Polarizer simulation method.

The local simulation of the polarizer define two results:

- The output taken by the photon (e or o is used)
- A coefficient proportional to the time taken for the polarizer crossing.

This is done in 2 steps:

- 1: Calculation of an amplitude value depending on the hidden variables of the photon and the angle of the polarizer.
- 2: A test with two threshold values equal to PI/4 and PI/2.

Step 1: **Calculation of the amplitude for the threshold test**.

The photon is modelled using 3 variables noted **p, q, r.**

With:

p : Angle of polarization.
q and r : Two other angles, whose origin is not established.

The value of these three variables are defined by the source during emission with random values between [0..PI] and an uniform distribution.
An amplitude value, noted S, is calculated from these three variables, and is defined as follows:

By setting:

d = p - a_pol

(p: photon polarization)
(a_pol: angle of the polarizer)

Note: “d” is the difference in polarization angle between p/angle of the polarizer. If the value is negative, the polarizer having a PI periodic operation, the positive value d + PI is used.

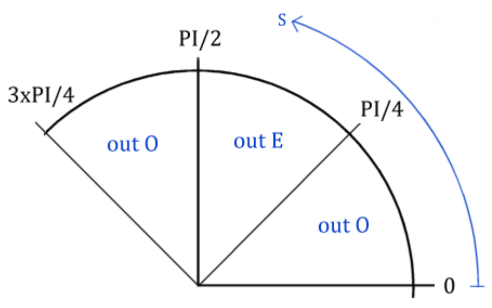
S is then defined by the sum:

S = d/2 + q/6 + r/12

The amplitude of this value varies between 0 and 3*PI/4.

Step 2: Determination of the polarizer output.

It is done with a test comparing the value S with two thresholds values PI/4 and PI/2.
If S between PI/4 and PI/2, the chosen output is e, otherwise it is o.



Coded in C language, this produces the following code:

```
#define OUT_O 0                  // value used to code o out
#define OUT_E 1                  // value used to code e out

int out;
float d, S;

d = pho.p - a_pol;               // a_pol is polarizer angle
if (d < 0)
  d = d + PI;                    // use positive modulus PI value

S = d/2 + pho.q/6 + pho.r/12;

if ((S >= PI/4) && (S < PI/2))
  out = OUT_E;                   // S in PI/4..PI/2 range
else
  out = OUT_O;
```

Note: pho.p, pho.q, pho.r represent the variables p, q, r associated with the photon.

Calculation of the polarizer crossing time.

When it leaves the polarizer, the polarization of the photon is adjusted to the e or o output angle.
This repolarization can reach a maximum of PI/2.

A transit time coefficient can be defined by calculating the variation in polarization occurred between its input and output.

The simulation default use a delay linearly dependent of this coefficient.
By defining in simulation 'st1 delay' parameter not 0, this produce a sigmoid-like shaped transmit time function with an inflection point in PI/4. That produce $\sin^2/\cos^2$ correlations if the size of the pairing window is not enlarged.

Physical interpretation:

It is difficult to know if this local model is only computational, or can represent something existing.
In the second case, it would be necessary to relate the quantities p, q, r to elements of an existing local theory.

We can notice that the output is determined to 2/3 by the difference in polarization. (p – a_diff)
The variables q and r allow for the last third to modulate the primary switching direction.
As the photon move in a 3 dimensional space, the q/6 + r/12 sum is maybe related to a physical magnitudes depending on the direction of the movement.

Note: Use of this algorithm is made on the following web page, describing methods for drawing circles.

http://pierrel5.free.fr/physique/circle/circles.htm (see "Method using randomness and correlations", co_cos() function.)

The code uses a stripped-down version of the algorithm, which makes it more readable than one used into the EPR simulation program.
It also uses the value 1 instead of PI, but that doesn't change the operation.

We can notice in this code that the time coefficients used to cross the polarizer (variables ra and rb) allow, with the help of a test, to determine in advance if the results o / e are correlated.
We can also notice, when there are two possible choices for the output o (S < PI/4, or S > PI/2), the output o taken is that corresponding to a minimum of repolarization, thus seeming to apply the principle of least action.

Conclusion:

This simulation allow to identify the parameter configurations allowing a local method to appear as non-local.
The two aspects studied are the correlation curves and the Eberhard inequality.

Concerning the correlation curves, the local method makes it possible to obtain $\sin^2$ and $\cos^2$ curves of perfect appearance.

This can happen in two ways:

- When the detection tends to less detect the most re-polarized particles. (Preconfigured experiment 6)
- When the pairing window is too narrow. (Preconfigured Experiment 7, 8, and 9)

In a real experiment, a lower rate of simple measurements allowing identical correlations to be obtained allow to rejects the validity of this local method.

Regarding the violation of Eberhard's inequality*, we can distinguish two cases:

1/ Pairing using mobile windows: (Preconfigured experiment 10).

With this method, only the too narrow window condition is sufficient to produce a stable inequality violation.
This confirms that the assumption of fair sampling cannot be made.
This pairing method is therefore very sensitive to out-of-window detections, and is not reliable.
However, this is no longer used in modern experiments.

2 / Pairing by fixed window: (Preconfigured experiment 11).

Under ideal conditions of detection, the local method can produce an inequality violation with a probability of 0.5.
This is due to the stochastic variations of the random sequences.
It is then necessary, especially if the amplitude of the result is close to 0, to repeat the same experiment a large number of times to ensure that the positivity is greater than 0.5, in order to identify the locality.

Under imperfect detection, two conditions can produce a false negative result.
An imperfect source is required, which on an emission request sometimes emits more than one pair of photons. (> = around 2% of requests)
As second condition, the measurements generating multiple detections must be counted as uu. (no detection).

A particular attention must therefore be made on these two conditions.
In this case, with the described local method, it seems impossible to stably obtain inequality violation.

* Also tested with CH, same conditions apply.