

LESO-PB

**User present : visual optimisation
Edificio, Subtask AD3**

Guillemin A., Morel N.

OFES, Technical Report

Novembre 1998

EDIFICIO, subtask AD3

User present: visual optimisation

Antoine Guillemin, Nicolas Morel, LESO-PB/EPFL, 25.11.1998

When the user enters the room, the controller switches in the visual optimisation mode. It means that the blinds (both alpha and beta with venetian blind, only alpha with not venetian blind) and the artificial light are controlled automatically. This paper describes six different algorithms (only for **not venetian** blinds) used for this visual optimisation. In each case a simulation with MATLAB SIMULINK is shown, it is used to test the algorithm.

But first of all, a small study is done concerning the interest of taking into account some thermal aspects also in the visual optimisation mode.

Important point:

If there are several blinds, each one has, at the beginning, the same control algorithm. They are differentiated by the adaptation to the user.

1. Thermal considerations

We want to answer the following question:

Is it important to take into account also some thermal aspects during the visual optimisation?

Cases studied

In order to answer this question, some different cases are studied.

For each case (1 to 4), the visual optimisation and the thermal optimisation are compared. The power balance of heat for the window is given in the visual optimisation case and the thermal optimisation case. They are values per square meter of window. They take into account the solar gains through window and the heat loss due to the difference of temperature between inside and outside. These values allow evaluating the difference, in energy, that exists between the visual and the thermal optimisation cases.

Summer

- | | | | |
|---------------|--------------------------------------|-------------------|---------------------------------------|
| 1. Sunny day | \Rightarrow (visual optimisation) | Blind half closed | $\Rightarrow P_w = 319 \text{ W/m}^2$ |
| | \Rightarrow (thermal optimisation) | Blind closed | $\Rightarrow P_w = 65 \text{ W/m}^2$ |
| 2. Cloudy day | \Rightarrow (visual) | Blind opened | $\Rightarrow P_w = 83 \text{ W/m}^2$ |
| | \Rightarrow (thermal) | Blind closed | $\Rightarrow P_w = 16 \text{ W/m}^2$ |

Winter

- | | | | |
|---------------|-------------------------|-------------------|---------------------------------------|
| 3. Sunny day | \Rightarrow (visual) | Blind half closed | $\Rightarrow P_w = 276 \text{ W/m}^2$ |
| | \Rightarrow (thermal) | Blind opened | $\Rightarrow P_w = 523 \text{ W/m}^2$ |
| 4. Cloudy day | \Rightarrow (visual) | Blind opened | $\Rightarrow P_w = 33 \text{ W/m}^2$ |
| | \Rightarrow (thermal) | Blind opened | $\Rightarrow P_w = 33 \text{ W/m}^2$ |

See appendix A for the calculations of the values presented here.

The thermal optimisation simply leads to accept solar gains in winter and to avoid them in summer.

Discussion

First, one can say that there is potentially more energy to save (or reject) during the sunny day (cases 1 and 3).

In case 1, the thermal aspects lead to close the blind and the visual ones give a half-closed blind position. The difference between the two power balances is more than 250 W/m^2 ! It could be interesting to close the blind more than that the visual aspects ask for. If one chooses a position of the blind of 0.2, 150 W/m^2 of solar heating could be avoided in comparison with the blind half-closed!

In case 3, the thermal aspects allow to gain 250 W/m^2 of heating power with a blind completely opened instead of a half-closed blind position. But in this case, it's not possible to consider a blind more open than that the visual aspects ask for, because of the big risk of glare.

In the case 2, the same kind of strategy could be done in order to take into account thermal aspects, but the gain will be limited at about 50 W/m^2 .

In the case 4, the visual aspects lead to the same blind position as the thermal aspects.

The night is a case that has not been calculated but it's clear that it leads to the same kind of results as the cases 2 or 4: possible gains but only small ones.

To summarise, one can say that it is sometimes possible to save energy (cooling or heating) in taking account some thermal aspects also during a visual optimisation period (user present). The best case seems to be in summer during sunny days. If one chooses a closer blind position (0.2 instead of 0.5) the avoided overheating could be of about 200 W/m^2 . To have an idea of the amount of cooling energy saved during one year, consider the following calculation:

200 W/m^2 for a window of 4 m^2 during 5 hours/day for 50 days:

$$200 \cdot 4 \cdot 5 \cdot 3600 \cdot 50 = 720 \text{ MJ} \text{ !!!!!!!}$$

In comparison, the total thermal energy consumption of an office room (in LESO) is 2500 MJ.

Conclusion

The answer to the studied question is now clear:

It's **very beneficial** to take into account also some thermal aspects during the visual optimisation.

2. Reference algorithm, no inside illuminance measurement

This algorithm comes from the DELTA project and it is shown because it allows having a reference when we compare the different algorithms.

The algorithm that controls the blind is taken directly from the DELTA final report (fuzzy logic rules). Additionally, one uses an artificial light control system (with dimming) in order to complete the level of total illuminance if the level of natural illuminance is 200 lux below the setpoint value.

The blind has four possible positions: open, half open, half closed, closed.

Theta is the solar incidence angle (angle between the perpendicular of the window and the light rays). Theta is a fuzzy variable, but one can say that an angle equal or less than 75° is considered as "low" value and angle equal or more than 85° is considered as "high" value.

Evdif is the direct illuminance on the facade. Evdif is also a fuzzy variable and values of less than 10 klux are considered as "low" values of Evdif and more than 30 klux are considered as "high" values.

A third fuzzy variable is Ehdif, which is the diffuse horizontal illuminance. Values of less than 30 klux are considered as "low" values of Ehdif and more than 30 klux are considered as "high" values.

Ehdif = low

<i>Theta \ Evdif</i>	low	mid	high
low	open	half closed	closed
mid	open	half open	half closed
high	open	open	open

Ehdif = high

<i>Theta \ Evdif</i>	low	mid	high
low	half closed	half closed	closed
mid	half closed	half closed	half closed
high	half closed	half closed	half closed

The SIMULINK model of this algorithm is shown in the appendix B.

Special feature: It's an "open loop" control.

Drawbacks: It uses pre-defined discrete values of the blind position. Moreover, it doesn't take into account various window (and room) characteristics (e.g. daylight factor), therefore it has to be adjusted again for every new room configuration. Finally, it doesn't work with an illuminance setpoint value so it's difficult to take into account, in this algorithm, different ambience definitions.

3. Discrete algorithm, one inside illuminance measurement

Inputs:

- Difference ($E_n - E_{set}$) between the natural illuminance (E_n) on desk (one has to subtract the **known** artificial part from the total illuminance) and the desired value (E_{set}), coming from ambiances definitions
- Solar incidence angle on the facade (θ)
- Global illuminance on facade (E_v)
- Fraction (F_{dif}) of diffuse illuminance : Diffuse illuminance on facade / Global illuminance on facade
- The season ($Season$), from the average of external temperature during the last 24 hours

Outputs:

- Variation of the blind position (*Varfinal*), in our case -- means two positions lower (from 5 possible positions, for example) and + means one position higher, etc ...
- Artificial illuminance needs (*Art*)

Each loop of the algorithm is as follow:

*****Determination of the variation (*vartemp*) of the blind position depending *****
 ***** of the fraction of diffuse illuminance *****

If *Fdiff* is high then *Vartemp* is calculated through the following table (fuzzy logic rules):

<i>En-Eset</i>	<i>Vartemp</i>
much higher	--
higher	-
equal	0
lower	+
much lower	++

If *Fdiff* is low then *Vartemp* is calculated through the two following tables (fuzzy logic rules). The first one gives the glare risk *Gr* in function of incidence angle and global vertical illuminance on the facade. The second one gives *Vartemp* in function of *En-Eset* and *Gr*. (Both tables can be combined into a single one if necessary.)

<i>Theta</i> □ <i>Ev</i>	low	medium	high	very high
low	no	possible	sure	sure
middle	no	possible	possible	sure
high	no	no	possible	possible

<i>En-Eset</i> □ <i>Gr</i>	no	possible	sure
much higher	--	--	--
higher	-	-	-
equal	0	0	0
lower	+	+	0
much lower	++	+	+

*****Determination of the variation (*Varfinal*) of the blind position depending of the season*****

If *season* is summer then *Varfinal* = One position lower than the *Vartemp*

If *season* is winter then *Varfinal* = One position higher than the *Vartemp*

*****Actions*****

Apply the *Varfinal* on the blind system

Update the *En* measurement

*****Determination of the artificial illuminance*****

If natural illuminance (*En*) on desk is still more than 200 lux lower than the desired value, the artificial light (*Art*) complete the illuminance to the desired level

Else no artificial light

End of the loop

The SIMULINK model of this algorithm is shown in the appendix C.

Special feature: It's a "closed loop" control with the measurement of the inside illuminance.

Drawbacks: It uses discrete (and fixed!) values of blind position. It deals with blind variation and not directly with blind position. That is not compatible with the nested loop control (level 1 and level 2 are not anymore well separated).

4. Continuous algorithm, one inside illuminance measurement

Inputs:

- The current illuminance (E_c) on desk
- The setpoint value of the illuminance (E_{set}), coming from ambiances definitions
- Solar incidence angle (θ)
- Global illuminance on facade (E_v)
- Fraction (F_{diff}) of diffuse illuminance : Diffuse illuminance on facade / Global illuminance on facade
- The season ($Season$), from the average of external temperature during the last 24 hours
- The blind position (α), possible positions are between 0 and 1

Outputs:

- Blind movement (B_{mov})
- Artificial illuminance needs (Art)

The algorithm works as follow:

*****Determination of the first variation ($Var1$) of the illuminance setpoint (taking account the glare risk)*****

If F_{diff} is low then $Var1$ is calculated through the following table (fuzzy logic rules):

$\theta \setminus E_v$	low	medium	high
low	0	- 10%	- 20%
middle	0	0	- 10%
high	0	0	0

If F_{diff} is high then $Var1$ is 0

*****The second variation ($Var2$) depends of the season (fuzzy logic rules)*****

Season	$Var2$
winter	+ 20%
mid-season	0
summer	- 20%

*****The final variation (Var_{total}) of the illuminance setpoint is then calculated*****

$$Var_{total} = Var1 + Var2$$

*****Determination of the new setpoint, called user setpoint (U_{set})*****

$$U_{set} = E_{set} * (1 + V_{artotal})$$

*****Comparison between current illuminance and the user setpoint*****

If $|E_c - U_{set}| < 150$ then end

Else

While $|E_c - U_{set}| > 10$ do

*****Determination of the blind movement (B_{mov})*****

If $(E_c - U_{set}) > 0$ and (Art is 0) then B_{mov} is down

If $(E_c - U_{set}) > 0$ and (Art is not 0) then subprogram ARTIFICIAL

If $(E_c - U_{set}) < 0$ then B_{mov} is up

*****Check limits*****

If B_{mov} is down and $alpha$ is 0 then end

Else if B_{mov} is up and $alpha$ is 1 then subprogram ARTIFICIAL

Else apply B_{mov} to the blind and E_c is updated

End

Subprogram ARTIFICIAL

***** Determination of the artificial illuminance*****

If $(E_c - U_{set}) < -200$ then the artificial light (Art) complete the illuminance to the U_{set} level

If $(E_c - U_{set}) > 0$ then the artificial light is decreased to the U_{set} level

The SIMULINK model of this algorithm is shown in the appendix D.

Special feature: It's a "closed loop" control with the measurement of the inside illuminance.

Drawbacks: It deals (as the previous algorithm) with a blind variation and not directly with a blind position. Furthermore, one can notice that it would be probably best to take relative values instead of absolute ones when one deals with illuminance level.

5. Sun-position algorithm, one inside illuminance measurement

This algorithm is a combination of the ideas of the three previous algorithms. The idea is to take into account not only the incidence angle of the solar radiation on the facade but the exact position of the sun relatively to the facade (see figure 1, the nine different positions of the sun). So, both the azimuth and the height of the sun are used in the algorithm. This allows having different behaviours for different kind of penetration of sun. If the sun illuminates the wall in front of the user or illuminates the user directly, the algorithm may give different blind position although the incidence angle is the same in the two cases. In particular, it gives the opportunity to adapt the system (through the user wishes) following the user position in the room!

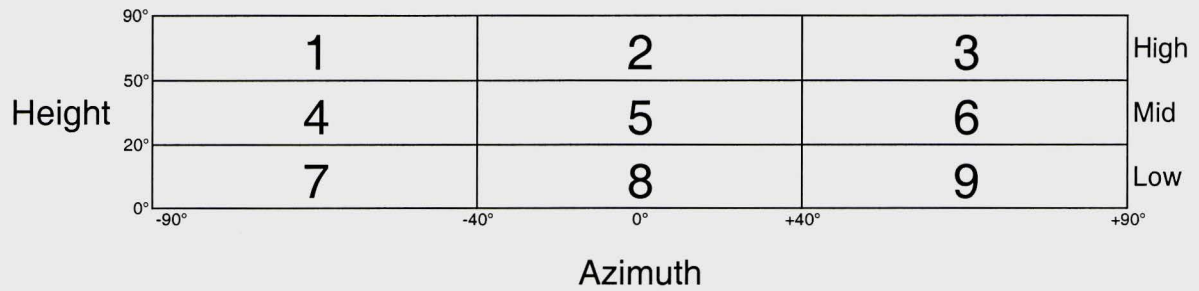


Figure 1: Sun position relative to the facade

The algorithm is split into two parts. The first one gives a maximum value for the blind position in order to avoid glare and the second one tries to find the blind position (below the maximum value) that leads to the inside illuminance corresponding to the ambience definition.

Maximum blind position

Inputs:

- Direct horizontal illuminance (*Ehdir*)
- The season (*Season*), from the average of external temperature during the last 24 hours
- The height of the sun (*Height*)
- The azimuth (relative to the facade orientation) of the sun (*Azimuth*)

Output:

- Maximum blind position (*Bmax*)

Rules:

There are 25 rules in the system. At the beginning (before the user adaptation) the rules concerning the same height of sun but with a different azimuth gives the same value of *Bmax* (see figure 1, the position of the sun 4, 5 and 6 gives the same result, for example). The idea is that the user adaptation will differentiate the rules with the same height of sun but with a different azimuth.

If *Ehdir* is high and *Season* is winter and "position of the sun is 1,2 or 3" then *Bmax* is 1 *

If *Ehdir* is high and *Season* is winter and "position of the sun is 4,5 or 6" then *Bmax* is 0.7 *

If *Ehdir* is high and *Season* is winter and "position of the sun is 7,8 or 9" then *Bmax* is 0.4 *

If *Ehdir* is high and *Season* is summer and "position of the sun is 1,2 or 3" then *Bmax* is 0.5 *

If *Ehdir* is high and *Season* is summer and "position of the sun is 4,5 or 6" then *Bmax* is 0.3 *

If *Ehdir* is high and *Season* is summer and "position of the sun is 7,8 or 9" then *Bmax* is 0.1 *

If *Ehdir* is high and *Azimuth* is $> +90^\circ$ then *Bmax* is 1

If *Ehdir* is high and *Azimuth* is $< -90^\circ$ then *Bmax* is 1

If *Ehdir* is low and *Height* is high then *Bmax* is 1

If *Ehdir* is low and *Height* is mid then *Bmax* is 1

If *Ehdir* is low and *Height* is low then *Bmax* is 1

If *Season* is summer and *Height* is negative then *Bmax* is 1

If *Season* is winter and *Height* is negative then *Bmax* is 0.1

* means that the rule correspond in fact to three rules

Blind position according to the inside illuminance measurement

The final position of the blind is simply determined by a closed loop control with the inside illuminance measurement. The only constraint is that the blind position must be lower than the maximum blind position previously calculated.

The SIMULINK model of this algorithm is shown in the appendix E.

Special feature: It's a "closed loop" control with the measurement of the inside illuminance.

Drawbacks: The quality of the control system is strongly user dependent. If the user doesn't express his wishes, the good capability of adaptation (concerning the azimuth) won't be used!

6. Illuminance ratio algorithm (I-ratio), three inside illuminance measurements

Two algorithms (I-Ratio 1 and I-Ratio 2) based on the measurement of three illuminance sensors have been developed. The difference between the two is that the season and the control of the artificial light are not taken into account directly in the fuzzy rules in the I-Ratio 2 algorithm. Nevertheless, in this algorithm the season is taken into account in a different way, by a change of the setpoint (see the Continuous algorithm) and the control of the artificial light is done in a classical way (see the others algorithms).

Inputs:

- Illuminance setpoint: Set
- Low limit parameter: Llim
- High limit parameter: Hlim

Fuzzy input variables:

- Illuminance: Too dark = $Llim * Set$; Too bright = $Hlim * Set$;
- Contrast (ratio max between sensors): Ok = 1 - 5 ; Too high = 10 - ... ;
- Total illuminance on facade (Tif): High = 10 - ... klux ;
- Season: Winter = $Text24 < 10$; Summer $Text24 > 10$;
- Artificial switch (ArtS), crisp value: On or Off ;
- Blind position (BP): Up, No or Down ;

Fuzzy output variables:

- Blind Move: -1 to 1 ;
- Artificial light change (Art): -1 (Decrease) to 1 (Increase) ;

Rules (I-Ratio 1):

If Illuminance	is	Too dark			then	Move is 0.5
If Illuminance	is	Too bright	and	ArtS is Off	then	Move is -0.5
If Illuminance	is	Too dark	and	BP is Up	then	Art is 1
If Illuminance	is	Too bright			then	Art is -1
If Contrast	is	Too high			then	Move is -1
If Tif is High			and	Season is Summer	then	Move is -0.1
If Contrast is Ok	and	Tif is not Zero	and	Season is Winter	then	Move is 0.1

Rules (I-Ratio 2):

If Illuminance	is	Too bright			then	Move is -1
If Illuminance	is	Too dark	and	Contrast is Ok	then	Move is 1
If Tif is high			and	Contrast is Too high	then	Move is -1

The SIMULINK model of these algorithms is shown in the appendix F.

Special feature: It uses three luxmeters and it can take into account the problem of glare with the measurement of the contrast between the three sensors. It's a double "closed loop" control with the measurement of the inside illuminance and the contrast.

Drawbacks: It gives a continuous blind position, so the blind moves too often. A simple discretisation of the output is not possible because this algorithm works in a dynamic way, that means the blind moves until a balance is found.

7. Comparison of the different algorithms

In order to compare all the algorithms, MATLAB simulations have been carried out. Each algorithm has been tested during one week with external weather conditions coming from synthetic values produced by the METEONORM program (ref. [1]). The simulations are done on the period that corresponds to the seven first days of July. The choice has been done in a way that different weather conditions are represented (sunny and cloudy days). The appendix H gives a more detailed picture of the weather conditions during this period. The night (from 21h00 to 7h00) the user is considered as absent, so the tested algorithm is stopped (no blind movements, no artificial light).

The physical software model of the room for the calculation of the inside illuminance is simply the illuminance on facade multiplied by a kind of daylight factor (0.05 for our case) and a blind transmission factor.

This illuminance blind transmission factor depends linearly of the blind position between a value of 1 (blind completely open) and 0.2 (blind completely closed).

The simulations give, as results, the extreme values of inside illuminance reached in the room, the integrated value of the difference between the setpoint value and the current value of illuminance, the electrical power consumption and the total number of blind movements during the simulation.

Results and discussion

The table below shows the results of the simulations for each algorithm. In the appendix I, the results of one day of simulation are given graphically.

Algorithm	Extreme values of inside illuminance [lux]	Integrated "error" of illuminance [lux]	Electrical energy consumption [MJ]	Number of blind movements
Reference	400 - 1500	690	13.6	16
Discrete	400 - 600	80	22.3	52
Continuous	280 - 900	600	8.7	38
Sun-position	380 - 800	230	13.5	42
I-Ratio 1	380 - 960	490	11.2	36
I-Ratio 2	300 - 680	590	8.6	44

Clearly, one sees that all the algorithms seem to work normally, without too big number of blind movements or too high electrical energy consumption. Concerning the inside illuminance level, all the algorithms keep a value not too far (< 300 lux of difference) from the setpoint value (600 lux), except the Reference algorithm, which is the only one with no closed-loop control.

It's hard to discuss these algorithms in a quantitative point of view, because no one of them is really bad. So, some comments are done in a qualitative point of view.

The I-Ratio algorithms use three luxmeters, but the others use only one luxmeter (placed on the desk). The positive point to have three luxmeters is that it's possible to take into account some glare aspects, but the drawback is that the position of the luxmeters is a very critical thing to obtain a good algorithm. It's important to notice that the simulations have not tested the algorithm behaviour (for the I-Ratio ones) with different values of the input variable "Contrast" (see part 6). The "Contrast" was chosen constant.

Here are the comments for each algorithm.

Reference: The main drawback of this algorithm is that the level of illuminance is far from the setpoint. There would be also some difficulties to take into account different ambience definitions.

Valuation: Not good

Discrete: This algorithm leads to nearly perfect visual conditions in the room. But this very small value of integrated error of illuminance is due, in fact, to a high use of the artificial light and a low position of the blind. Because of the low position of blind, the illuminance level is not very influenced by the outdoor conditions and could be kept very constant with the use of a big amount of artificial light. This behaviour is not bad because the simulation takes place in summer, and if we take some thermal considerations into account, the goal is precisely to reject a maximum of solar gains during this period. The drawbacks of this algorithm are, as mentioned earlier, the fact that the possible blind positions are pre-defined and fixed and also the fact that the algorithm deals with blind variation instead of blind position.

Valuation: Not good

Continuous: The drawback of the change on the setpoint to avoid glare is that it leads to reduce the illuminance level in the room. On the one hand, it's good because the blind goes to a lower position (so glare is avoided) but on the other hand the illuminance level in the room could be no more sufficient (the artificial light complete the indoor illuminance only to the reduced setpoint level).

Valuation: Not good

Sun-position: The main positive aspect of this algorithm is the fact that it is split into two parts. The first part to set the maximum blind position (to avoid glare and to take into account thermal aspects) and the second one to set the blind position in order to have the right illuminance level in the room. Because of this, there should be no difficulties to adapt the system to the user.

Valuation: Good and promising for the adaptation task

I-Ratio 1: This algorithm works in a dynamic way and the indispensable discretisation of the blind position is rather difficult to do without spoiling the quality of the algorithm.

Valuation: A bit too complicated, very probable difficulties in the adaptation task

I-Ratio 2: This algorithm works very simply (fuzzy controller for the blinds, and closed loop control for the artificial light) but can take into account some glare aspects. It could be considered as a safe solution.

Valuation: Fair

8. Conclusion

Five algorithms have been developed and tested for the visual optimisation task. The comparison of these algorithms with a reference algorithm (coming from a project especially dedicated to the blinds control, see ref. [2]) has shown that all the new algorithms give the same kind of results and even better than the reference algorithm.

Finally, two algorithms seem to be adequate for the visual optimisation task: the Sun-position algorithm and the I-Ratio 2 algorithm.

The more promising one is the one called Sun-position. It takes into account both the azimuth and the height of the sun. That allows having an algorithm, which gives different results for different penetrations of the sun in the room. So, for the EDIFICIO project we think that **we will use the "Sun-position" algorithm** for the control of the blind and the artificial light **when the user is present**.

The other interesting algorithm (I-Ratio 2) is kept as a spare solution.

9. References

- [1] Meteo Test: Meteonorm 95, v2.0, BEW, Berne, 1996
- [2] M. Bauer, J. Geiginger, W. Hegetschweiler, G. Sejkora, N. Morel, P. Wurmsdobler: DELTA, a blind controller using fuzzy logic, final OFEN Report, LESO, EPFL, 1996

Appendix A: Calculations

The following equation is used to quantify the solar gain through the window for each case:

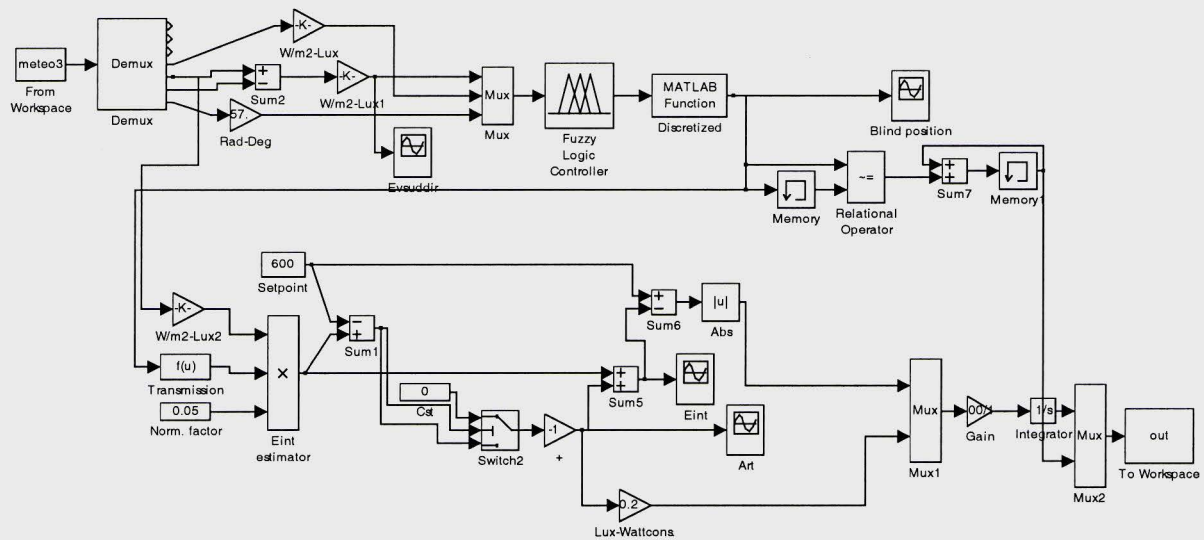
$$P_w = G_v g \alpha + G_v g_s (1 - \alpha) - [k \alpha + k (1 - \alpha)/(1 + R k)] (T_i - T_o)$$

Where	P_w [W/m ²]	Power balance of heat per square meter of window
	G_v [W/m ²]	Global vertical illuminance on facade
	g [-]	Solar transmission coefficient of window (energetic)
	g_s [-]	Solar transmission coefficient of blind (energetic)
	α [-]	Blind position, $0 \leq \alpha \leq 1$ ($\alpha=1$ means blind open)
	k [W/m ² K]	Heat-loss coefficient of window (convective and radiative)
	R [m ² K/W]	Thermal insulation coefficient for blind
	T_i [K]	Inside temperature
	T_o [K]	Outside temperature

Values used here:

$G_v = 100$ W/m² (cloudy day) or 800 W/m² (sunny day) ; $g = 0.7$; $g_s = 0.1$; $k = 2.5$ W/m²K ; $R = 0.15$ m²K/W ;
 $T_i = 293$ K ; $T_o = 298$ K (summer) or 278 K (winter)

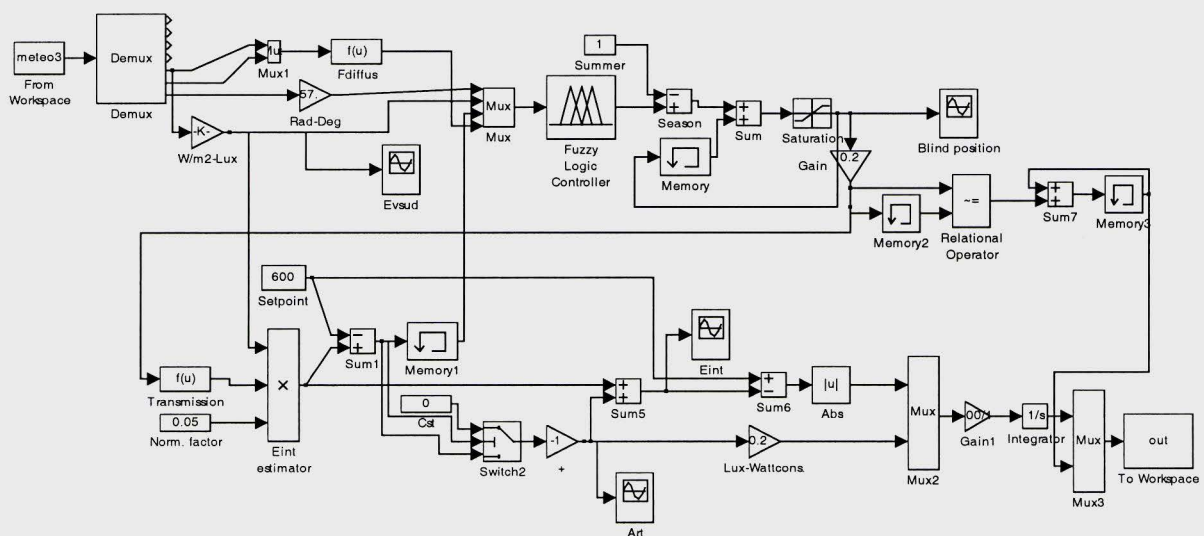
Appendix B: SIMULINK model of the Reference algorithm



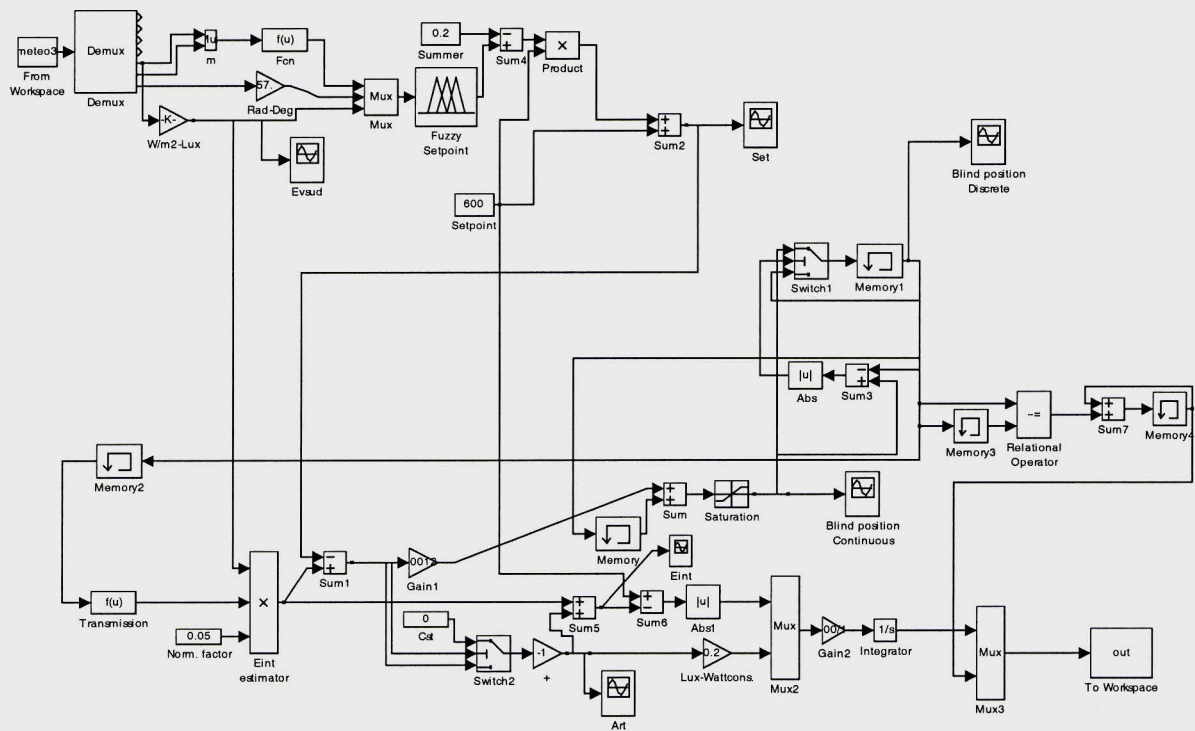
Outputs of the meteo3 block:

1. Time [days]): It's not an output of the Demux block (the one that follows the meteo3 block)!
2. Outside temperature [$^{\circ}\text{C}$]
3. Outside temperature (average on the last 24 hours) [$^{\circ}\text{C}$]
4. Global horizontal illuminance [W/m^2]
5. Diffuse horizontal illuminance [W/m^2]
6. Global vertical (south) illuminance [W/m^2]
7. Diffuse vertical (south) illuminance [W/m^2]
8. Solar incidence angle (south facade) [rad]

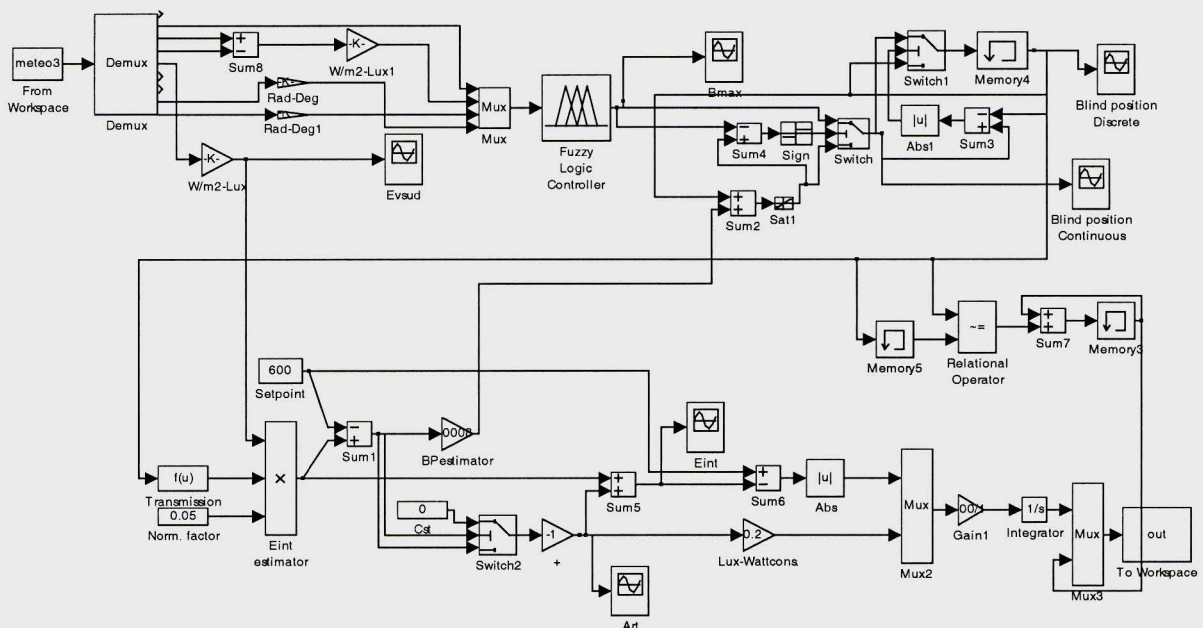
Appendix C: SIMULINK model of the Discrete algorithm



Appendix D: SIMULINK model of the Continuous algorithm



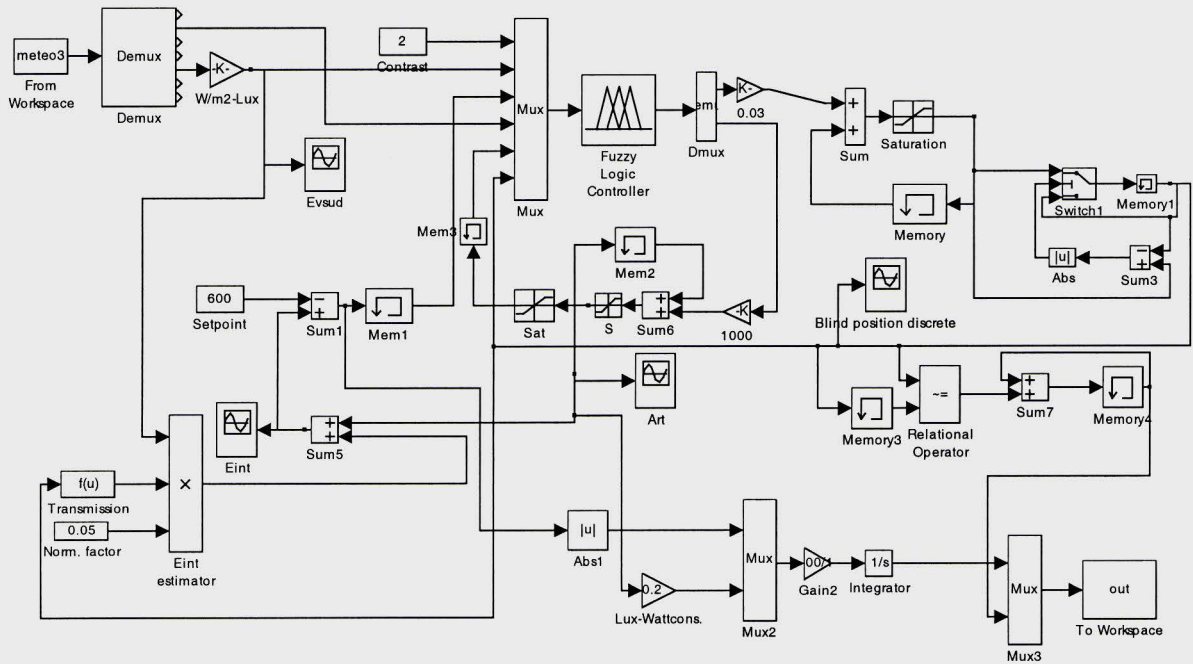
Appendix E: SIMULINK model of the Sun-position algorithm



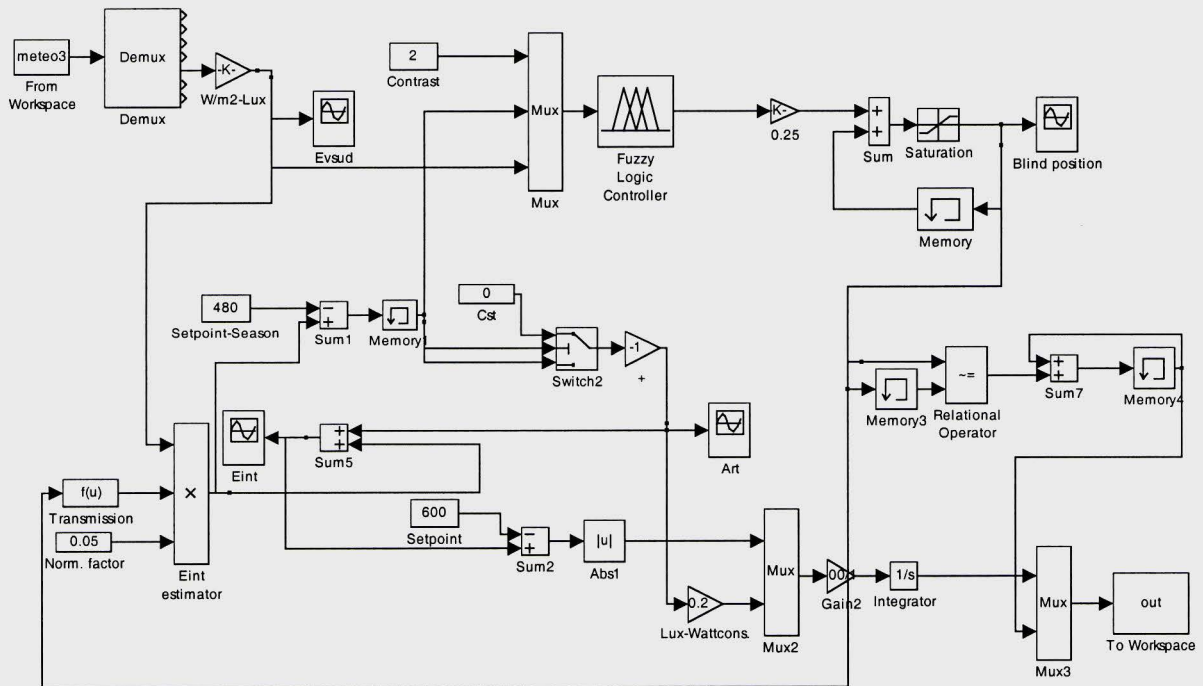
Supplementary outputs of the meteo3 block (see Appendix B):

9. Azimuth (relatively to the facade) [rad]
10. Sun Height [rad]

Appendix F: SIMULINK models of the I-Ratio 1 algorithm

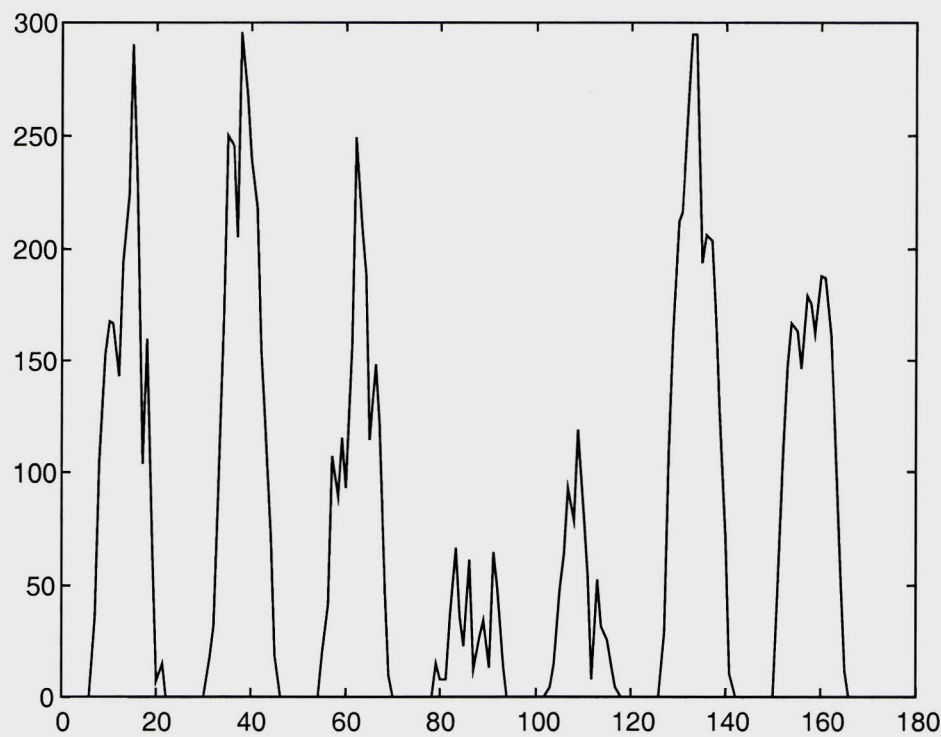


Appendix G: SIMULINK models of the I-Ratio 2 algorithm

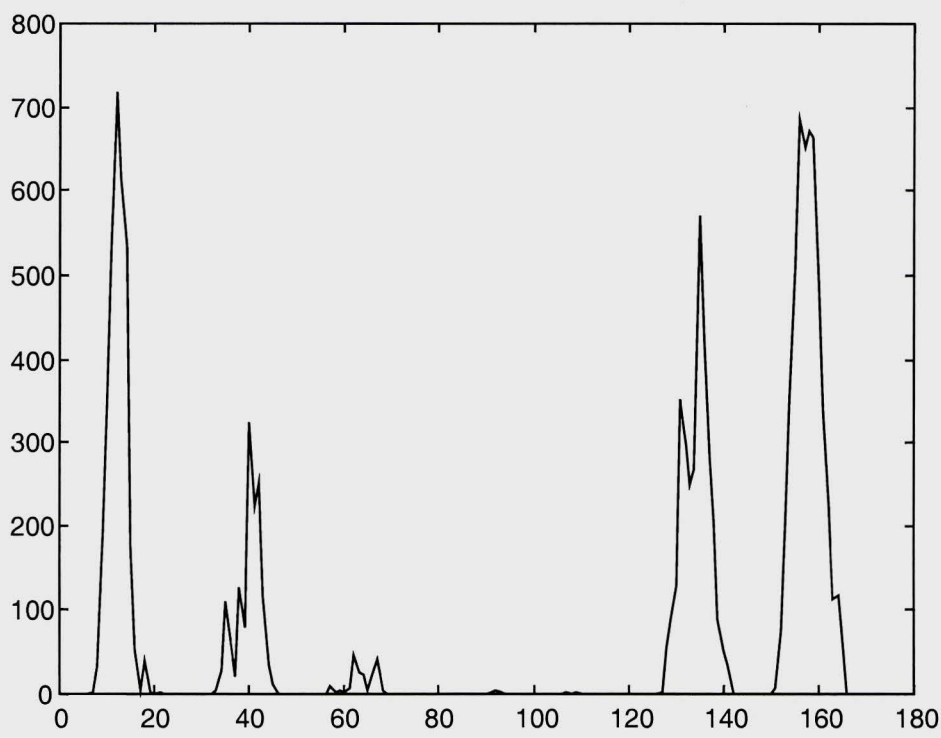


Appendix H: Weather conditions during simulations

Diffuse horizontal illuminance [W/m^2]

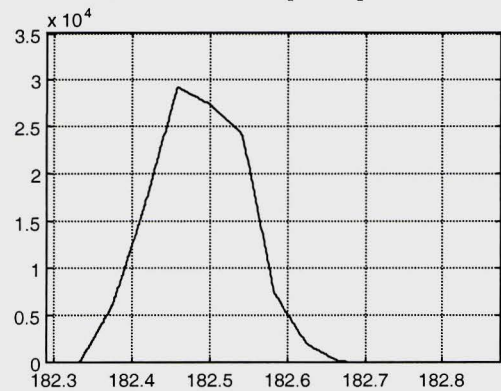


Direct horizontal illuminance [W/m^2]



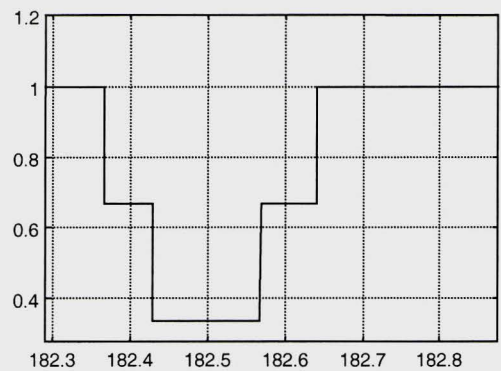
Appendix I: Results of simulations

Direct illuminance on facade [W/m²]

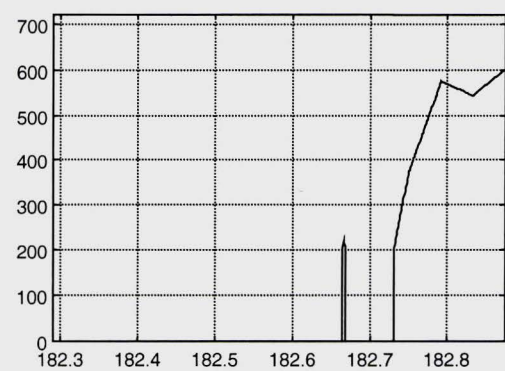


Reference algorithm:

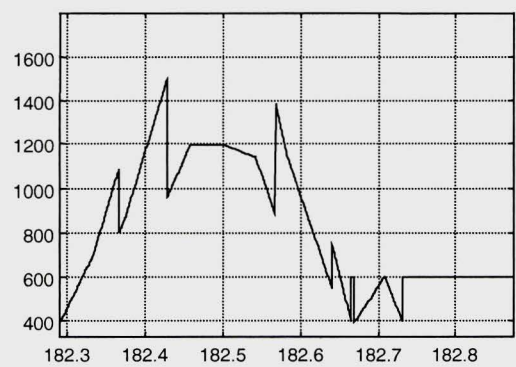
Blind position



Artificial light level

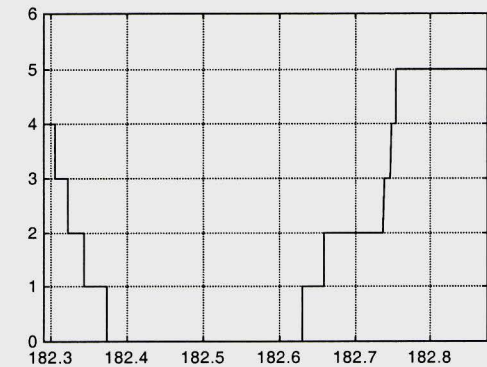


Inside illuminance

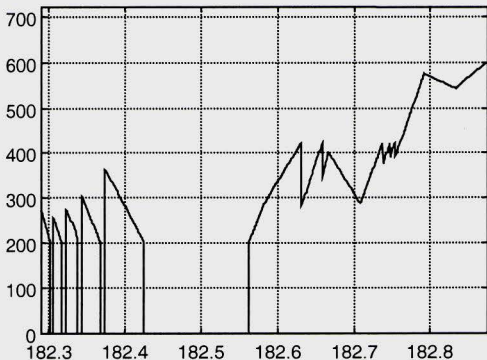


Discrete algorithm:

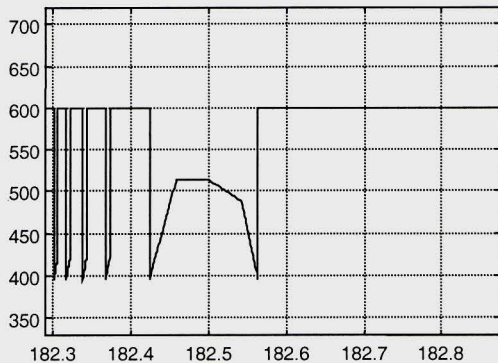
Blind position



Artificial light level

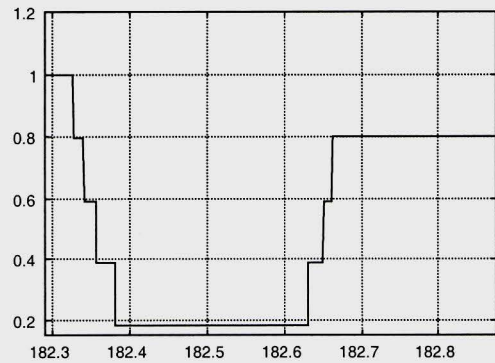


Inside illuminance

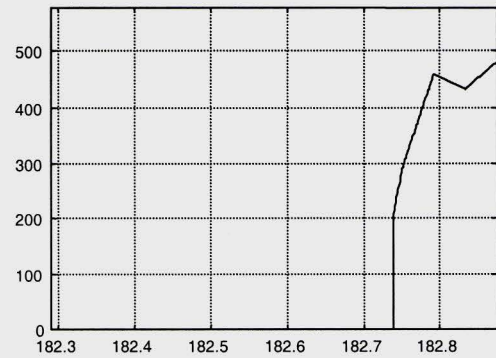


Continuous algorithm:

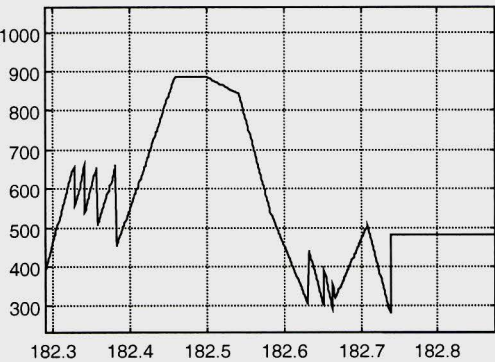
Blind position



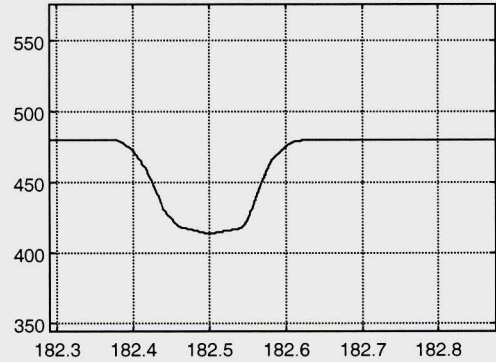
Artificial light level



Inside illuminance

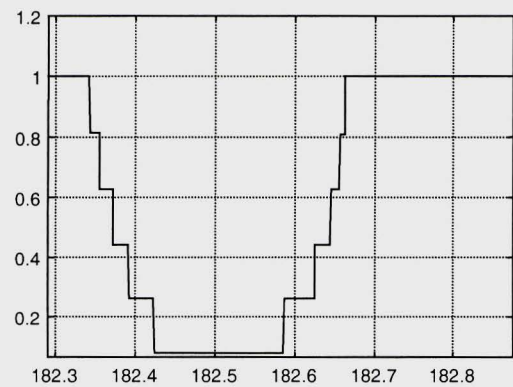


Illuminance setpoint

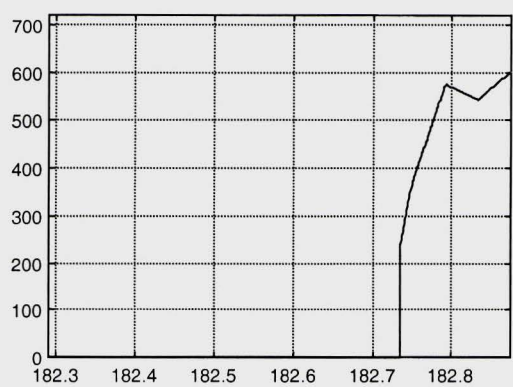


Sun-position algorithm:

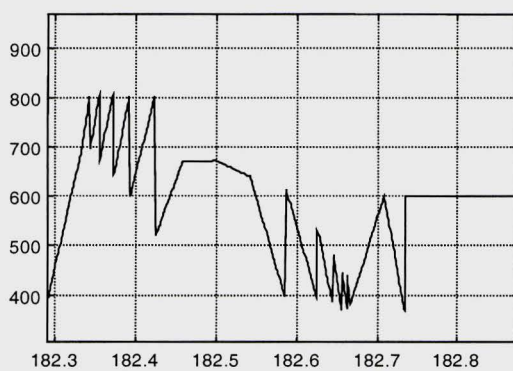
Blind position



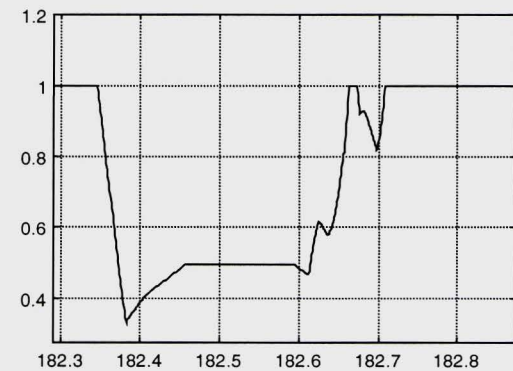
Artificial light level



Inside illuminance

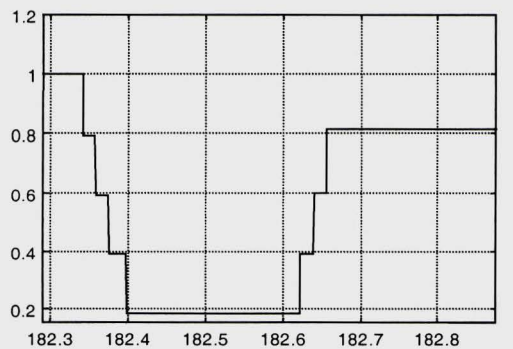


Maximum blind position

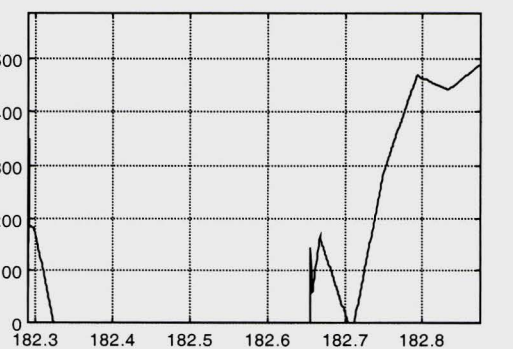


I-Ratio 1 algorithm:

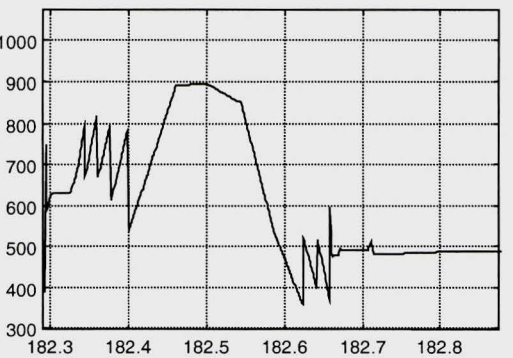
Blind position



Artificial light level



Inside illuminance

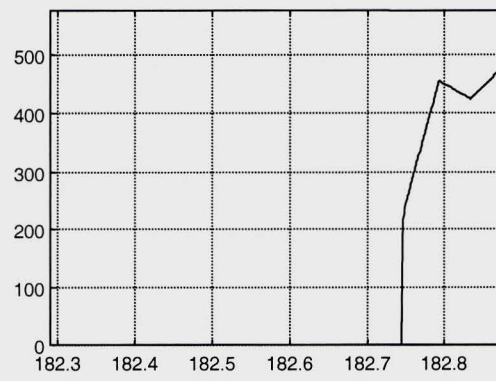


I-Ratio 2 algorithm:

Blind position



Artificial light level



Inside illuminance

