Linear Assignment Problem: Single-Particle Tracking with Particle Intensity

Anthony Lin[1]

[1]Massachusetts Institute of Technology, MA, United States
lin_a@mit.edu

Abstract. The Linear Assignment Problem (LAP) is often used to track particles between frames because of its high flexibility and speed compared to statistical-based tracking methods. This paper presents a modified LAP-based algorithm to improve the accuracy of single-particle tracking. By considering particle intensity in the cost function, the accuracy of particle linking increases significantly for cases where there is less Gaussian noise compared to the variance of the particle intensities. Beyond improvements in accuracy, this LAP-based algorithm performs significantly faster than current LAP-based single-particle tracking algorithms.

1 Method

1.1 Linear Assignment Problem

The Linear Assignment Problem (LAP) in single-particle tracking is used to optimally link particles across consecutive frames by minimizing a cost function, typically based on spatial distance (Jaqaman et al., 2008). Each detection in one frame is assigned to at most one detection in the next, ensuring consistent trajectories. LAP-based algorithm aims to find a perfect matching with the lowest-cost set of assignments to link these particles.

1.2 Frame-Linking Cost Matrix

The frame-linking cost matrix stores the costs of linking a particle from frame t to frame $t+1$, ending a track in frame t, and starting a track in frame $t+1$ (Fig. 1). It consists of four blocks: linking (top left), terminating (top right), initiating (bottom left), and auxiliary (bottom right) blocks. The linking block stores the cost of linking the particles. To speed up the computation, a maximum squared Euclidean distance threshold was set. Any two particles that are separated by more than this squared Euclidean distance will not be considered because it is highly unlikely for these two particles to link. For particles that do not exceed this threshold, the cost of particle i in the current frame to link with particle j in the next frame is defined by Eq. 1.

$$c_{ij} = \alpha \cdot d_{ij}^2 + \beta \cdot \Delta I_{ij}^2 \tag{1}$$

$$
\begin{array}{cc}
 & \begin{array}{cc} \hspace{6em}\text{Frame t+1}\hspace{6em} & \hspace{4em}\text{Frame t}\hspace{4em} \end{array} \\
\begin{array}{c} \text{Frame t} \\ \\ \\ \\ \text{Frame t+1} \end{array} &
\left[\begin{array}{ccccc:cccc}
c_{1,1} & c_{1,2} & c_{1,3} & \cdots & c_{1,n_{t+1}} & 100 & \times & \cdots & \times \\
c_{2,1} & \times & c_{2,3} & \cdots & \vdots & \times & \ddots & & \vdots \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & & \ddots & \times \\
c_{n_t,1} & \times & c_{n_t,3} & \cdots & c_{n_t,n_{t+1}} & \times & \cdots & \times & 100 \\
\hdashline
100 & \times & \cdots & \cdots & \times & -100 & -100 & \cdots & -100 \\
\times & \ddots & & & \vdots & -100 & \times & \cdots & \times \\
\vdots & & \ddots & & \vdots & -100 & -100 & \cdots & -100 \\
\vdots & & & \ddots & \times & \vdots & \vdots & \ddots & \vdots \\
\times & \cdots & \cdots & \times & 100 & -100 & \cdots & \cdots & -100
\end{array}\right]
\end{array}
$$

Fig. 1. Frame-linking cost matrix

d_{ij}^2 is the z-score normalized squared Euclidean distance between the two particles and ΔI_{ij}^2 is the z-score normalized squared intensity difference between the two particles defined below.

$$d_{ij}^2 = \frac{||\boldsymbol{x_j} - \boldsymbol{x_i}||^2 - \mu_{d^2}}{\sigma_{d^2}} \tag{2}$$

$$\Delta I_{ij}^2 = \frac{||I_j - I_i||^2 - \mu_{\Delta I^2}}{\sigma_{\Delta I^2}} \tag{3}$$

α and β are weighing terms for the squared Euclidean distance and squared intensity difference, respectively. They follow the following constraint.

$$\alpha + \beta = 1 \tag{4}$$

The terminating and initiating blocks are matrices with 100 along their diagonals. These values will always be greater than c_{ij}, so they will only be matched if there are no possible links between two particles. The auxiliary block is the same as the cost block, but is transposed and its values filled with -100. This block is added so there will be a perfect matching. These values will always be less than c_{ij}, so they will not affect the particle links.

1.3 Gap-Closing Cost Matrix

The gap-closing cost matrix stores the cost of closing a gap from the end of a track to the start of another track (Fig. 2). It also consists of four blocks: gap-linking (top left), terminating (top right), initiating (bottom left), and auxiliary (bottom right) blocks. The gap-linking block contains the cost for linking gaps. To speed up the computation, there are two constraints. First, a maximum frame window was set, which was the largest number of frames over which a particle can temporarily disappear. Beyond this frame window, the possibility of a gap

$$\begin{array}{cc} & \begin{array}{cc}\text{Start of Tracks} & \qquad\qquad \text{End of Tracks}\end{array} \\ \begin{array}{c}\text{End of Tracks} \\ \\ \\ \\ \text{Start of Tracks}\end{array} & \left[\begin{array}{cccc:cccc} g_{1,1} & g_{1,2} & \cdots & g_{1,n_S} & 100 & \times & \cdots & \times \\ g_{2,1} & \times & \cdots & \vdots & \times & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & & \ddots & \times \\ g_{n_E,1} & g_{n_E,2} & \cdots & g_{n_E,n_S} & \times & \cdots & \times & 100 \\ \hdashline 100 & \times & \cdots & \times & -100 & -100 & \cdots & -100 \\ \times & \ddots & & \vdots & -100 & \times & \cdots & -100 \\ \vdots & & \ddots & \times & \vdots & \vdots & \ddots & \vdots \\ \times & \cdots & \times & 100 & -100 & \cdots & \cdots & -100 \end{array}\right] \end{array}$$

Fig. 2. Gap-closing cost matrix

closing was not considered. Second, a maximum squared Euclidean distance threshold was set between the end of a track and the start of another track. Any two tracks that are separated by more than this squared Euclidean distance will not be considered because it is highly unlikely that these two tracks will link. For tracks that satisfy both of these constraints, the gap closing cost was defined by Eq. 5. This equation is the same as in Eq. 1, except that it is calculating the cost between the particle at the end of track i and the particle at the start of track j.

$$g_{ij} = \alpha \cdot d_{ij}^2 + \beta \cdot \Delta I_{ij}^2 \tag{5}$$

The terminating, initiating, and auxiliary blocks are the same as those in the frame-linking cost matrix. Any matching in the terminating, initiating, or auxiliary block will not result in a gap closure.

2 Validation

2.1 Simulation

To validate this algorithm, a simulation of particles undergoing Brownian motion was created. The simulation parameters were the following:

1. number of frames
2. region size
3. number of particles
4. diffusion coefficient
5. mean intensity of the particles
6. standard deviation of particle intensities
7. standard deviation of Gaussian noise for particle intensities

While there are many parameters to test, previous papers (Simon et al., 2024 and Shen et al., 2017) have determined the effects of duration, particle density and diffusion coefficient on the performance of particle tracking algorithms. Since this algorithm modifies previous LAP-based algorithms by also considering intensity, the validation will focus on parameters 6 and 7. For the rest of the simulations, the following parameters were fixed: number of frames = 100, region size = (256,256), number of particles = 150, diffusion coefficient = 1, and mean intensity = 100.

2.2 Baseline and Accuracy

Comparisons were made with the LapTrack algorithm in Fukai and Kawaguchi (2022). Their implementation is based on the LAP algorithm in Jaqaman et al. (2008), thus making it a valid comparison. Accuracy was defined as the percentage of correctly predicted particles to their tracks. After running the simulation 100 times, the mean accuracy and its 95% confidence interval was 0.7844 ± 0.0062 (Fig. 3).

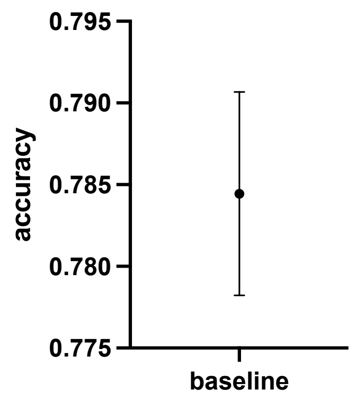

Fig. 3. Mean accuracy of the LapTrack algorithm in the simulation with 95% confidence intervals.

2.3 Ratio of Particle Intensity SD to Noise SD

The ratio of particle intensity standard deviation, σ_I, and Gaussian noise standard deviation, σ_n, determines the algorithm's accuracy. Setting $\alpha = 0.5$ and $\beta = 0.5$ to weigh each term equally, the algorithm's accuracy increases as the ratio increases (Fig. 4). Thus, a higher ratio suggests that particle intensity is a suitable predictor in particle linking. As long as this ratio is fixed, any particle intensity and Gaussian noise standard deviations will result in the same accuracy.

To roughly estimate this ratio in an experimental setting, one can run the algorithm by setting $\alpha = 0.9$ and $\beta = 0.1$. Assuming that the real particle

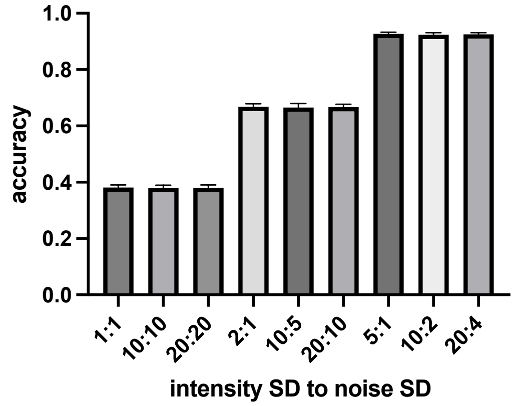

Fig. 4. Mean accuracy of the algorithm to different standard deviation ratios of particle intensity to noise in the simulation with $\alpha = 0.5$ and $\beta = 0.5$. Error bars represent the 95% confidence intervals.

intensity does not change significantly, the Gaussian noise is roughly the average standard deviation, σ'_n, of the particle intensity of each track. Then the ratio is roughly the following.

$$\frac{\sigma_I}{\sigma_n} \approx \sqrt{\frac{\sigma_I'^2 - \sigma_n'^2}{\sigma_n'^2}} \tag{6}$$

σ'_I is the average standard deviation for the observed particle intensity across the frames. As long as this ratio is large (> 1), it is beneficial to use the particle intensity as a predictor to link particles.

2.4 Weights

When the ratio of particle intensity standard deviation to Gaussian noise is 1:1, setting $\beta > 0$ led to worse accuracy (Fig. 5). However, for 2:1 and 5:1 ratios, there are improvements in the accuracy compared to the baseline (Fig. 6 and 7). Generally, $\beta = 0.1$ leads to significant improvements in accuracy. Setting $\beta > 0.1$ does not improve the accuracy significantly, and even performs worse for the 2:1 case. Therefore, it is appropriate to set $\beta = 0.1$ for most cases where the noise is low compared to the particle intensity signal.

2.5 Performance

This algorithm performs significantly faster than the baseline. Testing 150 particles for 1000 frames (150000 total particles), the algorithm took 1.391 seconds, while the baseline took 18.085 seconds (Fig. 8).

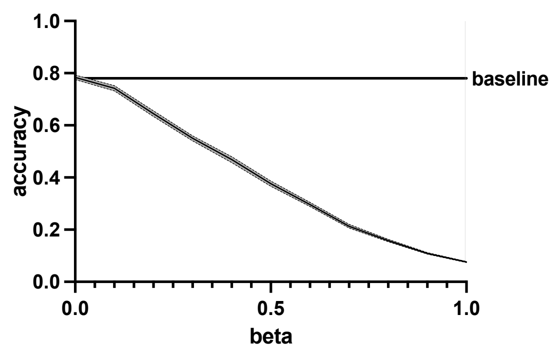

Fig. 5. Mean accuracy of the algorithm with varying β weights in the simulation at 1:1 particle intensity to noise standard deviation ratio. The horizontal line represent the baseline accuracy from the LapTrack algorithm. Error bars represent the 95% confidence intervals.

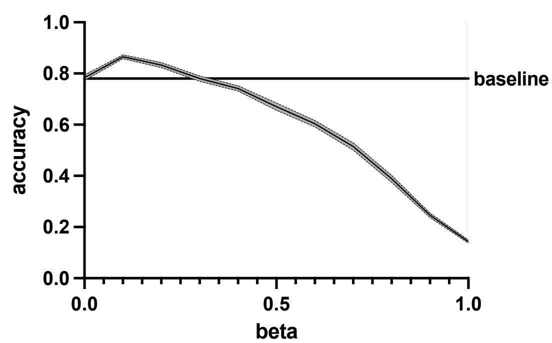

Fig. 6. Same as figure 5, but for 2:1 ratio.

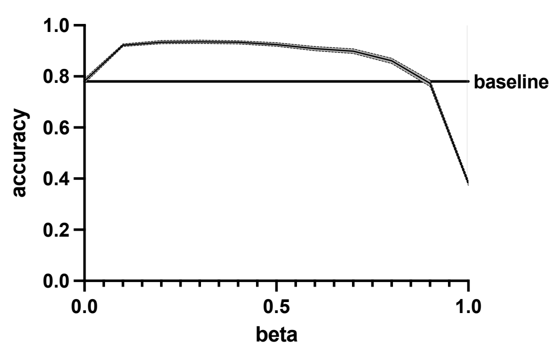

Fig. 7. Same as figure 5, but for 5:1 ratio.

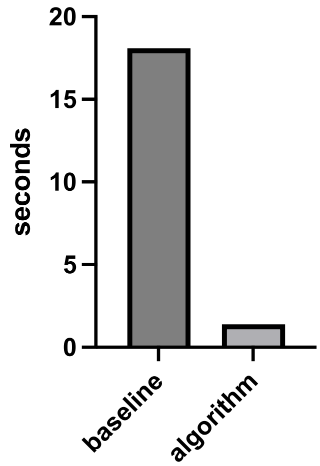

Fig. 8. Speed of baseline and the modified algorithm for linking 150000 particles.

3 Practical Considerations

In experimental conditions, the underlining ratios are unknown. Therefore, it is practical to run the algorithm with $\alpha = 0.9$ and $\beta = 0.1$, then estimate the particle intensity to noise standard deviation ratio. If the ratio is less than 1, then run the algorithm again with $\alpha = 1$ and $\beta = 0$. Otherwise, keep the results. However, the best way to get highly accurate results is with experiment setup. In dense conditions, the accuracy of any single-particle tracking algorithms are poor. Additionally, if the particles move greatly from frame to frame, the accuracy also becomes poor. Therefore, it is important to set up experiments with sparse to moderate particle density and use a high frame rate. In such cases, no tracking algorithm will be able to deliver reliable results, no matter how well-designed.

References

1. Hao Shen, Lawrence J. Tauzin, Rashad Baiyasi, Wenxiao Wang, Nicholas Moringo, Bo Shuang, and Christy F. Landes Chemical Reviews 2017 117 (11), 7331-7376 DOI: 10.1021/acs.chemrev.6b00815
2. Jaqaman, K., Loerke, D., Mettlen, M. et al. Robust single-particle tracking in live-cell time-lapse sequences. Nat Methods 5, 695–702 (2008). https://doi.org/10.1038/nmeth.1237
3. Simon, F., Weiss, L.E. & van Teeffelen, S. A guide to single-particle tracking. Nat Rev Methods Primers 4, 66 (2024). https://doi.org/10.1038/s43586-024-00341-3
4. Yohsuke T Fukai, Kyogo Kawaguchi, LapTrack: linear assignment particle tracking with tunable metrics, Bioinformatics, Volume 39, Issue 1, January 2023, btac799, https://doi.org/10.1093/bioinformatics/btac799