1 Fitting basics

The Time Tagger acquires clock time-tags y_i which are simply enumerated by an index x_i. The time-tags shall be fitted by a function

$$f(x) = ax + b \tag{1.1}$$

by least squares. For a set of N time-tags, we need to minimize

$$g(a, b) = \sum_{i=0}^{N-1} (y_i - f(x_i))^2 = \sum_{i=0}^{N-1} (y_i - ax_i - b)^2 \tag{1.2}$$

so the coefficients a and b need to fulfill the conditions

$$\partial_a g = -2 \sum_{i=0}^{N-1} (y_i - ax_i - b)\, x_i = 0 \tag{1.3}$$

$$\partial_b g = -2 \sum_{i=0}^{N-1} (y_i - ax_i - b) = 0. \tag{1.4}$$

Equation 1.4 directly results in

$$b = \frac{1}{N} \sum_{i=0}^{N-1} (y_i - ax_i) = \frac{S_y - aS_x}{N} \tag{1.5}$$

where we used the symbols

$$\begin{aligned} S_x &= \sum_{i=0}^{N-1} x_i \\ S_y &= \sum_{i=0}^{N-1} y_i \\ S_{xx} &= \sum_{i=0}^{N-1} x_i^2 \\ S_{xy} &= \sum_{i=0}^{N-1} x_i y_i. \end{aligned}$$

If we put 1.5 into 1.3, we obtain

$$S_{xy} - aS_{xx} - bS_x = S_{xy} - aS_{xx} - \frac{S_y - aS_x}{N} S_x = 0 \tag{1.6}$$

which results together with 1.5 in

$$a = \frac{S_x S_y - N S_{xy}}{S_x^2 - N S_{xx}} \tag{1.7}$$

$$b = \frac{S_{xy} S_x - S_y S_{xx}}{S_x^2 - N S_{xx}}. \tag{1.8}$$

2 Moving fit

The goal is now to use the offset b to correct the position of the current clock tag. For this purpose, the fit is performed over the last N time-tags. The current time-tag will always get the index $i = 0$ and will by definition be set to $x_0 \equiv 0$ and $y_0 \equiv 0$. The preceding tag is set to $x_1 = -1$ and so on, so the fit will always be performed over the x-range $[0, -1, \ldots, -(N-1)]$ which results in

$$S_x = \sum_{i=0}^{N-1} x_i = \sum_{i=0}^{N-1} -i = -\frac{N(N-1)}{2} \tag{2.1}$$

$$S_{xx} = \sum_{i=0}^{N-1} x_i^2 = \sum_{i=0}^{N-1} (-i)^2 = \frac{(N-1)N(2N-1)}{6}. \tag{2.2}$$

This simplifies expression 1.8

$$b = \frac{(2N-1)S_y - 3S_{xy}}{\sum_{i=1}^{N} i}. \tag{2.3}$$

Because it is not a good idea to perform the calculation of S_y and S_{xy} for every clock tag from the set of N tags, we will calculate it incrementally. Here we need to distinguish two cases: At the beginning of the fit, when less than N tags already arrived, the set will grow with every new tag. If there are N tags in the set, the oldest one needs to be dropped when a new tag arrives.

To keep numbers small, only relative values are used. Additionally, they are reduced by the expected period T.

2.1 Growing set of tags

We start with a set of size N. When the tag $i = n+1$ arrives, it will grow to a size of $N+1$ and the value S_y will change (Δ_+ indicates the growing case) by:

$$\Delta_+ S_y^{(n+1)} = \left(\sum_{i=0}^{N} y_{n+1-i} - y_{n+1} + iT \right) - \left(\sum_{i=0}^{N-1} y_{n-i} - y_n + iT \right) \tag{2.4}$$

$$= -N\left(y_{n+1} - y_n - T\right)$$

$$= -N d_{n+1} \tag{2.5}$$

with the reduced step size $d_{n+1} = y_{n+1} - y_n - T$. Similarly we can calculate the increment of S_{xy}:

$$\Delta_+ S_{xy}^{(n+1)} = \left(\sum_{i=0}^{N} -i\left(y_{n+1-i} - y_{n+1} + iT\right) \right) - \left(\sum_{i=0}^{N-1} -i\left(y_{n-i} - y_n + iT\right) \right) \tag{2.6}$$

$$
\begin{aligned}
&= \left(\sum_{i=0}^{N-1} -y_{n-i}\right) + y_{n+1}\left(\sum_{i=0}^{N} i\right) - y_n\left(\sum_{i=0}^{N-1} i\right) - N^2 T \\
&= -\left(\sum_{i=0}^{N-1} y_{n-i} - y_n + iT\right) + y_{n+1}\left(\sum_{i=0}^{N} i\right) - y_n\left(\sum_{i=0}^{N} i\right) - \left(N^2 - \sum_{i=0}^{N-1} i\right) T \\
&= -S_y^{(n)} + (y_{n+1} - y_n)\left(\sum_{i=0}^{N} i\right) - \left(N^2 - \frac{N(N-1)}{2}\right) T \\
&= -S_y^{(n)} + (y_{n+1} - y_n)\left(\sum_{i=0}^{N} i\right) - \left(\frac{N(N+1)}{2}\right) T \\
&= -S_y^{(n)} + (y_{n+1} - y_n - T)\left(\sum_{i=0}^{N} i\right) \\
&= -S_y^{(n)} + d_{n+1}\left(\sum_{i=0}^{N} i\right) \qquad (2.7)
\end{aligned}
$$

2.2 Constant set size

If the set is full, the calculation is very similar, but in both cases, eqs. 2.4 and 2.6, the sum in the first summand does not run up to N but only up to $N-1$. The summand for $i = N$ is simply subtracted:

$$
\begin{aligned}
\Delta S_y^{(n+1)} &= \left(\sum_{i=0}^{N-1} y_{n+1-i} - y_{n+1} + iT\right) - \left(\sum_{i=0}^{N-1} y_{n-i} - y_n + iT\right) \\
&= \Delta_+ S_y^{(n+1)} - (y_{n+1-N} - y_{n+1} + NT) \\
&= -N d_{n+1} + D_{n+1}, \qquad (2.8)
\end{aligned}
$$

$$
\begin{aligned}
\Delta S_{xy}^{(n+1)} &= \left(\sum_{i=0}^{N-1} -i\left(y_{n+1-i} - y_{n+1} + iT\right)\right) - \left(\sum_{i=0}^{N-1} -i\left(y_{n-i} - y_n + iT\right)\right) \\
&= \Delta_+ S_{xy}^{(n+1)} - \left(-N\left(y_{n+1-N} - y_{n+1} + NT\right)\right) \\
&= \Delta_+ S_{xy}^{(n+1)} - N D_{n+1}. \qquad (2.9)
\end{aligned}
$$

Here, D_{n+1} is the time between the newly added tag and the one that is dropped.

3 Python naming

The following names are used as attribute in the Python class `LinearClockApproximation` or as variables in its method `_process_clock_tag()`:

S_y	`self.y_sum`
S_{xy}	`self.xy_sum`
$\sum_{i=0}^{N} i$	`self.i_sum`
d_{N+1}	`step`
D_{N+1}	`front_to_end`