Track vertex reconstruction with neural networks at the first level trigger of Belle II

Sara **Neuhaus**[1,a], Sebastian **Skambraks**[1], and Christian **Kiesling**[1]

[1] *Max Planck Institute for Physics, Munich, Germany*

Abstract. The track trigger is one of the main components of the Belle II first level trigger, taking input from the Central Drift Chamber (CDC). It consists of several stages, first combining hits to track segments, followed by a 2D track finding in the transverse plane and finally a 3D track reconstruction. The results of the track trigger are the track multiplicity, the momentum vector of each track and the longitudinal displacement of the origin or production vertex of each track ("z-vertex"). The latter allows to reject background tracks from outside of the interaction region and thus to suppress a large fraction of the machine background. This contribution focuses on the track finding stage using Hough transforms and on the z-vertex reconstruction with neural networks. We describe the algorithms and show performance studies on simulated events.

1 Introduction

Belle II [1] is an upgrade of the Belle experiment, which is currently being constructed at KEK in Tsukuba, Japan. It is located at the asymmetric e^+e^- collider SuperKEKB [2], which will operate at center of mass energies at and around the $\Upsilon(4S)$ resonance. The $\Upsilon(4S)$ decays almost exclusively to $B\overline{B}$ pairs, which are nearly at rest in the center of mass system. The boost of the system allows to measure the difference between the two B decay times and thus to study time-dependent CP violation in the B meson system. SuperKEKB is designed to achieve an instantaneous luminosity of $8 \times 10^{35}\,\text{cm}^{-2}\,\text{s}^{-1}$ and deliver $50\,\text{ab}^{-1}$ of physics events in total.

The first level trigger is a pipelined hardware trigger implemented on FPGAs. It is restricted to a maximal trigger rate of 30 kHz and a fixed total latency of at most 5 µs. The average number of charged tracks for a $B\overline{B}$ event is ≈ 10. About 3 to 9 tracks are typically visible in the track trigger, while the background track multiplicity is relatively low, producing rarely more than two visible tracks. Therefore a trigger will generally be set for three or more tracks. However, there are many interesting physics channels that cannot be efficiently triggered if three tracks are required. Examples include B decays to invisible final states and τ pair events. The latter will be produced at SuperKEKB with a cross section of the same order of magnitude as the $\Upsilon(4S)$ cross section, so τ physics are also an important aspect of the Belle II physics program.

If the trigger threshold is lowered to two tracks, additional constraints are required to suppress background. A dominant source of background comes from the machine, that is from intra-bunch

[a] e-mail: sneuhaus@mpp.mpg.de

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (http://creativecommons.org/licenses/by/4.0/).

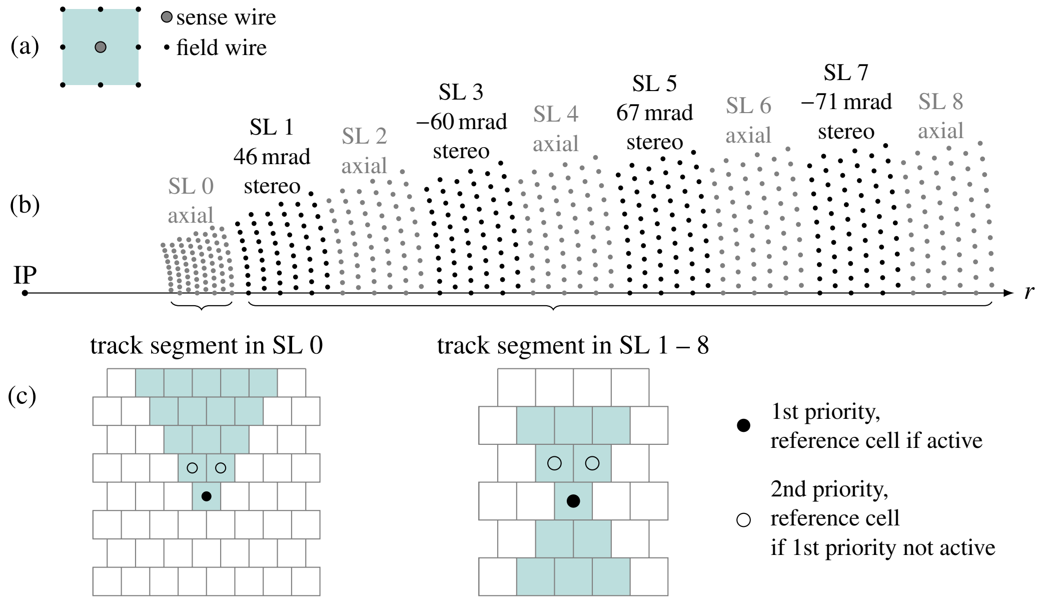

Figure 1. Wire configuration of the CDC. (a) Drift cell of one sense wire and eight field wires. (b) Arrangement of sense wires. Superlayers (SL) consist of several layers with the same wire orientation (axial or stereo). (c) In the track trigger, track segments are defined as a fixed arrangement of cells within a superlayer.

interactions (Touschek scattering [3]) and collisions with residual gas in the beampipe. This background can be suppressed by reconstructing the z-vertex of each track and rejecting tracks that do not originate at the interaction point, that is it requires a 3D track reconstruction. A second background source is Bhabha scattering, which needs to be suppressed by a factor of ≈ 100. A Bhabha veto will be based on tracks that are matched with clusters in the electromagnetic calorimeter. This matching algorithm also requires a 3D track reconstruction to obtain the polar angle and the total momentum of the track.

2 The track trigger

The input for the track trigger are hits from the Central Drift Chamber (CDC), a wire chamber with 14 336 sense wires arranged in 56 layers. The layers can be grouped into 9 so-called superlayers, which are defined by the wire orientation: 5 axial superlayers, where the wires are parallel to the z-axis, alternate with 4 stereo superlayers, where the wires are skewed by 45.4 mrad to 74.0 mrad to allow a 3D track reconstruction. Each superlayer consists of 6 layers, except for the innermost superlayer, which consists of 8 layers of smaller cells. The CDC geometry is illustrated in figure 1.

2.1 Track segment finder

The first stage of the track trigger is a data reduction step, where hits from the same superlayer are combined to so-called track segments. A track segment is defined as a fixed arrangement of wires, which is shown in figure 1c. A track segment hit is produced if there are hits in at least four of the

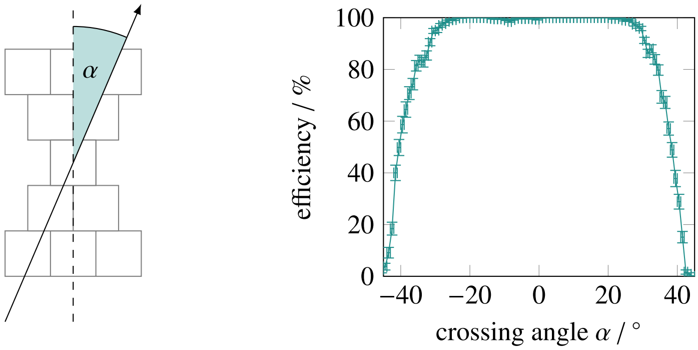

Figure 2. Track segment finder efficiency depending on the crossing angle α, measured on single track events. The efficiency is defined as the number of superlayers with a found track segment over the number of superlayers crossed by a track.

five layers within the track segment. The hit information for the following trigger stages contains a track segment number, the drift time of a reference cell in the center of the track segment, an ID that identifies the reference cell and a left/right passage. The latter tells whether the track passed left or right of the reference wire and is determined from the hit pattern within the track segment. For ambiguous hit patterns the left/right passage is undecided.

Besides compressing the input data, the track segment finder also suppresses noise, since isolated hits cannot produce a track segment. However, it also limits the acceptance of the track trigger. Figure 2 shows the track segment finder efficiency depending on the crossing angle α of the track. An efficiency of $\geq 90\,\%$ is reached for crossing angles smaller than 30°. For crossing angles larger than 45° the track segment finder is not sensitive.

2.2 2D track finder

Tracks are first searched only in the transverse plane by a Hough transform [4], which takes axial track segment hits as input. The position of the reference wire is transformed into a curve in a parameter space that is spanned by the azimuthal angle φ and the track curvature ρ, which is inversely proportional to the transverse momentum. Tracks are found as crossing points of several curves in the parameter space, with the track parameters given by the crossing point coordinates. The transformation is given by

$$\rho(\varphi) = \frac{2}{r_{\mathrm{TS}}} \cdot \sin\left(\varphi - \varphi_{\mathrm{TS}}\right)$$

where $(r_{\mathrm{TS}}, \varphi_{\mathrm{TS}})$ are the polar coordinates of the reference wire. The transformation is illustrated in figure 3. Two crossing points are found for a single track, one corresponding to a track going clockwise, the other corresponding to a track going counter-clockwise. However, due to the limited track segment finder acceptance, curling tracks cannot be found in the first level trigger. Therefore, the crossing point ambiguity can be resolved by removing half of each Hough curve, namely the half that corresponds to tracks curling back through the point. The condition for the remaining outgoing half is given by

$$\varphi \in [\varphi_{\mathrm{TS}} - 90^\circ, \varphi_{\mathrm{TS}} + 90^\circ]$$

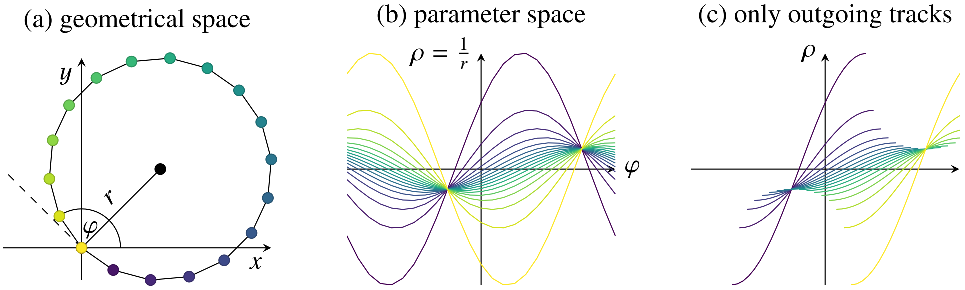

Figure 3. Illustration of the Hough transform. (a) Points on a circle in geometrical space. (b) Corresponding Hough curves in parameter space. (c) The ambiguity of the crossing point is resolved by removing the falling half of each curve.

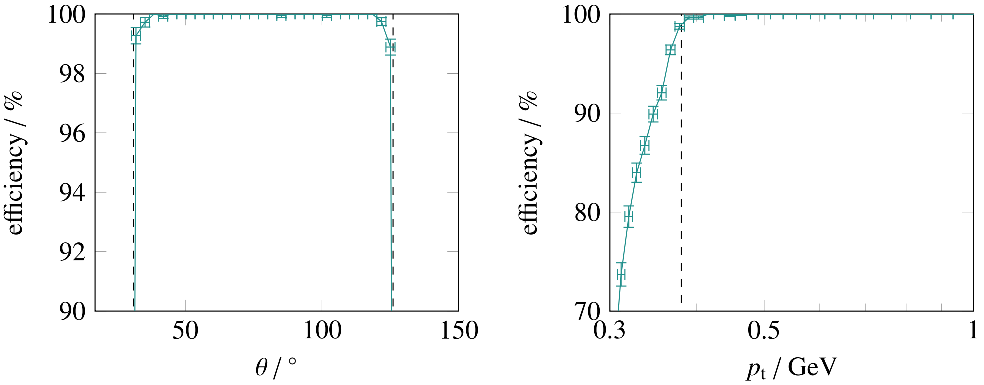

Figure 4. Track finding efficiency depending on the polar angle θ and the transverse momentum p_t, measured on single track events. Within the dashed lines the efficiency is above 99 %. For the θ efficiency, only tracks within the p_t region of full efficiency are taken into account, and vice-versa.

which corresponds to the half of the sine that has a rising slope. The crossing point of the curves for an outgoing track is then unique. The charge of the track is obtained from the crossing point as sign(ρ).

To find the crossing points, a grid of 160×34 cells is defined in the parameter space. For each grid cell, the number of crossing Hough curves from different superlayers is counted in parallel. Crossing point candidates are obtained as grid cells with at least four curves. Neighboring candidates are clustered. The track parameters φ and ρ are then given by the cluster center. The efficiency is shown in figure 4. The polar angle acceptance is given by $\theta \in [31°, 126°]$ and is directly related to the CDC geometry and the requirement of hitting at least four axial superlayers. The dependence on the transverse momentum p_t is related to the crossing angle and the track segment finder efficiency. The full efficiency is obtained for $p_t > 0.38$ GeV, which corresponds to a crossing angle of about 30° in the fourth axial superlayer. The number of grid cells was optimized to get a high track parameter resolution while maintaining a low clone rate. Clones occur when clusters have gaps, such that two clusters are found instead of one. With the selected setup, the clone rate is only 0.16 %.

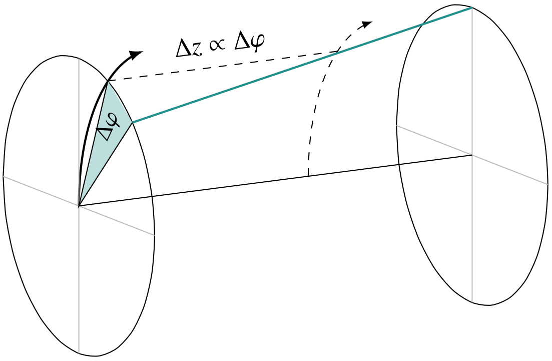

Figure 5. z-coordinates are obtained by projecting the distance $\Delta\varphi$ of a stereo wire end point from the 2D track along the stero wire.

2.3 Track vertex reconstruction

The last stage of the track trigger is a full 3D reconstruction for each track found by the Hough finder. To include the drift times in this stage, the event time needs to be known. It is determined in parallel to the track finding by comparing the "fastest timings" of all track segments, which are given by the drift time of the first hit in the track segment. Longitudinal hit coordinates can be obtained from the stereo superlayers by calculating the crossing point of a wire with the 2D track, as illustrated in figure 5. Then the longitudinal coordinates of all stereo superlayers can be combined to reconstruct the polar angle and the z-vertex of the track. The method presented here employs neural networks to estimate the 3D track parameters. Earlier versions are documented in [5–7]. In addition, an independent algorithm based on a least squares fit is being developed [8], which will run in parallel to the neural network. The optimal way to combine the results of the two algorithms for the final trigger decision has yet to be studied.

For the input of the neural network, one hit is selected for each superlayer. Hit candidates have to be within a certain φ region around the 2D track and within a certain time window. If there are several hit candidates, hits with known left/right passage and the shortest drift time are selected. Then three input values are calculated for each hit: the distance $\Delta\varphi$ between the 2D track and the wire end point, the drift time with a sign given by the left/right passage and the crossing angle α of the 2D track, which contains information about the track curvature. Using a separate input for the drift time allows the network to learn nonlinear corrections for the drift velocity and to take the crossing angle into account. Both axial and stereo hits are used. Although axial hits contain no 3D information, they allow to improve the 2D track estimate, which was obtained without drift times.

In total 27 input values are obtained, which are scaled to a common interval of $[-1, 1]$ and fed into a Multi Layer Perceptron (MLP) [9] with a single hidden layer of 127 nodes and two output nodes, as shown in figure 6. The three layer feed-forward structure is particularly suited for parallel execution with a deterministic runtime, as required by the trigger. The size of the hidden layer was optimized such that a further increase does not significantly improve the resolution of the output values. Each hidden and output neuron computes a weighted sum of its inputs and and evaluates it with an activation function:

$$y_j = \tanh\left(\sum_{i=0}^{N} x_i \cdot w_{ij}\right)$$

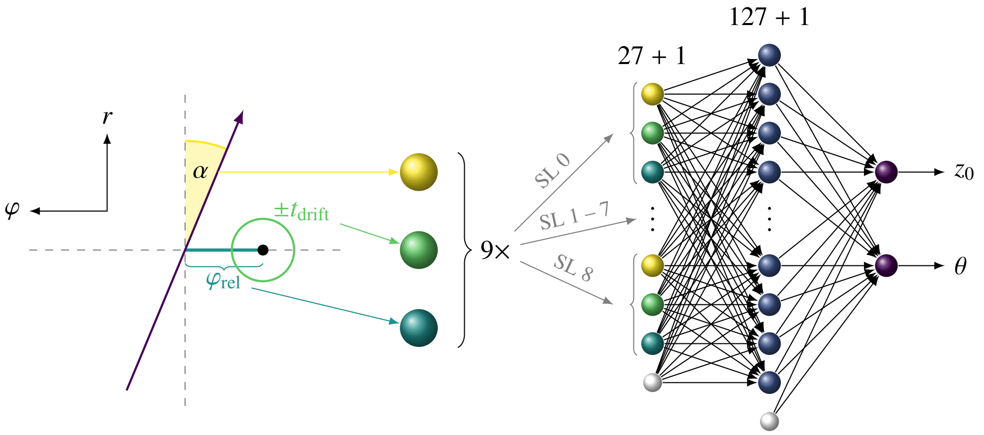

Figure 6. Illustration of the input and the structure of the MLP. The small white nodes denote bias nodes with a fixed input of 1.

Here, the x_i are the N input values for the neuron with output y_j, w_{ij} are the weights and the first weight w_{0j} acts as a constant bias, with $x_0 = 1$. The weights are trained using the backpropagation algorithm iRprop$^-$ [10] as implemented in the open source library FANN [11], with the z-vertex and the polar angle θ as continuous targets. To match the co-domain of the activation function, the target values are scaled to the interval $[-1, 1]$. To obtain the track parameters with the correct units, the inverse scaling is applied to the network outputs. Note that the scaling procedure requires a cut-off during the training, using only tracks with $|z| \leq 50\,\text{cm}$.

For superlayers where no hits pass the hit selection criteria, the corresponding input values are missing. For axial layers, a default input tuple is defined as $(0, 0, 0)$, which can be interpreted as "no 2D correction for this superlayer". For stereo layers, no reasonable default values exist, as any input would be interpreted as some z-coordinate. Therefore, independent networks are trained for the case of missing stereo hits, which receive only 24 input nodes. In total five such "expert" networks are trained, one for 4 stereo hits and four for 3 stereo hits, depending on which hit is missing. For tracks with less than 3 stereo hits no 3D reconstruction is possible, which happens for $\approx 1\,\%$ of the tracks.

3 Performance

For single tracks with a momentum distribution uniform in the solid angle and the track curvature, an average resolution of 2.9 cm for the z-vertex and 3.1° for the polar angle is obtained. The resolution depends on the transverse momentum, with a z-vertex resolution of 2.3 cm for high momentum tracks and 4.3 cm for momenta at the edge of the track finding acceptance. In the following, a cut is applied to the estimated z-vertices of physics and background tracks, to demonstrate the rejection power and efficiency of a z-vertex trigger.

3.1 Background rejection

For the machine background simulation, particles are generated whose energy and momentum deviate slightly from the design values, with the exact distribution depending on the background type. Then

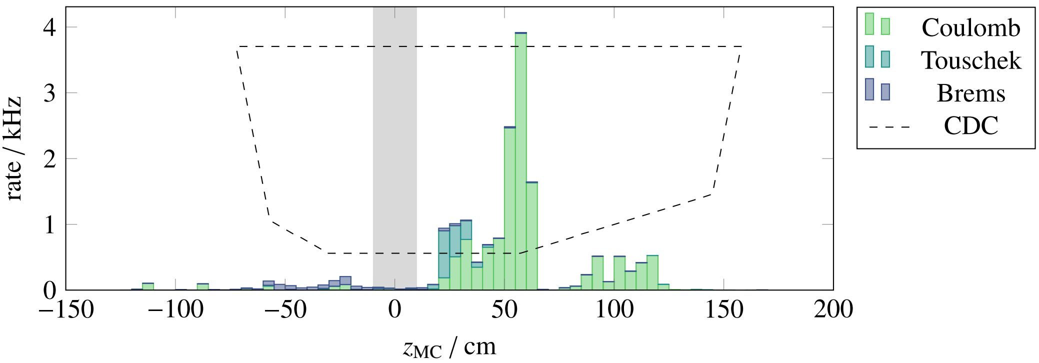

Figure 7. True z-vertex distribution of background tracks found by the trigger. The outline of the CDC is shown as a scale reference. The shaded region shows ±10 cm around the interaction point.

the particles are tracked through the accelerator ring. When particles deviate far enough from the design orbit that they hit the beampipe or other machine elements, the loss position and momentum are recorded. The initial energy and momentum deviations are simulated for three types of reactions: Touschek scattering within the bunches, elastic scattering with residual gas ("Coulomb") and inelastic scattering ("Brems"). Then a full detector simulation is done for particles that are lost in the vicinity of the detector. Figure 7 shows the distribution of true z-vertices for tracks found in the trigger. Mostly, those tracks correspond to secondary particles that are created when a lost particle is scattered in some detector structure. The integrated background track rate without a z-vertex veto is 17 kHz, which is more than half of the maximal total trigger rate. By rejecting tracks whose estimated z-vertex is outside of ±10 cm, the background rate can be reduced to 0.2 kHz. Note that a very efficient rejection is achieved also for tracks with $|z| > 50$ cm, which were not included in the training.

3.2 Track trigger efficiency

The efficiency is measured both for generic $\mathrm{B\overline{B}}$ events and for $\tau^+\tau^-$ events, which are studied as an example for events with low multiplicity. In addition to generic τ decays, a specific channel is investigated, namely the flavor violating decay $\tau{\rightarrow}\mu\gamma$. Table 1 summarizes the track trigger efficiency for different trigger conditions. For three or more tracks no additional constraint is imposed. For events with two tracks, different z-vertex constraints are compared: either both tracks are required to come from the interaction point (IP) or at least one of them, where an IP track is characterized by an estimated z-vertex within ±10 cm. For comparison, the pure two track efficiency is also shown. For two IP tracks, the efficiency is relatively low compared to a pure two track trigger. The losses are due to tracks with low momentum, missing hits or wrong hit selection. Opening the z-vertex cut reduces these losses only slightly and increases the background rate. A better trade-off is obtained by requiring only one of the two tracks to originate at the interaction point. Then the efficiency is very close to a pure two track trigger.

Table 1 also shows the importance of a low multiplicity trigger for τ physics studies. Compared to a three track trigger, the efficiency is increased by a factor of 3 for generic τ events and a factor of 4.6 for the flavor violating decay mode.

Table 1. Efficiency for example event types under different trigger conditions. A certain total number of tracks is required as well as a certain number of tracks with a z-vertex within ±10 cm ("IP tracks"). The last line shows the combined efficiency of ≥ 3 tracks or 2 tracks with IP constraint. The efficiency is calculated relative to the total number of simulated events.

trigger condition		efficiency		
# tracks	# IP tracks	$B\overline{B}$	$\tau^+\tau^-$	$\tau\to\mu\gamma$
≥ 3	≥ 0	93.9 %	18.7 %	12.4 %
2	≥ 0	4.6 %	38.7 %	44.4 %
2	≥ 1	4.4 %	38.1 %	44.1 %
2	2	2.9 %	33.5 %	40.0 %
3 or 2 with ≥ 1 IP		98.3 %	56.8 %	56.5 %

4 Conclusion

The first level track trigger of the Belle II experiment will offer a full 3D reconstruction. In particular, the z-vertex reconstruction allows to suppress the expected machine background by two orders of magnitude. This reduction is essential to open the trigger for low multiplicity events. For a hypothetical decay channel $\tau\to\mu\gamma$ an efficiency of 56.5 % is obtained for a two track trigger. This efficiency will be further increased to about 90 % by including other subtriggers, such as the calorimeter trigger and a combined track-calorimeter logic.

The neural network logic has been implemented on FPGA and verified on simulated events [12]. The neural network trigger is ready to be installed and will be tested on cosmic rays during the summer of 2017.

Acknowledgements

This work has been supported by the German Federal Ministry of Education and Research (BMBF) under grant number 05H15WOKBA. The machine background simulations have been carried out on the computing facilities of the Computational Center for Particle and Astrophysics (C2PAP) within the Excellence Cluster Universe.

References

[1] T. Abe, I. Adachi, K. Adamczyk, S. Ahn, H. Aihara, K. Akai, M. Aloi, L. Andricek, K. Aoki, Y. Arai et al. (Belle II) (2010), Belle II Technical Design Report, `http://arxiv.org/abs/1011.0352`

[2] Y. Ohnishi, T. Abe, T. Adachi, K. Akai, Y. Arimoto, K. Ebihara, K. Egawa, J. Flanagan, H. Fukuma, Y. Funakoshi et al., Progress of Theoretical and Experimental Physics **2013**, 03A011 (2013)

[3] C. Bernardini, G.F. Corazza, G. Di Giugno, G. Ghigo, J. Haissinski, P. Marin, R. Querzoli, B. Touschek, Phys. Rev. Lett. **10**, 407 (1963)

[4] R.O. Duda, P.E. Hart, Commun. ACM **15**, 11 (1972)

[5] S. Skambraks, Diploma thesis, Ludwig-Maximilians-Universität, München (2013), `https://publications.mppmu.mpg.de/2013/MPP-2013-377/FullText.pdf`

[6] S. Skambraks, F. Abudinen, Y. Chen, M. Feindt, R. Frühwirth, M. Heck, C. Kiesling, A. Knoll, S. Neuhaus, S. Paul et al., Nuclear Science, IEEE Transactions on **62**, 1732 (2015)

[7] S. Neuhaus, S. Skambraks, F. Abudinen, Y. Chen, M. Feindt, R. Frühwirth, M. Heck, C. Kiesling, A. Knoll, S. Paul et al., Journal of Physics: Conference Series **608**, 012052 (2015)

[8] J.B. Kim, Master thesis, Korea University (2012), `http://dcollection.korea.ac.kr/jsp/common/DcLoOrgPer.jsp?sItemId=000000033908`

[9] R. Hecht-Nielsen, *Theory of the backpropagation neural network*, in *1989 IJCNN, International Joint Conference on Neural Networks* (1989), pp. 593–605

[10] C. Igel, M. Hüsken, *Improving the Rprop Learning Algorithm*, in *Proceedings of the Second International Symposium on Neural Computation* (2000), pp. 115 – 121

[11] S. Nissen, Report, Department of Computer Science University of Copenhagen (DIKU) **31**, 29 (2003), code available at `https://github.com/libfann/fann`

[12] S. Bähr, S. Skambraks, S. Neuhaus, C. Kiesling, J. Becker, *A neural network on FPGAs for the z-vertex track trigger in Belle II*, in *Topical Workshop on Electronics for Particle Physics (TWEPP) 2016* (2017), Vol. 12, p. C03065