

Technical Report: Measuring fidelity of DTs Robotic Arm

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

Manuel Wimmer
CDL-Mint
Johannes Kepler University (Austria)

Contents

I	Niryo Ned in the CLD-Mint	1
I-A	System description	1
I-B	Scenarios	1
I-C	Gap Tuning	2
	I-C1 Simple Gap	2
	I-C2 Affine Gap	2
I-D	Fidelity assessment	7
	I-D1 Scenario I	8
	I-D2 Scenario II	11
	I-D3 Scenario III	14
	I-D4 Scenario IV	17
	I-D5 Scenario V	20
II	Acknowledgments	23
References		23

I Niryo Ned in the CLD-Mint

A. System description

The Christian Doppler Laboratory for Model-Integrated Smart Production (CDL-MINT) at Johannes Kepler University (Linz, Austria) is currently focusing a part of its research on the operation and validation of Digital Twins (DT) within the manufacturing context. To ensure the effectiveness of their new researched techniques before implementing them in real production environments, they are building a Digital Twin System of a robotic arm, simulating the arm models employed in manufacturing processes. The robotic arm used for this purpose is the Niryo Ned robot arm.

The Nyrio Ned is a lightweight and versatile 6-axis robotic manipulator. Its mechanical structure enables agile and portable operation. With precise servo motor actuation, it delivers accurate and smooth movements within a reach of up to 70 cm. The arm supports various end effectors and grippers, allowing for task-specific functionalities and easy interchangeability. It can be controlled through a user-friendly web interface, ROS integration, and Python programming.

The digital replica for this arm that they plan to use as Digital Twin is... **Paula**
Need further information about this!

Their objective is to develop a Digital Twin (DT) based on these two systems. To achieve this, they need to evaluate the extent to which the digital replica accurately represents the physical system.

B. Scenarios

Paula We have five scenarios in the GitHub, could I get a high level description of each one of those? Do they cover a good range of behaviors?

Paula Do we want a set of samples of the same scenario to study variability as we did in the Elevator use case?



Fig. 1. Niryo Ned Arm **Paula** any photos in the CDL-Mint lab?

C. Gap Tuning

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, 0.22]	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

1) Simple Gap

2) Affine Gap

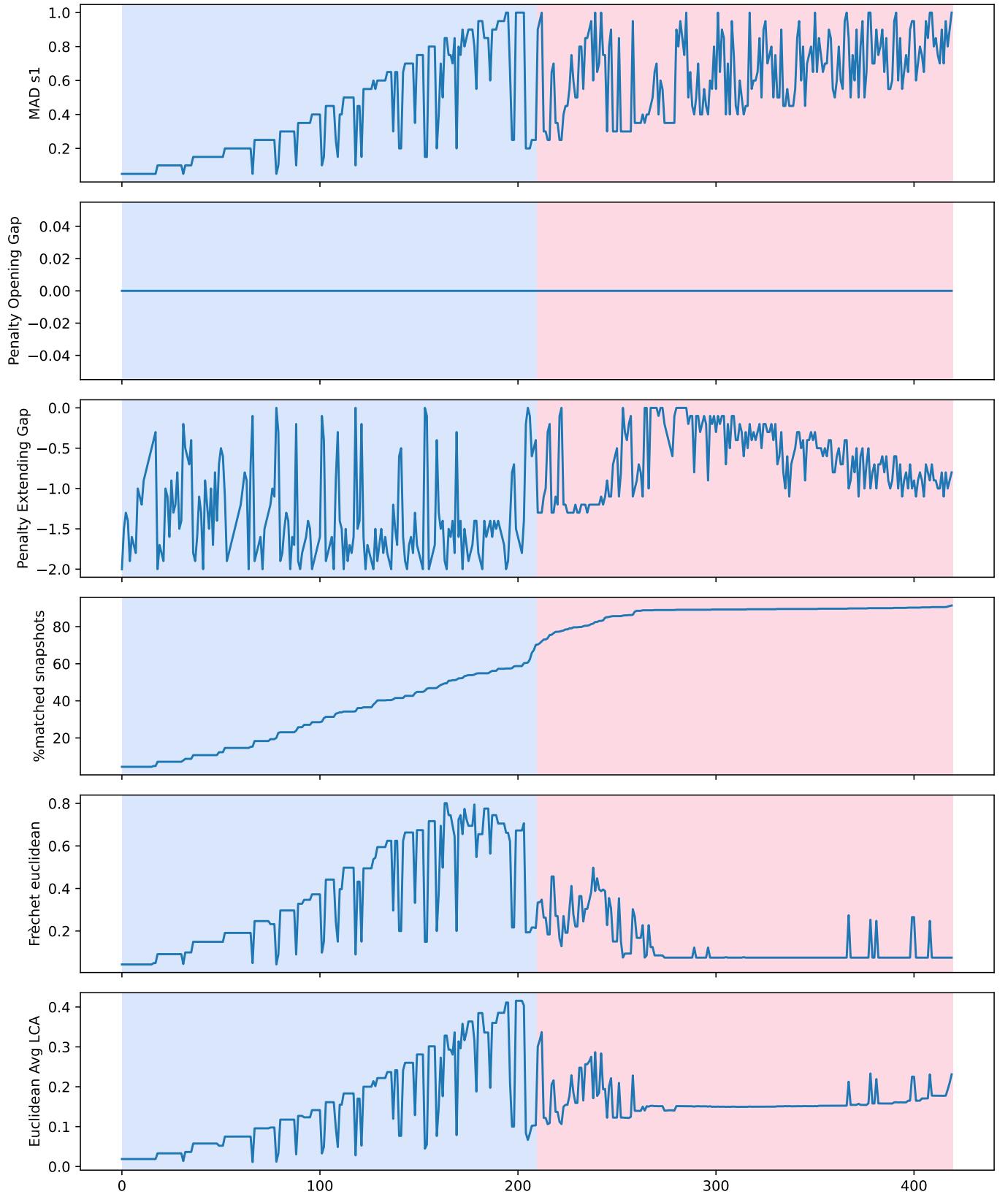


Fig. 2. Analysis of alignment statistics for simple gap in ascending order by % of matched snapshots.

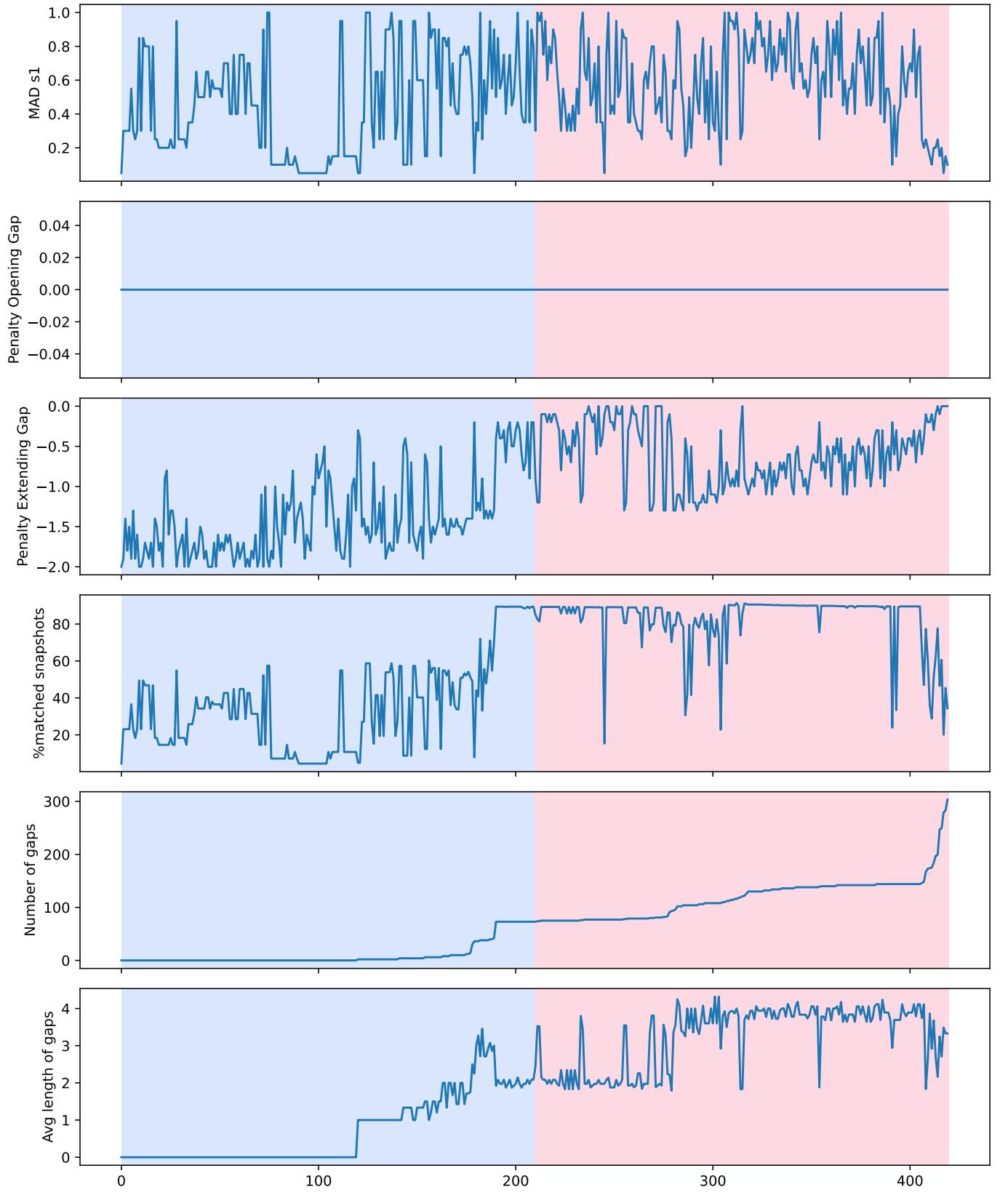


Fig. 3. Analysis of alignment statistics for simple gap in ascending order by number of gaps.

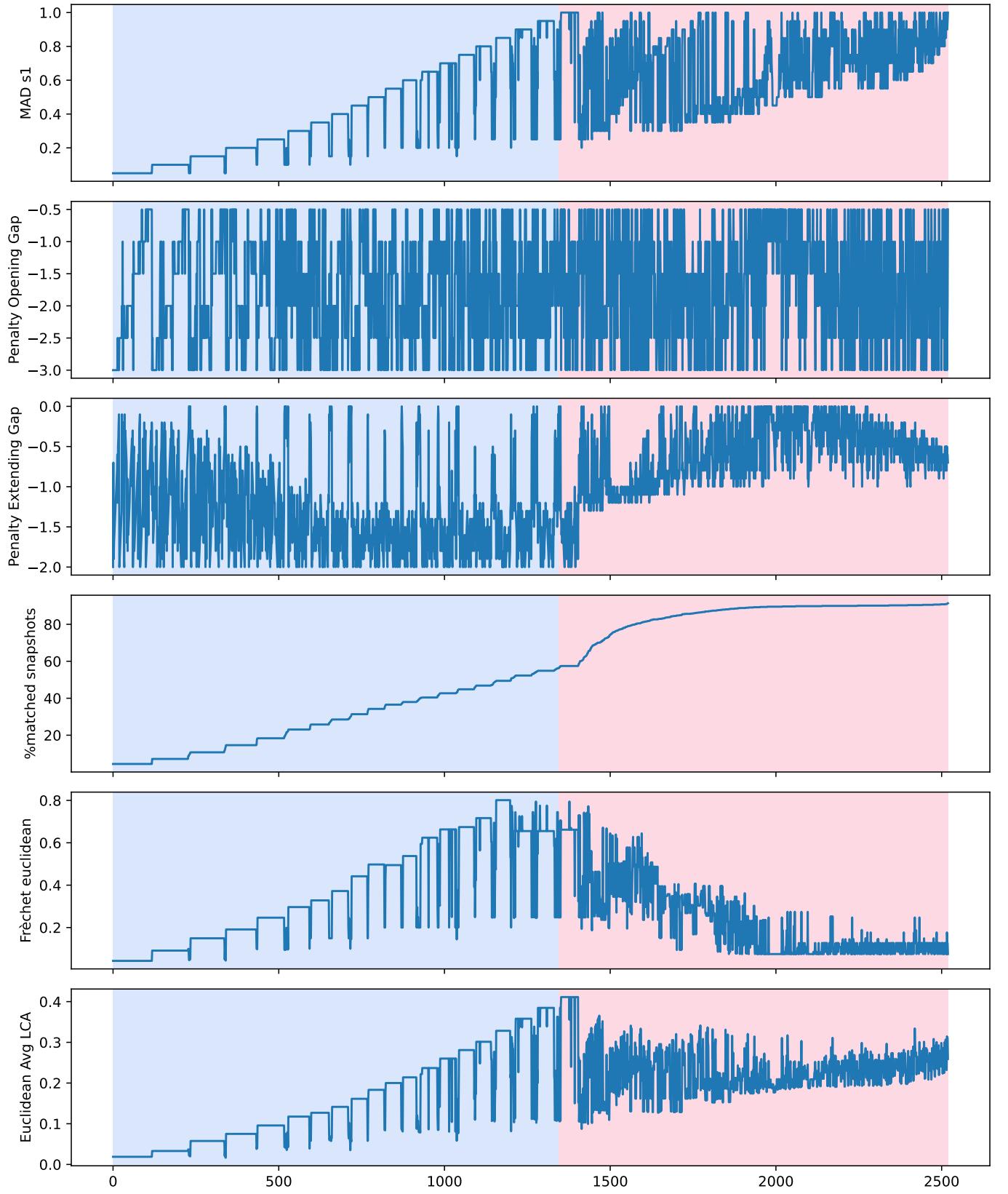


Fig. 4. Analysis of alignment statistics for affine gap in ascending order by % of matched snapshots.

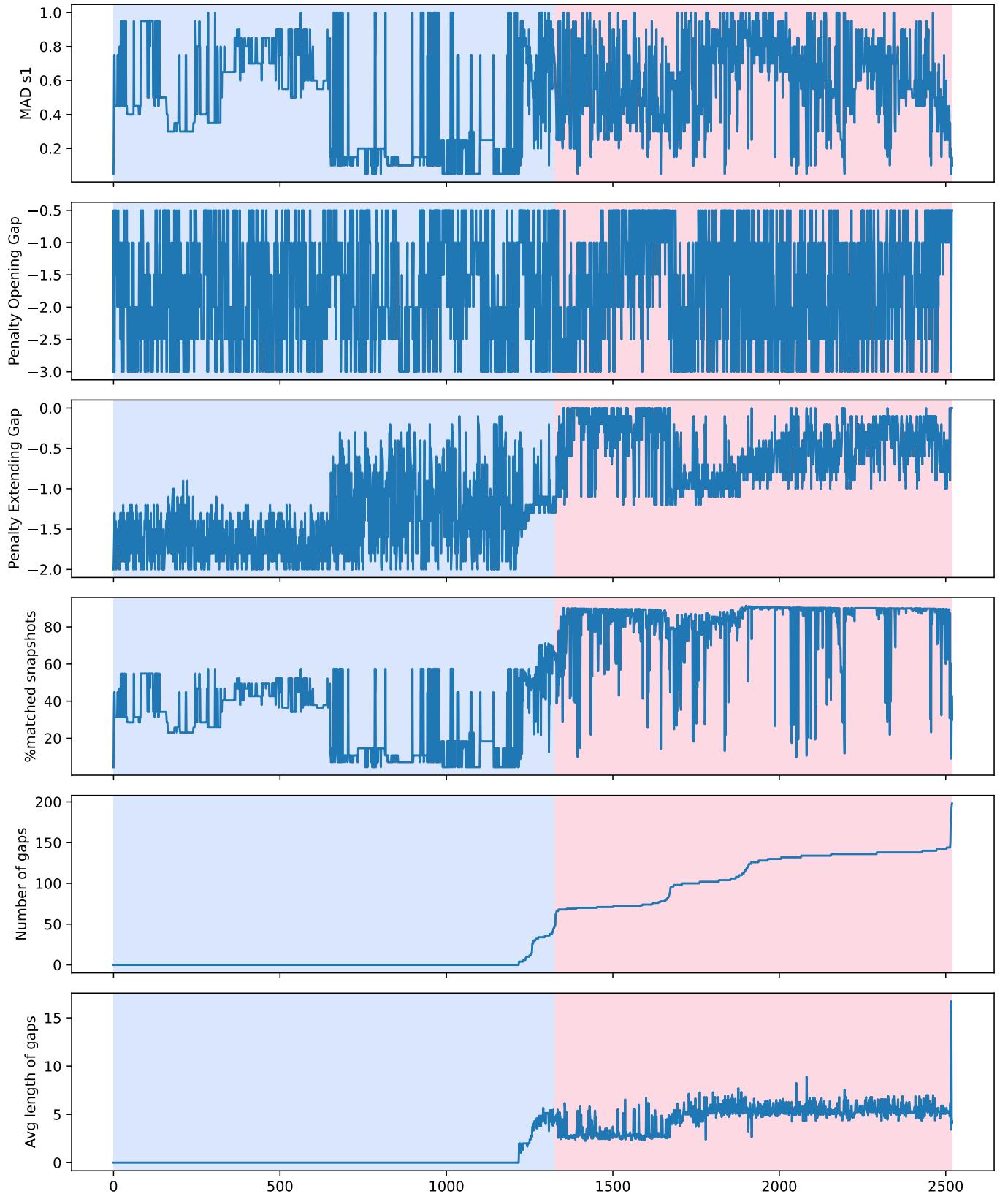


Fig. 5. Analysis of alignment statistics for affine gap in ascending order by number of gaps.

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 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.
- **Low Complexity Areas Weight (LCAW):** 0.005
- **Affine Gap Weights:** (-1, -0.1)

The specific and detailed guidelines on how to set the configuration values for LCAW and Affine Gap are available in the previous section. 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.

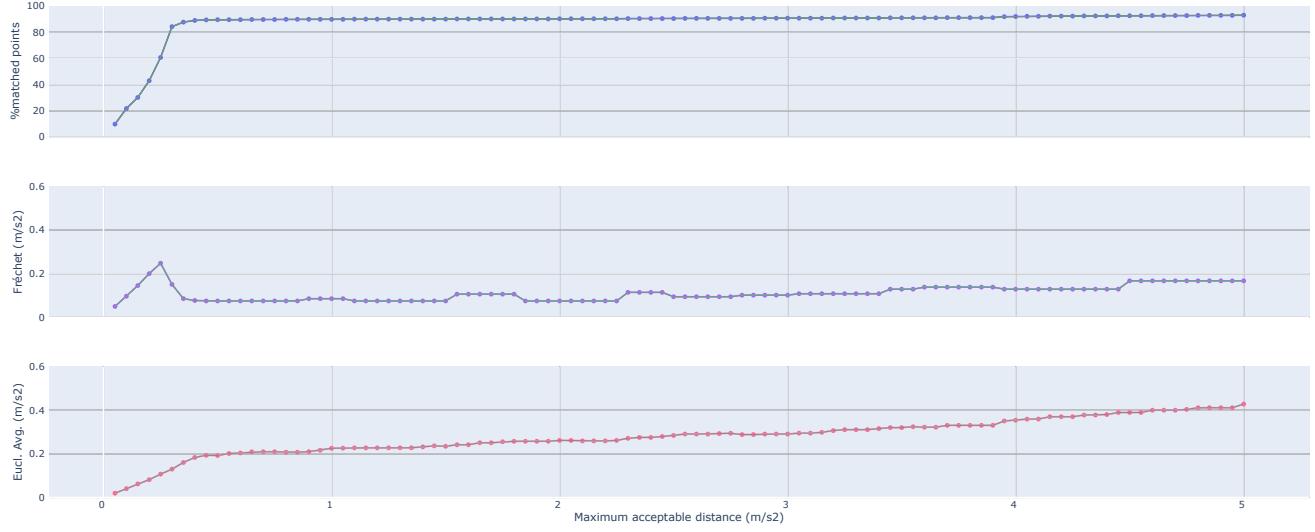


Fig. 6. Analysis of fidelity metrics based on MAD for the scenario 1.

1) Scenario I

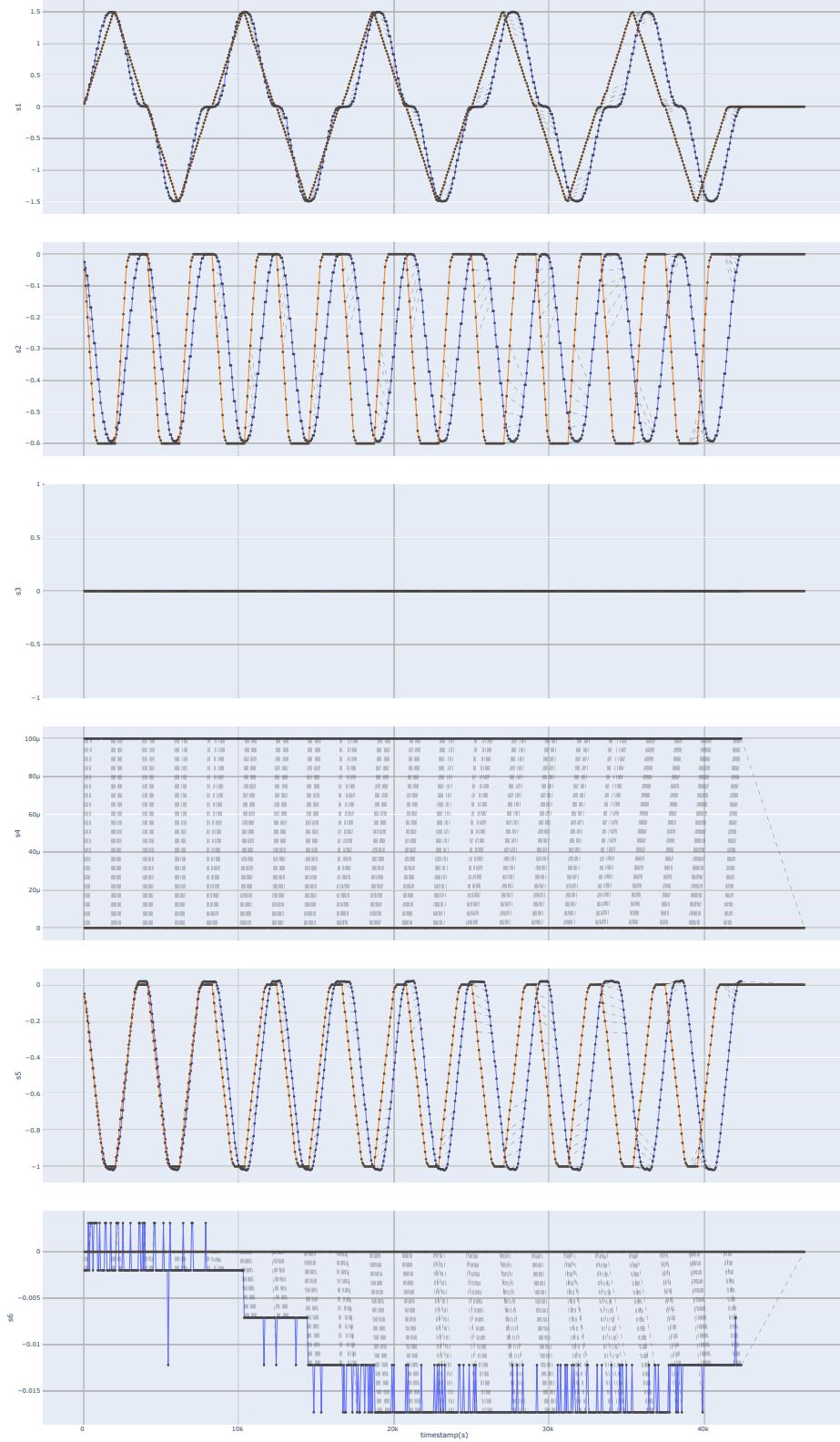


Fig. 7. Alignment of the scenario 1 with a MAD of 0.15 degrees

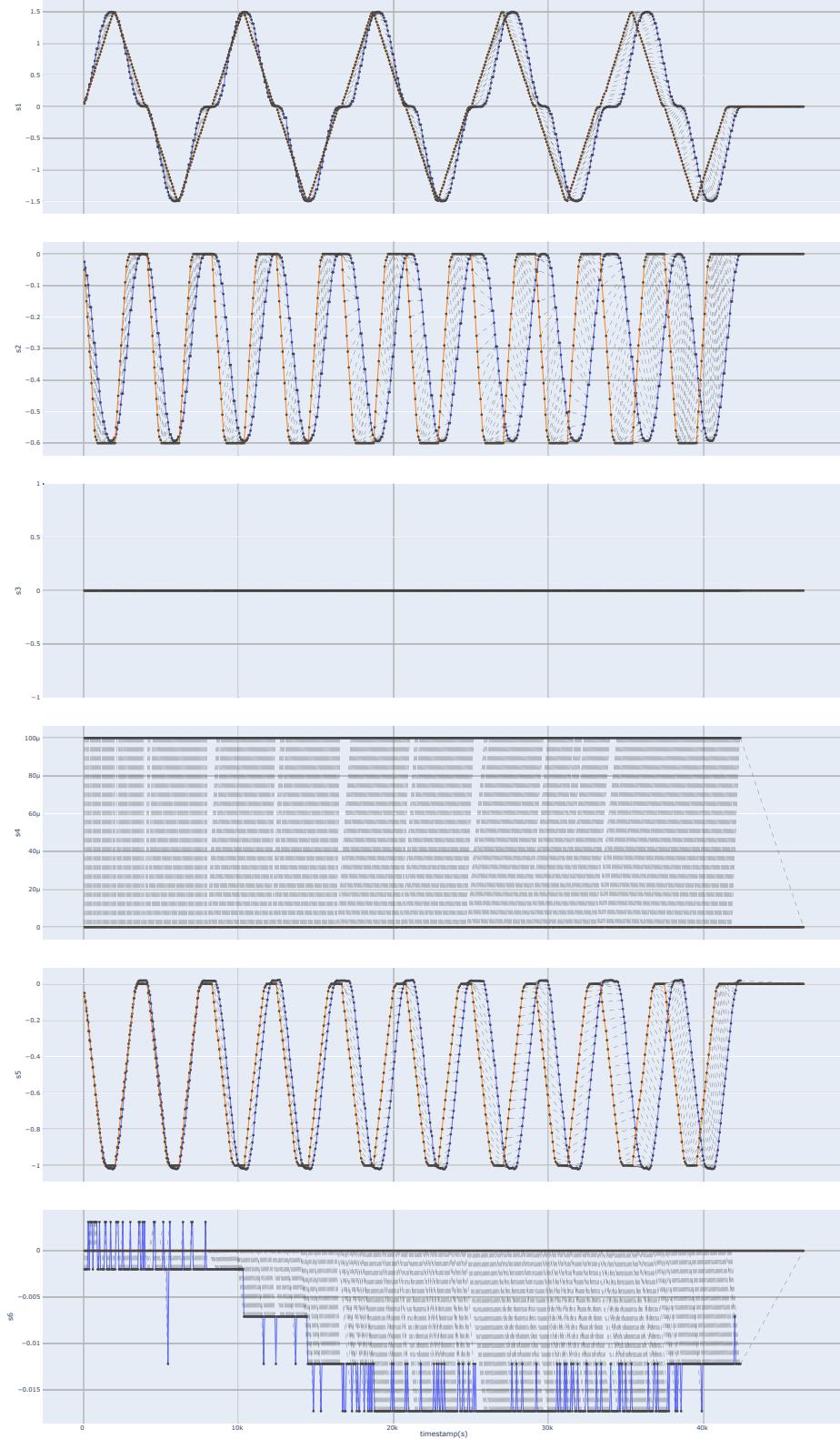


Fig. 8. Alignment of the scenario 1 with a MAD of 1.15 degrees

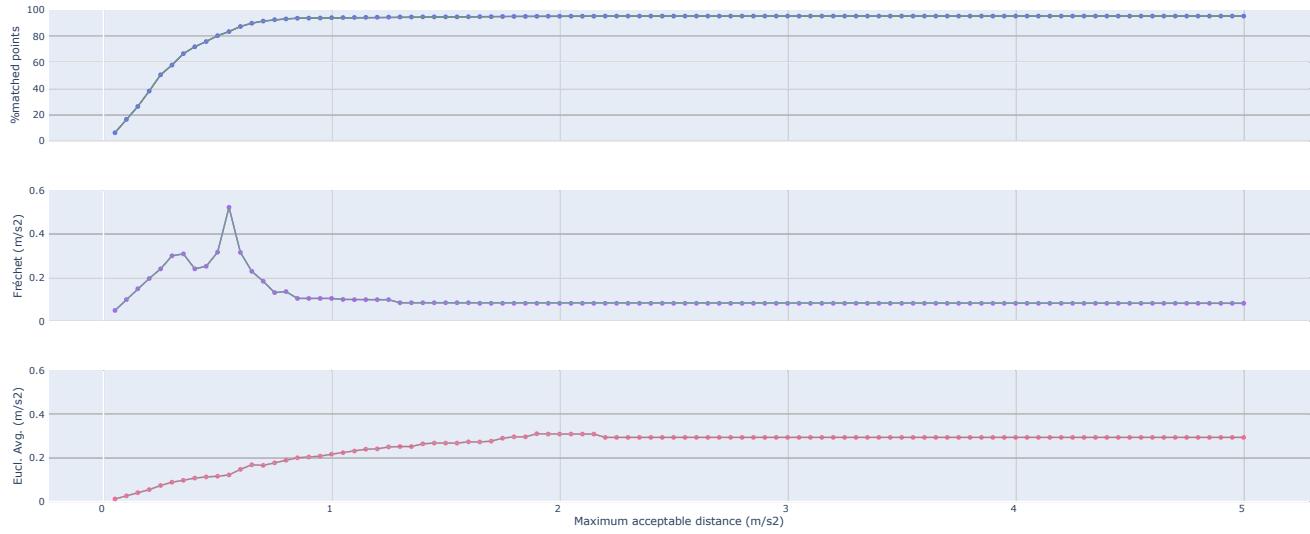


Fig. 9. Analysis of fidelity metrics based on MAD for the scenario 2.

2) Scenario II

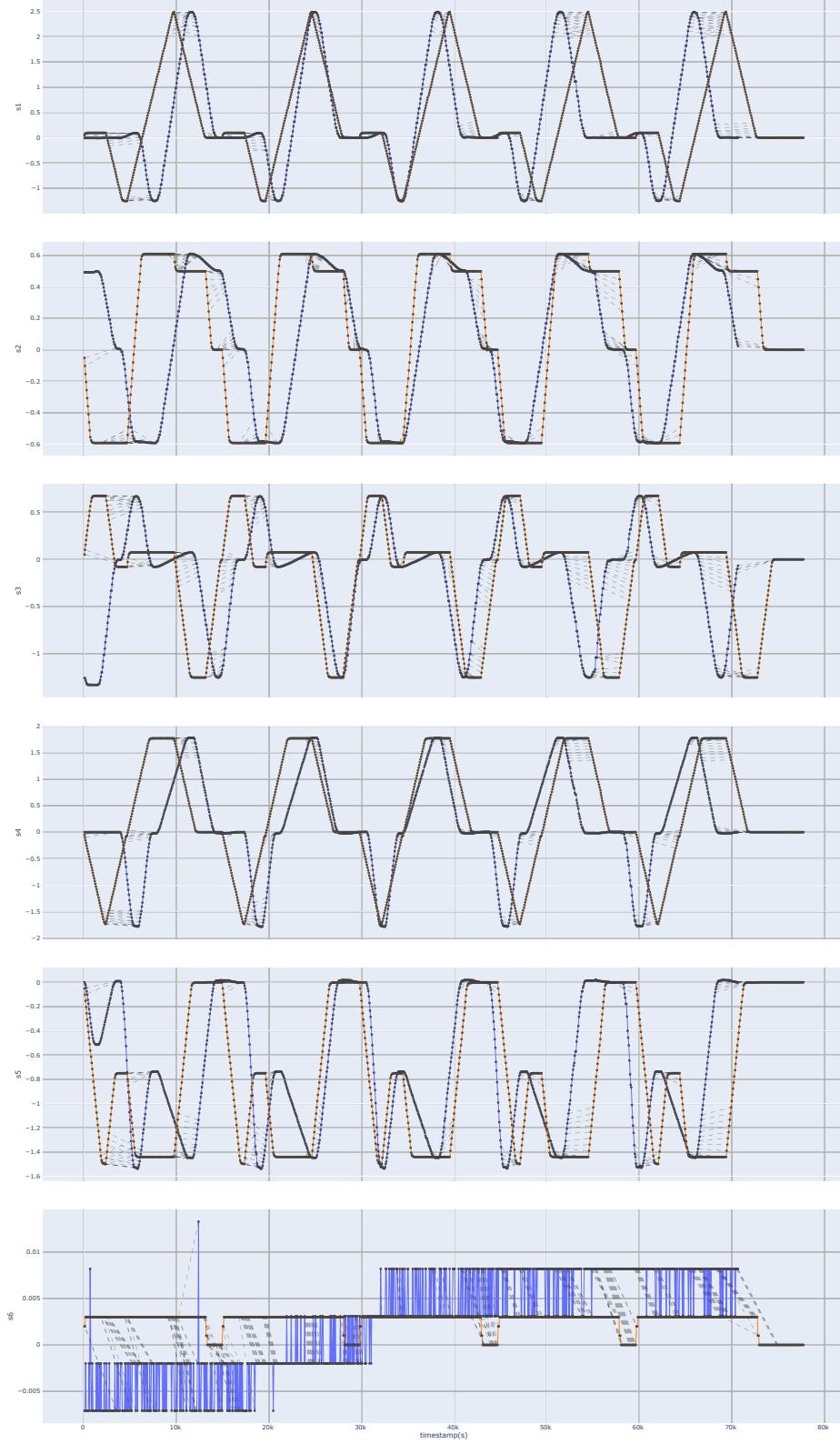


Fig. 10. Alignment of the scenario 2 with a MAD of 0.15 degrees

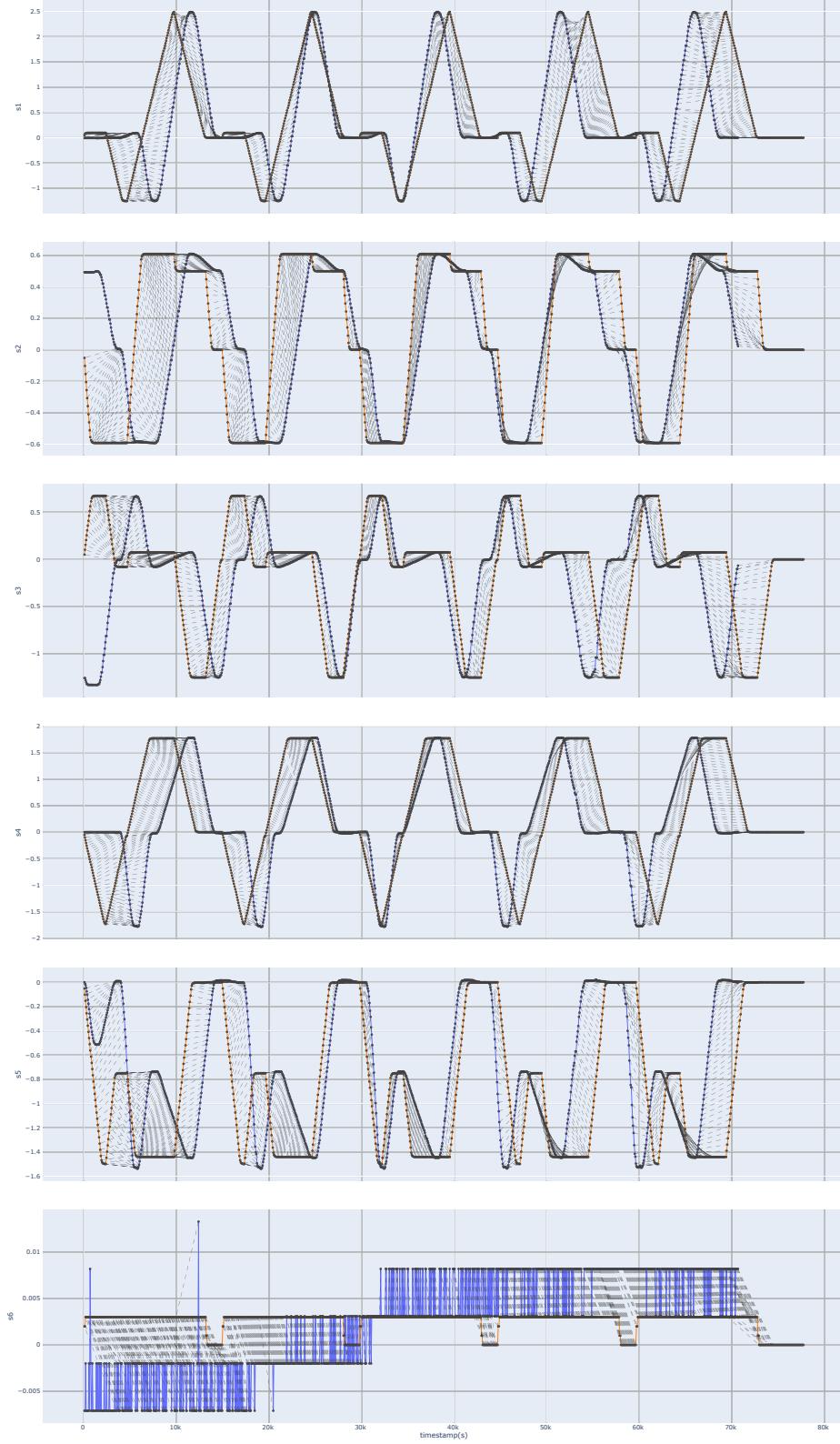


Fig. 11. Alignment of the scenario 2 with a MAD of 1.15 degrees

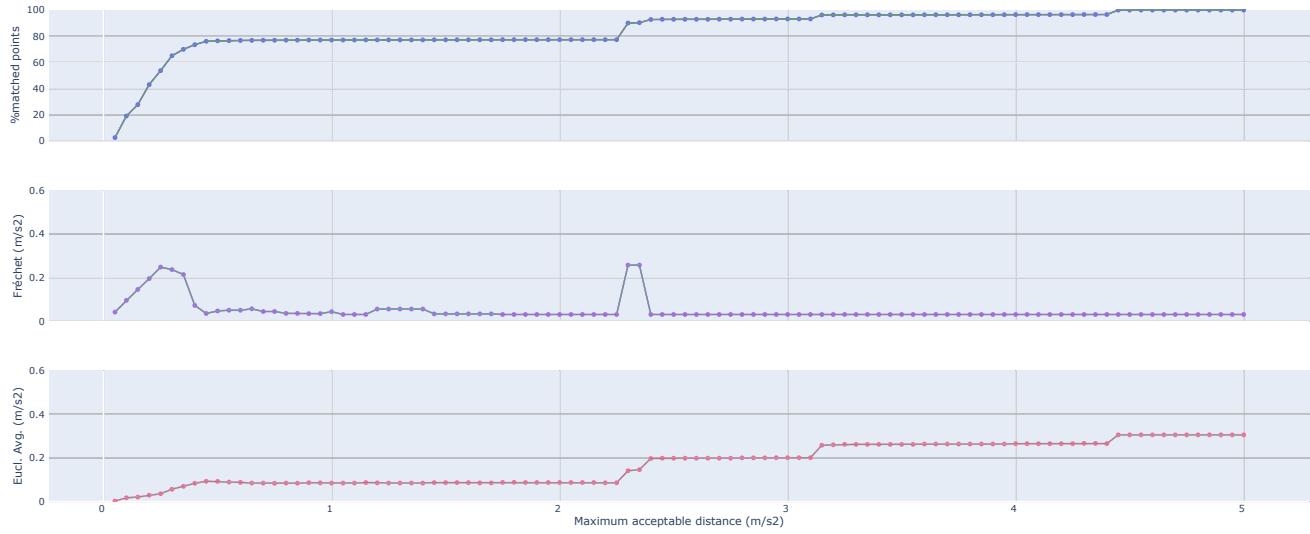


Fig. 12. Analysis of fidelity metrics based on MAD for the scenario 3.

3) Scenario III

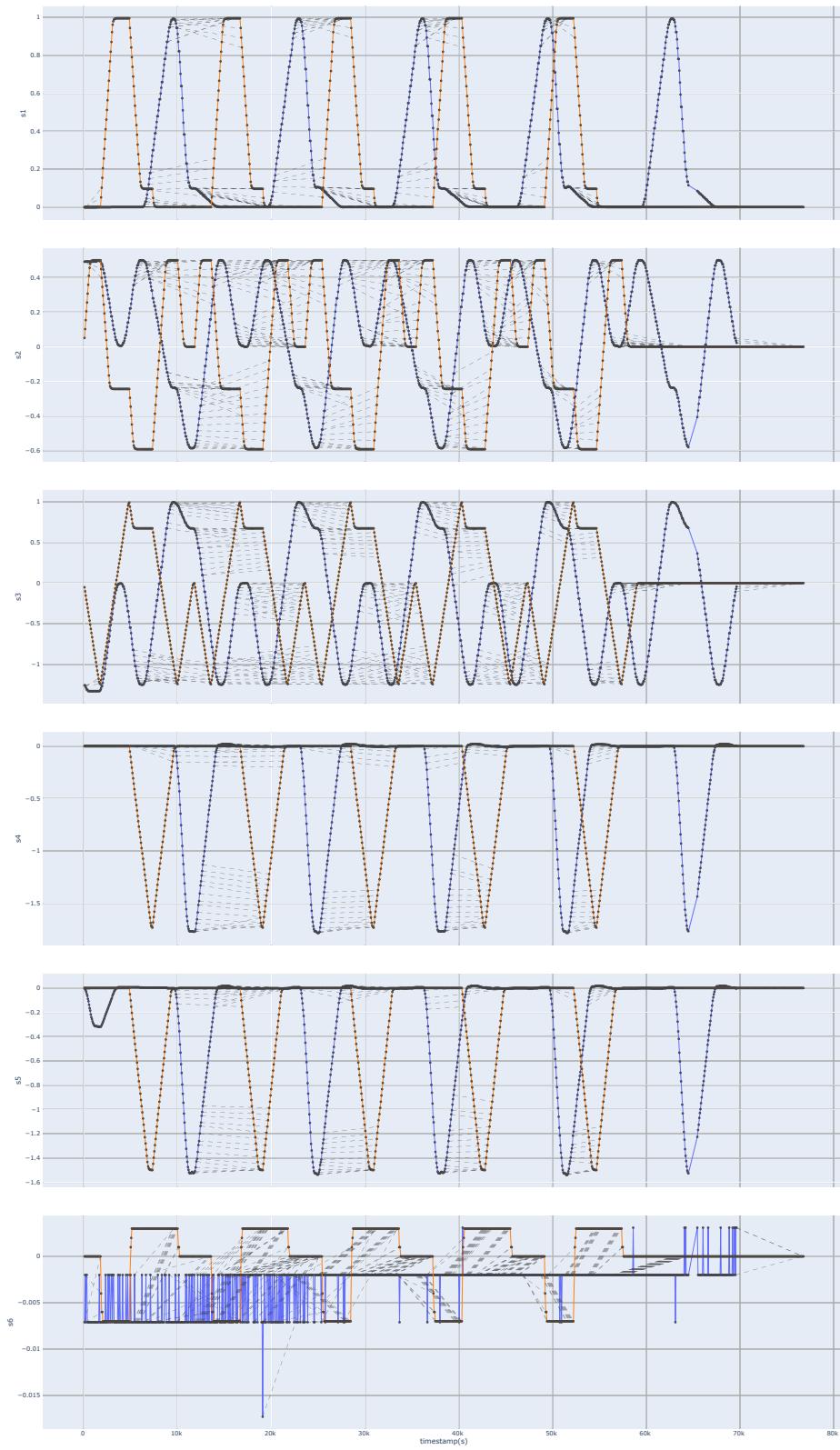


Fig. 13. Alignment of the scenario 2 with a MAD of 0.15 degrees

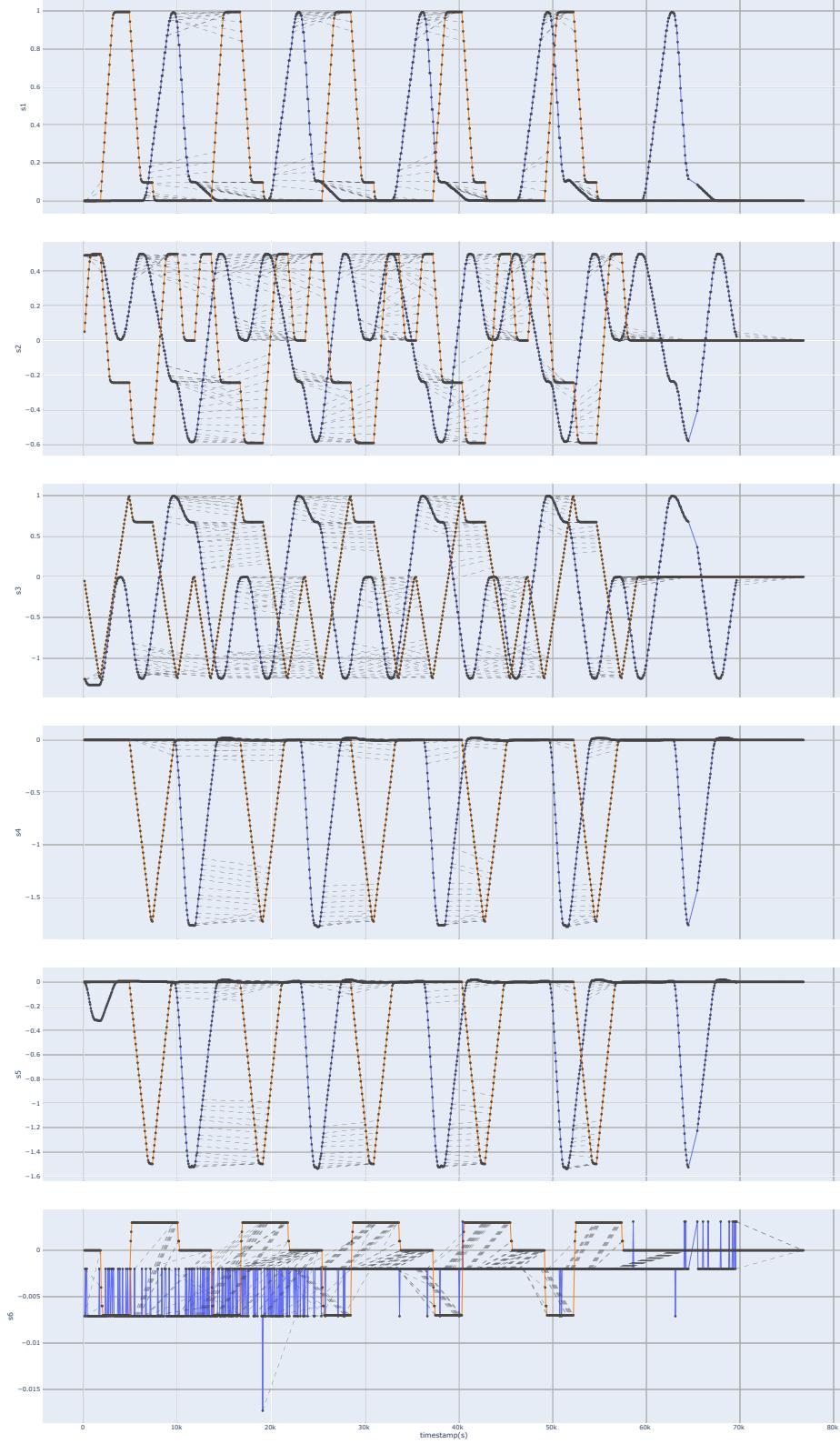


Fig. 14. Alignment of the scenario 2 with a MAD of 1.15 degrees

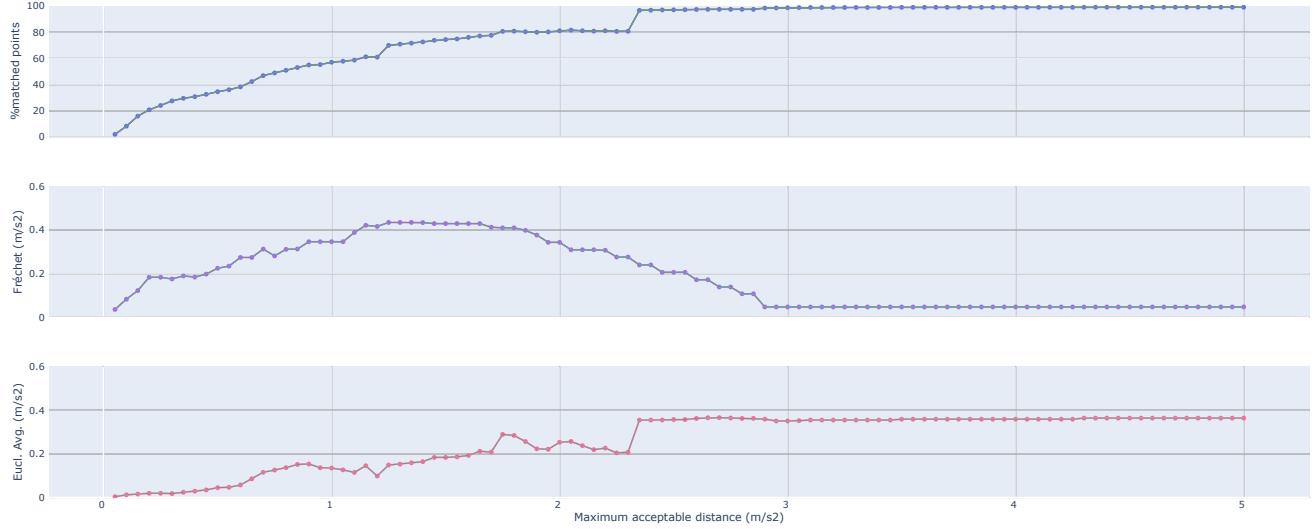


Fig. 15. Analysis of fidelity metrics based on MAD for the scenario 4.

4) Scenario IV

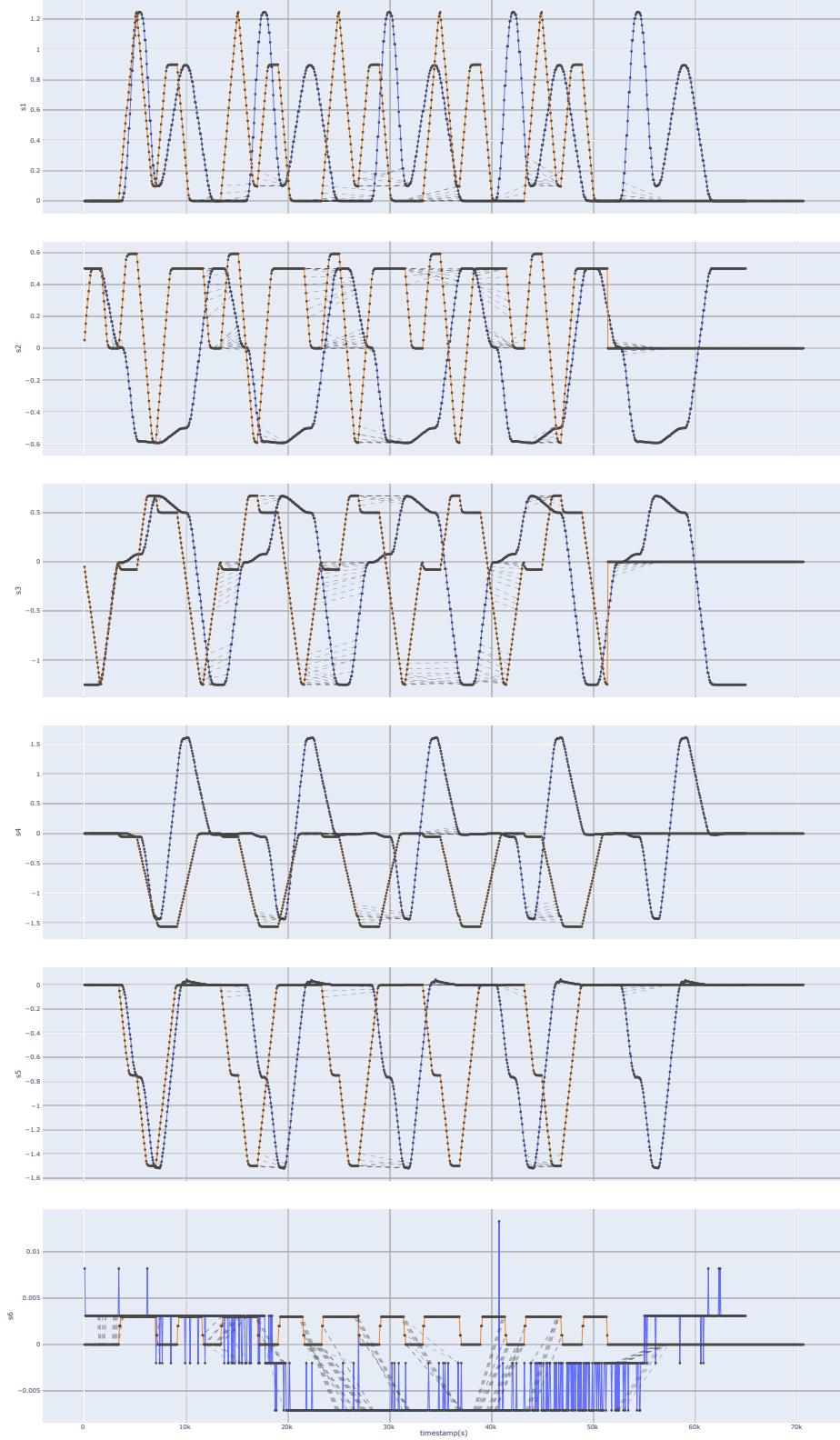


Fig. 16. Alignment of the scenario 4 with a MAD of 0.15 degrees

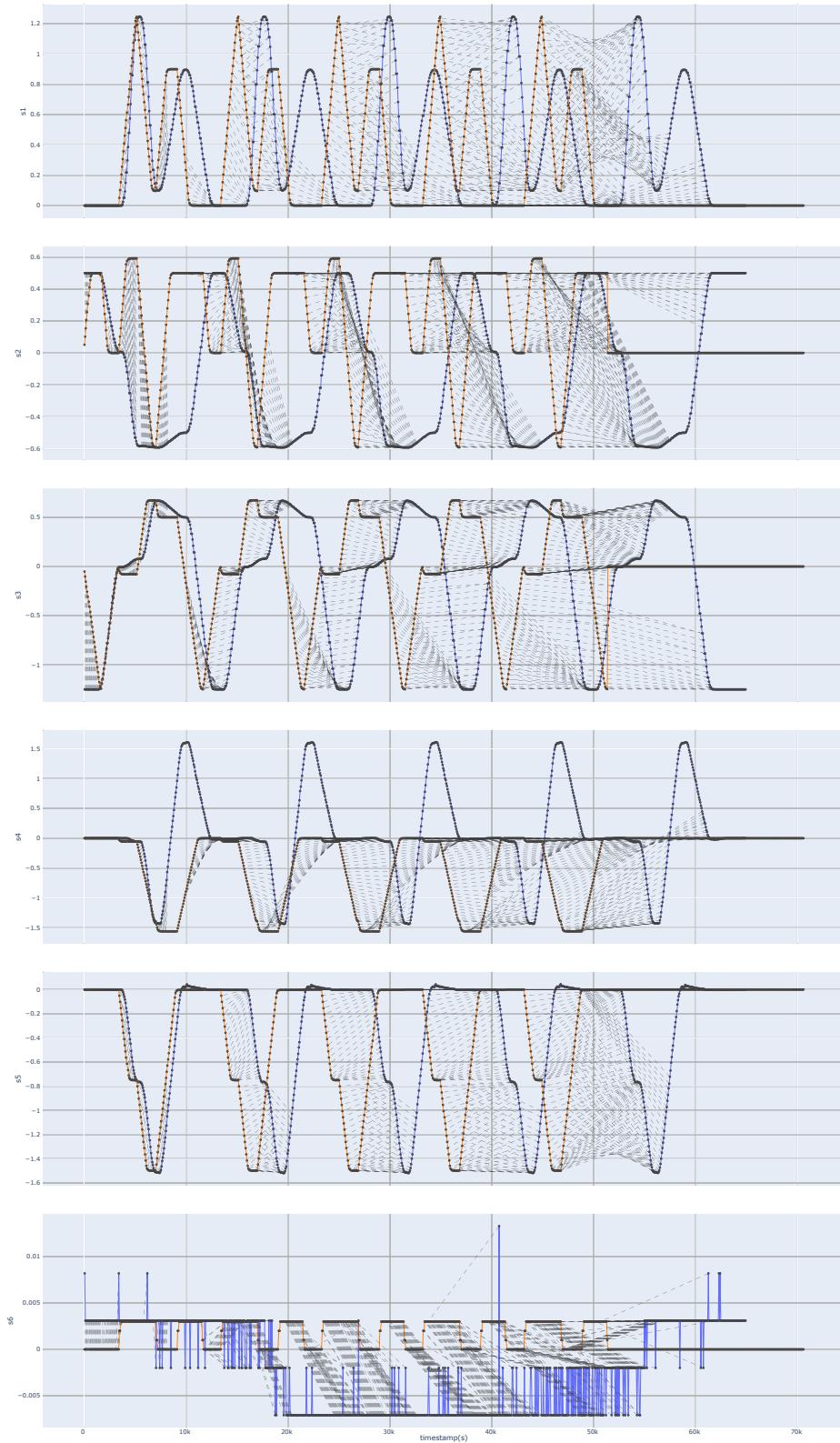


Fig. 17. Alignment of the scenario 4 with a MAD of 1.15 degrees

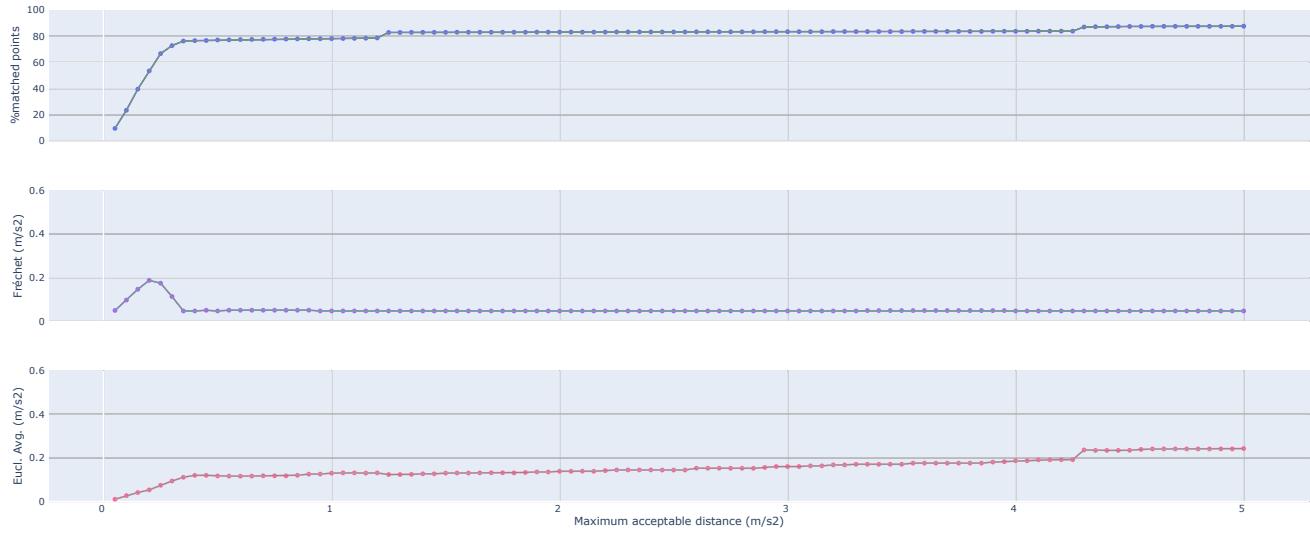


Fig. 18. Analysis of fidelity metrics based on MAD for the scenario 5.

5) Scenario V

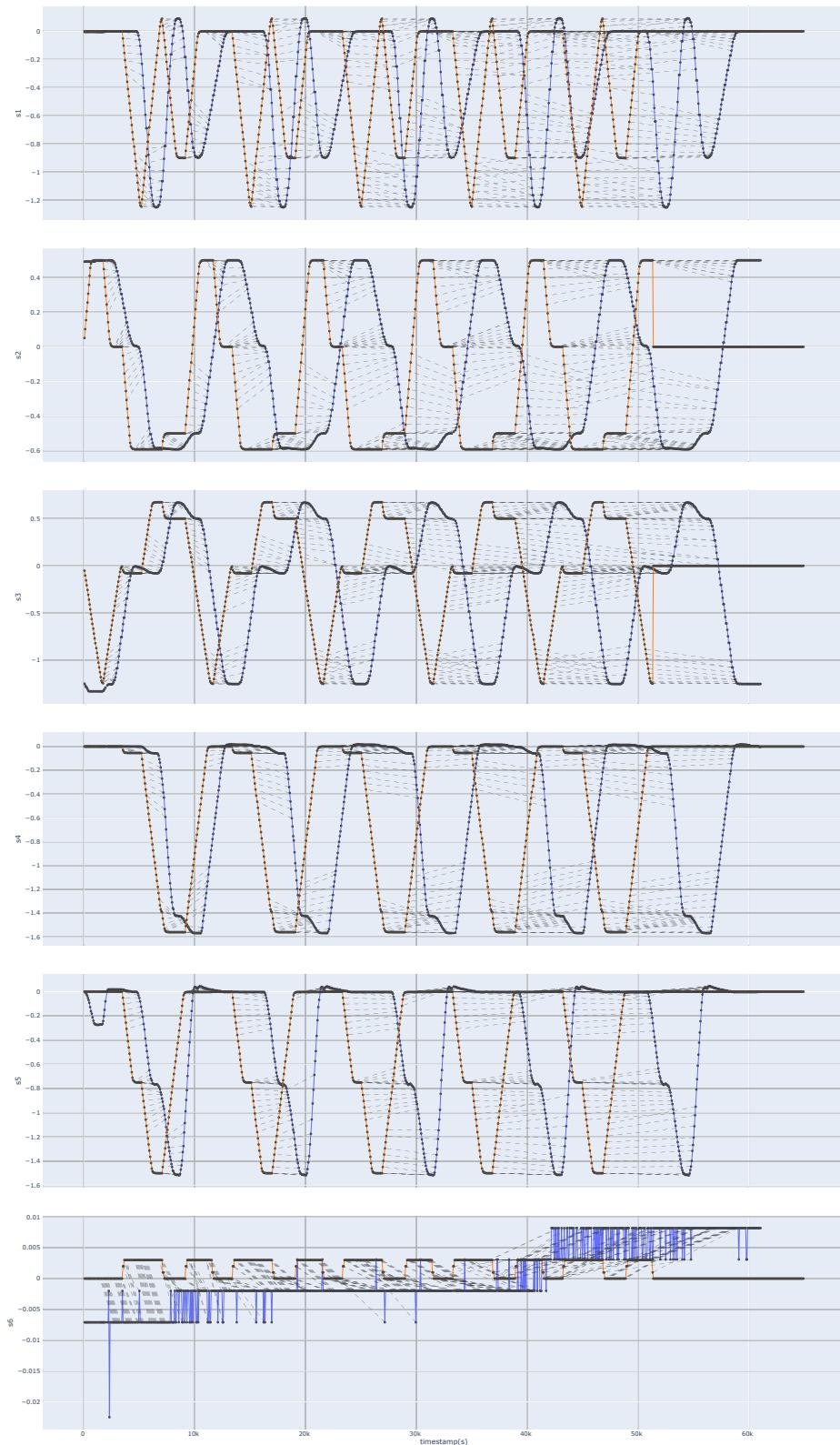


Fig. 19. Alignment of the scenario 4 with a MAD of 0.15 degrees

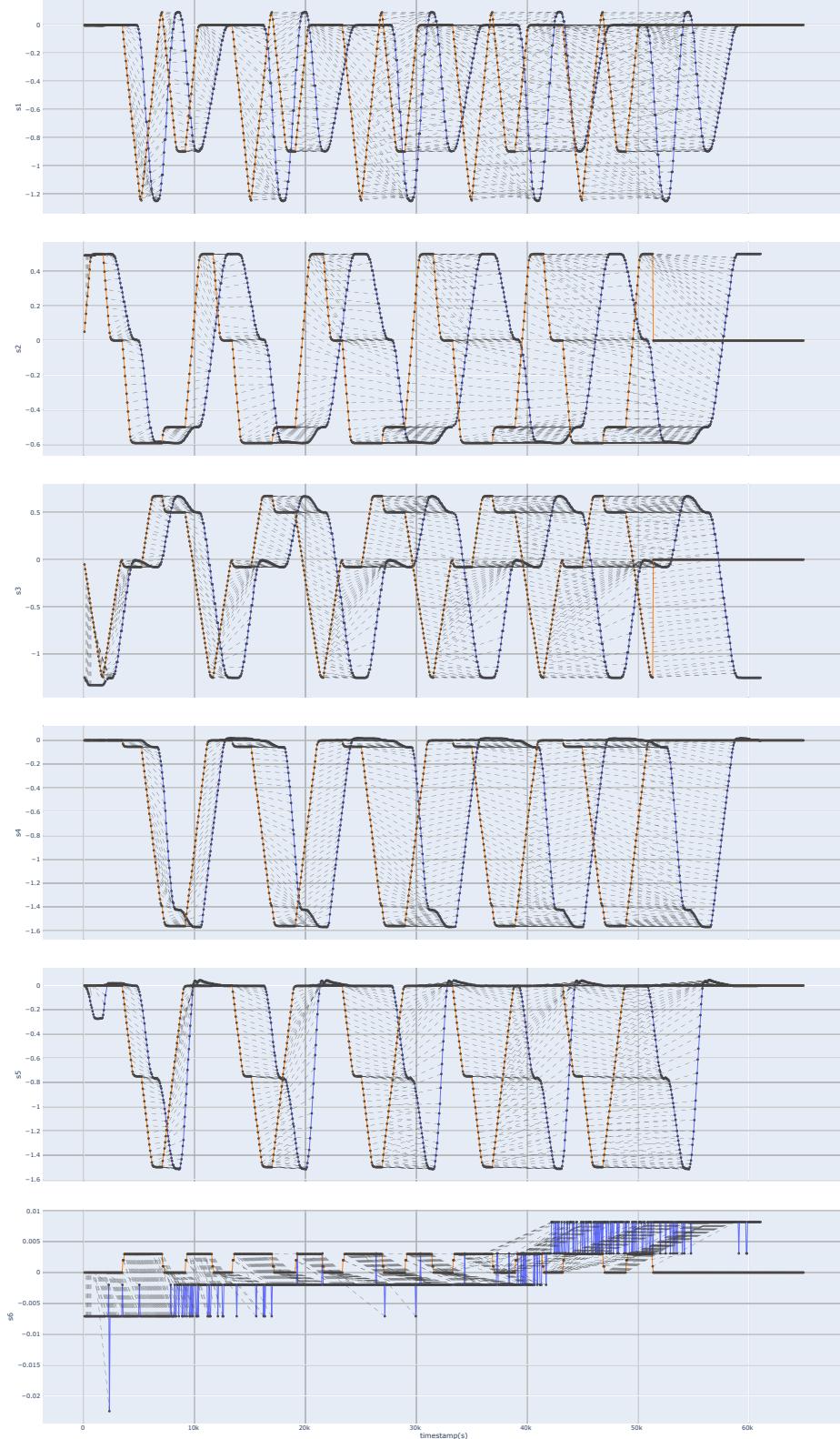


Fig. 20. Alignment of the scenario 4 with a MAD of 1.15 degrees

II Acknowledgments

Paula *Thanks to Manuel Wimmer and his team for the data*

References