# Detailed Description of MADROB Protocol and Performance Indicators

## 1 Terminology

The starting side is the side of the door where the robot is at the start of the benchmark. The destination side is the side of the door that the robot must reach by going through the door.

The terms clockwise/CW and counterclockwise/CCW identify the two possible directions of rotation of the door panel when observed from above, with respect to its central (closed) position.

The same terms are also used to identify the two sides of the testbed area that are separated by the wall to which the door is affixed. Precisely, CW is the side entered by the door panel when it rotates in the CW direction; CCW the side it enters when it rotates in CCW direction. In the following, the values of starting side and destination side can be either CW or CCW.

In the execution of the benchmark movement of the door can be constrained so that it only opens when rotated either CW or CCW. The opening side of the testbed area is the side towards which the door is allowed to move. The value of opening side can be either CW or CCW. Such value is independent from the value of starting side.

The threshold is the ideal line on the ground corresponding to the projection of the door panel when locked. The threshold separates the CW and CCW sides of the testbed. A robot crosses the threshold when it goes from a pose where its projection on the ground is entirely on one side of the threshold to a pose where its projection is entirely on the other side.

## 2 Procedure

In order to execute the benchmark, the robot must execute, in order, the following steps:

- Touch the door handle.
- Unlock the door.
- Open the door.
- Cross the threshold.

- Close the door, without crossing the threshold again.
- Lock the door, without crossing the threshold again.

For the benchmark to start, the robot must first be set in a start pose specified by the referees. Once this precondition is satisfied, the benchmark starts in a time instant specified by the referees. The benchmark stops as soon as any of the following events occurs: the last step of the benchmark is completed by the robot; the timeout has elapsed. The benchmark can also be manually interrupted by a referee (via the GUI). In this case, the execution is invalid.

# 3 Pre-Processed Data

## 3.1 Event

The event time series contains detectable events and their timestamps.

The possible events are:

- benchmark start: identifies the start of benchmark execution. This event is always present. The timing of this event is affected by a delay due to the people operating the robot having to manually start the robot.
- door opens: when the condition $|door\ angular\ position| < th_{closed}$ changes from being false to true. The value of the threshold $th_{closed}$ is $0.03 rad$.
- door closes: when the condition $|door\ angular\ position| > th_{closed}$ changes from being true to false.
- handle is touched: event occurs when force signal exceeds a predefined threshold $th_{force}$. Only the first occurrence of this event is recorded. We set $th_{force}$ to 200 grams-force.
- humanoid moves to cw side, humanoid moves to ccw side: last detection of humanoid moving from one side to the other by passage sensors. Only one occurrence of these events can be present.
- humanoid approaches the door on cw side, humanoid approaches the door on ccw side: first detection of a humanoid under the passage sensors. Only one occurrence of these events can be present.
- benchmark stop: identifies the end of benchmark execution. This event is always present.

The events "humanoid approaches the door on [cw, ccw] side" and "humanoid moves to [cw, ccw] side" are computed using 10 of the 16 laser proximity sensors fitted above the door to detect the presence of a humanoid on each side. Five sensors are fitted on each side of the door (cw/ccw).

Knowing the height of the sensors we can compute the height of any obstacle under each sensor.

In order to detect when a humanoid is present underneath the sensors, we need to compare the height reading with a threshold. We set this threshold to 50cm in order to account for noise that could affect the sensors in certain conditions (for example operating at excessive temperature or while exposed to direct sunlight). The event "humanoid approaches the door on $x$ side" is recorded when the height reading on any sensor raises above the threshold for the first time. The side $x$ (cw/ccw) is the same side of the sensor that detected the first raise. The event "humanoid approaches the door on $y$ side" is recorded when the height reading on each sensor falls under the threshold for the last time. The side $y$ (cw/ccw) is the same side of the sensor that detected the last fall.

When the door panel passes underneath a sensor it produces a false reading equal to the height of the door panel. We simply ignore the reading of the sensors when the door is between the range of angles at which it is close to being detected by it. The ranges are manually estimated for each sensor. As a side note, the angle ranges in which the sensors are ignored are such that we were not able to pass through the door undetected. This is because the sensors are only ignored when the door is almost closed and by the time the robot or person reaches the door, they have already being detected. Although the testbed is fitted with 16 sensors (8 on each side), the six sensors closer to the door's hinge are not needed for the computation of these events.

### 3.2 Wrench of handle

The data contains two columns: time (ros time in seconds since epoch), force_x $[N]$ The force applied on the door handle is simply copied from the raw data from the ros topic /madrob/handle/force.

### 3.3 Joint state of door panel

The data contains four columns: time (ros time in seconds since epoch), position (panel angular position) $[rad]$, velocity (panel angular velocity) $[rad/s]$, acceleration (panel angular acceleration) $[rad/s^2]$.

The position is simply copied from the raw data from the ros topic /madrob/door/angle. The velocity and acceleration must be computed from the position after applying a filter to reduce noise. We apply a moving average on the position (see pandas.core.window.Rolling.mean) then compute the second order derivative of the position using the Savitzky-Golay filter (see scipy.signal.savgol_filter).

These two filtering operations depend from the following parameters: moving average window size, Savitzky-Golay window size and Savitzky-Golay polynomial order. The size of the moving average window and Savitzky-Golay window are computed by rounding up to the closest odd number of samples falling in a time window of 0.2 seconds (assuming a sampling frequency of 50Hz, the window size will be 9 samples). This value was chosen as a compromise between noise reduction and preserving the maximum velocity and acceleration values. The Savitzky-Golay polynomial order is set to 2, the lowest possible value enabling the computation of angular velocity and acceleration. Higher values would introduce artefacts and require a larger window size.

# 4 Performance Indicators

## 4.1 Execution time

Overall duration of benchmark execution. The output is the difference between the time of the events "benchmark stop" and "benchmark start". This performance indicator uses the start benchmark event to take into account the time employed by the humanoid to perceive the door, plan its actions and move through the door. This timing is affected by some human factor since the robot and the benchmark must be manually started at the same time. Ideally we would want to measure the execution time from the moment the robot starts executing the task, but due to limitations in the communication (the task must be started manually in the robot) there can be a small delay between the time recorded in the benchmarking software and the actual start time of the robot task.

## 4.2 Time to handle

Time elapsed from the start of the benchmark to the first time the handle is touched by the robot/humanoid. This PI accounts for the time the robot takes to perceive the door, plan its actions and start opening the door. The value is computed as the time elapsed from the event "benchmark start" to the first event "handle is touched".

## 4.3 Door occupation time

Time elapsed between the humanoid approaching the door from the starting side and the humanoid leaving the destination side (measured by sensors detecting when the humanoid is present in the proximity of the door on each side).

## 4.4 Passage time

Time elapsed between the event "handle is touched" and when the humanoid closes the door after reaching the destination side. This PI is similar to the PIs Door occupation time and Overall execution time, but avoids using the timing of the benchmark start, that rely on manual timing (starting benchmark and the robot at the same time).

## 4.5 Unsafety of door operation

This PI is a measurement of the safety of door operation by the robot based on the maximum angular acceleration of the door panel. High angular accelerations correspond to sudden movements and high forces exerted on any object

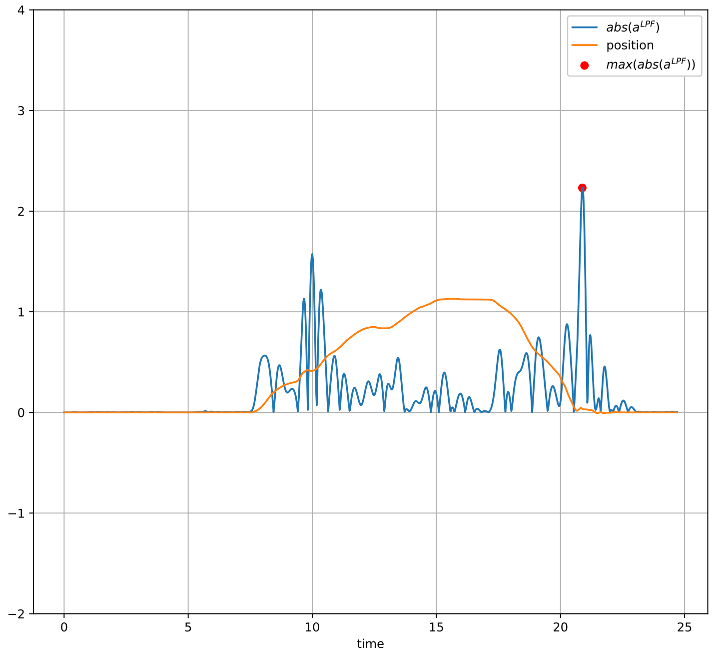


Figure 1: Plot of the position (orange), the low-pass filtered acceleration $a^{LFP}$ (blue) of the door panel and its maximum value (red) used to compute the value of the PI. The data is collected in a test run in which a person operates the door.

or person touching the door panel. The lower door acceleration, the lower the risk of serious incidents. High angular acceleration can indicate tremors in the robot's hand effectors. However, these are not as significant for safety because of the limited range of motion they impose to the door panel, therefore low-pass filtering is applied to the door angular acceleration (as opposed to the PI Smoothness of door actuation).

The value is computed as $max(abs(a_i^{LPF}))$, where $a_i^{LPF}$ are the low-pass filtered values of angular acceleration in the Joint state pre-processed data. See Fig. 1. The low-pass filter is implemented with scipy.signal.iirfilter and scipy.signal.lfilter. The critical frequency of the filter, set to $2Hz$, has been chosen empirically.

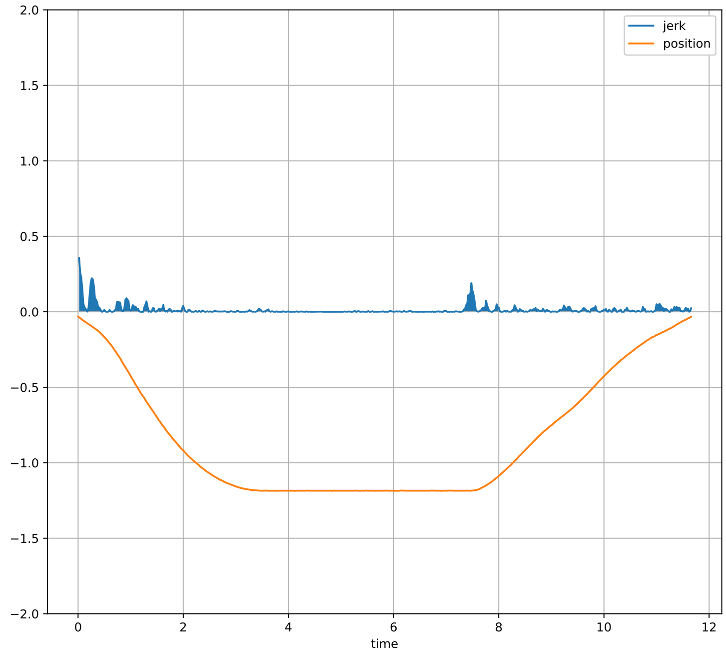


Figure 2: Plot of the position (orange) and jerk (blue) of the door panel resulting in a very high smoothness value of 12.8. The data is collected in a test run in which a person operates the door.

### 4.6 Smoothness of door actuation

This PI is a measurement of the smoothness of the actuation of the door panel based on its angular jerk. To operate the door smoothly the humanoid should minimise the jerk of the door panel. Angular accelerations and decelerations can indicate bumps against the handle/panel and unnecessary corrections by the humanoid in the actuation of the door panel. The value is computed as:

$$\frac{10}{\frac{1}{N}\sum_{i=1}^{N}|j_i|} \tag{1}$$

where $j_i$ is the jerk, computed by deriving the angular acceleration from the Joint state pre-processed data with respect to time. We only consider values of jerk for which the door is considered not closed (door angle greater than a small angle threshold). The value is multiplied by 10 to rescale the values in a more comfortable range. Three plots with examples of data from tests in which a person uses the door are shown in Figures 2, 3 and 4.

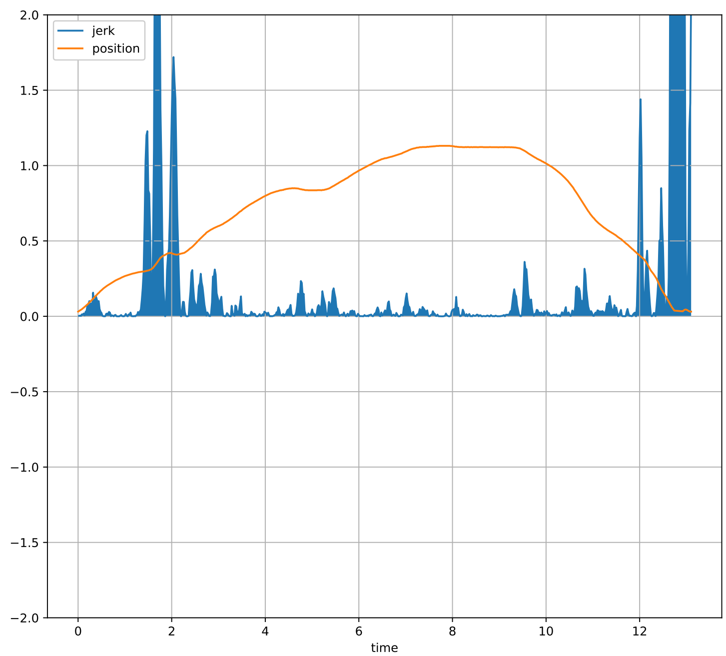


Figure 3: Plot of the position (orange) and jerk (blue) of the door panel resulting in a modest smoothness value of 3.9. The data is collected in a test run in which a person operates the door.

## 4.7 Roughness of actuation

This PI measures how rough (or undelicate) the robot is in operating the door based on the maximum force applied to the handle. High force applied to the handle is a symptom of bad handling of the door and a risk for the integrity of the door. The value is computed as $max_i(|f_i|)$, where $f_i$ are the values of force from the wrench pre-processed data.

## 4.8 Capability level

Number of steps of the benchmark actually completed by the robot. Each step is considered completed only after all the steps preceding it have been completed as well. The result is an integer ranging from 0 to the number of steps composing the benchmark procedure.

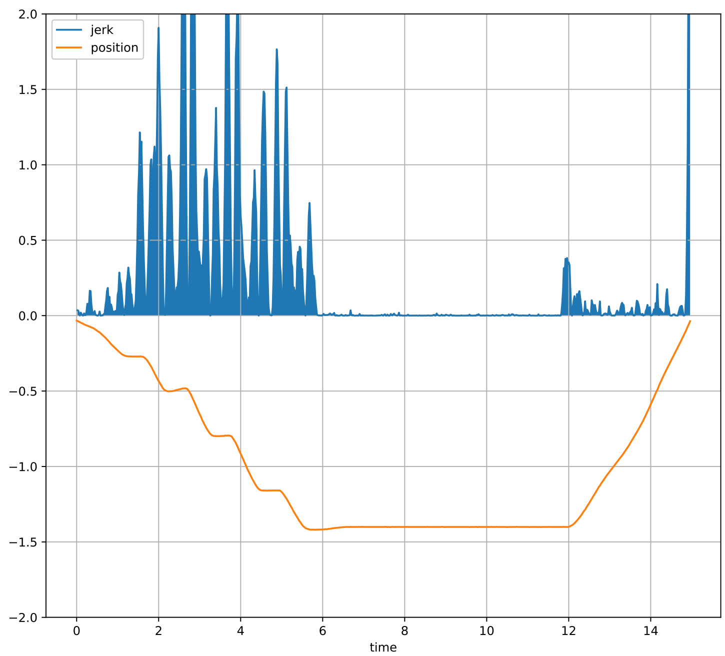


Figure 4: Plot of the position (orange) and jerk (blue) of the door panel resulting in a very low smoothness value of 3.2. The data is collected in a test run in which a person operates the door.

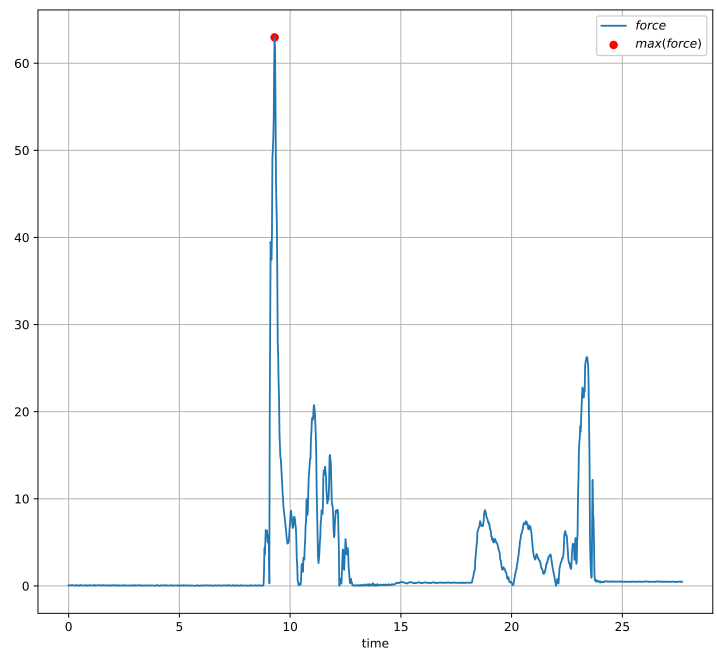


Figure 5: Plot of the force acting on the handle (blue) and its maximum value (red) used to compute the value of the PI. The data is collected in a test run in which a person operates the door.