Technical Report: Measuring fidelity of DTs Incubator

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

Manuel Wimmer
CDL-Mint
Johannes Kepler University (Austria)

Contents

I Incubator in the Aarhus University

A. System description

At Aarhus University in Denmark, they are building a digital twin for their studies in the field, using an incubator as the foundation case study. They built this system, shown in Figure 1, as a simple yet representative thermal incubator system. The main goal of the digital twin system of the incubator is to reach a certain temperature within a box and regulate it regardless of the content inside.

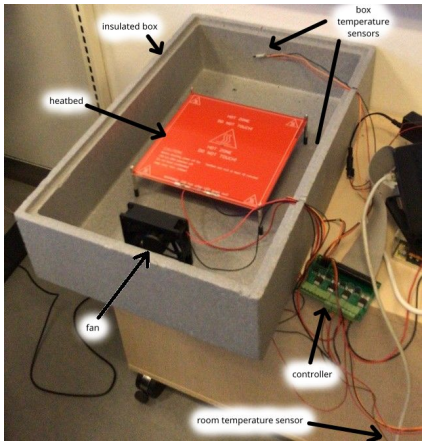

Fig. 1. Incubator in the University of Aarhus (Denmark)

The physical twin of the incubator includes a set of components which form a plant that is controlled by a Raspberry Pi. The system includes an insulated box fitted with a heatbed, and complete with a software system for communication, a controller, and simulation models. The system utilizes two DHT22 sensors (available at https://components101.com/sensors/dht22-pinout-specs-datasheet) to monitor the temperature inside the incubator, while another sensor measures the room temperature. The sensors are programmed to provide readings once every two seconds. The controller, which operates every three seconds, calculates the average temperature readings from the two sensors inside the incubator to determine its internal temperature.

Two simulation models were developed to act as the digital twin. They can predict the temperature inside the incubator, one with two free parameters and one with four free parameters.

1) 2-parameter model

The two parameter model for the incubator is a simple linear model which predicts the temperature inside using two parameters: the heat transfer coefficient and thermal resistance. It assumes that these parameters remain constant throughout the operation. To determine the values of these parameters, experimental data from the physical incubator system was used for calibration. The calibration process involved minimizing the difference between the predicted temperatures from the model and the actual measured temperatures using a least-squares method. The two-parameter model was found to be accurate in predicting the incubator's temperature. However, it should be noted that the model has limitations in capturing non-linear dynamics and transient behavior of the system.

2) 4-parameter model

The four-parameter temperature prediction model for the incubator is an advanced non-linear model designed to capture the system's non-linear dynamics and transient behavior. This model considers the heat transfer coefficient, thermal resistance, thermal capacitance, and a time constant as its key parameters. To determine the optimal values for these parameters, experimental data from the physical incubator system was used for calibration. The calibration process involved minimizing the discrepancy between the predicted temperatures from the model and the actual measured temperatures using a least-squares method, as in the two-parameter model. The four-parameter model exhibited greater accuracy compared to the simpler two-parameter model, particularly during transient behavior. However, it comes with the trade-off of requiring more computational resources and being more complex in nature.

More detailed information about the incubator system description and implementation are available in [1] and the complete implementation of the system in [2].

B. Scenarios

The objective of the incubator is to maintain the temperature within specific boundaries. For the scenarios we are working with, the upper limit will be set at 35 degrees Celsius, which is the desired temperature, while the lower limit will be 30 degrees Celsius. As a result, the temperature inside the incubator will fluctuate within these limits.

1) *Heating time 3 s - Heating gap 2 s (Ht3Hg2)*

In Figure 2, we can observe the heating and cooling patterns for the first scenario: *Heating time 3 s - Heating gap 2 s (Ht3Hg2)*. The heating pattern is within the temperature range of [35, 30] degrees Celsius for the physical twin and the two digital twins of the incubator. In this particular scenario, the heater is activated during the heating process, following a pattern of heating for 3 seconds, turning off for 2 seconds, and then reactivating if the controller detects that the target temperature has not been reached. These short decision-making periods help maintain the temperature within tight boundaries, ensuring that the system consistently operates within the specified range.

Fig. 2. Traces for scenario Ht3Hg2 (In order: Physical Twin, 2-P model, 4-P model)

2) Heating time 30 s - Heating gap 20 s (Ht30Hg20)

In Figure 3, we can observe the heating patterns of the second scenario: *Heating time 30 s - Heating gap 20 s (Ht30Hg20)*. In this scenario, the controller activates the heater for 30 seconds if it detects that the temperature is below the upper limit, and then turns it off for 20 seconds. It reactivates the heater if, after these 20 seconds, the temperature is still below the limit. This spaced-out control intervals result in slower decision-making and cause the system to move outside the specified temperature limits, leading to poorer performance based on the problem parameters. Additionally, this pattern creates a sawtooth pattern in the two-parameter model, which is unable to smooth out the temperature changes caused by this control scheme.

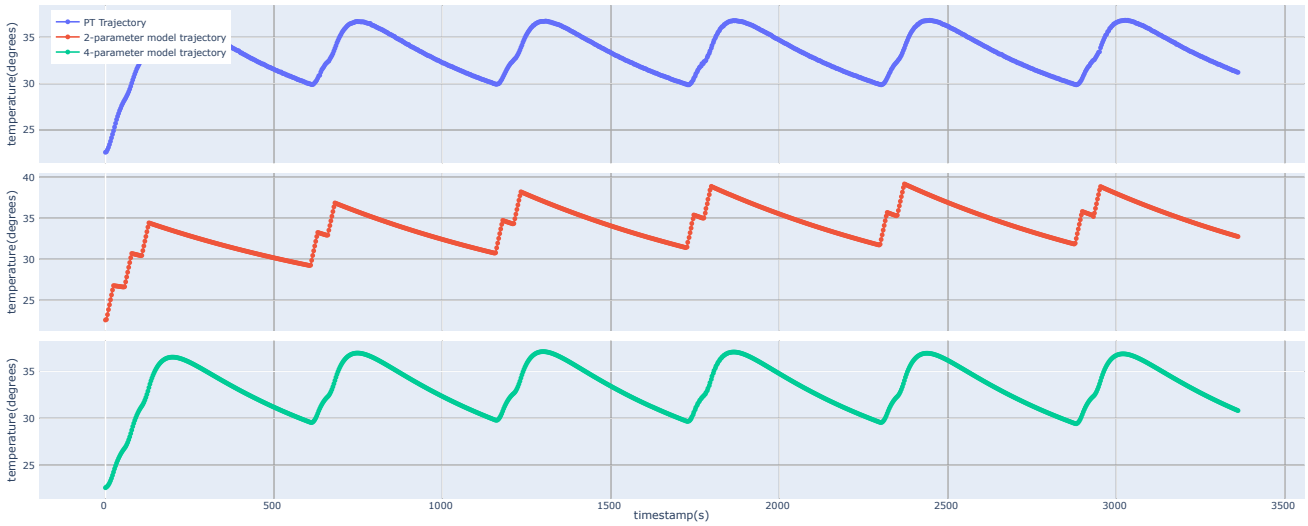

Fig. 3. Traces for scenario Ht30Hg20 (In order: Physical Twin, 2-P model, 4-P model)

C. Gap Tunning

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 **??** Paula *añadir referencia a mano* .

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

Parameter	**Range**	**Increments**
Maximum Acceptable Distance (MAD)	[0.10, 0.2]	0.02
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 882 alignments applied to Scenario Ht3Hg2 for each model (2-P and 4-P). 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 4 and 5. 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 breakpoints, dividing the values into groups based on their values.

1) Simple gap

The statistics for the percentage of matched snapshots, Frèchet distance, and average Euclidean distance between aligned points are depicted in Figures 4 and 5, for the 2-Parameter model and Figures 6 and 7, for the 4-Parameter model. These figures specifically focus on the 126 input configurations where the P_{op} value is set to 0, implying that the cost of opening and extending a gap is the same.

TABLE I. ANALYSIS OF THE INFLUENCE OF SIMPLE GAP PENALTY ON THE PERCENTAGE OF ALIGNED SNAPSHOTS.

Model	R-squared	F-statistic	Coef. MAD	P-value MAD	Coef. Init_gap	P-value Init_gap	Coef. Cont_gap	P-value Cont_gap
Simple 2-P (segment)	0.530	26.549	31.651 ± 4.375	0.000	0.000 ± 0.000	0.000	1.311 ± 0.604	0.035
Simple 2-P (all)	0.876	411.616	69.583 ± 14.951	0.000	0.000 ± 0.000	0.000	25.074 ± 0.886	0.000
Simple 4-P (segment)	0.450	4.917	26.559 ± 9.615	0.000	0.000 ± 0.000	0.000	6.244 ± 2.547	0.031
Simple 4-P (all)	0.983	3397.827	62.053 ± 3.099	0.000	0.000 ± 0.000	0.000	14.678 ± 0.184	0.000

The first two figures, 4 and 6, presents the samples sorted along the x-axis based on the increasing percentage of matched snapshots. In the other two figures, 5 and 7, instead of arranging the samples based on the percentage of matched snapshots, they are sorted in increasing order of the number of gaps in the alignment.

Let us analyze the configuration results for both models.

I. 2-Parameters model

The change point, which signifies a shift in the values, is observed at the peak of the percentage of aligned snapshots. This peak includes the samples with the highest percentage of aligned snapshots, which are in the range [70,80] %, shaded in pink in Figure 8. This range of achieves 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 -1.0. 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 5. In this case, we can observe a similar breakpoint as in Figure 4, obtaining the alignments with the highest percentage of matched snapshots in the range [70, 80]%. Once a certain number of gaps is reached (around 300), the alignments become satisfactory. This happens again for values of P_{ex} greater than -1.0, 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-C1. In this table, we have the values for the analysis of all samples (Simple 2-P (all)) and specifically for the red-colored segment with the most optimal values (Simple 2-P (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 are relatively high: if we change one unit the MAD value, the % of aligned snapshot should change approximately the value of the coefficient which is between 30%. However, by examining Figure 4, 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 relevance but the coefficients are 0, which means that it barely affect the percentage of aligned snapshots, since it doesn't play any role in this analysis.
- $\mathbf{P_{ex}}$ 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 [-1, 0) has little impact on the percentage of aligned snapshots.

The conclusion is that we can achieve satisfactory results for the incubator example with $\mathbf{P_{ex}}$ **values between [-1.0, 0)**, regardless of the specific variations within that range.

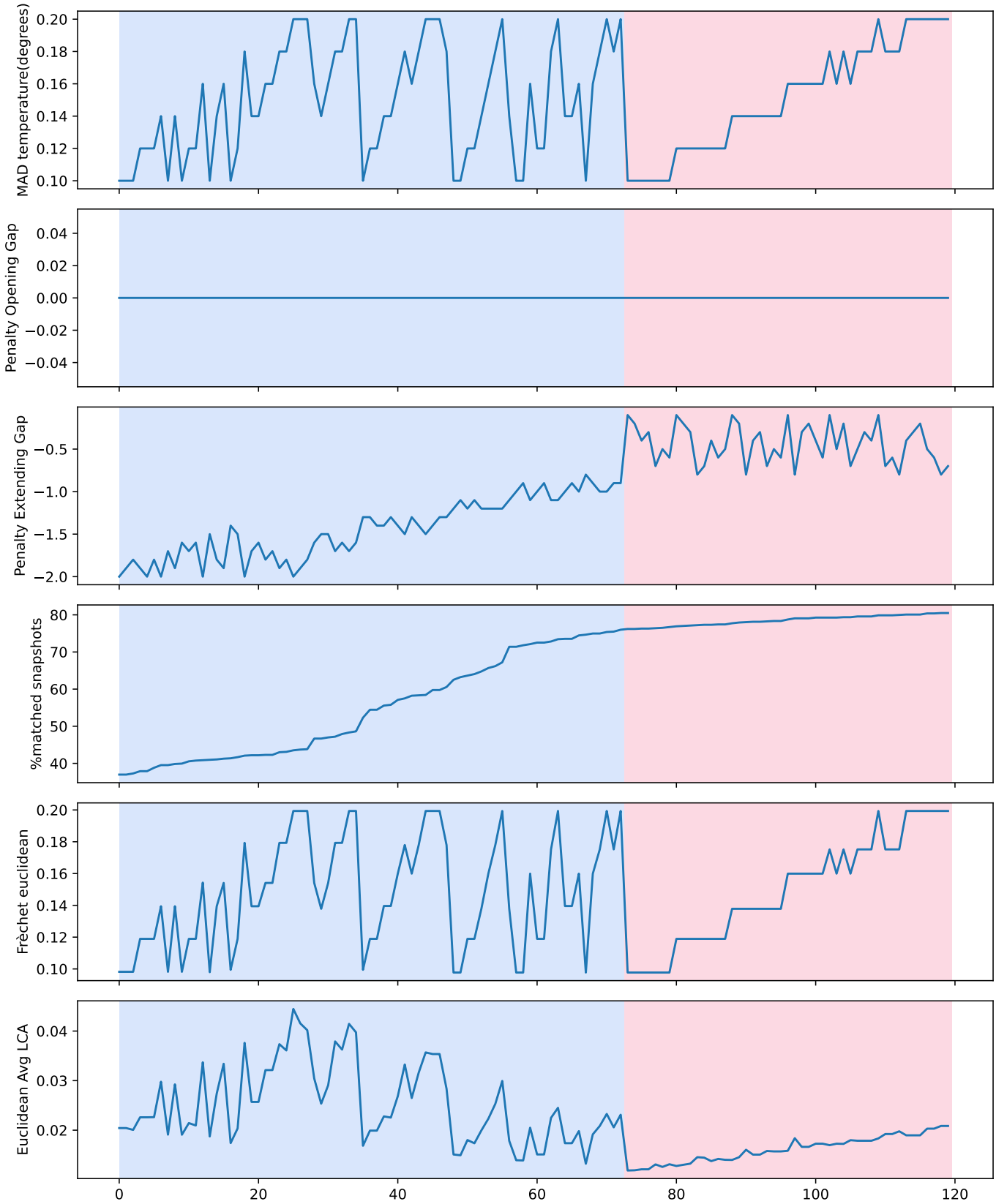

Fig. 4. Analysis of alignment statistics for simple gap in ascending order by % of matched snapshots for 2-Parameters model.

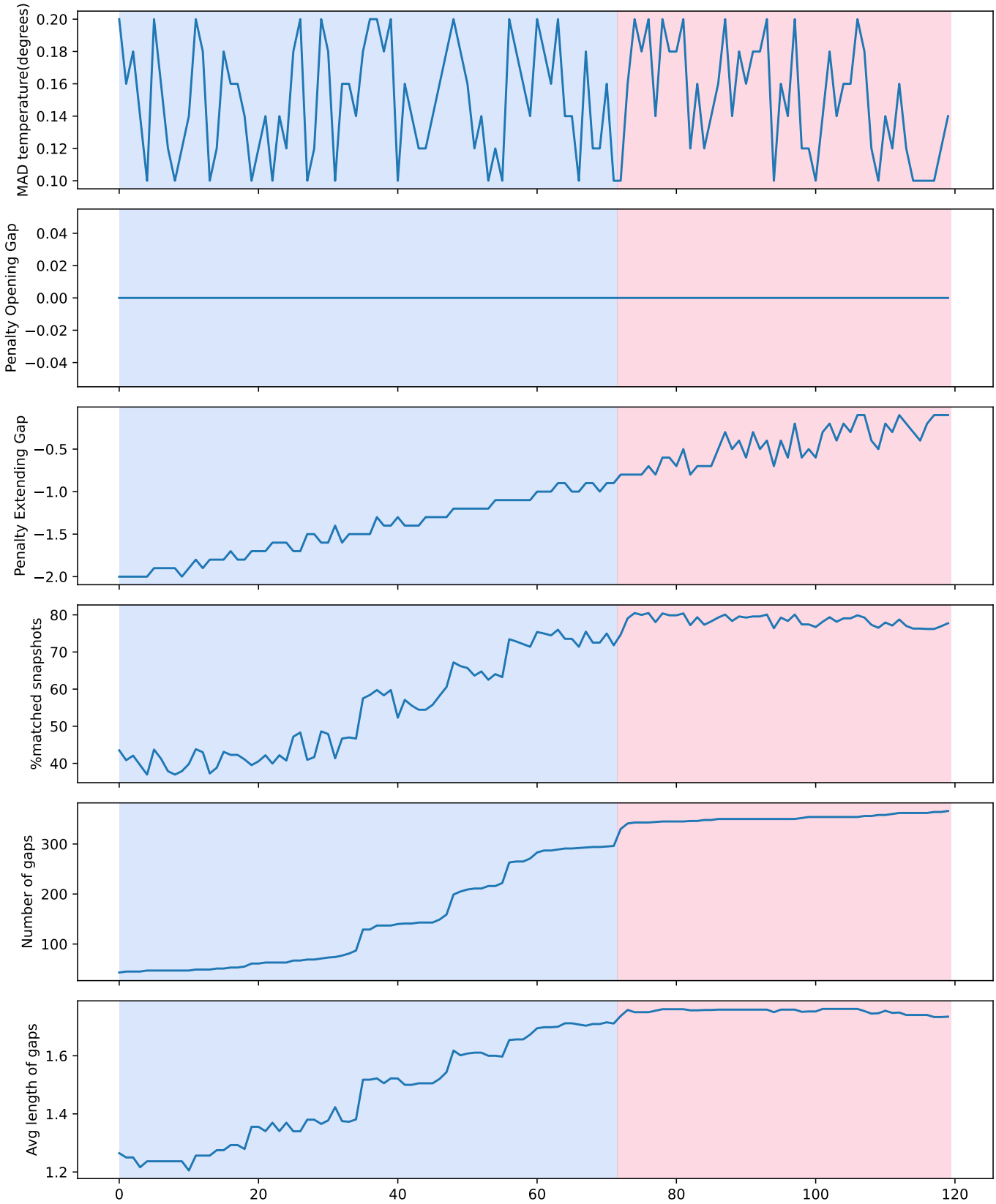

Fig. 5. Analysis of alignment statistics for simple gap in ascending order by number of gaps for 2-Parameters model.

II. 4-Parameters model

The analysis of the 4-Parameter Model closely resembles that of the 2-Parameter Model. In this case, as shown in Figure 6, the change point is not as prominent, but there is a shift in trends around 70% of aligned snapshots. The percentage of aligned points progressively increases as we decrease P_{ex}. However, similar to the previous case, we observe that the best results are achieved with a P_{ex} greater than -1.0. With these configurations, we attain a percentage of aligned snapshots ranging from 70% to 80%. Furthermore, we significantly reduce the average Euclidean distance between points to below 0.025, which is in contrast to the remaining samples. 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.

As in the previous example, the pink 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.

We can further analyse this example this more easily by looking at Figure 7. In this case, we can observe a similar change point as in Figure 6, obtaining the alignments with the highest percentage of matched snapshots in the range [70, 80]%. Once a certain number of gaps is reached (around 200), with P_{ex} greater than -1.0, the alignments become satisfactory. By further increasing this value up to -0.5, we can achieve an even higher number of aligned snapshots with over 300 gaps in the alignment. This pattern repeats for values of P_{ex} greater than -1.0, indicating that the penalty for introducing a gap is low enough for the algorithm to prioritize an alignment that includes a sufficient number of gaps. These gaps play a crucial role in characterizing the behavior of our system compared to the simulation, allowing for the inclusion of delays, among other factors.

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-C1. In this table, we have the values for the analysis of all samples (Simple 4-P (all)) and specifically for the red-colored segment with the most optimal values (Simple 4-P (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 are relatively high: if we change one unit the MAD value, the % of aligned snapshot should change approximately the value of the coefficient which is between 25%. However, by examining Figure 4, 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 relevance but the coefficients are 0, which means that it barely affect the percentage of aligned snapshots, since it doesn't play any role in this analysis.
- $\mathbf{P_{ex}}$ 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 [-1, 0) has little impact on the percentage of aligned snapshots.

The conclusion is that we can achieve satisfactory results for the incubator example with $\mathbf{P_{ex}}$ **values between [-1.0, 0)**, regardless of the specific variations within that range.

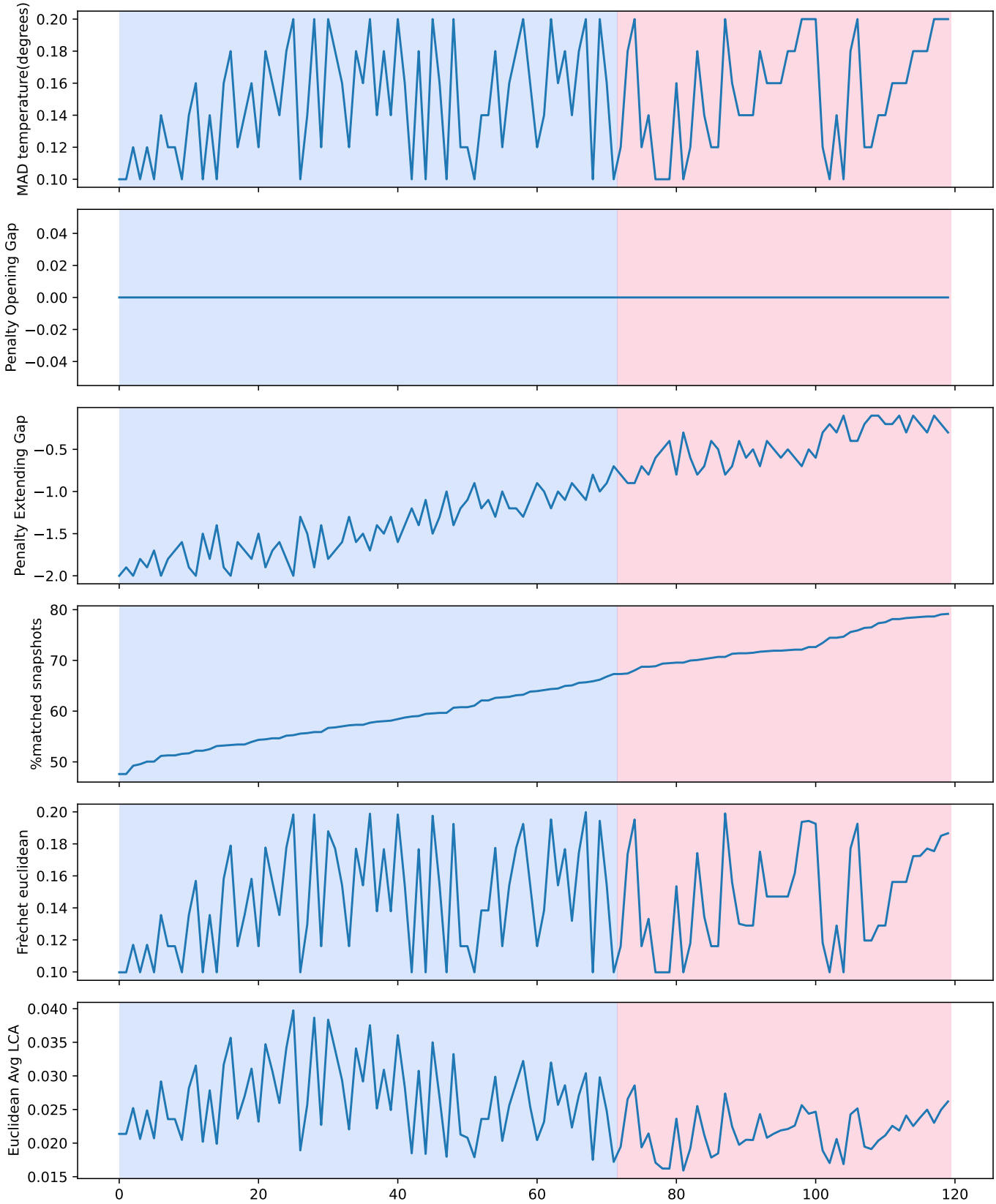

Fig. 6. Analysis of alignment statistics for simple gap in ascending order by % of matched snapshots for 4-Parameters model.

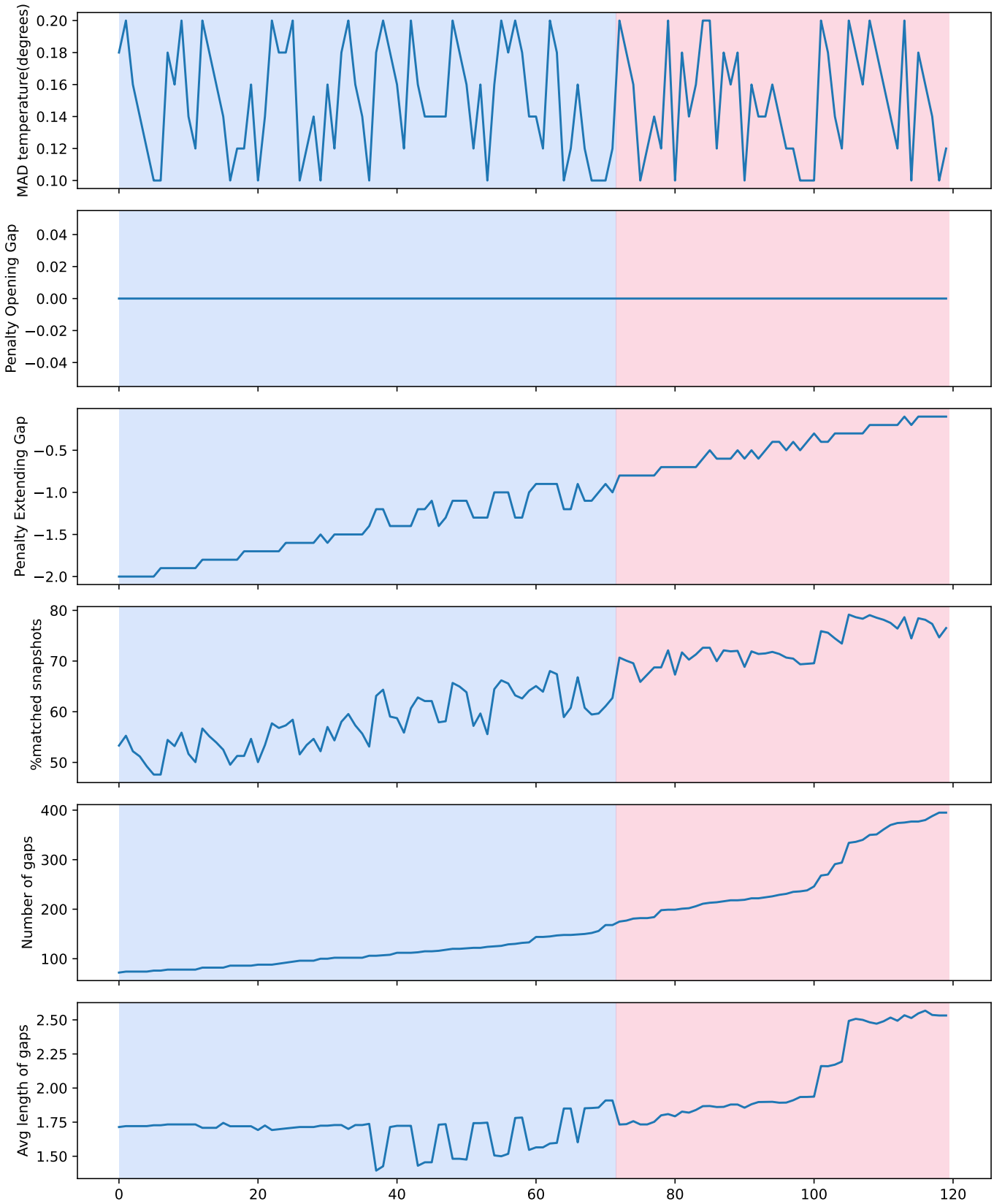

Fig. 7. Analysis of alignment statistics for simple gap in ascending order by number of gaps for 4-Parameters model.

TABLE II. ANALYSIS OF THE INFLUENCE OF AFFINE GAP PENALTY ON THE PERCENTAGE OF ALIGNED SNAPSHOTS.

Model	R-squared	F-statistic	Coef. MAD	P-value MAD	Coef. Init_gap	P-value Init_gap	Coef. Cont_gap	P-value Cont_gap
Affine 2-P (segment)	0.862	165.112	33.468 ± 1.655	0.000	0.563 ± 0.039	0.000	0.371 ± 0.200	0.068
Affine 2-P (all)	0.867	1558.220	91.015 ± 6.439	0.000	3.468 ± 0.258	0.000	24.992 ± 0.381	0.000
Affine 4-P (segment)	0.918	14.833	23.919 ± 4.247	0.005	1.427 ± 0.258	0.005	1.192 ± 0.792	0.207
Affine 4-P (all)	0.960	5662.082	85.357 ± 2.293	0.000	3.401 ± 0.092	0.000	16.204 ± 0.136	0.000

2) Affine gap

I. 2-Parameters model

The analysis for the Affine Gap approach is similar to that performed for the simple approach. In Figures 8 and 17, 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 8, 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 to not be too high in order to include them and obtain relevant alignments.

Similarly, in Figure 17, 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.

Regarding the analysis of the statistical significance of the data, the results are in Table I-C2 (Affine-All, Affine-Segment), and are similar to the previous ones.

- **MAD** has 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 **??**, 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 low for the segment (< 1). This means that a one unit change in the P_{op} produces a change smaller than 1 in the % of matched snapshots in the segment.
- $\mathbf{P_{ex}}$ 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.

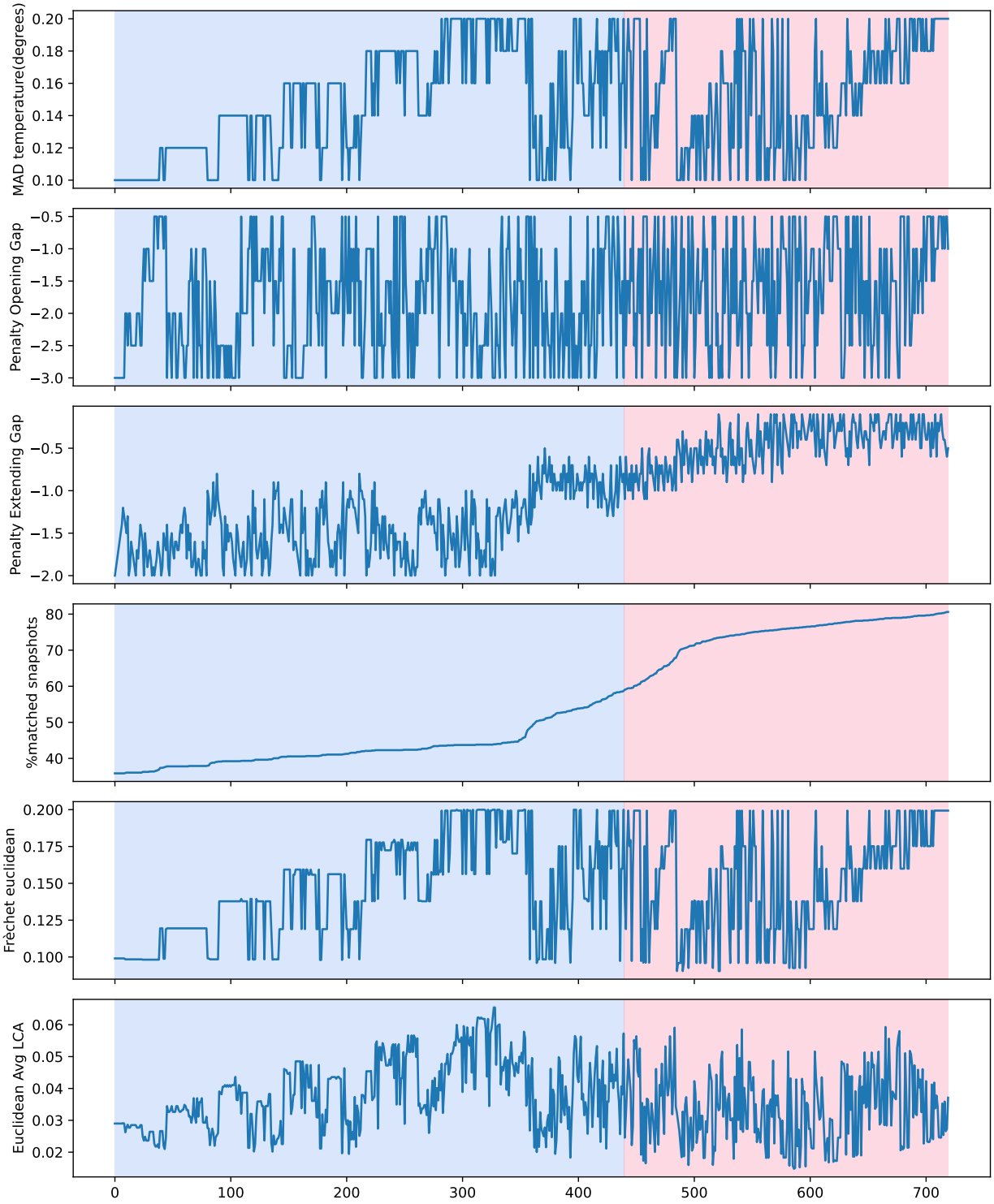

Fig. 8. Analysis of alignment statistics for affine gap in ascending order by % of matched snapshots for 2-Parameters model.

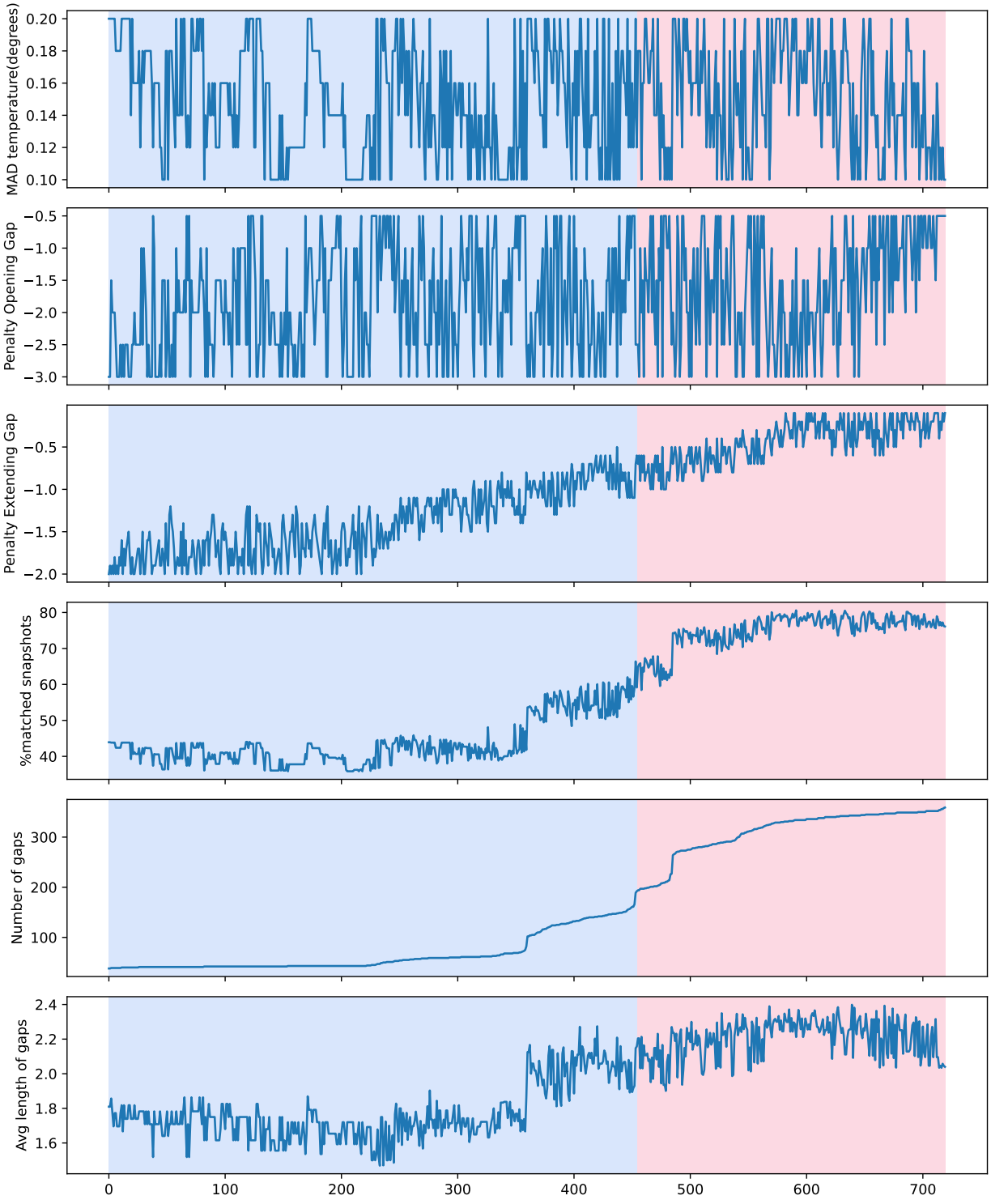

Fig. 9. Analysis of alignment statistics for affine gap in ascending order by number of gaps for 2-Parameters model.

II. 4-Parameters model

The analysis for the 4-Parameters model is similar to the one performed to the 2-Parameter model. In Figures **??** and 11, 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 10, 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 to not be too high in order to include them and obtain relevant alignments.

Similarly, in Figure 17, 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.

Regarding the analysis of the statistical significance of the data, the results are in Table I-C1 (Affine-All, Affine-Segment), and are similar to the previous ones.

- **MAD** has 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 **??**, 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 low for the segment (< 2). This means that a one unit change in the P_{op} produces a change smaller than 2 in the % of matched snapshots in the segment.
- $\mathbf{P_{ex}}$ 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.

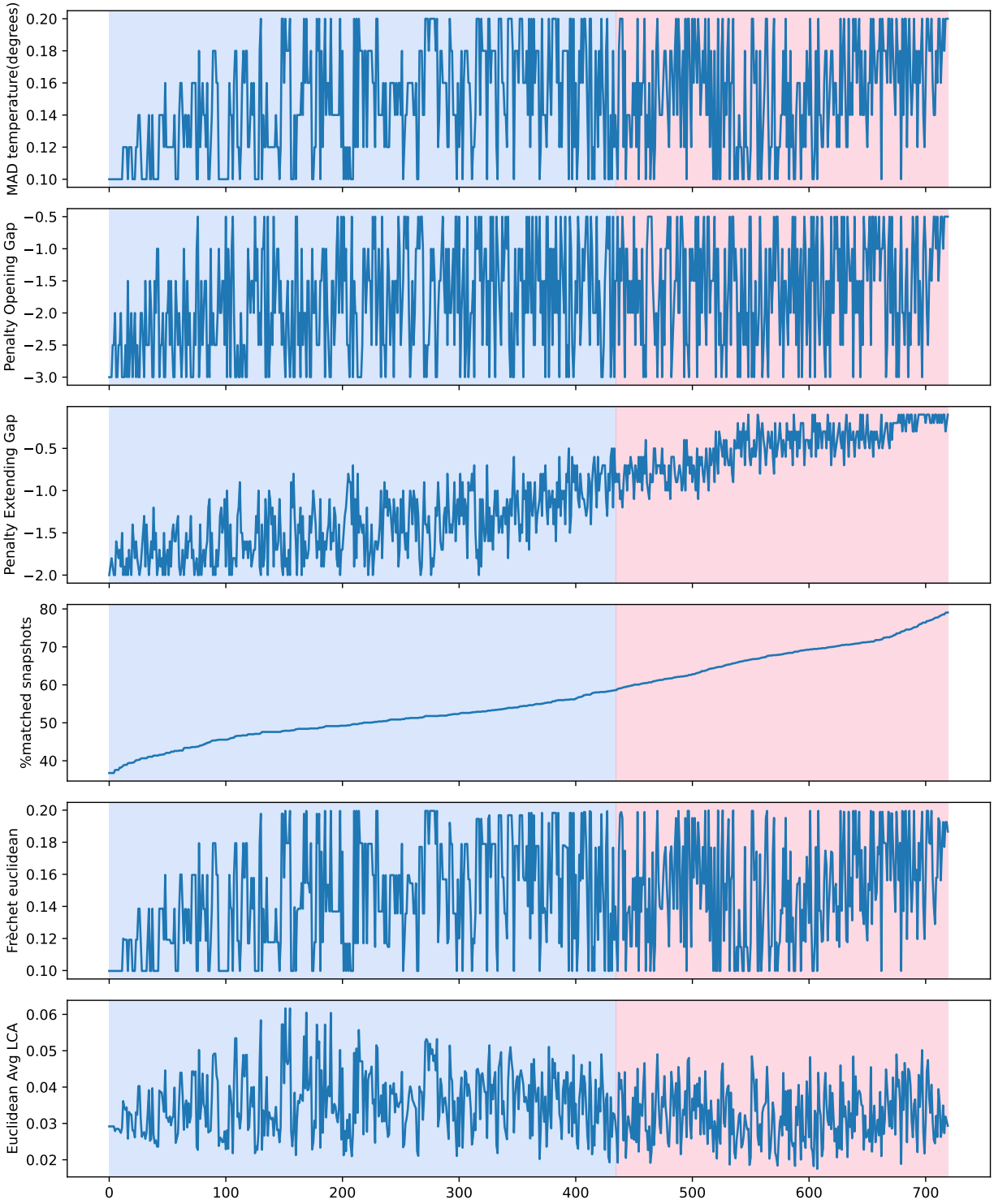

Fig. 10. Analysis of alignment statistics for affine gap in ascending order by % of matched snapshots for 4-Parameters model.

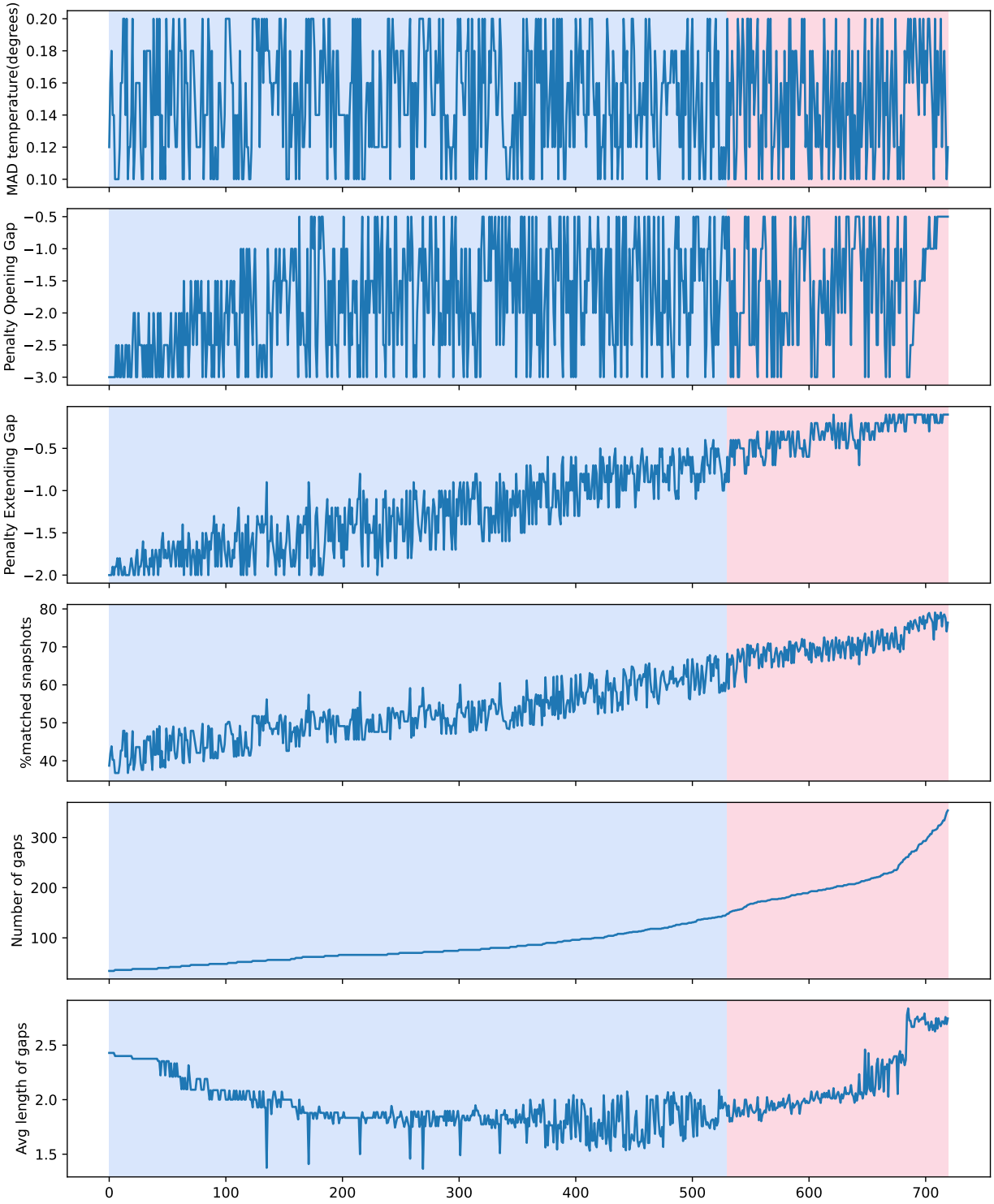

Fig. 11. Analysis of alignment statistics for affine gap in ascending order by number of gaps for 4-Parameters model.

D. Fidelity assessment

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 the some of the fidelity metrics we have defined in section **??** Paula *poner a mano* : the percentage of aligned snapshots, the Frèchet distance, and the average Euclidean distance (in the relevant area) between aligned snapshots.

To interpret these metrics, we need to consider what the values would be for the alignment of two identical traces. In that case, for any value of MAD:

- the **percentage of aligned points** 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 perfect alignment as a reference.

The alignment algorithm applies the following configuration for all scenarios:

- **Maximum Acceptable Distance (MAD):** [0.02, 0.30] with increments of 0.02.
- **Affine Gap Weights:** (-1, -0.1)

The specific and detailed guidelines on how to set the configuration values for Affine Gap are available in the previous section. In the the incubator experiment we do not consider any Low Complexity Areas. As for MAD, it was empirically established by determining where the plateau of fidelity metrics was achieved, for illustrative purposes. To establish a single value in a practical example, we need to reason about the maximum distance we want to allow for aligning two snapshots. We will further develop this idea in the subsequent sections based on the data. In the following sections, we will assess the level of fidelity of both models and compare their suitability as a digital twin depending on the requirements we impose on the system.

TABLE III. FIDELITY RESULTS FOR SCENARIO HT3HG2.

mad_temperature(degrees)	% matched	2-P Model Frèchet	Avg. Euclidean	% matched	4-P Model Frèchet	Avg. Euclidean
0.2	79.9796	0.1994	0.0338	78.5495	0.1926	0.0349
0.4	82.5332	0.2543	0.0547	83.9632	0.3928	0.0646
0.6	83.7589	0.3104	0.0756	87.1297	0.5493	0.0958
0.8	85.2911	0.4311	0.1004	88.5598	0.5178	0.1154
1	86.4147	0.4941	0.1226	89.7855	0.5913	0.1358
1.2	88.049	0.6191	0.1549	90.9091	0.6713	0.159
1.4	89.1726	0.7126	0.1806	91.8284	0.7346	0.1789
1.6	90.5005	0.7751	0.2146	92.4413	0.7796	0.1945
1.8	91.522	0.9001	0.2411	93.3606	0.8252	0.2196
2	92.5434	0.9626	0.272	94.1777	0.8846	0.2439
2.2	93.3606	1.0561	0.2961	94.4842	0.9139	0.2546
2.4	94.382	1.1811	0.3303	94.9949	0.9428	0.2764
2.6	94.8927	1.2126	0.3468	95.4035	0.9715	0.293
2.8	95.4035	1.3376	0.3665	95.7099	0.9856	0.3079
3	95.7099	1.4001	0.379	96.0163	1.0142	0.3284
3.2	96.4249	1.4311	0.4129	96.2206	1.028	0.3368
3.4	96.6292	1.5251	0.4219	96.3228	1.0422	0.3423
3.6	96.9356	1.6191	0.4356	96.3228	1.0422	0.3423
3.8	97.1399	1.6501	0.4455	96.3228	1.0422	0.3423
4	97.4464	1.7436	0.4604	96.4249	1.0422	0.3496
4.2	97.6507	1.7751	0.4707	96.6292	1.0559	0.3675
4.4	97.9571	1.8691	0.4875	96.7314	1.0559	0.3771
4.6	98.1614	1.8691	0.499	96.9356	1.0699	0.3944
4.8	98.4678	1.9316	0.5173	96.9356	1.0699	0.3944
5	98.7743	1.9626	0.5361	97.0378	1.0699	0.4017

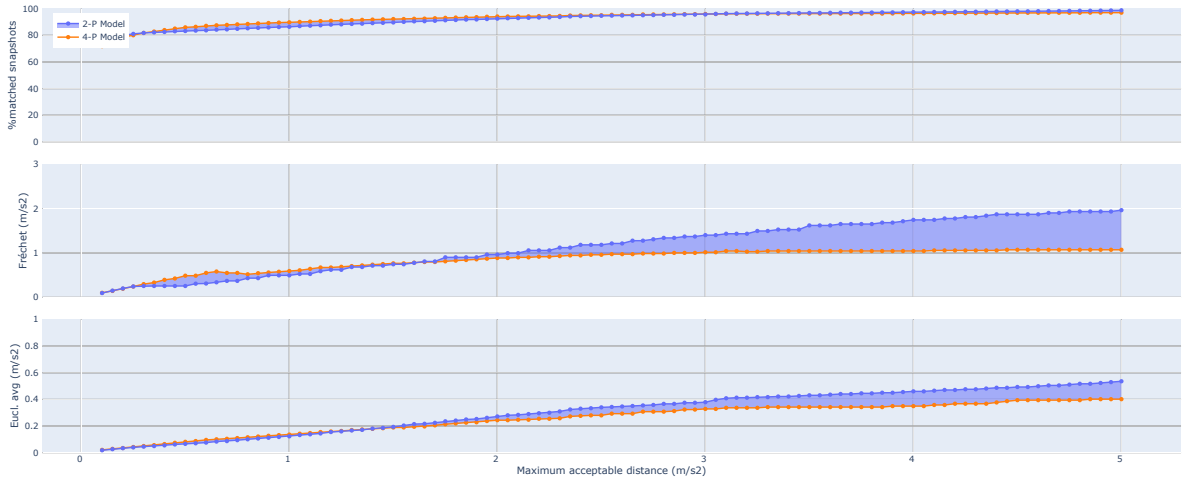

Fig. 12. Statistical comparison of the 2-P Model against the 4-P Model for scenario Ht3Hg2.

1) Heating time 3 s - Heating gap 2 s (Ht3Hg2)

To illustrate this scenario, we included alignment figures for both models with two MAD values (0.2 and 1.2). The alignments for the 2-Parameter model are shown in Figures 13 and 14, while for the 4-Parameter model, they are shown in Figures 15 and 16. To enhance the visualization of the alignments, a constant offset of 3 degrees has been added to all digital twin snapshots, preventing overlapping of the graphs.

Looking at Figures 13 and 15, we can observe that for both models, the cooling and heating processes are aligned. However, there are gaps in the transitions between them. With a MAD of 0.2 degrees Celsius, both models achieve approximately 80% aligned snapshots, with the 2-Parameter model having 1% more aligned snapshots (see Table III and Figure 2). By examining Figure 13, we can see that this is because there are fewer gaps in some transitions, such as the one occurring from timestamp 1000. On the other hand, Figure 15 shows a noticeable gap in the same transition at timestamp 1000. A similar trend can be observed in the other transitions in these two figures.

However, as the MAD value increases (see Table III and Figure 2), we can see that the 4-Parameter model is capable of aligning a greater number of snapshots, surpassing the 2-Parameter model by 2% to 4%. Nonetheless, despite having a higher percentage of aligned points, the 4-Parameter model also exhibits a greater Frèchet distance and a higher average distance, as the incorporated points have lower quality compared to those in the case of the 2-Parameter model.

Based on these results, we can conclude that the 2-Parameter model performs well in scenarios with rapid temperature changes. Although it may not accurately replicate smooth transitions, it can provide a good approximation of the system's temperature. On the other hand, the 4-Parameter model, despite achieving smoother transitions, does not produce values that are precise enough to be considered faithful.

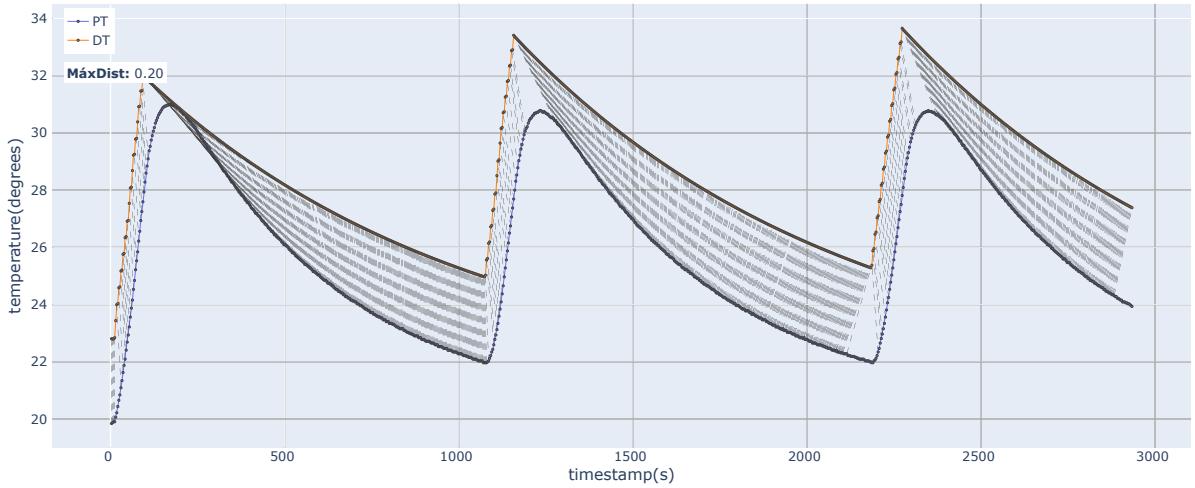

Fig. 13. Alignment of 2-P Model for scenario Ht3Hg2 with MAD 0.2.

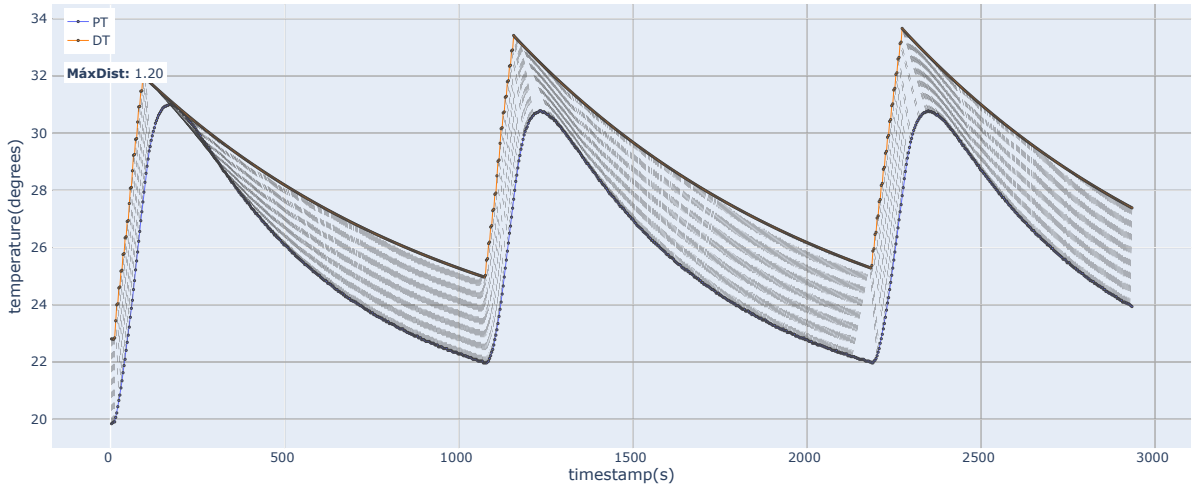

Fig. 14. Alignment of 2-P Model for scenario Ht3Hg2 with MAD 1.2.

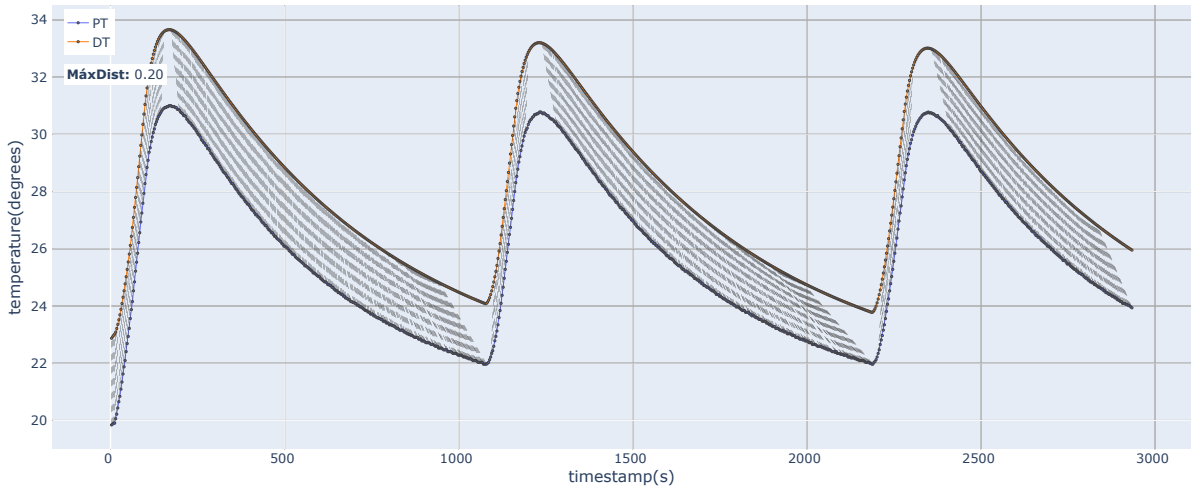

Fig. 15. Alignment of 4-P Model for scenario Ht3Hg2 with MAD 0.2.

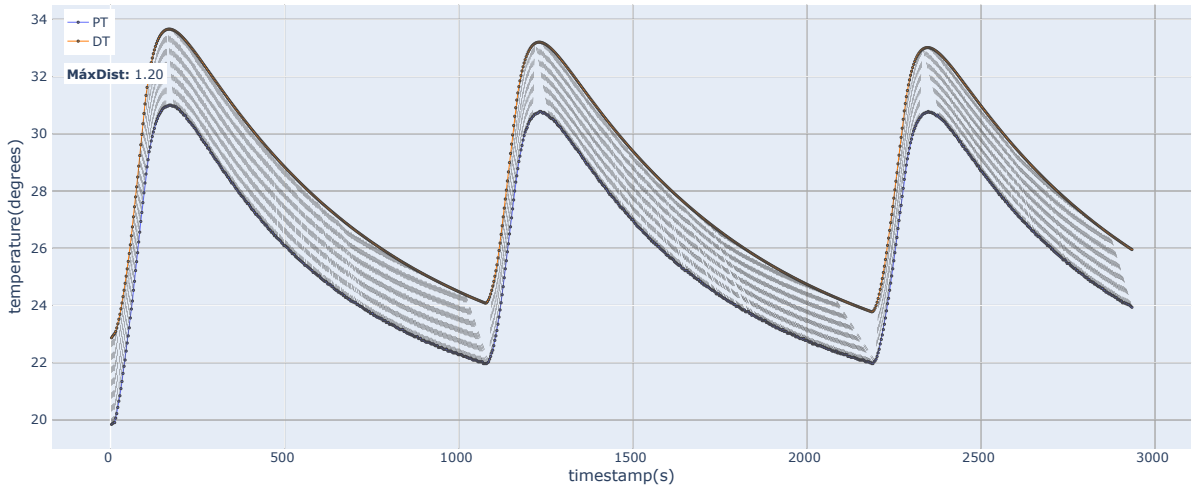

Fig. 16. Alignment of 4-P Model for scenario Ht3Hg2 with MAD 1.2.

TABLE IV. FIDELITY RESULTS FOR SCENARIO HT30HG20.

mad_temperature(degrees)	% matched	2-P Model Frèchet	Avg. Euclidean	% matched	4-P Model Frèchet	Avg. Euclidean
0.2	54.1481	0.1796	0.0428	89.6521	0.1956	0.0352
0.4	57.8055	0.3901	0.0833	93.8448	0.347	0.0585
0.6	62.9795	0.5982	0.13	95.7181	0.5645	0.0776
0.8	67.4398	0.7778	0.1729	96.521	0.658	0.089
1	72.2569	0.8907	0.2281	97.0562	0.658	0.1006
1.2	75.3791	0.9124	0.2801	97.5914	0.658	0.114
1.4	77.2525	0.9747	0.3199	98.0375	0.658	0.1274
1.6	80.1963	1.0601	0.3741	98.2159	0.658	0.1343
1.8	81.8912	1.2247	0.4228	98.3943	0.658	0.141
2	83.7645	1.4437	0.4748	98.6619	0.658	0.1528
2.2	85.1918	1.6098	0.5189	98.8403	0.658	0.1605
2.4	85.9946	1.5929	0.5529	98.8403	0.658	0.1611
2.6	87.0651	1.5929	0.5898	98.9295	0.658	0.1661
2.8	87.868	1.5929	0.6211	99.0187	0.658	0.1717
3	88.5816	1.6098	0.6546	99.0187	0.658	0.1717
3.2	88.9384	1.6307	0.6717	99.1079	0.658	0.1777
3.4	89.2953	1.6932	0.6908	99.1079	0.658	0.1777
3.6	89.7413	1.8966	0.7148	99.1079	0.658	0.1777
3.8	90.0981	1.8966	0.7346	99.1079	0.658	0.1777
4	90.6334	2.0155	0.7663	99.1079	0.658	0.1777
4.2	90.9902	2.0682	0.7865	99.1079	0.658	0.1777
4.4	91.5254	2.1312	0.8217	99.1971	0.6895	0.1843
4.6	91.8822	2.2557	0.8437	99.1971	0.6895	0.1843
4.8	92.5067	2.2872	0.8859	99.3756	0.721	0.2003
5	93.1311	2.3497	0.9291	99.3756	0.721	0.2003

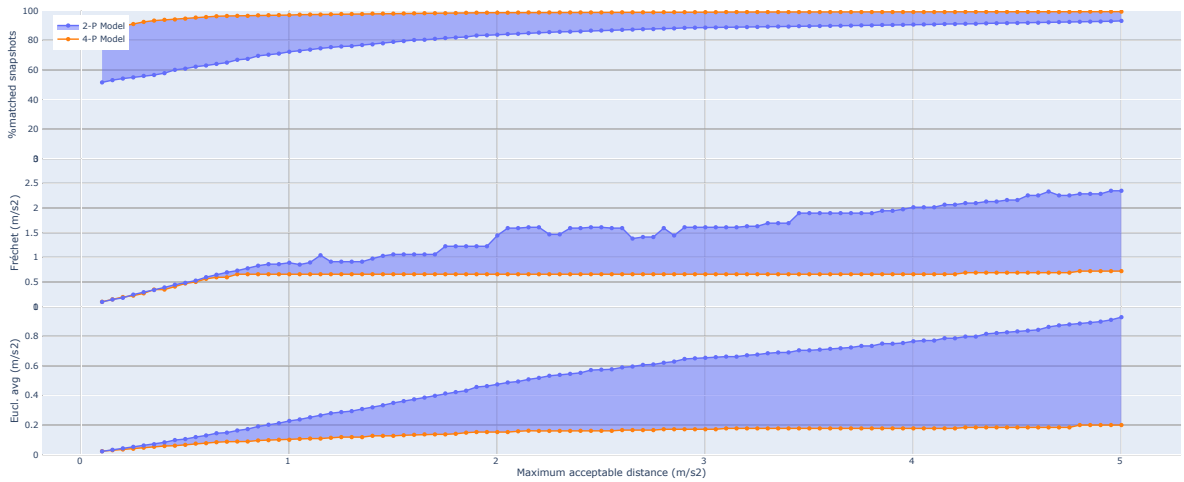

Fig. 17. Statistical comparison of the 2-P Model against the 4-P Model for scenario Ht30Hg20.

2) Heating time 30 s - Heating gap 20 s (Ht30Hg20)

To illustrate this scenario, alignment figures have been included for both models with two Mean Absolute Difference (MAD) values (0.2 and 1.2). The alignments for the 2-Parameter model can be found in Figures 18 and 19, while for the 4-Parameter model, they are shown in Figures 20 and 21. In order to better visualize the alignment of the points, a constant offset of 3 degrees has been added to all snapshots of the digital twin, preventing overlapping of the graphs.

Looking at Figures 18 and 20, we can see that the cooling and heating frequency is now faster. This is due to the heater being kept on for a longer period of time, allowing the system to reach the temperature more quickly. As a result, the temperature transitions are different from those in the previous scenario. In Figure 18, we can observe that the 2-Parameter model is unable to accurately emulate this faster growth during the the 30-second heating and smooth cooling during the 20-second shutdown of the system before it turns back on. We can see spikes resembling a sawtooth pattern, which does not resemble the real behavior. However, in Figure 20, the 4-Parameter model is capable of emulating these changes, and the transitions under these conditions are more accurate. This alignment achieves 89% of aligned points compared to 54% obtained by the other model (see Table IV). This trend persists as we increase the MAD, and although the 2-Parameter model can align a greater number of points, it reaches a plateau at around 90% of the points at the cost of significantly increasing the Frèchet and Euclidean distances, exceeding 2 degrees and nearly 1 degree, respectively. In contrast, the other model keeps both values below 0.7 and 0.2, respectively. This contrast can be observed in Figure 19 and Figure 21: more points are aligned, but it is clear that the 4-Parameter model achieves nearly perfect alignment, while the other model does not.

Based on the results, we can conclude that the 4-Parameter model performs significantly better in a scenario where temperature changes are faster compared to the previous scenario. The 2-Parameter model is unable to accurately emulate the smooth changes during longer periods of heating and cooling. Therefore, depending on the specific scenario, we can decide which model to employ for emulating the system's temperature.

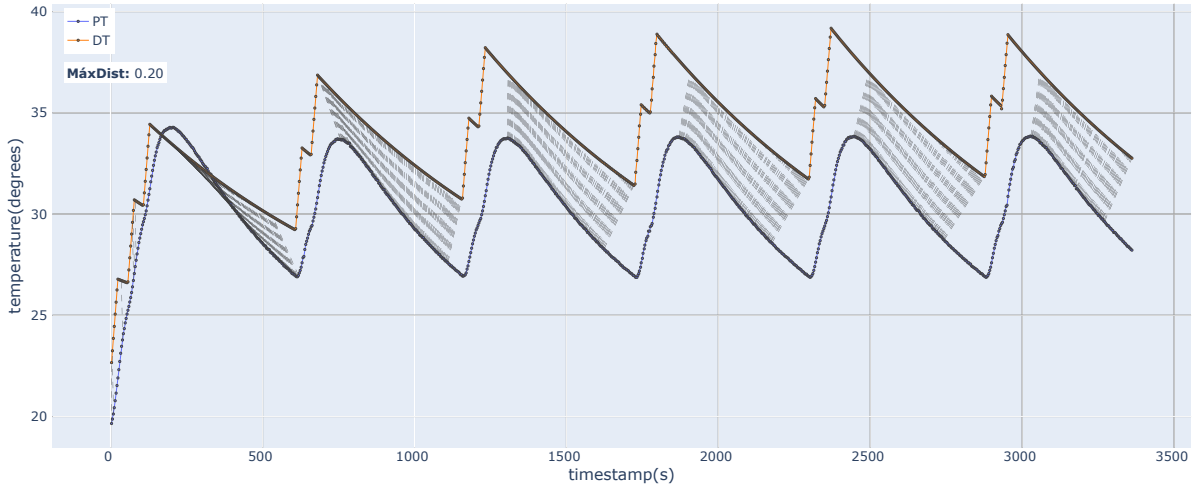

Fig. 18. Alignment of 2-P Model for scenario Ht30Hg20 with MAD 0.2.

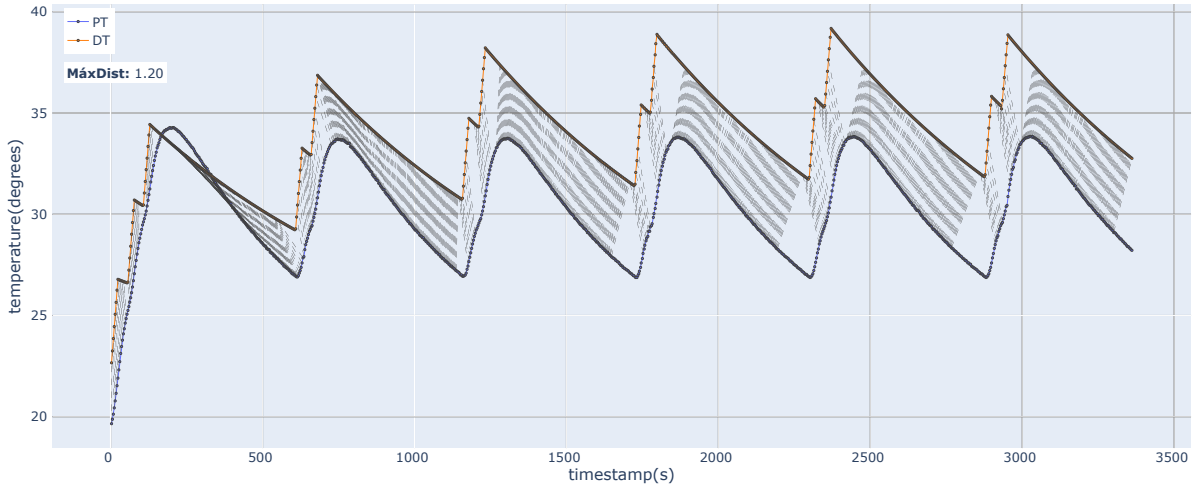

Fig. 19. Alignment of 2-P Model for scenario Ht30Hg20 with MAD 1.2.

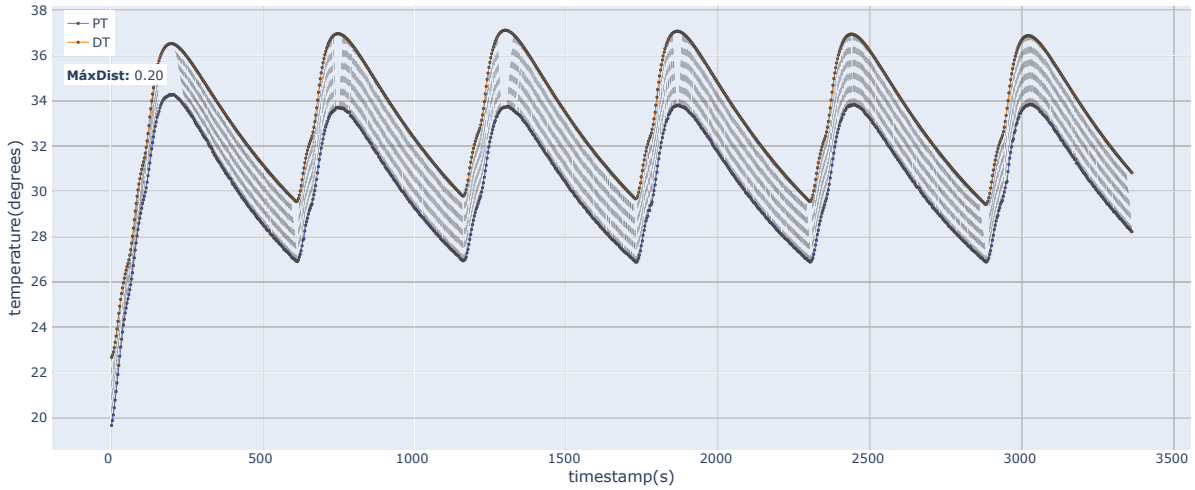

Fig. 20. Alignment of 4-P Model for scenario Ht30Hg20 with MAD 0.2.

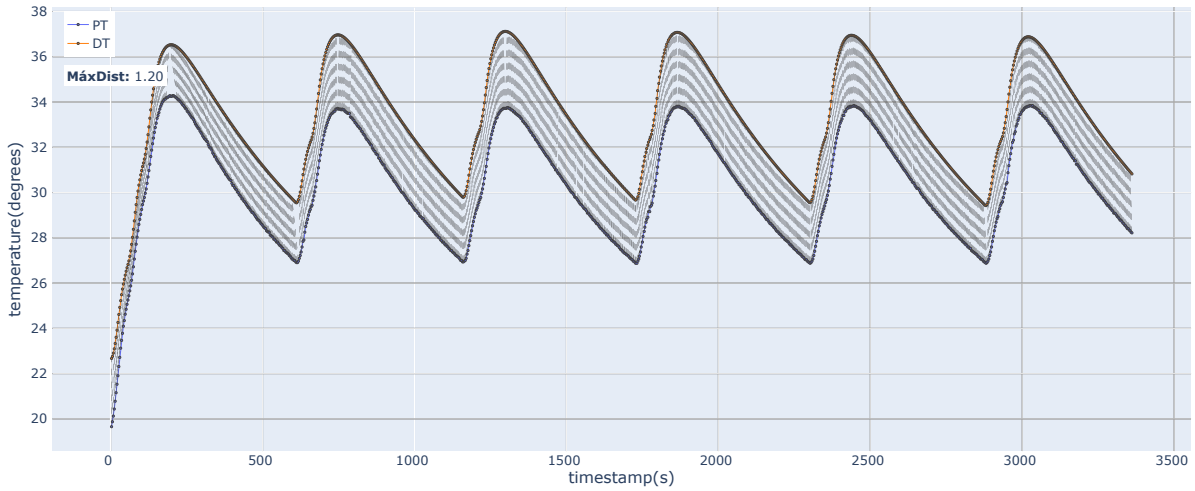

Fig. 21. Alignment of 4-P Model for scenario Ht30Hg20 with MAD 1.2.

II Acknowledgments

Paula *Thanks to Claudio Gomes and his team for the data*

References

[1] H. Feng, C. Gomes, C. Thule, K. Lausdahl, M. Sandberg, and P. G. Larsen, "The incubator case study for digital twin engineering," 2021.

[2] "Example digital twin: The incubator," 2023. [Online]. Available: https://github.com/INTO-CPS-Association/example_digital-twin_incubator

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