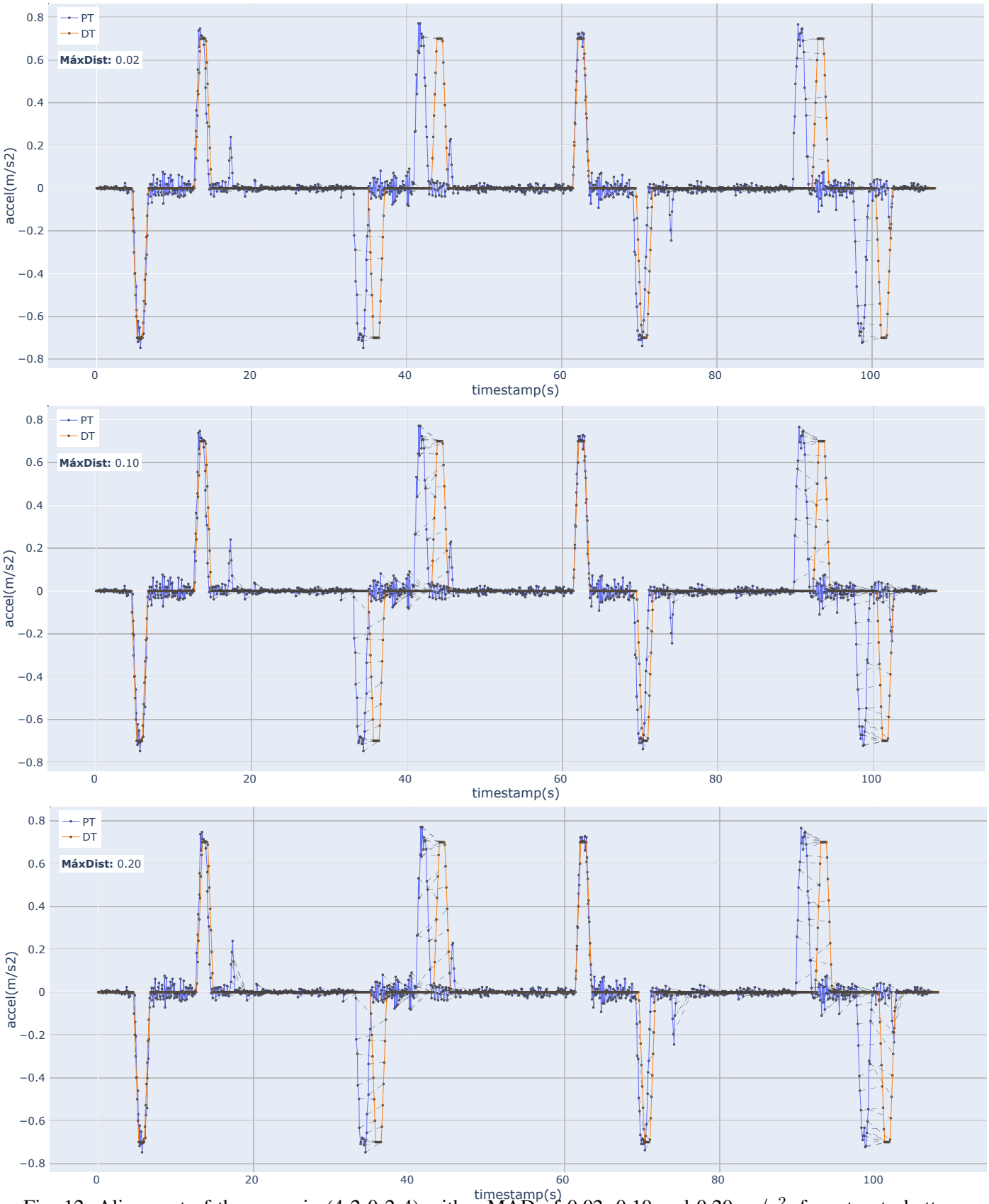


Fig. 12: Alignment of the scenario (4-2-0-2-4) with a MAD of 0.02, 0.10 and 0.20 m/s^2, from top to bottom.

3) Scenario (4-3-2-1-0-1-2-3-4)

To analyze the scenario (4-3-2-1-0-1-2-3-4), we follow the same procedure as in the two previous scenarios: for a MAD range of [0.02, 0.30], we align a single trace from a deterministic simulator with the samples of the real system's behavior for this scenario. In this case, we have three samples, and given fifteen distinct MAD values, we will analyze the results of 45 alignments taking into account only the relevant areas. The statistics of the Low Complexity Areas are not taken into account.

Figure 13 and Table IV present the aggregated results of all the alignments. Once again, the figure includes three gray lines representing each of the samples. The thick-colored lines represent the mean between the samples, and the shaded region represents the standard deviation.

The fidelity metric results indicate that the plateau is reached in this case for a MAD of approximately $0.14m/s^2$, where we have 91.6% of the aligned snapshots, a Frèchet distance of 0.1 m/s^2, and an average Euclidean distance in the relevant areas of 0.04 m/s^2. These values are similar to those obtained in the previous two scenarios, demonstrating consistency in the fidelity level of the simulator across different scenarios.

If we analyze the specific alignments for one of the samples, we obtain the following results:

- **MAD $\mathbf{0.02 m/s^2}$.** (Figura 14, top) The MAD is too strict, and only 22.6% of the snapshots can be aligned. The acceleration curves are not fully aligned. This value is even smaller than the sensitivity of the accelerometer (0.05 m/s^2). This alignment is considered unsatisfactory.
- **MAD $\mathbf{0.10 m/s^2}$.** (Figura 14, middle) We can observe that more of the snapshots in the acceleration curves are aligned, and we achieve 88.3% aligned snapshots, with a Frèchet distance of 0.1 and an average Euclidean of 0.03.
- **MAD $\mathbf{0.20 m/s^2}$.** (Figura 14, bottom) The snapshots of the acceleration curves are aligned. However, we can see that some points in the braking pattern start to align with points where the elevator is stopped. This MAD is too high for a proper evaluation of fidelity.

The results are similar to those obtained in two the previous scenarios. Based on this, we can draw two conclusions:

- The plateau in the metrics is reached at a **MAD value that is approximately 2 or 3 times the accuracy of the measuring instrument**: our accelerometer has an accuracy of 0.05 m/s^2. With this value, the alignments are satisfactory, as presented in Figure 12.
- **The percentage of aligned snapshots is between 90 and 95%** ($\pm 2\%$), which is enough to consider the distance metrics. Since the Euclidean distance is below the accuracy of the accelerometer, we could say that the Elevate simulator is faithful enough for this scenario.

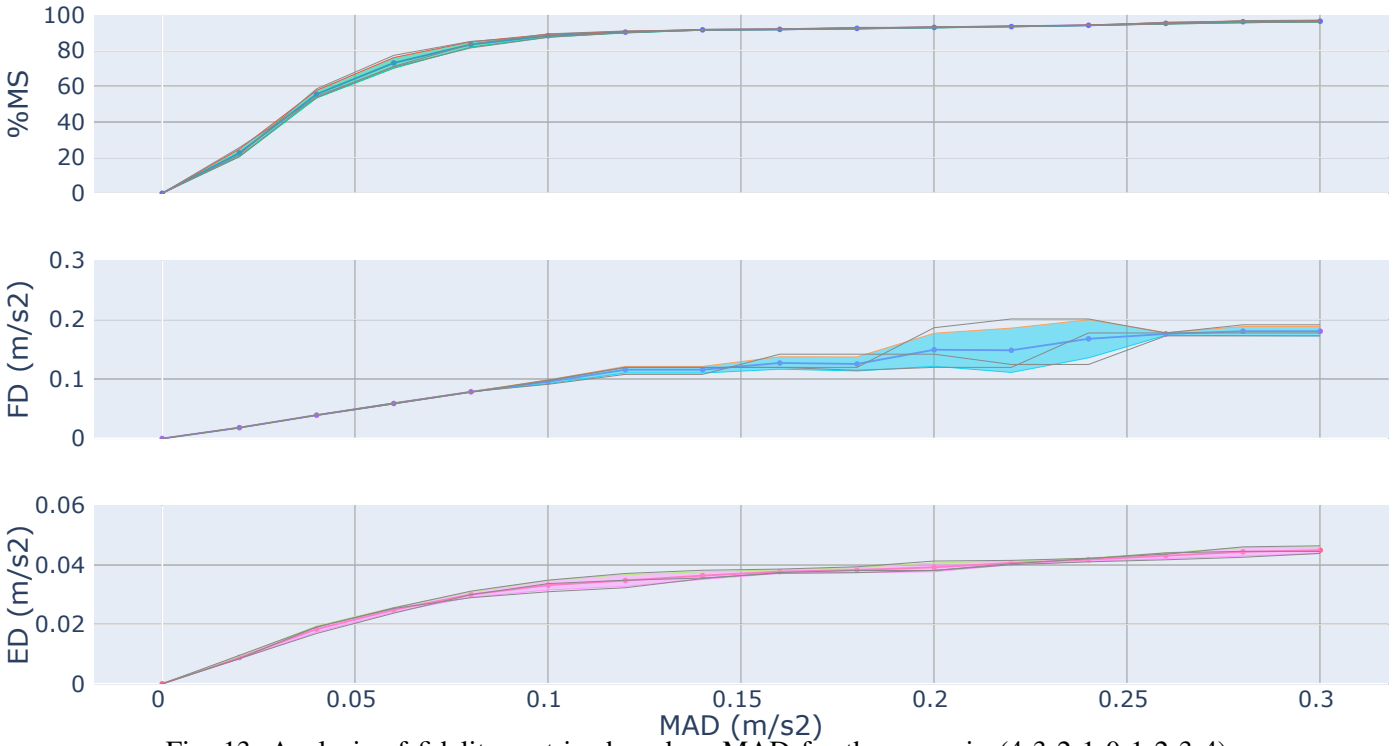


Fig. 13: Analysis of fidelity metrics based on MAD for the scenario (4-3-2-1-0-1-2-3-4)

TABLE IV: Results of the fidelity metrics for the scenario (4-3-2-1-0-1-2-3-4).

MAD accel(m/s2)	% matched	Frèchet	Avg. Euclidean
0.02	22.6084 ± 2.1580	0.0183 ± 0.0003	0.0089 ± 0.0005
0.04	55.5485 ± 2.1395	0.0394 ± 0.0003	0.0184 ± 0.0010
0.06	73.0685 ± 3.0030	0.0590 ± 0.0007	0.0249 ± 0.0008
0.08	83.4355 ± 1.4990	0.0786 ± 0.0001	0.0300 ± 0.0009
0.10	88.3360 ± 0.8090	0.0956 ± 0.0031	0.0332 ± 0.0016
0.12	90.3349 ± 0.4808	0.1158 ± 0.0055	0.0348 ± 0.0020
0.14	91.6169 ± 0.1915	0.1158 ± 0.0055	0.0364 ± 0.0013
0.16	91.8905 ± 0.1964	0.1271 ± 0.0105	0.0377 ± 0.0006
0.18	92.4402 ± 0.3973	0.1254 ± 0.0118	0.0383 ± 0.0008
0.20	92.9793 ± 0.2675	0.1493 ± 0.0277	0.0392 ± 0.0015
0.22	93.4370 ± 0.1761	0.1485 ± 0.0373	0.0407 ± 0.0006
0.24	94.1684 ± 0.1669	0.1678 ± 0.0320	0.0418 ± 0.0005
0.26	95.2590 ± 0.4628	0.1759 ± 0.0023	0.0431 ± 0.0010
0.28	96.1621 ± 0.5165	0.1805 ± 0.0079	0.0444 ± 0.0014
0.30	96.4408 ± 0.4683	0.1805 ± 0.0079	0.0449 ± 0.0011

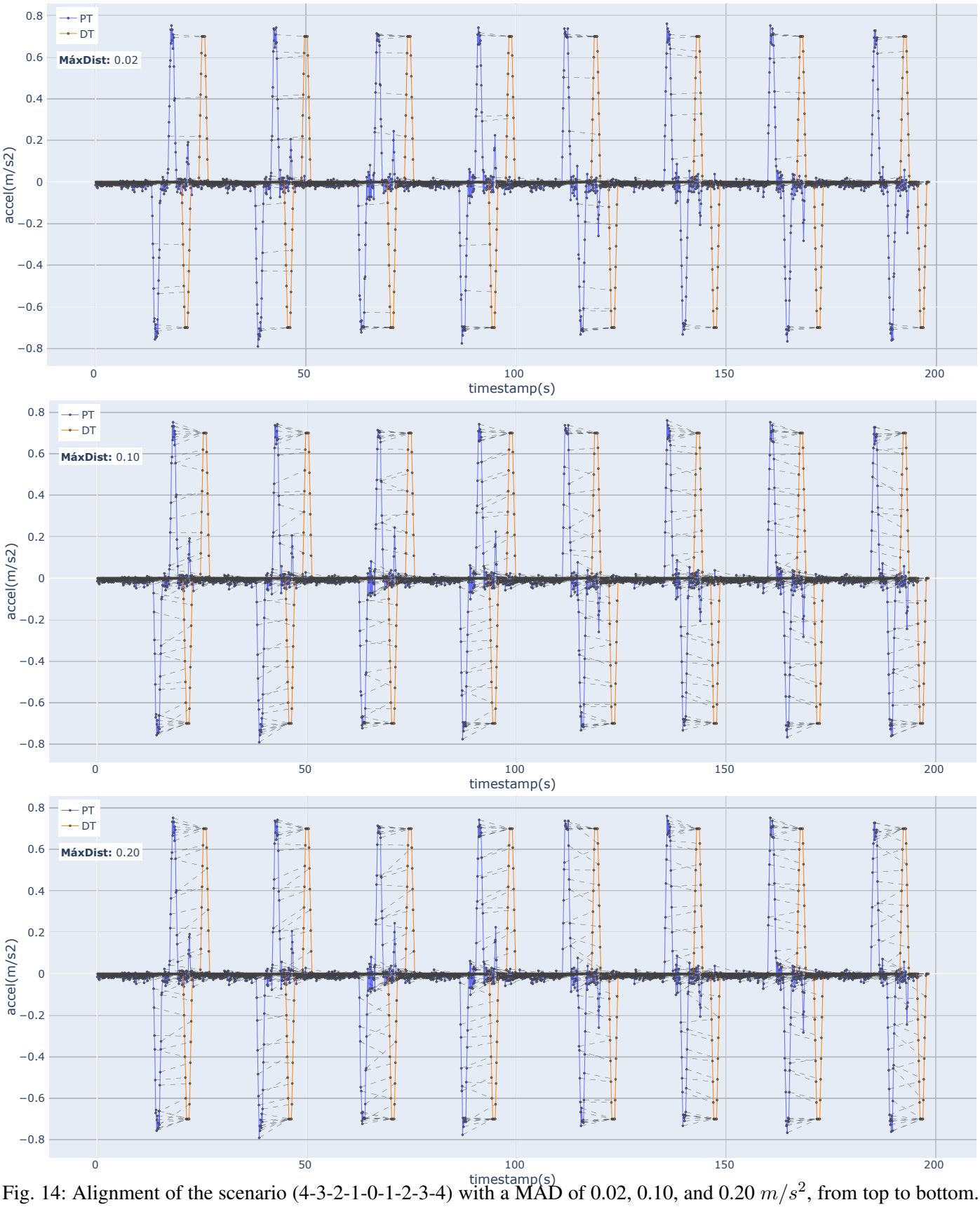


Fig. 14: Alignment of the scenario (4-3-2-1-0-1-2-3-4) with a MAD of 0.02, 0.10, and 0.20 m/s^2, from top to bottom.

II Acknowledgments

We would like to express our gratitude to Aitor Arrieta from the University of Mondragón (Spain) for providing us with the elevator operational data for our analysis. Thanks to this information, we were able to develop the algorithm further and gain insights into the challenges of validating real-world digital twins .

References

[1] Peters Research, "Elevate software," 2023. [Online]. Available: https://peters-research.com/index.php/elevate/

[2] P. Muñoz, J. Troya, M. Wimmer, and A. Vallecillo, "Using trace alignments for measuring the fidelity of a physical and a digital twin: General concepts," 2023. [Online]. Available: https://github.com/atenearesearchgroup/fidelity-measure-for-dts/blob/main/docs/Technical_Report_General_Concepts.pdf

[3] I. Korf, M. Yandell, and J. A. Bedell, *BLAST - an essential guide to the basic local alignment search tool*. O'Reilly, 2003.

[4] D. A. Snow, Ed., *Plant Engineer's Reference Book*, 2nd ed. Elsevier, 2003.