Law of Malus and principle of least action.

Pierre Leroy

Abstract: This document proposes to apply a principle of least action to the law of Malus in the context of using a photon model with local variables.
It shows that in this context, a principle of least action does not seem to apply. It proposes an interpretive hypothesis allowing to assume that this principle really applies.

Introduction.

Applied to light, the principle of least action can be used to explain the trajectory of a light ray.

This principle assumes that the dynamic variables evolving along the trajectory are always done in order to minimize an action value dependent on these variables, which has the consequence of defining a unique trajectory.

Since this principle applies to photons, it is assumed that it should also apply to a polarization process allowing the production of Malus' law.

It will then allow, when several possibilities of evolution of the polarization appear possible, to predict which one will be selected.

This situation arises when interacting through a beam splitter polarizer.

In order to approach this subject within the framework of objective physics, a local deterministic polarizer model is used to study how an action value could be defined and minimized from its operating principle.

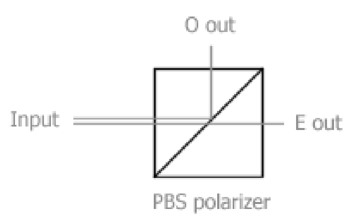


Image1 : Beam splitter polarizer. A principle of least action applied to the polarization process would predict the output taken by a photon as being that which requires the least action.

1. Questions about the law of Malus.

Does a principle of least action seem to apply with the law of Malus ?

Before we can address this question in a local and deterministic framework, we must first determine the supposed effect of a polarizer on a photon.

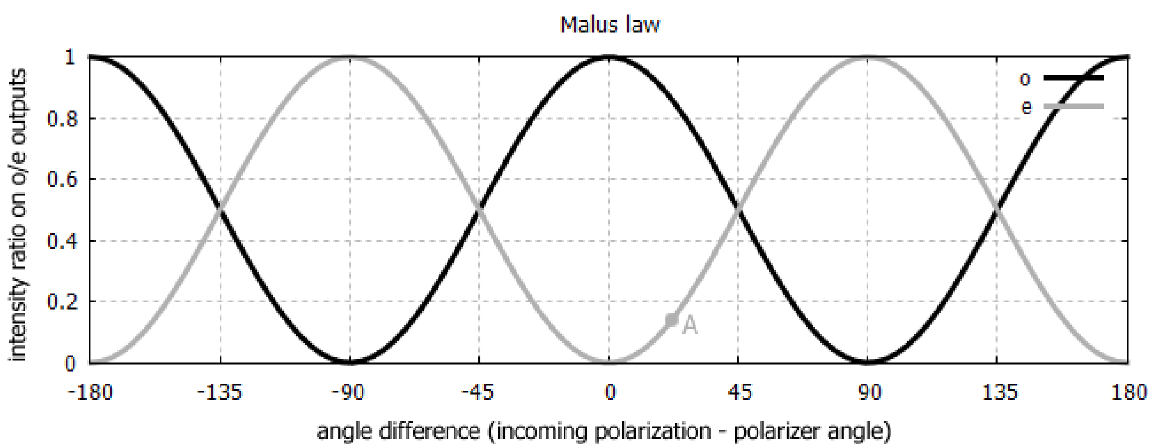
In this framework, it is considered that the polarization value of the photon is at each instant defined by real physical states and that an angle value can then be deduced from these states during an interaction with a polarizer.

It is also considered that this interaction modifies the state of the photon, and that a polarizer does not act only as a filter which would only select two categories of photons by possibly modifying only their trajectory.

In the case of simulating a beam splitter polarizer type, each output is associated with a polarization angle and these angles are offset by π/2.

When a photon passes through the polarizer, its polarization is then aligned with the selected output and can therefore evolve in two different ways between its input and its output.

We can then assume that the selected exit is the one that requires the least action.



Graph 1 : Malus's laws applied to a polarizer with two outputs for an incoming linearly polarized light beam. The o and e curves represent the proportion of photons directed to the o and e outputs as a function of the difference in polarization angle between the incoming beam and the polarizer angle. o = $\cos(ad)^2$, e = $\sin(ad)^2$ with ad = angle difference.

By observing the theoretical Malus law of graph 1, and assuming that the action necessary for the polarization process only depends on the polarization of the photon, can this graph respect a principle of least action?

Assuming that a polarizer only has the effect of aligning the incoming polarization with the output polarization, we see that it is not always the polarizer output that requires minimal adjustment that is selected.

For example for point A in gray on graph 1, the difference in incoming polarization with that of the output axis o is 22.5 degrees, and that with the output axis e is 90 – 22.5 = 67 .5 degrees.

However, a proportion of the photons exits through output e, although this output requires adjustment of its polarization greater than that necessary for output through o.

If the action depended only on polarization, then the output axis of the polarizer should always be the one that requires the least adjustment of the incoming polarization of the photon.

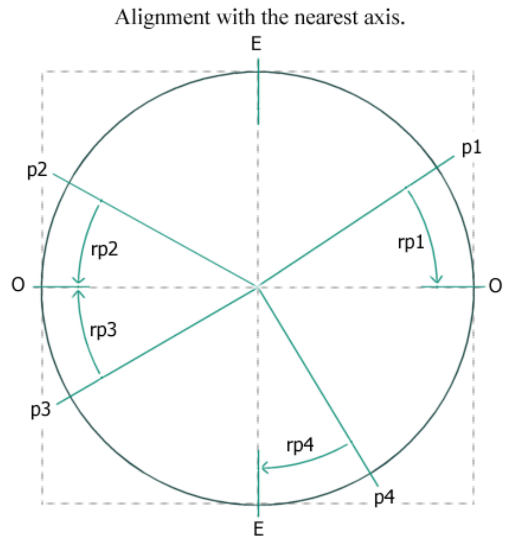


Image 2 : By representing the output axis o of the polarizer horizontally and the output axis e vertically, the polarization of any incoming photon p1..p4 could align with one of the two output axis with a rotation rp1 ..rp4 always less or equal to π/4.

Malus's law therefore does not seem to respect a principle of least action depending solely on the polarization value. It is therefore necessary to consider that it also depends on other variables to apply this principle.

2. A polarizer model.

In order to study a principle of least action not only depending on polarization, a local deterministic simulation model is used. [1]

This model can precisely produce Malus' law through one or more successive polarizers without having to use a random source, and therefore seems plausible in the context of deterministic physics.

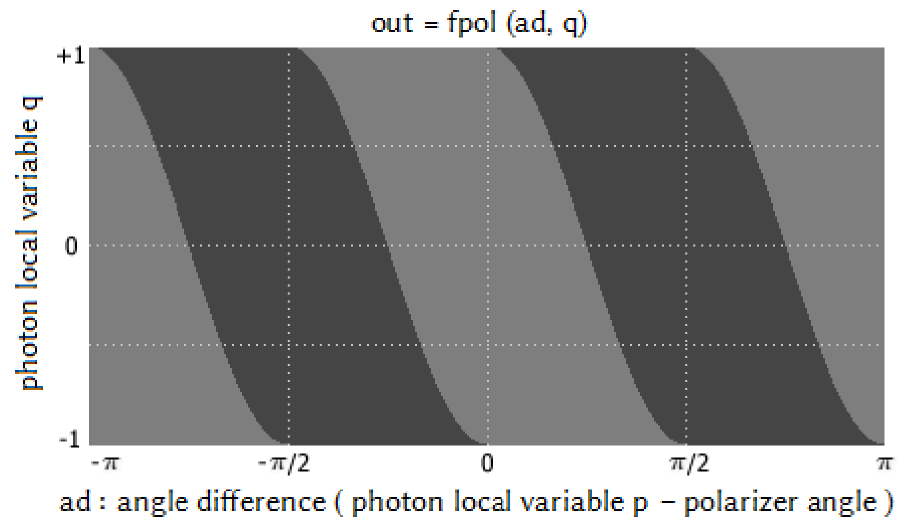
It uses two local variables noted p and q associated with the photon with p representing the polarization angle.

The variable q can take any value with respect to p, but also changes depending on the polarizer output selected.

The choice of polarizer output then depends only on the following two parameters:

- The initial angle difference between p and the polarizer angle.
- The value of q.

By noting “ad” the angle difference, the operation of the polarizer can be summarized by graph 2.



Graph 2 : Polarizer output selection function as a function of ad and q.
The light gray regions designate an output on o and those in dark gray an output on e.
The variable “ad” varies from [-π..π] and designates the angle difference between the local variable p of the photon and the angle of the polarizer.
The variable “q” varies between [−1..1] and designates the value of the local variable q of the photon.

Note that for certain angle differences, ad = 0, ± π/2 and ± π, the value of q has no influence on the selected output, while for angle differences ± π/4 and ± 3*π/4 the selected output depends only on the sign of q.

3. Evaluation of action during polarization.

With this model the values of p and q are altered differently depending on the o/e output selected.

In order to define an action dependent on these two parameters, it is necessary to evaluate the variations of the variables p and q, which will then be denoted “rp” and “dq”.

Variation of p.

The variable p is aligned with the axis angle of the selected output of the polarizer.
The variation of p, denoted “rp”, can then take two possible values.

By noting p_out the polarization angle associated with the selected output o or e, we can write.

rp = p – p_out (eq. 1)

This value varies in the interval ± π/2.

Variation of q.

This depends on rp and the initial value of q.

It can be evaluated from the equation defining the new value of q at the output of the polarizer described in graph 2 of the polarizer document. [1]

This output value, noted q2, is defined as a function of rp as follows:

- if $rp > 0$ $q2 = (q + 1) / \cos(rp)^2 - 1$
- if $rp < 0$ $q2 = (q - 1) / \cos(rp)^2 + 1$

We can then evaluate the variation dq by noting $dq = q2 - q$, and we obtain:

- if $rp > 0$ $dq = (q + 1) / \cos(rp)^2 - 1 - q$
- if $rp < 0$ $dq = (q - 1) / \cos(rp)^2 + 1 - q$

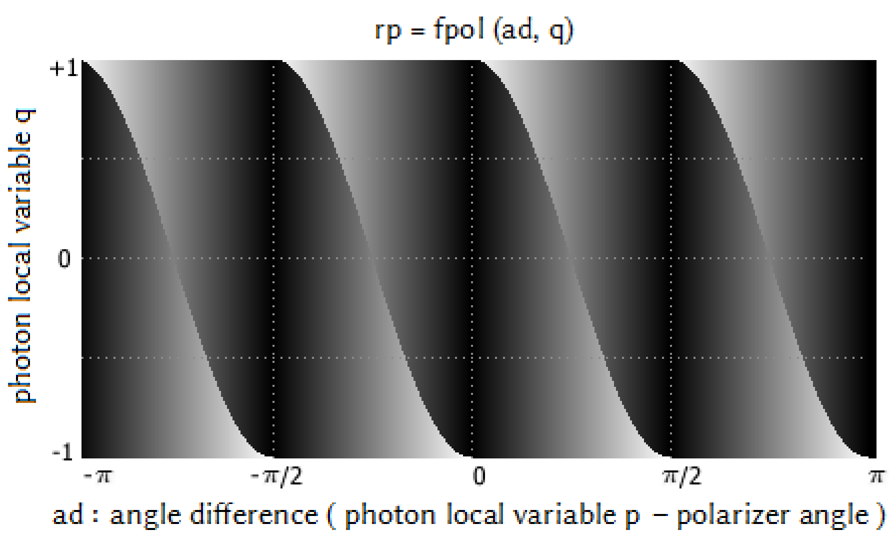
These two conditions can be simplified into a single equation of the form:

$dq = (q + \mathrm{sign}(rp)) * \tan(rp)^2$ (eq. 2)

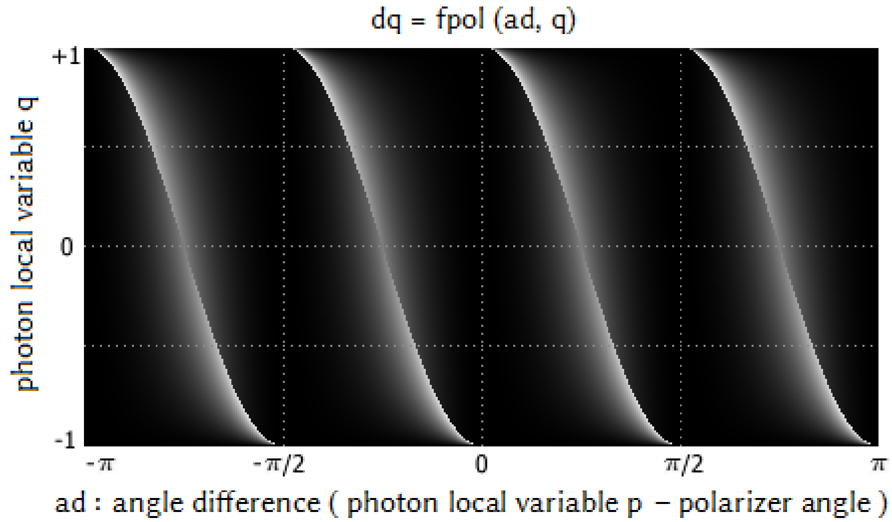
With sign(x) a function which returns -1 if $x < 0$, +1 if $x > 0$, and 0 if $x = 0$.

Although q is defined recurrently, its value always remains in the interval [-1..1] and dq varies in the interval [-2..2].

The following two graphs display the value of rp and dq as a function of ad and q for the output taken by the photon.



Graph 3a : Value |rp| as a function of ad and q in variations of gray intensity.
In black rp = 0, and white |rp| = π/2.



Graph 3b : Value |dq| as a function of ad and q in variations of gray intensity.
In black dq = 0, and white |dq| = 2.

We can notice from these two graphs that when the incoming polarization of the photon is aligned with one of the outputs of the polarizer (ad = 0, ± π/2 and ± π) the p and q values of the photon are not altered. It is then assumed that the action is minimal in these cases.

4. Study of non-optimal action.

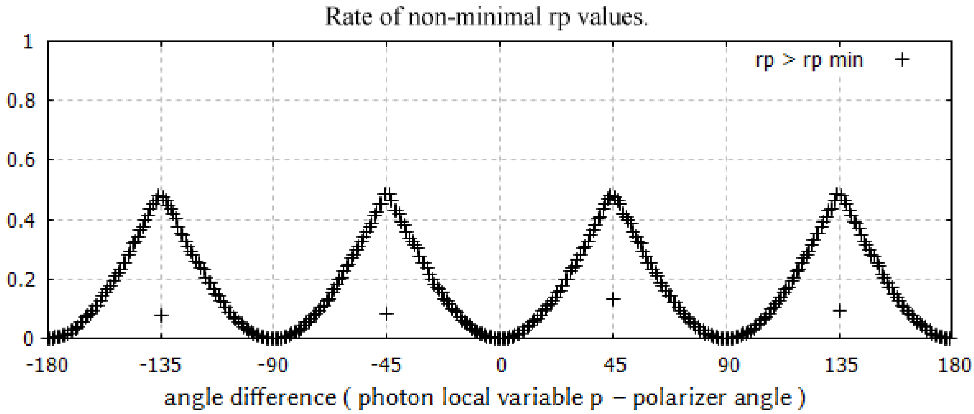
We showed previously that a minimum action value could not be established solely as a function of p and therefore the value rp.

To visualize this effect with the polarizer model, a simulation of Malus's law is done and the rp value is evaluated for the two possible outputs.

By knowing the output that must be taken to correctly produce Malus' law, we can then evaluate the proportion of interactions that do not use the output associated with a minimum value of |rp|.

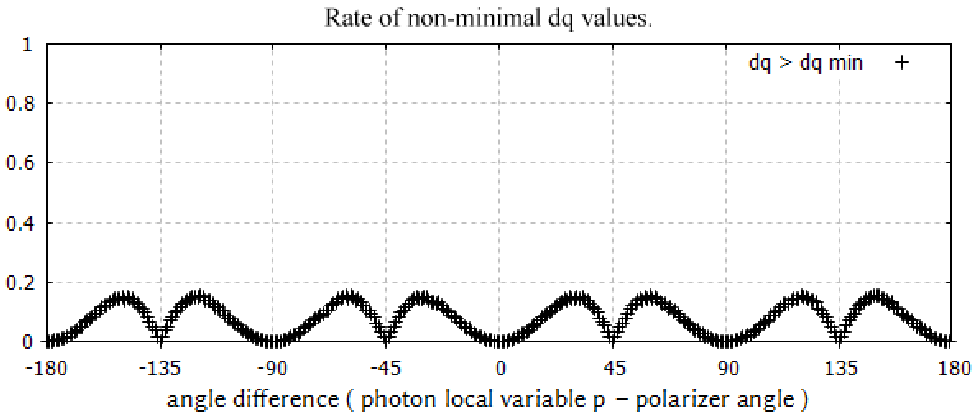
This means that the polarization of the incoming photon is not aligned with the "nearest" axis of the polarizer. (see image 2).

The following graph displays this proportion as a function of ad.



Graph 4a : Proportion of non-minimal |rp| values used to produce Malus's law.

It is also possible to do this same simulation by comparing the value |dq| used, which produces the following graph.

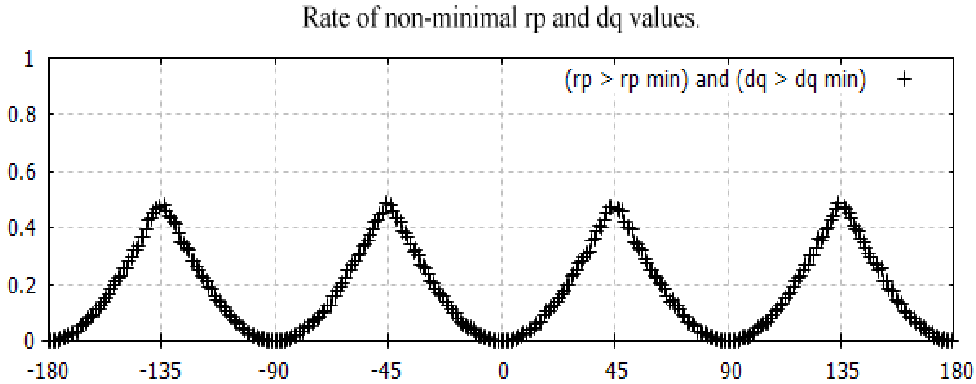


Graph 4b : Proportion of non-minimal |dq| values used to produce Malus's law.

We observe that as for |rp|, the value |dq| used is not always that of the smallest amplitude.

But these two graphs represent proportions, and it is possible, however, that for all measurements, at least the value |rp| or |dq| used is minimal.

To verify this hypothesis the following graph displays the proportion of interactions using neither the minimum value of |rp| nor the minimum value of |dq|.



Graph 4c: Proportion of |rp| and |dq| non-minimal values used to produce Malus's law.

We see in this graph that certain measurements are produced using at the same time the largest value for |rp| and |dq|.

It then seems difficult to define an action value depending only on rp and dq and which would always be minimal to produce Malus' law.
Indeed, the output to choose must sometimes use the minimum values for rp and dq, sometimes the maximum values, sometimes only one of the two.

An action equation dependent on rp and dq being able to do this would not be monotonically increasing as a function of |rp| and |dq| which seems physically unlikely to define the action.

Arrived at this point, we must therefore assume that if a principle of least action applies, it also depends on other variables which apparently have no effect on the choice of the polarizer output since this can be defined only from the variables p and q of the photon.

The proposed solution assumes that an additional mechanism must be considered during the polarization interaction that can minimize the action in certain circumstances.
This mechanism does not act on the selected polarizer output, but on a supposed state of detectability of the photon.

Thus, if Malus's law does not seem to respect the least action, this would only be an appearance produced by the measurement of a subset of detectable photons.

5. Two sets of photons.

The previously mentioned detectability state can be presented in two different ways.

- Either we consider that there may exist a state in which the photon is not detectable.
- Either we consider that there exist two distinct sets of photons of that only one can produce measurements.

Although the two approaches are logically identical, the second has the advantage of not requiring attaching an additional variable to the photon.

In addition, some theories assume that such sets can exist, such as supersymmetry, or the theory of mirror universes. These theories would help explain the existence of invisible matter or energy.

A notion of two sets will therefore be used in the remainder of this document, which also makes it possible to produce a more intuitive explanation.

Applied to the principle of least action, the following things are then assumed:

- A set change occurs under certain conditions of interaction.
- For all emitted and detected photons in the same set, the polarization interaction always selects the output for which the |rp| and |dq| values are minimal.

Thus, in each set, a principle of least action can be applied by considering only rp and dq.

We can then assume that for any photon, the total action depends on rp, dq, and an action value associated with the set change, and that the total action is always minimal for all polarization interactions.

The consequences produced are that any photon emitted in the measurable set can change set and therefore seem to disappear because it has become undetectable. Conversely, a photon emitted in the non-measurable set may seem to appear spontaneously in the measurable set.

Then only the measurement in a single set of photons coming from two distinct sets statistically reveals the Malus law, giving the appearance that a principle of least action does not apply in the measured set.

In this hypothesis, the polarizer model used until now that use a single set is then incomplete and must be modified.

6. Changing the polarizer model.

Modifying the model will consist of identifying the condition that must produce a set change.

Following the logic explained previously, the selected polarizer output must always be the one associated with a minimum variation of p and q, i.e. the smallest value for |rp| and |dq|.

This test is simple to do for |rp|.

Indeed if |rp| for the selected output is less than π/4, this necessarily implies that |rp| for the other output is greater than π/4, because the angle difference between the output polarizations is π/2.

Regarding |dq| this is more difficult.

Equation 2, $dq = (q + \text{sign}(rp)) * \tan(rp)^2$, must always produce the value |dq| minimum for the selected output, and it depends on the value (q + sign(rp)).

However, we observe in simulation that <u>if the correct polarizer output is chosen</u> , if |rp| is minimal, then |dq| is always minimal. The output selection method used by the model automatically produces this unexpected effect.

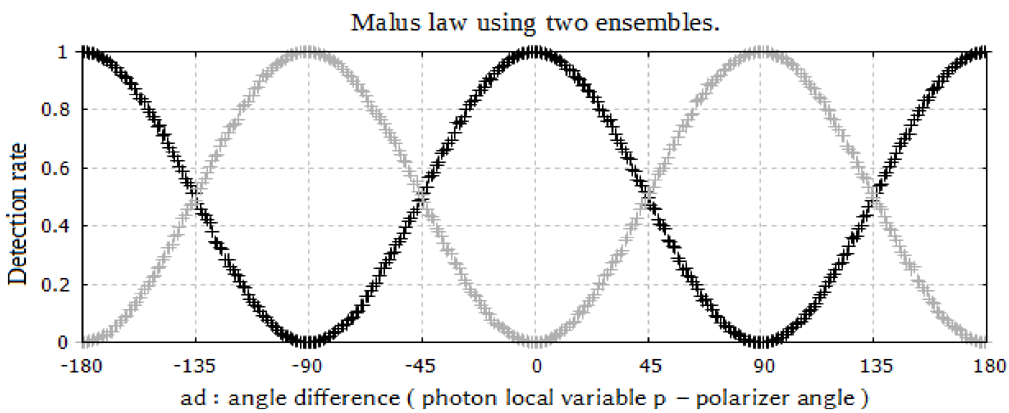
It is therefore possible to know whether an set change of the photon should occur by only testing with $|rp| > \pi/4$ or $\tan(rp)^2 > 1$.

<u>Test of the modified model.</u>

The operation of the polarizer model is modified by introducing the test on $\tan(rp)^2$ described previously, and when $\tan(rp)^2 >= 1$, the photon switches to the other possible set.

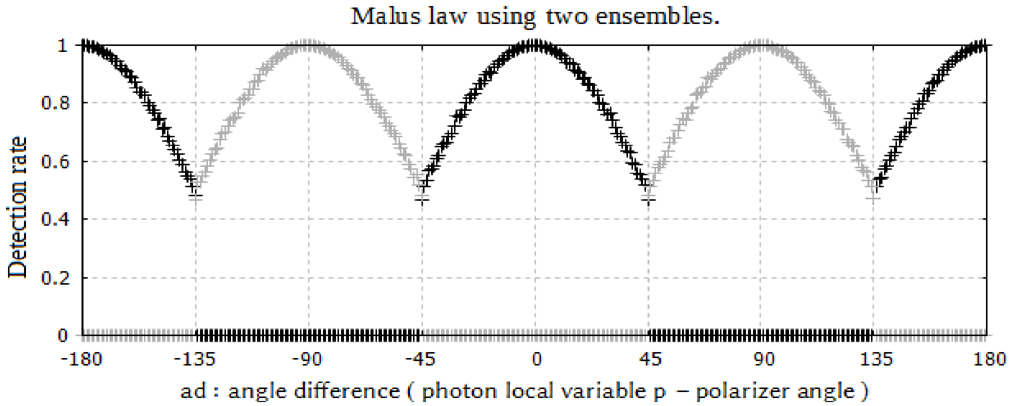
To carry out a simulation, it is then necessary to consider that a photon source emits photons equally in both sets, and that only one of the two sets can produce detections.

The following graphs plot Malus's law using this modification.



Graph 6a : Malus law produced using two sets of photons.
The black and gray curves display the detection rates on outputs o and e.
Emissions of photons are made in equal quantities in both sets, but detection is done on only one of the sets. (The results are identical whatever the choice of the detected set). We obtain curves in $\cos(ad)^2$ and $\sin(ad)^2$ for the outputs o and e conforming to Malus' law.

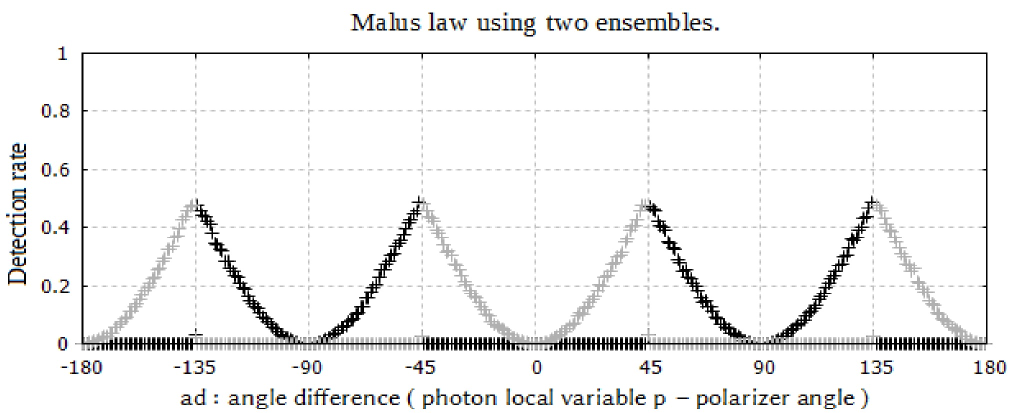
In order to verify that a principle of least action minimizes $|rp|$ for the chosen output, it is possible to trace only the detections made in the measurement set for the photons emitted in this same set. This produces the following graph.



Graph 6b : Detections made on a single set only for the photons emitted in this set. There is no mixing of photon detections from two different sets.

We then notice that the resulting transmission law only contains photons which have been repolarized by a value less than π/4, which are then always adjusted to the “closest” output axis of the polarizer, like this is depicted in picture 2.

Finally the following graph displays the proportion of photons which have changed of set.



Graph 6c : Rate of photons having changed set.

We note that graph 6a which describes the observable Malus law is indeed produced by the sum of graphs 6b and 6c, and results from the mixture of photon detections coming from two different sets.

7. Review of the principle of least action and the law of Malus.

The use of two sets of photons allows us to suppose a way in which a principle of least action could be applied with the law of Malus.

However, this has a fairly high cost by requiring either physically considering two sets of photons, or a principle defining a detectability state of the photon.

This hypothesis, however, has the advantage of producing another consequence which makes it possible to partially explain another important problem in a deterministic and local way.

It concerns correlation measurements of detections of pairs of photons produced by twin photon sources.

This refers to EPR type experiments, the effect called 'spooky action', and the violation of a Bell inequality called CHSH.

Considering two sets produces the following consequences on the correlation measurements:

- When a pair of photons is emitted in a set, it is not theoretically possible to detect this pair in certain polarizer angle configurations.
- The deterministic rejection of certain photon pair detections produces an unfair sampling of possible coincidence measurements.
  This rejection can produce the appearance that a non-local effect is acting and produce a violation of Bell's inequalities that do not take into account simple measurements.

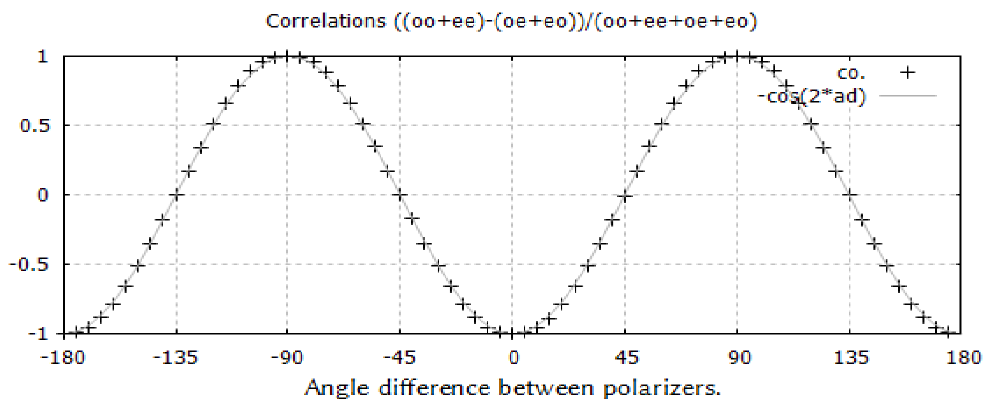
However, it remains that this is not sufficient to explain the violation of Bell's inequalities taking into account simple measurements such as that of Eberhard.

8. Effect of two sets on correlation measurements.

The following graph simulates an EPR type correlation experiment, and displays the detection correlations obtained by simulating two sets.

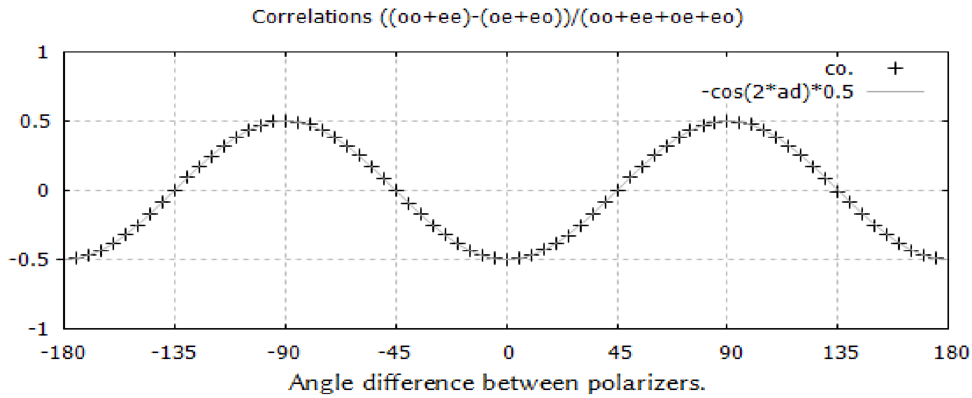
An emission of twin photons is simulated.

To reveal the statistical correlations produced by twin photons, it is necessary that the two photons of each pair be emitted in the same set, and also that for each emission the variables p and q of the two photons have identical values or offseted by a constant value.



Graph 8a : Correlations obtained by simulating two sets and pairs of twin photons.
The black crosses represent the simulation results.
The solid gray curve represents the exact -cos(2.ad) function.

The next graph simulates the correlations obtained if the choice of the emission set is random for each photon of a pair and their q values are initialized with independent random values.
Only the variable p is common to both photons (Simulated here with a constant shift of π/2).
This can be produced experimentally using a polarized source of free photons.



Graph 8b : Correlations obtained by simulating two sets and pairs of polarized free photons.

9. Values of CHSH inequality.

The twin photon simulation used to produce graph 8a produces a numerical value of CHSH varying around the value 2.91xxx.
The variations x depend on the seed of the RNG generator and the number of measurements made.
It would be possible to obtain a more precise value with a high quality RNG generator and using a large number of measurements.

The value is not 2*sqrt(2) (2.8284…). It seems possible that it is 3/2 + sqrt(2) (2.9142…)
The correlation curve obtained is close to the function -cos(2*ad) but does not exactly define this function.

The simulation of polarized free photons used to produce graph 8b produces a numerical value of CHSH varying around the value 1.41xxx. It is probable that the exact value is root of 2.
The correlation curve obtained seems to define exactly the function (-1/2)*cos(2*ad).

10. Review of correlations.

As is visible on the correlation graph 8a and a CHSH value greater than 2, if reality were composed of two sets of photons of which only one would be detectable, it could take on a non-local appearance.

By considering that not everything is measurable allow us to produce an explanation of the experimental results producing a CHSH value greater than 2 without requiring to assume that an instantaneous and non-local effect applies between the photons.

Conclusion.

It seems that a principle of least action cannot be applied to the law of Malus if it is considered that all the photons emitted by a source can be detected.

A solution to apply this principle seems possible by considering two sets of photons with only one allowing detection measurements to be produced.

The principle of least action would then be respected in each set, but not apparently in the statistical measurement produced by the mixture of the two sets made on a single set.

The hypothesis of a double set also makes it possible to provide a local explanation for certain detection correlation experiments producing apparently non-local results.

Thus, considering the possibility of the existence of two sets of photons has many explanatory advantages.

Reference:

[1] Malus'law photon by photon, a deterministic method.
https://www.vixra.org/abs/2201.0019

Annex:

Source code 1. (C language)

Program 1 simulates Malus' law by considering two sets and produces graphs 6a to 6c.
It is also downloadable here: http://pierrel5.free.fr/physique/ml/de_ml.c

```
// Malus law using two photons ensembles.
#include <stdio.h>
#include <math.h>

#define PI 3.14159265358979323846

// ----------------------------------------------
// random generator

// RNG seed
unsigned long r_seed = -123;

// return signed random value in -1..1 range
double rand1s(void)
{
  r_seed = r_seed * 214013L + 2531011L;
  return ((short)(r_seed >> 16))*(1.0/0x8000);
}

// ------------------------------------------------
// simulate polarizer

#define OUT_O 1              // value to code o output
```

```
#define OUT_E 2               // value to code e output

// define photon local variables type
struct lv_t
{
  int e;                      // photon set 0/1
  double p;                   // polarization
  double q;                   // undefined
};

// sign function, return -1 if x < 0, +1 if x > 0, 0 if x = 0
double sign(double x)
{
  return (x > 0) ? 1 : (x < 0) ? -1 : 0;
}

// simulate polarizer.
// update photon local variables.
// return polarizer output OUT_O or OUT_E value.
int sim_polarizer(double pol_angle, struct lv_t *lv)
{
  int out;
  double rp;
  double tan_rp, tan_rp2, dq;

  double ad = fmod(lv->p - pol_angle, PI);
  double s = ad + acos(-lv->q)*0.5;

       if (s >=  PI)    { out = OUT_O; rp = ad -   PI; }
  else if (s >=  PI/2)  { out = OUT_E; rp = ad - PI/2; }
  else if (s >=  0)     { out = OUT_O; rp = ad;        }
  else if (s >= -PI/2)  { out = OUT_E; rp = ad + PI/2; }
  else                  { out = OUT_O; rp = ad +   PI; }

  // eval dq
  tan_rp = tan(rp);
  tan_rp2 = tan_rp * tan_rp;
  dq = (lv->q + sign(rp)) * tan_rp2;

  // update local variables
  lv->p += rp;
  lv->q += dq;

  // check if ensemble change required
  if (tan_rp2 >= 1)
  {
    if (lv->e == 0)
      lv->e = 1;
    else
      lv->e = 0;
  }
  return out;
}

#define N_EMIT 10000

// simulate Malus law using two ensembles, produce a graph data file for gnuplot.
void main(void)
{
  int e_id = 0;               // emission ensemble id 0/1
  FILE *f = fopen("ml_2e.txt", "wb");
  if (f)
  {
    int ad;
    for (ad=-180; ad<180; ad++)
    {
      double ad_rad = (double)ad*PI/180.0;
```

```
      int n_O = 0;
      int n_E = 0;

      int i;
      for (i=0; i<N_EMIT; i++)
      {
        int out;
        struct lv_t lv;
        double pol_angle = rand1s()*PI;

        e_id = !e_id;        // alternate 0/1 emission ensemble id

        // simulate emission, init photon local variables
        lv.p = pol_angle + ad_rad;
        lv.q = rand1s();
        lv.e = e_id;         // set emit ensemble

        // simulate polarizer using photon lv, get polarizer output
        out = sim_polarizer(pol_angle, &lv);

        // count detections
        if (lv.e == 1)                    // graph 6a
        // if ((lv.e == 1) && (e_id == 1))  // graph 6b
        // if ((lv.e == 1) && (e_id == 0))  // graph 6c
        {
          if (out == OUT_O)
            n_O++;
          else
            n_E++;
        }
      }

      // save O/E detections ratio in the data file for gnuplot
      fprintf(f, "%d %.3f %.3f\n", ad, (double)n_O / (N_EMIT/2),
                                       (double)n_E / (N_EMIT/2));
    }
    fclose(f);
  }
}

// gnuplot command line
// cd 'C:\dir_to_file'                       // set path of ml_2e.txt file
// set grid;set xrange [-180:180];set yrange [0:1];set xtics 45;plot "ml_2e.txt"
using 1:2 with points pt 1 lc rgb '#000000' title "", "ml_2e.txt" using 1:3 with
points pt 1 lc rgb '#b0b0b0' title ""
```

Source code 2. (C language)

Program 2 simulates a correlation experiment by considering two sets and produces graphs 8a and 8b. It is also downloadable here: http://pierrel5.free.fr/physique/ml/de_co.c

```
// EPR correlations simulation using two photons ensembles.
#include <stdio.h>
#include <math.h>

#define PI 3.14159265358979323846
#define DEG_TO_RAD(d) (((d)*PI)/180.0)

// ------------------------------------------------
// simulation configuration
```

```
#define E_DETECT 1                  // set that can produce detections
#define SIM_FREE_PHO 0              // define 1 to simulate free photons

// -------------------------------------------------
// random generator

// RNG seed
unsigned long r_seed;

// return signed random value in -1..1 range
double rand1s(void)
{
  r_seed = r_seed * 214013L + 2531011L;
  return ((short)(r_seed >> 16))*(1.0/0x8000);
}

// return random boolean state 0/1
int rand_bool(void)
{
  r_seed = r_seed * 214013L + 2531011L;
  return (short)(r_seed >> 16) >= 0;
}

// ---------------------------------------------------
// simulate polarizer

#define OUT_U 0              // coded value for undetected
#define OUT_O 1              // coded value for o output
#define OUT_E 2              // coded value for e output

// define photon local variables type
struct lv_t
{
  int e;                     // 0/1 ensemble id
  double p;                  // polarization
  double q;
};

// sign function, return -1 if x < 0, +1 if x > 0, 0 if x = 0
double sign(double x)
{
  return (x > 0) ? 1 : (x < 0) ? -1 : 0;
}

// simulate polarizer.
// update photon local variables.
// return polarizer output OUT_O or OUT_E value.
int sim_polarizer(double pol_angle, struct lv_t *lv)
{
  int out;
  double rp;
  double tan_rp, tan_rp2, dq;

  double ad = fmod(lv->p - pol_angle, PI);
  double s = ad + acos(-lv->q)*0.5;

       if (s >=  PI)    { out = OUT_O; rp = ad -   PI; }
  else if (s >=  PI/2)  { out = OUT_E; rp = ad - PI/2; }
  else if (s >=  0)     { out = OUT_O; rp = ad;        }
  else if (s >= -PI/2)  { out = OUT_E; rp = ad + PI/2; }
  else                  { out = OUT_O; rp = ad +   PI; }

  // eval dq
  tan_rp = tan(rp);
  tan_rp2 = tan_rp * tan_rp;
  dq = (lv->q + sign(rp)) * tan_rp2;
```

```
  // update local variables
  lv->p += rp;
  lv->q += dq;

  // check if ensemble change required
  if (tan_rp2 >= 1)
  {
    if (lv->e == 0)
      lv->e = 1;
    else
      lv->e = 0;
  }
  return out;
}

// ------------------------------------------------
// EPR simulation

// EPR pair counters
struct ctr_t
{
  int oo, ee, oe, eo;
};

// single epr test (1 pair emission), update EPR counters
void epr_single_test(double a_pol_angle, double b_pol_angle, struct ctr_t *ctr)
{
  // simulate emission, init Alice/Bob photons local variables
  struct lv_t a_lv, b_lv;
  int a_out, b_out;

  // init local variables Alice
  a_lv.p = rand1s()*PI;       // init polarization angle
  a_lv.q = rand1s();          // random q [-1..1]
  a_lv.e = rand_bool();       // random ensemble 0 or 1

  // init local variables Bob
  b_lv.p = a_lv.p + PI/2;     // same as Alice + PI/2.

#if SIM_FREE_PHO              // simulate free photons
  b_lv.q = rand1s();          // Bob q independant of Alice q value
  b_lv.e = rand_bool();       // random ensemble 0 or 1
#else                         // simulate emission of twin photons
  b_lv.q = a_lv.q;            // same q
  b_lv.e = a_lv.e;            // same ensemble
#endif

  // simulate polarizers
  a_out = sim_polarizer(a_pol_angle, &a_lv);
  b_out = sim_polarizer(b_pol_angle, &b_lv);

  // test if photon is in detectable ensemble
  if (a_lv.e != E_DETECT)
    a_out = OUT_U;

  if (b_lv.e != E_DETECT)
    b_out = OUT_U;

  // update pair detection counters
  if (a_out == OUT_O)
  {
    if      (b_out == OUT_O)  ctr->oo++;
    else if (b_out == OUT_E)  ctr->oe++;
  }
  else
  if (a_out == OUT_E)
```

```
  {
    if      (b_out == OUT_O)  ctr->eo++;
    else if (b_out == OUT_E)  ctr->ee++;
  }
}

// -------------------------------------------------------
// correlation graph

#define GR_STEP_DEG 5        // step angle diff for graph

void def_co_graph(void)
{
  FILE *f;
  struct ctr_t ctr_graph[360] = { 0 }; // counter list
  int a, ad;

  for (a=-180; a<180; a++)
  {
    double an_pol_a = DEG_TO_RAD(a);
    for (ad=-180; ad<180; ad+=GR_STEP_DEG)
    {
      double an_pol_b = an_pol_a + DEG_TO_RAD(ad);
      struct ctr_t *ctr = &ctr_graph[180+ad];
      int i;

      // do some epr test, update counters
      for (i=0; i<1000; i++)
        epr_single_test(an_pol_a, an_pol_b, ctr);
    }
  }

  // save data file for gnuplot graph
  f = fopen("co_2e.txt", "wt");
  if (f)
  {
    for (ad=-180; ad<180; ad+=GR_STEP_DEG)
    {
      struct ctr_t *ctr = &ctr_graph[180+ad];
      int n_eq = ctr->oo + ctr->ee;
      int n_nq = ctr->oe + ctr->eo;
      int n_tot = n_eq + n_nq;
      double E = n_tot ? (double)(n_eq - n_nq)/n_tot : 0;
      fprintf(f, "%d %.3f\n", ad, E);
    }
    fclose(f);
  }
}

// -------------------------------------------------------
// eval CHSH
// https://en.wikipedia.org/wiki/CHSH_inequality

double eval_E(double an_pol_a, double an_pol_b, int N)
{
  struct ctr_t ctr = { 0 };
  int i, n_eq, n_nq, n_tot;
  double E;

  // do N epr test, update counters
  for (i=0; i<N; i++)
    epr_single_test(an_pol_a, an_pol_b, &ctr);

  n_eq = ctr.oo + ctr.ee;
  n_nq = ctr.oe + ctr.eo;
  n_tot = n_eq + n_nq;
  E = n_tot ? (double)(n_eq - n_nq)/n_tot : 0;
```

```
  return E;
}

void eval_CHSH(int N)
{
  double a1 = DEG_TO_RAD(0.0);
  double a2 = DEG_TO_RAD(45.0);
  double b1 = DEG_TO_RAD(22.5);
  double b2 = DEG_TO_RAD(67.5);

  double E_a1b1 = eval_E(a1, b1, N);
  double E_a1b2 = eval_E(a1, b2, N);
  double E_a2b1 = eval_E(a2, b1, N);
  double E_a2b2 = eval_E(a2, b2, N);

  double S = E_a1b1 - E_a1b2 + E_a2b1 + E_a2b2;
  double CHSH = fabs(S);
  printf("CHSH: %f\n", CHSH);
}

// main program entry
int main(void)
{
  r_seed = -123;    // init a RNG seed

  // define correlations graph
  printf("Define correlations graph, please wait..\n");
  def_co_graph();

  // eval CHSH
  printf("Eval CHSH..\n");
  eval_CHSH(10000000);
}

// gnuplot command lines
// cd 'C:\dir_to_file'   // set path of co_2e.txt file
// graph 6a  (SIM_FREE_PHO = 0)
// set title "Correlations ((oo+ee)-(oe+eo))/(oo+ee+oe+eo)";set grid;set xrange
[-180:180];set yrange [-1:1];set xtics 45;plot "co_2e.txt" using 1:2 with points
lc rgb '#000000' title "co.",-cos(2*x*pi/180.0) ls 7 lc rgb '#b0b0b0' title "-
cos(2*ad)"
// graph 6b  (SIM_FREE_PHO = 1)
// set title "Correlations ((oo+ee)-(oe+eo))/(oo+ee+oe+eo)";set grid;set xrange
[-180:180];set yrange [-1:1];set xtics 45;plot "co_2e.txt" using 1:2 with points
lc rgb '#000000' title "co.",-cos(2*x*pi/180.0)*0.5 ls 7 lc rgb '#b0b0b0' title
"-cos(2*ad)*0.5"
```

Version 1.0 November 20, 2023.