Full Event Interpretation at Belle II
Why we need Full Event Interpretation

- Interesting physics can be obtained from several challenging modes with missing neutrinos ($B \rightarrow D^{(*)}\tau\nu$, $B \rightarrow l\nu$, $B \rightarrow X_u l\nu$, $B \rightarrow h\nu\bar{\nu}$)
Introduction

Why we need Full Event Interpretation

- Interesting physics can be obtained from several challenging modes with missing neutrinos ($B \rightarrow D^{(*)} \tau \nu$, $B \rightarrow l \nu$, $B \rightarrow X \nu$, $B \rightarrow h \nu \bar{\nu}$)
Tag-side $B$ reconstruction

- Collide $e^+$ and $e^-$ at the energy to make $\Upsilon(4S)$ particles
Tag-side $B$ reconstruction

- Collide $e^+$ and $e^-$ at the energy to make $\Upsilon(4S)$ particles.
- $\Upsilon(4S)$ decays to $B^+B^-$ and $B^0\bar{B}^0$ 96% of the time.
Tag-side $B$ reconstruction

- Collide $e^+$ and $e^-$ at the energy to make $\Upsilon(4S)$ particles.
- $\Upsilon(4S)$ decays to $B^+B^-$ and $B^0\bar{B}^0$ 96% of the time.
- Reconstruct one $B$ meson as tag-side ($B_{tag}$) hadronic or SL

\[ B^- \rightarrow e^- \]
\[ B_{tag}^+ \rightarrow \bar{D}^0 \]
\[ \pi^+ \rightarrow K^+ \]

\[ \Upsilon(4S) \]

\[ e^+ \rightarrow e^- \]

\[ \nu_l \rightarrow \bar{\nu}_l \]
Collide $e^+$ and $e^-$ at the energy to make $\Upsilon(4S)$ particles.

$\Upsilon(4S)$ decays to $B^+B^-$ and $B^0\bar{B}^0$ 96% of the time.

Reconstruct one $B$ meson as tag-side ($B_{\text{tag}}$) hadronic or SL.

Study remaining $B$ meson as signal ($B_{\text{sig}}$).
Collide $e^+$ and $e^-$ at the energy to make $\Upsilon(4S)$ particles.

$\Upsilon(4S)$ decays to $B^+B^-$ and $B^0\bar{B}^0$ 96% of the time.

Reconstruct one $B$ meson as tag-side ($B_{\text{tag}}$) hadronic or SL

Study remaining $B$ meson as signal ($B_{\text{sig}}$)

Flavour constraints:

$B_{\text{tag}}^+ \rightarrow B_{\text{sig}}^-$

Kinematic constraints:

$p_{\nu} = p_{e^+e^-} - p_{\mu^-} - p_{B^+}$
Collide $e^+$ and $e^-$ at the energy to make $\Upsilon(4S)$ particles.

$\Upsilon(4S)$ decays to $B^+B^-$ and $B^0\bar{B}^0$ 96% of the time.

Reconstruct one $B$ meson as tag-side ($B_{tag}$) hadronic or SL.

Study remaining $B$ meson as signal ($B_{sig}$).

Flavour constraints:

$B_{tag}^+ \rightarrow B_{sig}^- $

Kinematic constraints:

$p_\nu = p_{e^+e^-} - p_{\mu^-} - p_{B^+}$
Which tag-side reconstruction?

Tagging techniques

<table>
<thead>
<tr>
<th>Inclusive</th>
<th>Semileptonic</th>
<th>Hadronic</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B \rightarrow \text{anything}$</td>
<td>$B \rightarrow D^*(\ell \nu_\ell)$</td>
<td>$B \rightarrow \text{hadrons}$</td>
</tr>
<tr>
<td>$\epsilon \approx O(100%)$</td>
<td>$\epsilon \approx O(1%)$</td>
<td>$\epsilon \approx O(0.1%)$</td>
</tr>
</tbody>
</table>

Purity

Efficiency

Very large statistics; Also very large background

Mid-range reconstruction efficiency; Less information about $B_{\text{tag}}$ due to neutrino

Cleaner sample

Knowledge of $p(B_{\text{sig}})$; Lower tag-side efficiency
The Task

- Must reconstruct a substantial number of modes (e.g. O(50) hadronic modes of $B$ mesons)
- Face combinatorics which scale as the factorial of the number of tracks $\Rightarrow$ Require selections early on
The Task

- Must reconstruct a substantial number of modes (e.g. $O(50)$ hadronic modes of $B$ mesons)
- Face combinatorics which scale as the factorial of the number of tracks $\rightarrow$ Require selections early on
Combinatorics

- ~10 tracks in this event
- Let’s assume 5 positively charged and 5 negatively charged.
- Now let’s reconstruct $D^0 \rightarrow K^- \pi^+ \pi^+ \pi^-$
- $\binom{5}{2}^2 = 100$ possible combinations
- Reconstructing $B^+ \rightarrow (D^0 \rightarrow K^- \pi^+ \pi^+ \pi^-) \pi^+$ introduces $\binom{3}{1} \times 100 = 300$ combinations.
Utilises $O(200)$ decay channels with classifiers (BDTs) trained for each.

Reconstructs $O(10000)$ unique decays chains in six stages.
Classifiers

- Describe decay channels with features $x_i$.
- Labelled training datasets obtained from simulation.
- Train classifiers for each channel, $\mathcal{P}(x_i)$
- The FEI by default uses Boosted Decision Trees.

![Decision Tree Diagram]

![Scatter Plots]
Classifiers

- Describe decay channels with features $x_i$.
- Labelled training datasets obtained from simulation.
- Train classifiers for each channel, $\mathcal{P}(x_i)$
- The FEI by default uses Boosted Decision Trees.
Classifiers

- Describe decay channels with features $x_i$.
- Labelled training datasets obtained from simulation.
- Train classifiers for each channel, $P(x_i)$
- The FEI by default uses Boosted Decision Trees.
Classifiers

- Describe decay channels with features $x_i$.
- Labelled training datasets obtained from simulation.
- Train classifiers for each channel, $\mathcal{P}(x_i)$
- The FEI by default uses Boosted Decision Trees.

```
x = 2
y = 5

Is y negative?
```

```
x < 0
y < 0
x < 4
y < 0
x < 8
```

```
x < -5
y < 3
0.1
0.2
0.3
0.8
0.4
0.7
0.5
0.9
```

Graphs showing data distribution in $x$ and $y$.
Classifiers

- Describe decay channels with features $x_i$.
- Labelled training datasets obtained from simulation.
- Train classifiers for each channel, $P(x_i)$
- The FEI by default uses Boosted Decision Trees.

![Decision Tree Diagram]

[x = 2
y = 5

Returned probability is 0.7]

![Scatter Plots]

[x = 2
y = 5

Returned probability is 0.7]
The Algorithm

- Particle candidates assigned from tracks and clusters after a precuts + Best Candidate Selection (BCS).

- For each particle, a pre-trained BDT is applied and post cuts + BCS are made.

- Stable particles are combined to reconstruct decays of intermediate particles. After precuts + BCS, a vertex fit is performed.

- Intermediate classifiers use daughter kinematics and classifiers.

- Intermediates and stable particles are combined into a $B^-$ candidate.

- $B^-$ classifier takes daughter classifiers and kinematics as inputs.
Particle candidates assigned from tracks and clusters after a precuts + Best Candidate Selection (BCS).

For each particle a pre-trained BDT is applied and post cuts + BCS are made.
The Algorithm

- Particle candidates assigned from tracks and clusters after a \textit{precuts} + Best Candidate Selection (BCS).

- For each particle a pre-trained BDT is applied and post cuts + BCS are made.

- Stable particles are \textit{combined} to reconstruct decays of intermediate particles. After \textit{precuts} + BCS a \textit{vertex fit} is performed.
The Algorithm

- Particle candidates assigned from tracks and clusters after a precuts + Best Candidate Selection (BCS).
- For each particle a pre-trained BDT is applied and post cuts + BCS are made.
- Stable particles are combined to reconstruct decays of intermediate particles. After precuts + BCS a vertex fit is performed.
- Intermediate classifiers use daughter kinematics and classifiers.
The Full Event Interpretation

The Algorithm

- Particle candidates assigned from tracks and clusters after a precuts + Best Candidate Selection (BCS).
- For each particle a pre-trained BDT is applied and post cuts + BCS are made.
- Stable particles are combined to reconstruct decays of intermediate particles. After precuts + BCS a vertex fit is performed.
- Intermediate classifiers use daughter kinematics and classifiers.
- Intermediates and stable particles are combined into a $B$ candidate.
The Algorithm

- Particle candidates assigned from tracks and clusters after a precuts + Best Candidate Selection (BCS).
- For each particle a pre-trained BDT is applied and post cuts + BCS are made.
- Stable particles are combined to reconstruct decays of intermediate particles. After precuts + BCS a vertex fit is performed.
- Intermediate classifiers use daughter kinematics and classifiers.
- Intermediates and stable particles are combined into a $B$ candidate.
- $B$ classifier takes daughter classifiers and kinematics as inputs.
The Full Event Interpretation

The Algorithm

- Same $B^+ \rightarrow D^0 \pi^+$ classifier.
- Different decay chain as $D^0 \rightarrow K^0_s \pi^0$.
- $D^0 \rightarrow K^0_s \pi^0$ has its own classifier.
The Full Event Interpretation

The Algorithm

- Different \( B^+ \rightarrow D^0 \pi^+ \pi^0 \) decay with its own classifier.
- Original \( D \) decay chain as \( D^0 \rightarrow K^- \pi^+ \).

\[
\begin{align*}
\gamma & \rightarrow \pi^0 \\
\gamma & \rightarrow \pi^0 \\
\pi^+ & \rightarrow B^+ \\
\pi^+ & \rightarrow D^0 \\
K^- & \rightarrow D^0
\end{align*}
\]
Both training and application phases can be distributed via a map reduce approach.

For training:
- $O(100M)$ simulated $\Upsilon(4S) \to B\bar{B}$ events
- Monte carlo is partitioned and processed at different nodes.
- At each of the reconstruction phases training data is generated.
- Training data of each stage is subsequently merged and classifiers trained.
The Full Event Interpretation

Need for speed

- Utilise FastBDT:
  - Computes cumulative probability histograms (CPH) of nodes in the same level simultaneously.
  - Stores data as an array of structs.

  arxiv1609.06119, Keck, T.

- Utilise FastFit (GitHub link):
  - Uses eigen libraries to gain from vectorisation.
  - Overall factor of 2.7 speed up in the FEI

- In application 38% of the time is spent on vertex fitting, 27% on particle combination and 15% on classifier inference.
Specific vs Generic FEI

- **Generic FEI** - Reconstruct signal after reconstructing a tag-side $B$ candidate.
- **Specific FEI** - Reconstruct a tag-side $B$ candidate after reconstructing signal

![Diagram of B decay processes]

- $B_{tag}^{+} \rightarrow \bar{D}^0 \pi^+ \pi^+ \pi^-$
- $B_{sig}^{-} \rightarrow \tau^- \bar{\nu}_\tau \ell e^- \bar{\nu}_e \nu_\tau$
- $\rho_{miss}$

Fig. 1: Schematic overview of a $(4S)$ decay: (Left) tag-side (
\( B_{tag} \)) and (right) a typical signal-side decay (\( B_{sig} \)). Tags can be easily discarded and the final purity depends strongly on the considered signal decay-channel.
How does one quantify tagging performance?

- **Tagging efficiency**
  \[ N_{\text{tag}} / N_{\gamma(4S)} \]

- **Tagging side efficiency**
  \[ N_{\text{correct}} / N_{\gamma(4S)} \]

- **Purity**
  \[ N_{\text{correct}} / N_{\text{tag}} \]
How does one quantify tagging performance?

- **Tagging efficiency** = \( \frac{N_{\text{tag}}}{N_{\gamma(4S)}} \)
How does one quantify tagging performance?

- **tagging efficiency** $= \frac{N_{\text{tag}}}{N_{\gamma(4S)}}$
- **tag-side efficiency** $= \frac{N_{\text{correct}}}{N_{\gamma(4S)}}$
How does one quantify tagging performance?

- **tagging efficiency** = \( \frac{N_{\text{tag}}}{N_{\gamma(4S)}} \)
- **tag-side efficiency** = \( \frac{N_{\text{correct}}}{N_{\gamma(4S)}} \)
- **purity** = \( \frac{N_{\text{correct}}}{N_{\text{tag}}} \)
Tagging performance in Belle data

$$m_{bc} = \sqrt{E_B^2 - p_B^2}$$

$$E_B = \sqrt{s}/2$$

Different event topologies

- Spherical
- Jet-like

$B$-mesons Continuum

![Graphs showing tagging performance](image-url)
Tagging performance

Tag-side efficiency again purity in Belle data

Maximum tag-side efficiency

<table>
<thead>
<tr>
<th>Tag</th>
<th>FR</th>
<th>SER</th>
<th>FEI Belle</th>
<th>FEI Belle II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hadronic $B^+$</td>
<td>0.28%</td>
<td>0.4%</td>
<td>0.76%</td>
<td>0.66%</td>
</tr>
<tr>
<td>Hadronic $B^0$</td>
<td>0.18%</td>
<td>0.2%</td>
<td>0.46%</td>
<td>0.38%</td>
</tr>
<tr>
<td>SL $B^+$</td>
<td>0.31%</td>
<td>0.3%</td>
<td>1.80%</td>
<td>1.45%</td>
</tr>
<tr>
<td>SL $B^0$</td>
<td>0.34%</td>
<td>0.6%</td>
<td>2.04%</td>
<td>1.94%</td>
</tr>
</tbody>
</table>

FR = Full Reconstruction (Belle Algorithm), SER = Semi-Exclusive Reconstruction (BaBar Algorithm)
The Full Event Interpretation (FEI) is an algorithm for tag-side $B$ reconstruction at Belle 2.

It trains $O(200)$ decay channel classifiers which are used in the reconstruction of $O(10000)$ decay chains.

The FEI outperforms its predecessors with a higher tag-side efficiency.

The FEI is an essential to the Belle II physics program and resolving the $B$ physics anomalies.

Machine learning algorithms for the Belle II experiment and their validation on Belle data - Thomas Keck https://publikationen.bibliothek.kit.edu/1000078149

Analysis Software and Full Event Interpretation for the Belle II Experiment - Christian Pulvermacher http://ekp-invenio.physik.uni-karlsruhe.de/record/48741

Log in to kekcc with port forwarding:
ssh username@login.cc.kek.jp -L 8300:localhost:8300

Note that you should choose a different port.

Clone repository at https://stash.desy.de/users/sutclw/repos/feitutorial/browse:
git clone ssh://git@stash.desy.de:7999/∼sutclw/feitutorial.git

Within the tutorial directory run: source setup_basf2_rel3.sh

Run jupyter note book with the following command:
jupyter-notebook --port 8300 --no-browser

If problems see:
https://confluence.desy.de/display/BI/Running+Jupyter+Notebook+on+KEKCC