

Deterministic Annealing for Vertex Finding at CMS

E. Chabanat, N. Estre, IPNL-Université Claude Bernard Lyon 1, Villeurbanne, France

Abstract

The CMS detector and other LHC experiments offer a new challenge for the vertex reconstruction, the elaboration of efficient algorithms at high-luminosity beam collisions. This note presents a new approach, the deterministic annealing. Deterministic annealing is a heuristic algorithm which comes from information theory. The principle is described in analogy to statistical physics. The simulated performance for vertex identification, with the CMS detector, is presented. The results are compared to those obtained with the CMS reference algorithm.

INTRODUCTION

The vertex finding problem can be seen as a clustering problem, in which each vertex is a cluster of tracks. The vertex reconstruction can be decomposed into two sequential steps, (i) the search of the best partition of tracks (the clustering), and (ii) the fitting of each set of tracks to a common vertex. The DA (Deterministic Annealing [1]) approach to clustering has demonstrated substantial performance improvements over standard supervised and unsupervised learning methods in variety of important applications including compression, pattern recognition and classification, and statistical regression. For the vertex finding, this method offers an important feature, the ability to find vertices in a noisy environment without a previous knowledge of the number of vertices to be found.

THE DETERMINISTIC ANNEALING

The main algorithm

Let X or $\{x\}$ be a set of inputs (each x can be a representation of a track) and Y the best representation of X (it can be the output of a fit). The set Y is often called the set of prototypes $\{y\}$. In a probabilistic framework, the distortion measurement can be defined as:

$$D = \sum_{x,y} p(x)p(y|x)d(x,y(x)), \quad (1)$$

where $d(x,y(x))$ is a distance between x and its associated prototype y , $p(x)$ is the weight of the input x and $p(y|x)$ the conditional probability of y given x . The smallest distortion search is recasted as that of seeking the distribution $p(y|x)$ which minimizes F

$$F = D - TS, \quad (2)$$

where S is the joint entropy

$$S(X,Y) = - \sum_x \sum_y p(x,y) \log p(x,y). \quad (3)$$

T is a Lagrange multiplier called “temperature”, in analogy with statistical physics. Only the main parts of the algorithm are given here, see reference [1] for more detailed calculations. In order to determine the analytical form of the association probabilities, the functional derivative of F with respect to $p(y|x)$ is calculated and gives the Gibbs distribution when equals zero.

$$p(y|x) = \frac{\exp\left(-\frac{d(x,y)}{T}\right)}{Z_x} \quad (4)$$

with the canonical partition function

$$Z_x = \sum_y \exp\left(-\frac{d(x,y)}{T}\right). \quad (5)$$

These equations lead, minimizing F with respect to y , to the prototypes positions

$$y = \frac{\sum_x x p(x)p(y|x)}{\sum_x p(x)p(y|x)} \quad (6)$$

where $p(y|x)$ is the Gibbs distribution given by (4).

Phase transitions

Last section shows how to obtain the positions of the set of the prototypes Y . One of the main advantages of this algorithm is that there is no need for a prior knowledge of the good number of prototypes. The phase transitions mechanism allows, starting from a unique prototype at high temperature, to create new prototypes progressively while temperature decreases, reaching critical temperatures along the annealing schedule. First, the case of very high temperature is considered. The association probabilities (4) are uniform and the equation (6) becomes

$$y = \frac{1}{N_x} \sum_x x. \quad (7)$$

The only prototype is then placed at the center of gravity of the x distribution. Hence, at high temperature, the output set collapses on a single cluster containing all the input set with very uniform association probabilities. Instead of considering the number of prototypes as being the number

of the searched clusters, let's consider that the number of prototypes is indefinite and that the clusters are represented by a set of effective prototypes. The number of clusters is then given by the number of effective prototypes and the number of prototypes never change. This definition allows to consider variable numbers of clusters with a constant set Y of prototypes. The phase of the system is described by the number of effective prototypes.

We know from statistical physics that phase transitions appears when the minimum of the free energy is no longer stable when applying some perturbation on the system. If $Y + \epsilon\Psi = \{y + \epsilon\psi_y\}$ denotes the perturbed set of prototypes, where ψ_y is the perturbation applied to the prototype y , the condition for a phase transition is obtained by the following requirement on the second derivative of the free energy with respect to the perturbation strength ϵ :

$$\frac{d^2}{d\epsilon^2} F^*(Y + \epsilon\Psi)|_{\epsilon=0} = 0 \quad \forall \Psi. \quad (8)$$

After some differentiations [1], a prototype y_0 undergoes a phase transition and splits if:

$$\det \left[I - \frac{2}{T} C_{x|y_0} \right] = 0 \quad (9)$$

where $C_{x|y} = \sum_x p(x|y)(x-y)(x-y)^T$ is the covariance matrix of the posterior distribution $p(x|y)$.

The critical temperature T_c is therefore determined as

$$T_c = 2\lambda_{\max} \quad (10)$$

where λ_{\max} is the largest eigenvalue of $C_{x|y_0}$. In other words, a phase transition occurs as the temperature is lowered to twice the variance along the principal axis of the cluster. It can be further shown that the prototype splits along the direction of its principal axis. An illustration of this is given in Fig. 1: starting at high temperature (a), T is lowered and phase transitions (shown in (b),(d),(e)) are computed. At low temperature, the clustering stops and the best solution is found (f).

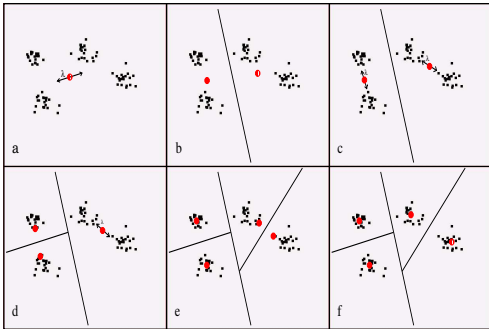


Figure 1: Illustration of a deterministic annealing process for the clustering of a set of points.

APPLICATION TO VERTEX FINDING

Choice of the event topology representation

DA has been implemented in the CMS reconstruction framework ORCA [3]. The first step of the algorithm is to characterize the system topology. The simplest approach is to define the distance between an input and a prototype as the minimal distance between the track and the prototype. However, some tests performed with the Vertex Gun [4] facility (which provides a fully controlled environment with non-realistic vertices) showed a very poor efficiency with this choice. It can be understood, looking at the event topology: for instance, two unphysical vertices, distant from one centimeter and containing respectively nines and four tracks are presented in Fig. 2; the minimal distance d between tracks coming from different vertices is many orders of magnitude smaller than the distance between the two vertices, which is the real dimension of the problem. With this definition of the input-output distance and without taking into account track errors, DA cannot be efficient.

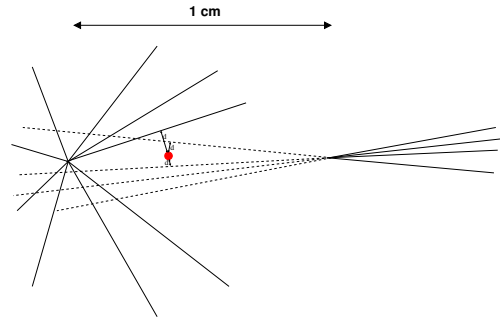


Figure 2: Example of the minimal tracks-prototype distance of two simulated vertices.

A possible solution to this topological problem is the *Apex Point* [2] clustering approach where each track is replaced by a point representative of the surrounding track density. The principle of the computation of the apex point for a track is the following:

- The points of closest approach of any surrounding track to the selected one are calculated.
- Each point of closest approach is transformed into a one dimensional point along the selected track trajectory. It is affected with a weight equal to the inverse of the distance between the two tracks.
- An *Apex Finder* [2] performs, for each track, the search of the apex point(s). An example of concrete implementation is the Minimum Two Values algorithm (MTV). It determines the two neighbour points with the largest weight sum and returns the weighted mean value of their position. It is possible to search for more than one candidate; in this case, the apex finder is called the *Multi State Apex Finder* [2].

- The apex point is re-transformed into a 3 dimensional point belonging to the selected track.

Implementation of the DA algorithm

The implementation of the vertex reconstruction with deterministic annealing on apex points consists in the following steps:

- given a set of reconstructed tracks, all apex points are calculated using the *MTV* [2] algorithm.
- DA is applied to the set of apex points. At the beginning, a single prototype is associated to all apex points. As the temperature is decreased by a definite cooling factor (see below), the conditions of phase transitions are checked, and, if necessary, a new prototype is created. At each step, assignment probabilities and prototypes positions are updated. At the end of the process, when the minimal temperature value T_{\min} is reached, prototypes are fixed and apex points are definitely assigned to one prototype if the weight of assignment is higher than a cut value. Only prototypes which have been associated to more than two apex points are accepted.
- For each prototype, the set of associated apex points is replaced by their corresponding tracks. Each prototype becomes then a vertex seed: a point associated to a set of tracks.
- All vertex seeds are fitted.

With this implementation, DA reconstructs all vertices (primary and secondaries) at the same time, without a previous knowledge of the number of vertex to be found.

SIMULATION RESULTS

Simulated vertices characteristics

Simulated primary and secondary vertices must be distinguished in order to be able to define the reconstruction efficiency for both primary and secondary reconstructed vertices. The ORCA [3] standard method is used:

- a reconstructable vertex is defined as primary if its transverse position is compatible with the beam line with a probability higher than 5%.
- a reconstructable vertex is defined as secondary if its transverse position, is compatible with the beam line with a probability lower than 1%.

Vertex reconstruction performance

In order to optimize the performance with respect to the free parameters of the algorithm, DA has been tested in a realistic environment of displaced vertices. The test framework is the following:

- 4000 $b\bar{b}$ events ($E_t = 100$ GeV, $\eta < 1.4$, without interaction pile-up) have been generated with Pythia [5] (version 6.215) and the detector response has been simulated with the CMS 133 layout [6]. The events reconstruction has been performed using the ORCA_7.6.1 [3] software.
- For each event:
 - tracks are reconstructed with the deterministic annealing filter (DAF) [7]
 - vertex seeds provided by the deterministic annealing process are fitted with the Adaptive Fitter [9]

To evaluate the performance of the algorithm, the following criteria and estimators are used:

- A reconstructed vertex is associated to a simulated vertex if its purity, defined as the ratio of the number of correctly assigned reconstructed tracks to the number of simulated tracks, is strictly higher than 50%.
- **The vertex reconstruction efficiency** is the ratio of the number of reconstructed vertices associated to a simulated vertex to the number of reconstructable simulated vertices. The vertex reconstruction efficiency is estimated separately for primary and secondary vertices.
- **The vertex fake rate** is the ratio of the number of non-associated reconstructed vertices to the total number of reconstructed vertices.

Performances study in a global vertex search

This section presents the results obtained for 4000 $b\bar{b}$ events generated in the detector central region, without interaction pile-up. The reconstruction is done globally, *i.e.* all reconstructed tracks are taken into account by the vertex reconstructor in order to find both primary and secondaries vertices. The summary of the vertex identification performance is shown in table 1 in comparison with the PVR algorithm results. While similar efficiencies are reached with both methods, the lower fake rate of 17% for the DA compared to 44% with the PVR is a significant improvement.

Table 1: Vertex finding performances of the DA and PVR algorithm in a global reconstruction setup

	DA	PVR
PV efficiency	92.75 ± 0.42 %	94.97 ± 0.35 %
SV efficiency	27.80 ± 0.43 %	29.19 ± 0.44 %
Fake rate	17.12 ± 0.42 %	44.01 ± 0.45 %

Primary vertex reconstruction study Resolution values are summarized in table 2. DA produces better resolutions ($19.5 \mu\text{m}$ in transverse plan and $23.5 \mu\text{m}$ on z axis) than PVR: an improvement of $4 \mu\text{m}$ is obtained in transverse plan and $6 \mu\text{m}$ on the z axis.

Table 2: Primary vertex reconstruction resolutions, with DA and PVR.

	DA	PVR
trans. plane res.	$19.5 \pm 0.4 \mu\text{m}$	$23.5 \pm 0.4 \mu\text{m}$
z axis res.	$23.6 \pm 0.4 \mu\text{m}$	$30.2 \pm 0.5 \mu\text{m}$

Secondary vertex reconstruction study Resolutions values are summarized in table 3. DA produces better resolutions (the mean value of σ_1 on the x and y axis is equal to $90\mu\text{m}$ in transverse plan and $78\mu\text{m}$ on z axis) than PVR (an improvement of $7\mu\text{m}$ is obtained in transverse plan and $30\mu\text{m}$ on the z axis).

Table 3: Secondary vertex reconstruction resolution, with DA and PVR.

2-gaussian fit	DA	PVR
σ_1 in transverse	$90 \pm 5 \mu\text{m}$	$97 \pm 4 \mu\text{m}$
σ_1 on z axis	$78 \pm 4 \mu\text{m}$	$110 \pm 7 \mu\text{m}$
σ_2 (tails) transverse	$400 \pm 15 \mu\text{m}$	$445 \pm 16 \mu\text{m}$
σ_2 (tails) z axis	$425 \pm 26 \mu\text{m}$	$500 \pm 25 \mu\text{m}$

Fake vertex reconstruction study A vertex can be considered as fake for two reasons:

- if its purity is lower or equal to 50%, in this case the vertex is called a *real fake*
- if it has been associated to a previously associated simulated vertex, in this case the vertex is called a *twin fake*

The table 4 presents the detailed ratio of real and twin fakes for DA and PVR.

Table 4: Detailed contributions of real fakes and twin fakes to the total fake rate

	Real fake rate	Twin fake rate	Total fake rate
DA	$15.3 \pm 0.4 \%$	$1.8 \pm 0.1 \%$	$17.1 \pm 0.4 \%$
PVR	$37.2 \pm 0.4 \%$	$6.8 \pm 0.2 \%$	$44.0 \pm 0.4 \%$

The main fake production is the real fake production: 90% (resp. 84%) of fakes are real fakes for DA (resp. PVR).

CONCLUSION

The deterministic annealing approach to the vertex finding gives a powerful framework to improve vertex reconstruction. In the first implementation, the deterministic an-

nealing algorithm provides the same efficiency as the classical PVR algorithm for global vertex reconstruction. DA provides a fake rate which is about twice lower.

With the same vertex fitter, DA produces better spatial resolutions than PVR for both primary and secondary vertices and produces less distribution tails.

The limiting factor for this implementation of DA vertex finding are apex point definitions (with ideal apex DA is able to find up to 95% of secondary vertices and produces a very low fake rate). To improve performance a possibility could be to try another approach than the apex point one, and to better account for track errors.

An other application could be the reconstruction of primary vertices with pixel tracks by deterministic annealing.

REFERENCES

- [1] K. Rose, Deterministic Annealing for clustering, compression, classification, regression and related optimization problems, Proceedings IEE, vol. 86, num. 11, p. 2210-2239, 1998.
- [2] R. Frühwirth et al., New vertex reconstruction algorithms for CMS, ePrint physics/0306012.
- [3] <http://cmsdoc.cern.ch/orca/>.
- [4] W. Waltenberger, private communications.
- [5] T. Sjöstrand et al., Pythia 6.2 Physics and Manual, hep-ph/0108264, April 2002.
- [6] C. Charlot et al., CMS Simulation Facilities, CMS TN 93-63, CERN, Geneva.
- [7] R. Frühwirth, T. Todorov and M. Winkler, A comparative study of the Deterministic Annealing Filter at CMS, CMS IN 2003/043.
- [8] The TriDAS project - Technical Design Report, Volume 2: Data acquisition and High Level Trigger, CERN/LHCC 02-26 (2002), ISBN 92-9083-111-4, chapter 14.4, pages 270-276.
- [9] J. D'Hondt, P. Vanlaer, R. Frühwirth and W. Waltenberger, Sensitivity of Robust Vertex Fitting Algorithms, CMS NOTE 2004/002, to be published in IEEE - Transactions on Nuclear Science.