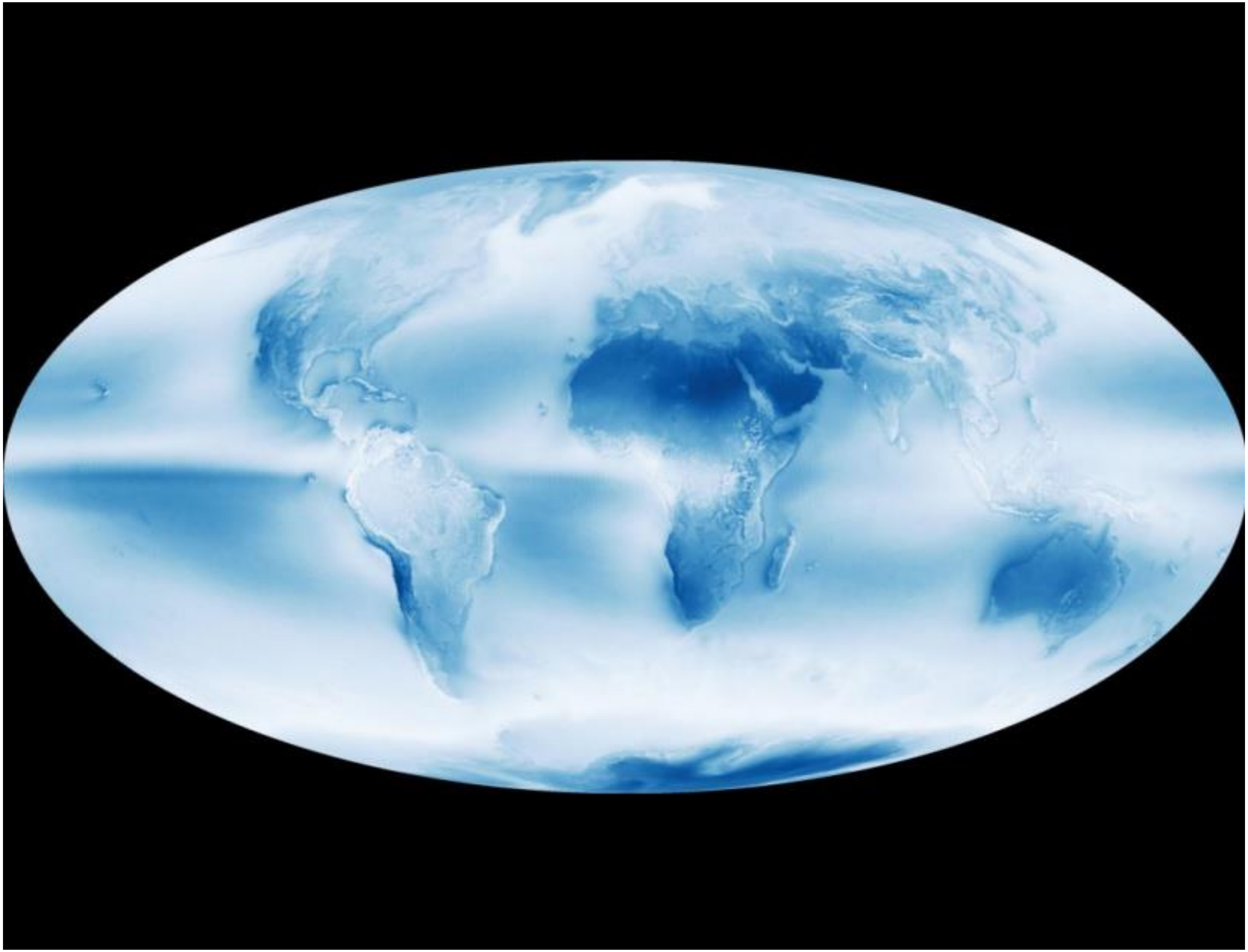
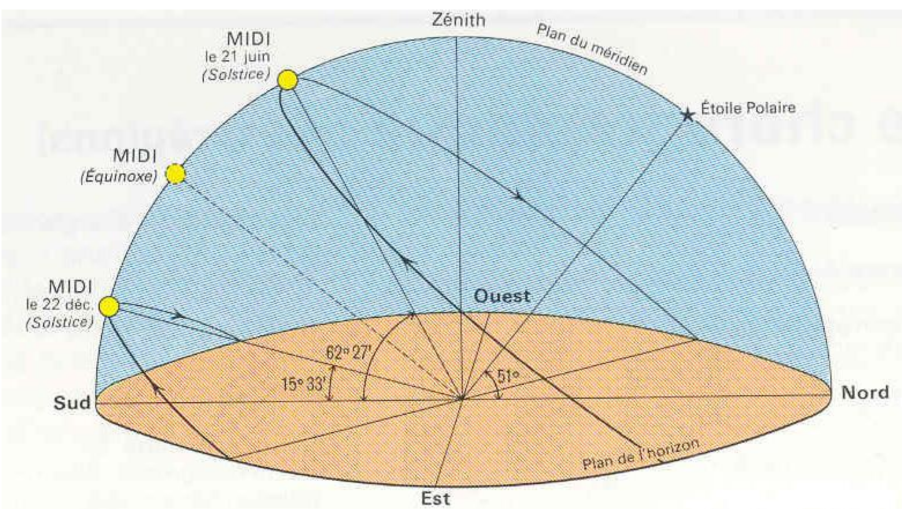
Benoit Beckers

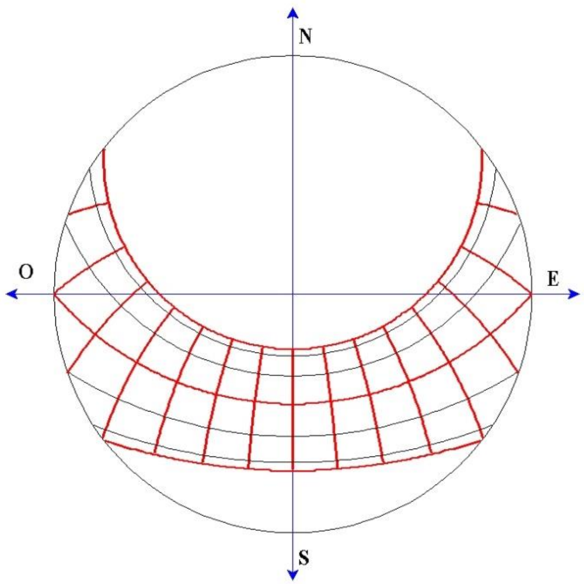
Architecture et Physique Urbaine - ISA BTP
Université de Pau et des Pays de l'Adour
Allée du Parc Montaury, 64600 Anglet (France)
benoit.beckers@univ-pau.fr, www.heliodon.net

Finite element simulation of thermal exchanges using isoparametric models

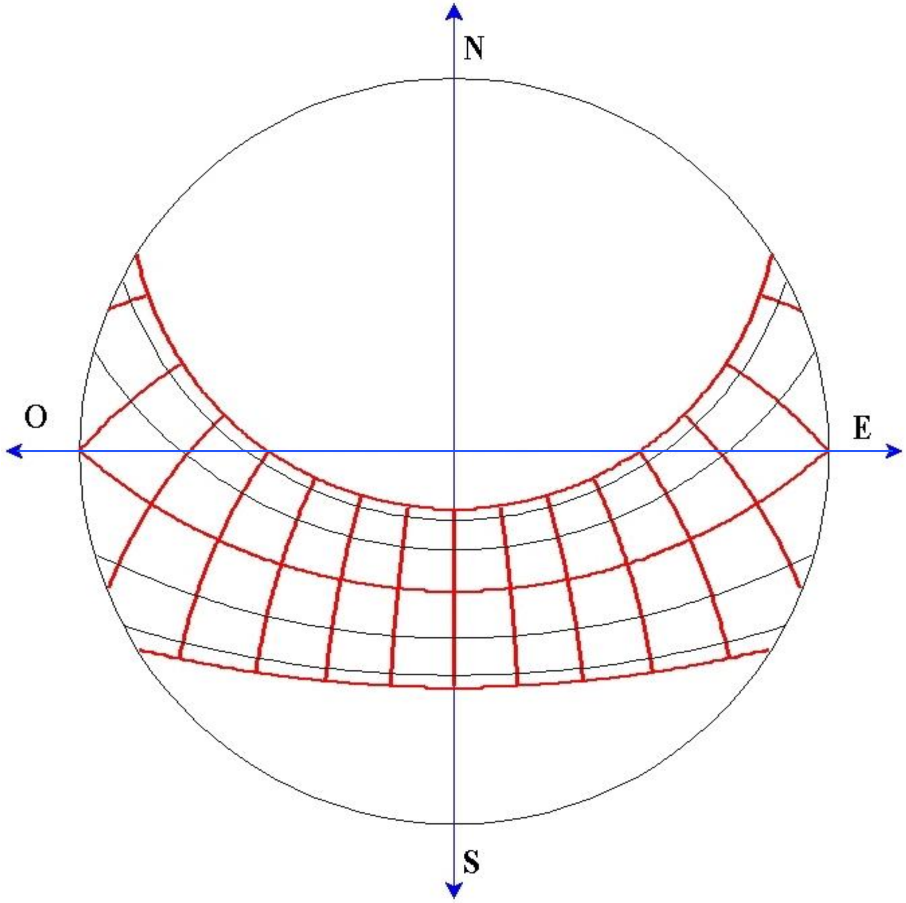


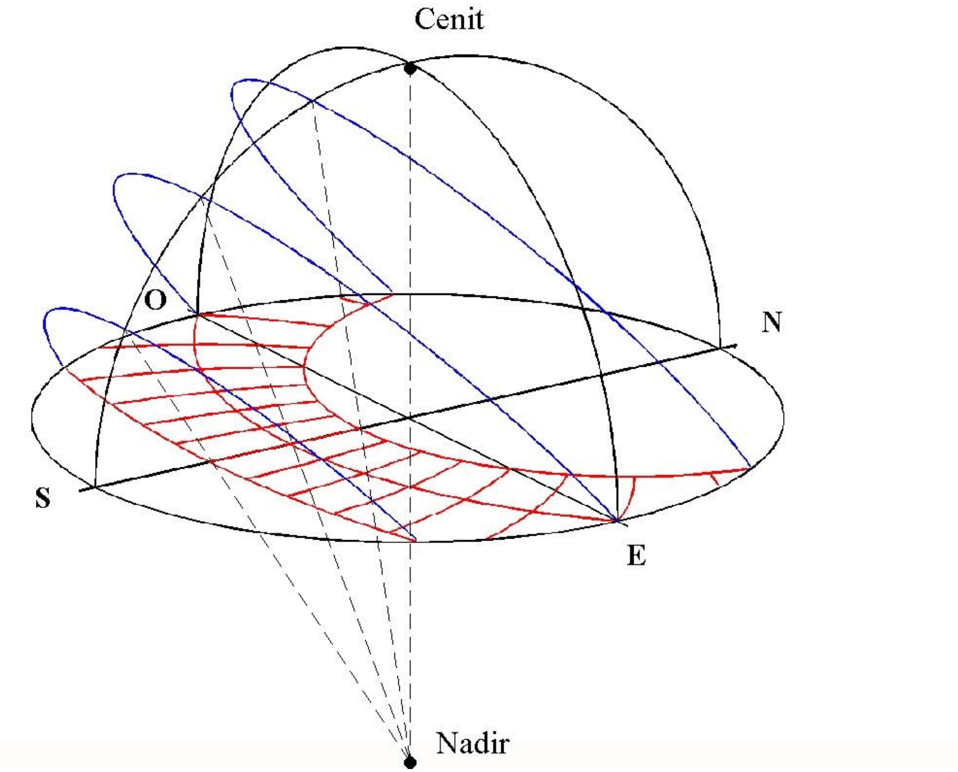
El Diagrama Solar



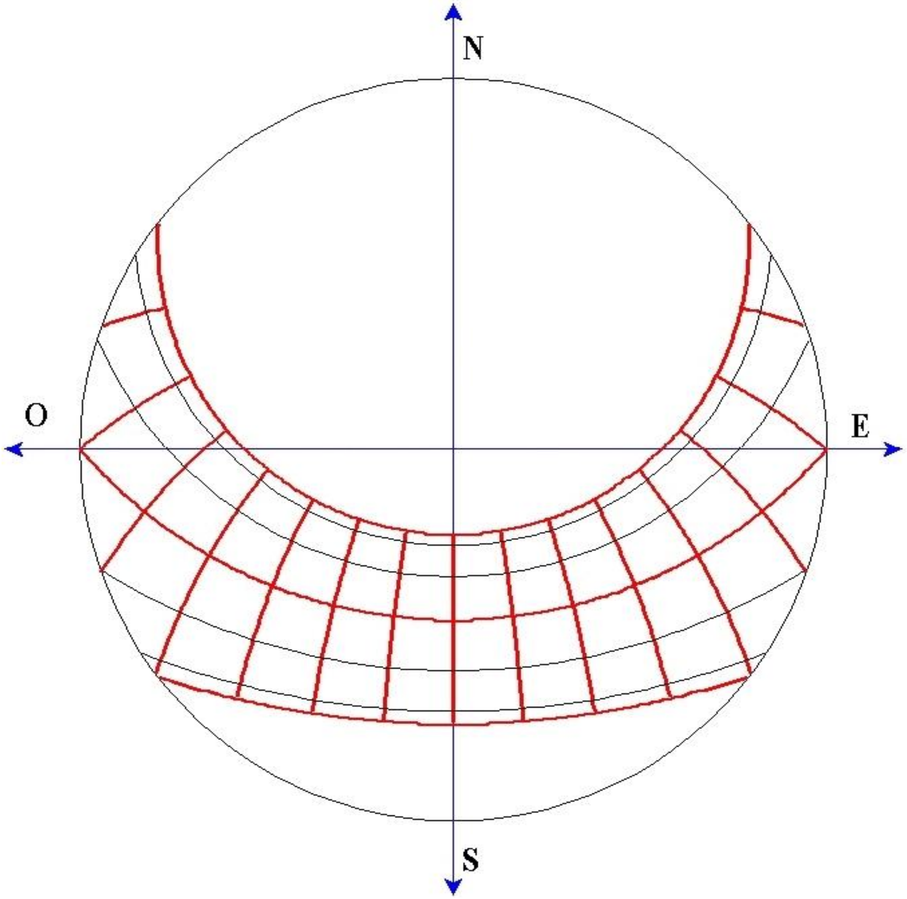


Barcelona 41:18:07 N

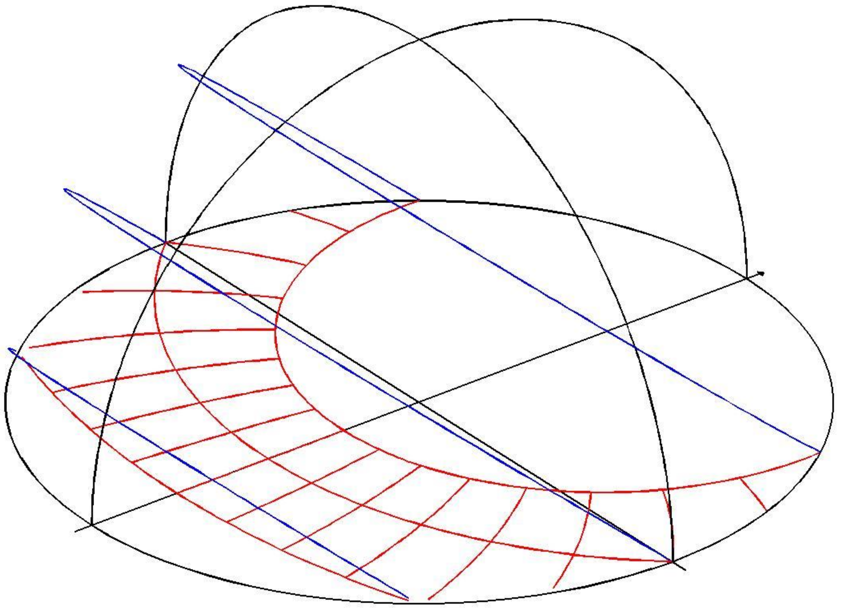




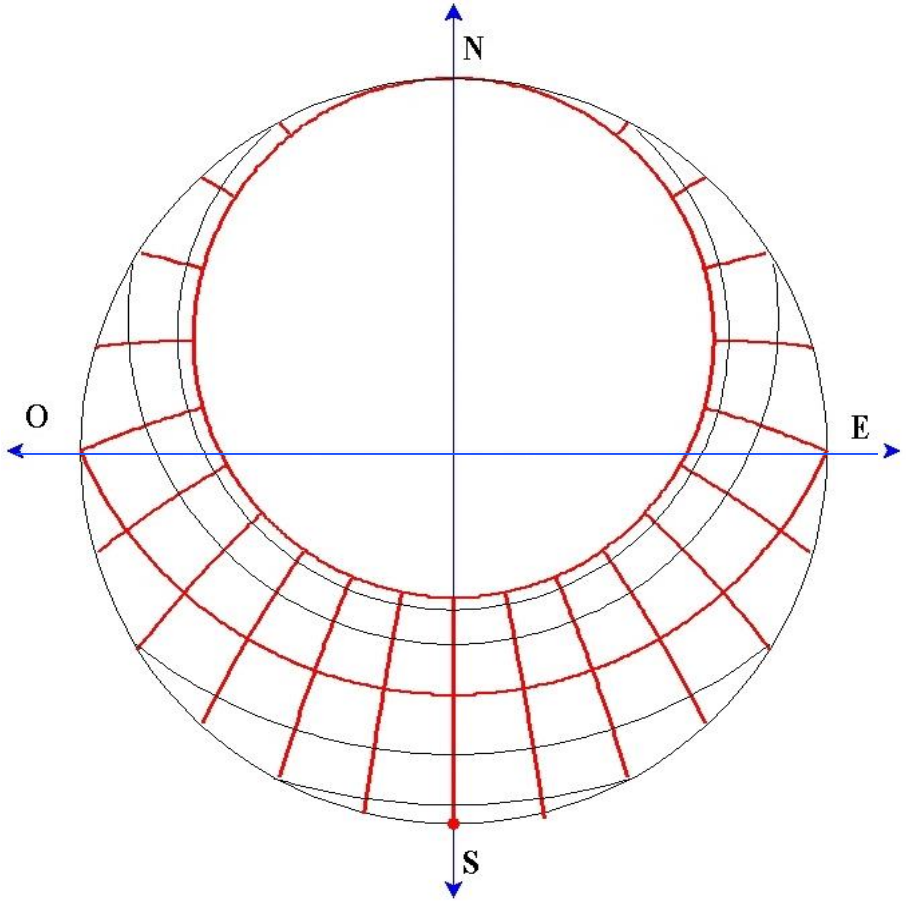
Paris 49.5° N

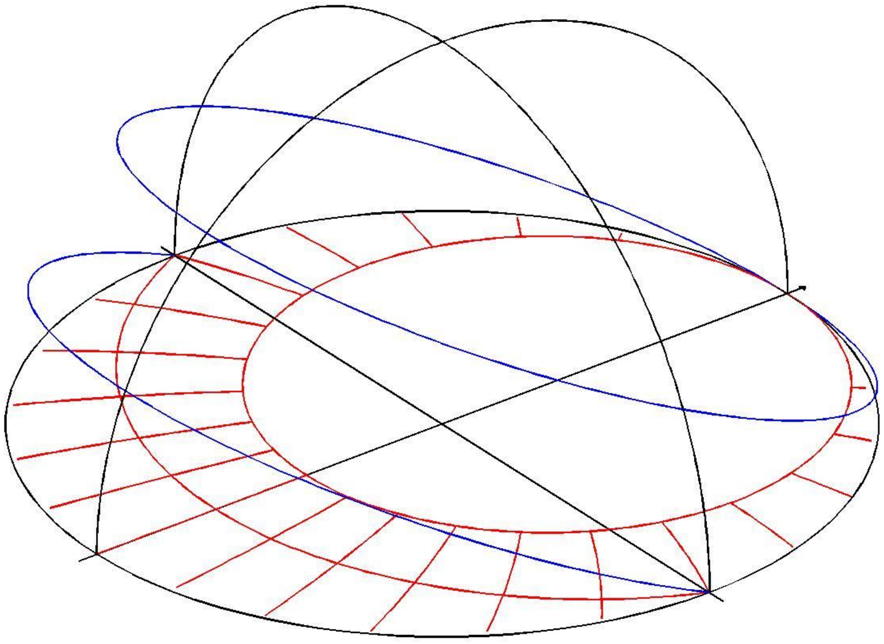


print



Círculo polar **66.5 N**



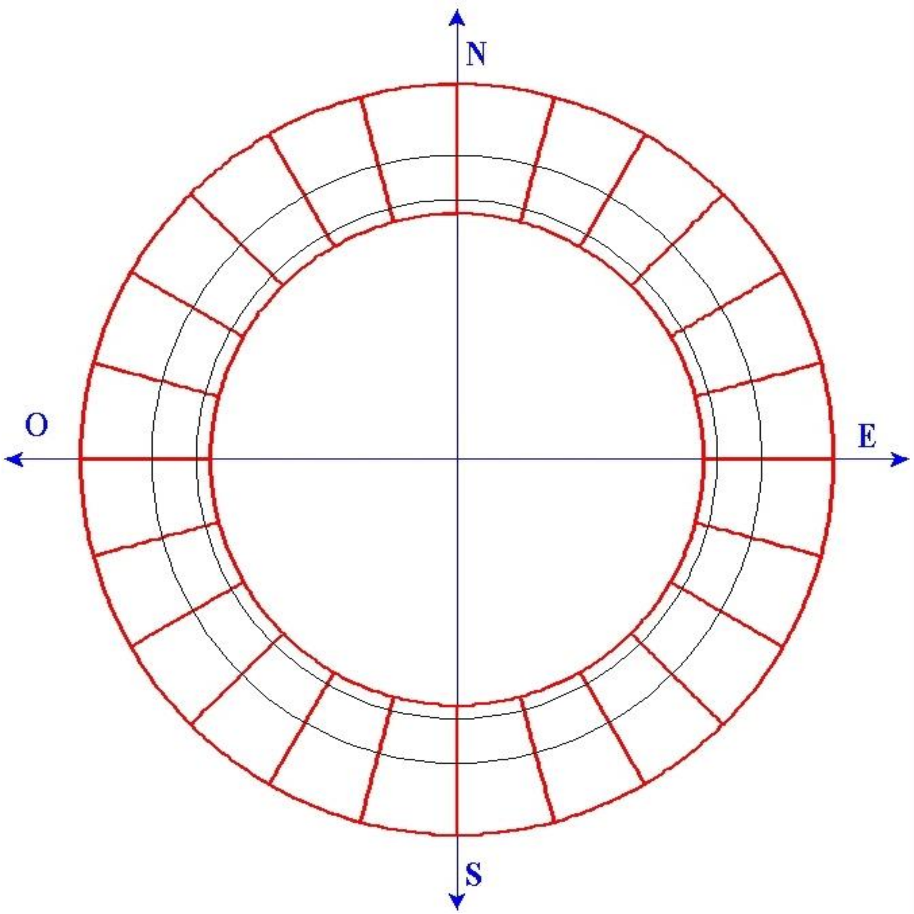


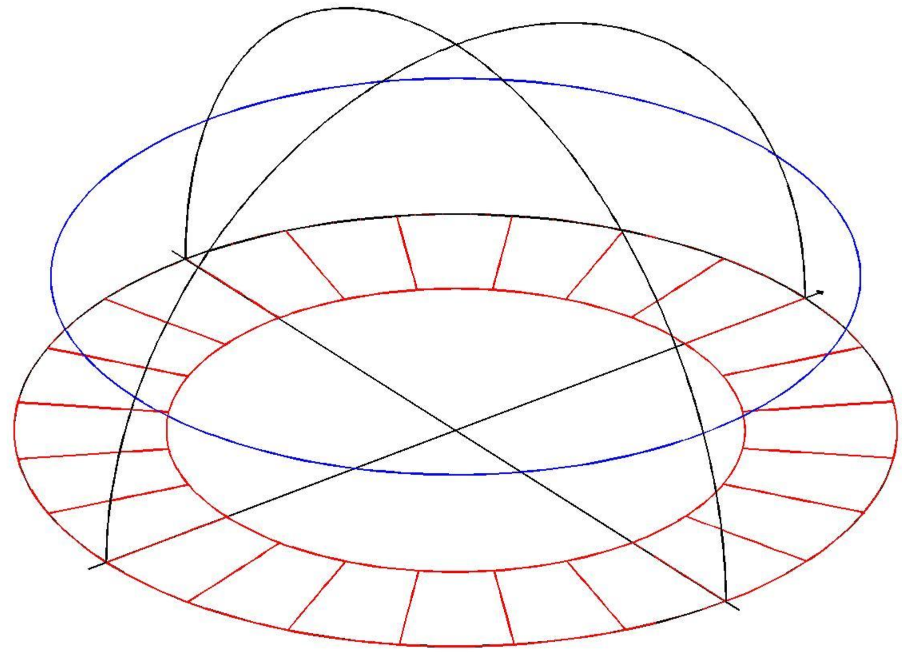
print

NORDKAPP
71° 10' 21" N

Midnattssolen sett fra Nordkapp
Midnight sun seen from the North Cape
Mitternachtssonne am Nordkap

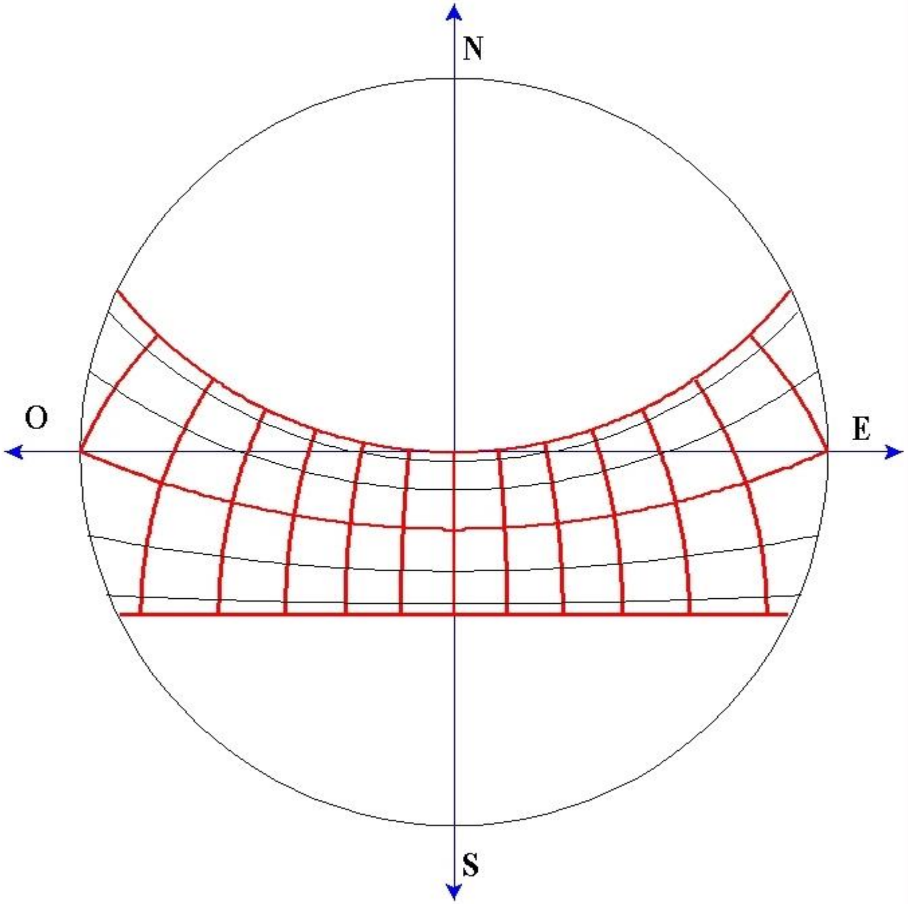
Polo Norte   90° N

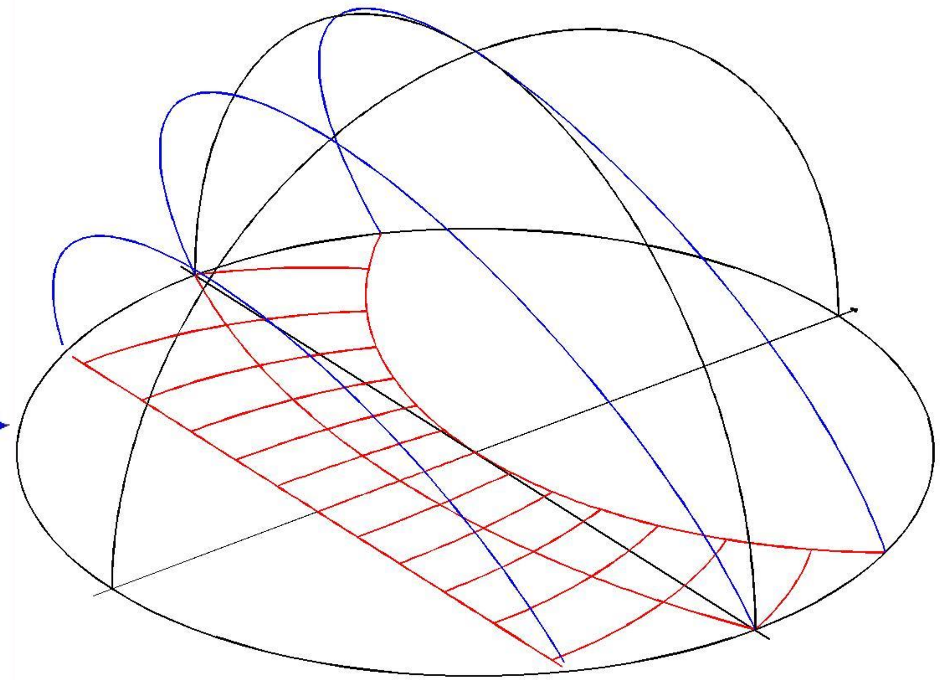


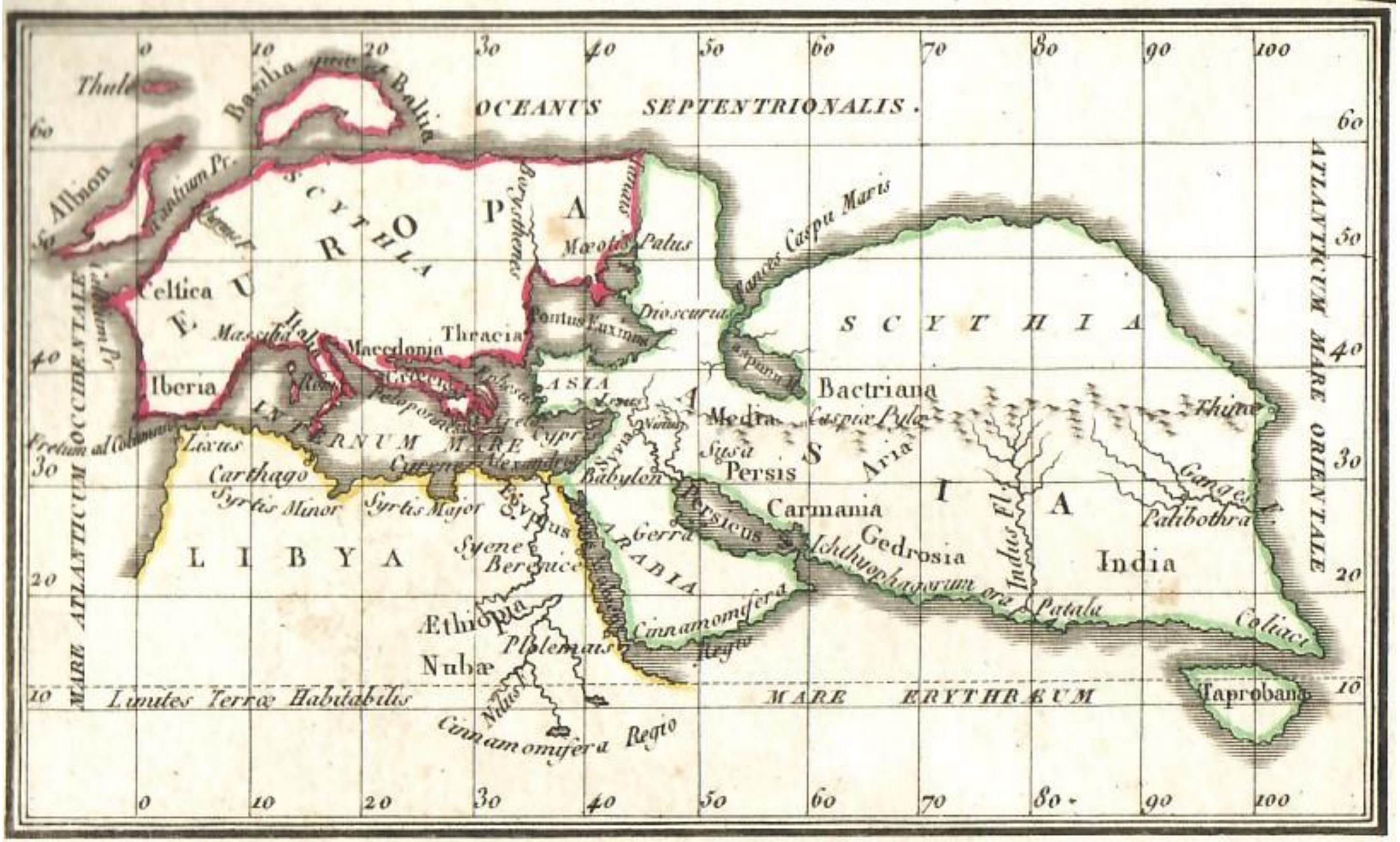


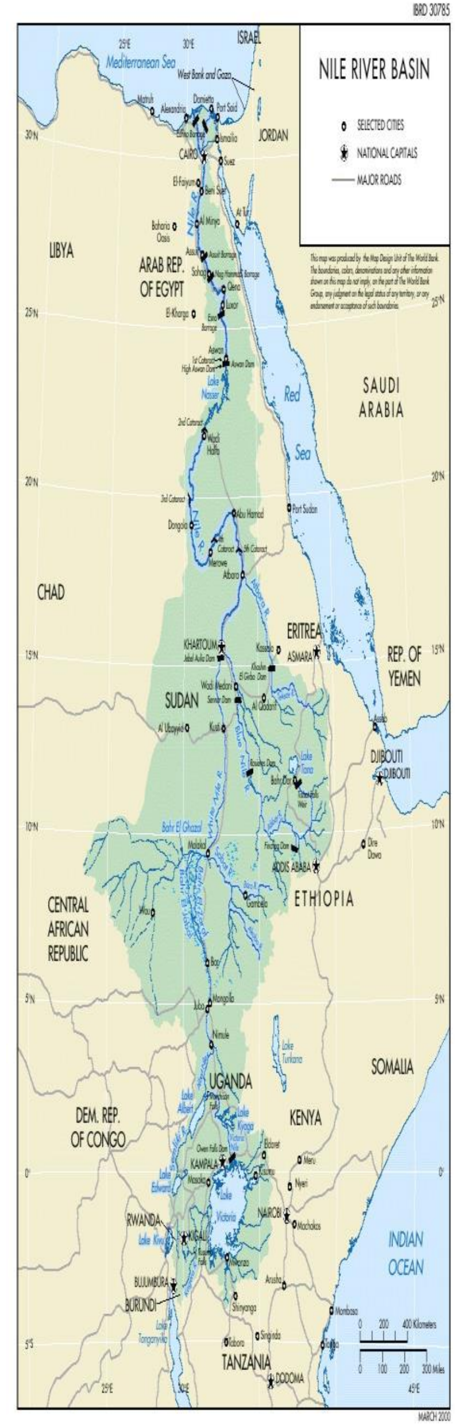
print

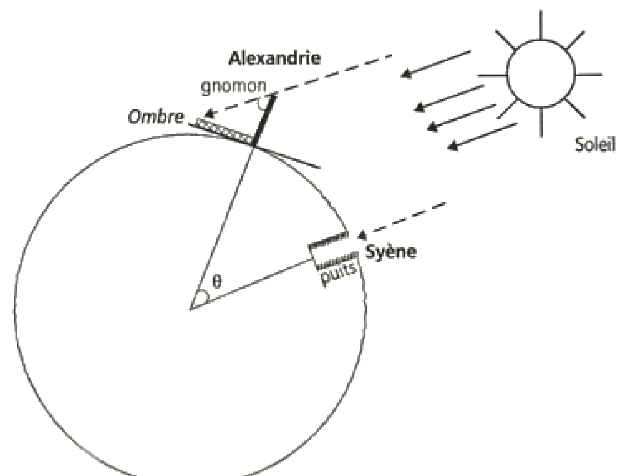
Trópico del Cáncer 23.5° N



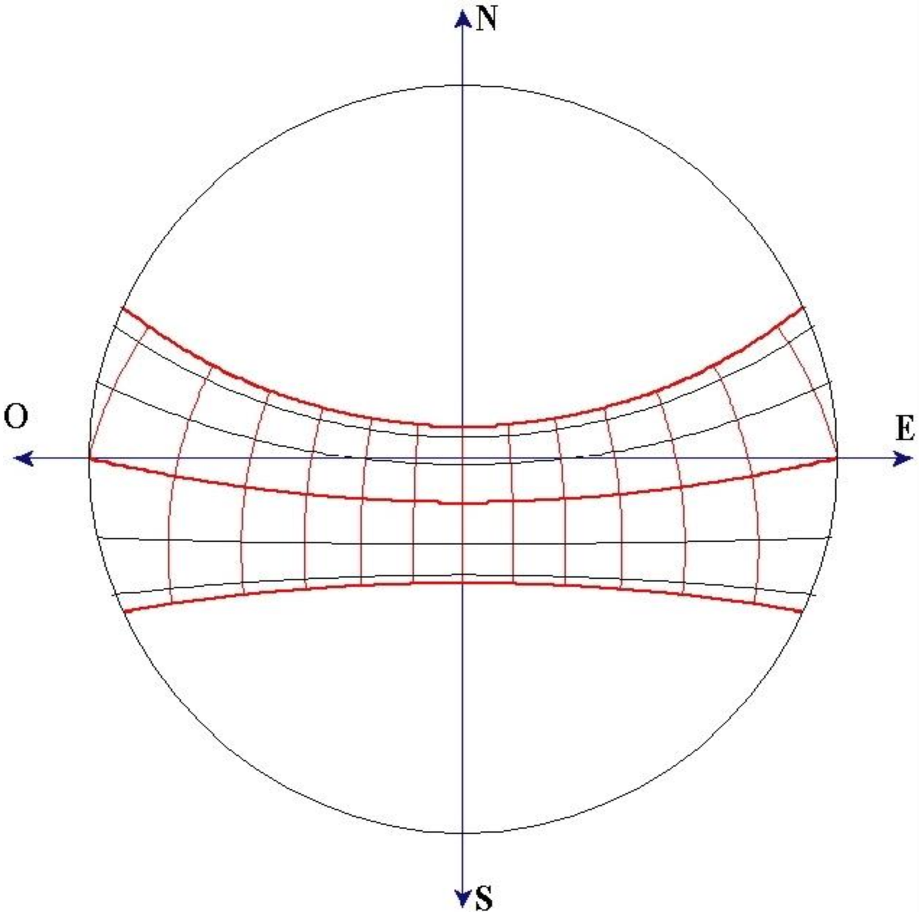


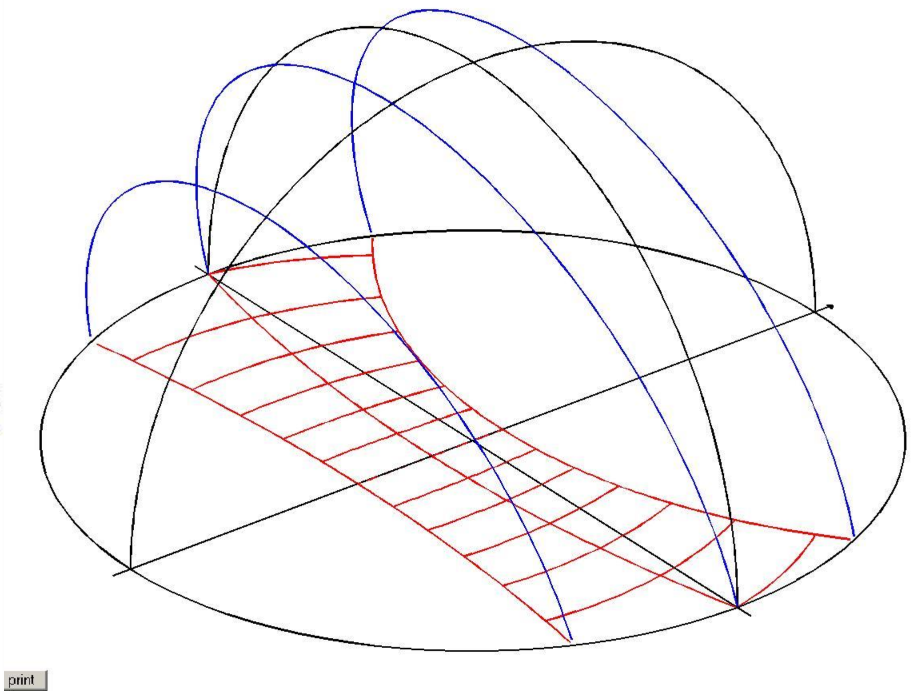

OCEANUS SEPTENTRIONALIS.
Thule
Basilia quæ et Baltia
Albion
Cantium Pr.
EUROPA
SCYTHIA
Borysthenes
Tanais
Mæotis Palus
Celtica
Massilia
Italia
Macedonia
Thracia
Pontus Euxinus
Dioscurias
Fauces Caspii Maris
SCYTHIA
Caspium M.
Iberia
Roma
Peloponnesus
ASIA
INTERNUM MARE
Fretum ad Columnas
Lixus
Cyrene
Alexandria
Cyprus
Media
Caspiæ Pylæ
Bactriana
Thinæ
Carthago
Syrtis Minor
Syrtis Major
Ægyptus
Babylon
Susa
Persis
Aria
ASIA
Persicus
Carmania
Gerra
Gedrosia
Ichthyophagorum ora
Indus Fl.
Ganges F.
Palibothra
India
LIBYA
Syene
Berenice
ARABIA
Æthiopia
Nubæ
Ptolemais
Cinnamomifera Regio
Patala
Coliaci
Taprobana
Nilus
Limites Terræ Habitabilis
MARE ERYTHRÆUM
MARE ATLANTICUM OCCIDENTALE
ATLANTICUM MARE ORIENTALE
Cinnamomifera Regio


IBRD 30785
NILE RIVER BASIN
SELECTED CITIES
NATIONAL CAPITALS
MAJOR ROADS
Mediterranean Sea
ISRAEL
West Bank and Gaza
JORDAN
CAIRO
LIBYA
ARAB REP. OF EGYPT
Red Sea
SAUDI ARABIA
CHAD
ERITREA
ASMARA
REP. OF YEMEN
KHARTOUM
SUDAN
DJIBOUTI
ETHIOPIA
ADDIS ABABA
CENTRAL AFRICAN REPUBLIC
SOMALIA
UGANDA
KENYA
DEM. REP. OF CONGO
KAMPALA
NAIROBI
RWANDA
KIGALI
BUJUMBURA
BURUNDI
TANZANIA
DODOMA
INDIAN OCEAN
MARCH 2000

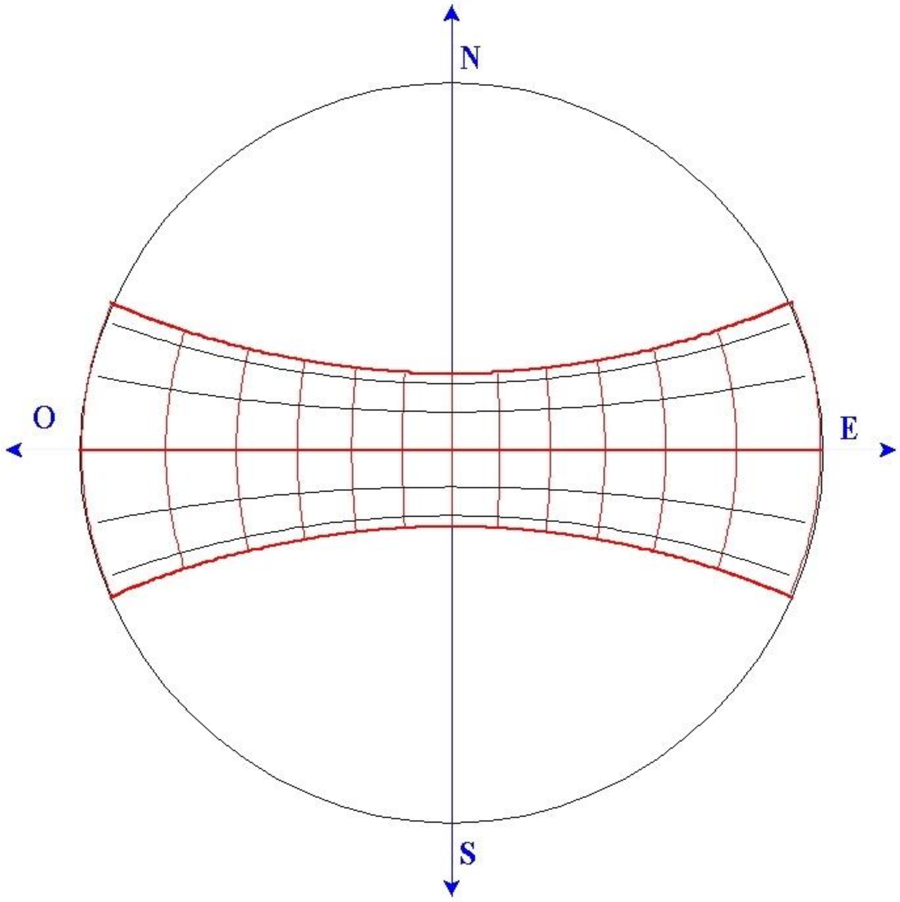

Alexandrie
gnomon
Ombre
Soleil
Syène
puits
θ

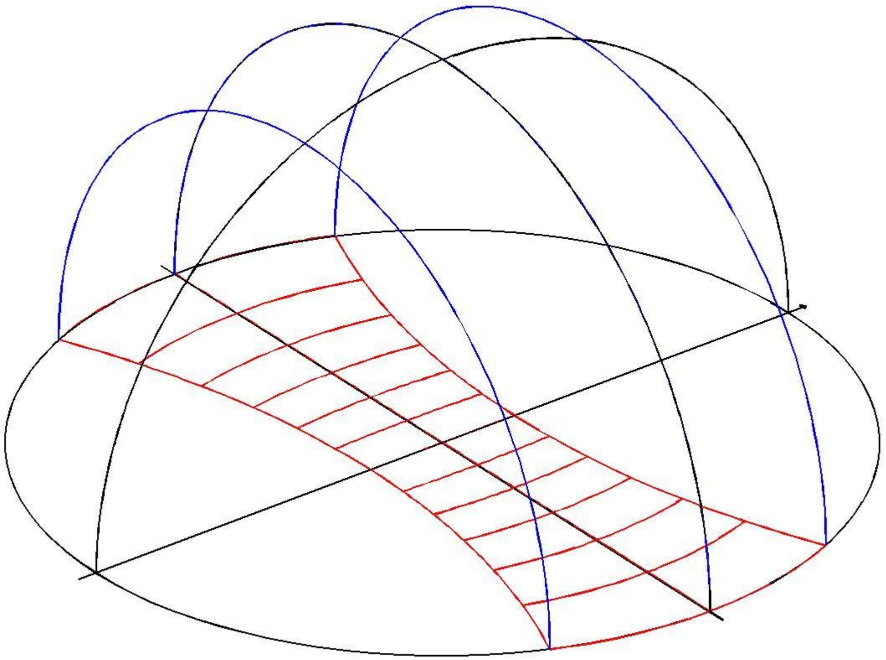
San Salvador 13:42:00 N



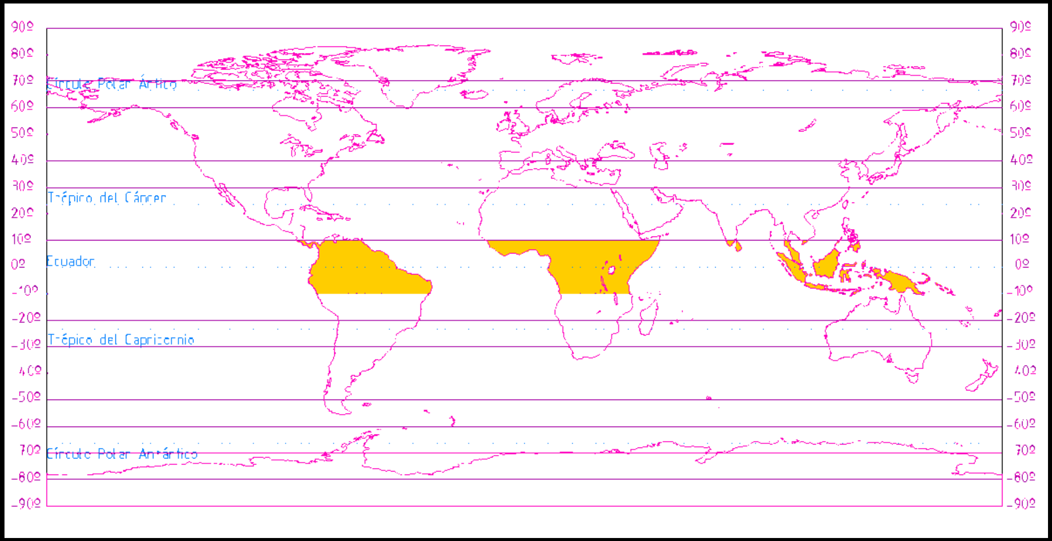


Ecuador 0°





Zone équatoriale. Une architecture qui imite les arbres



9°36'N

Purus, Pérou 9°26'S

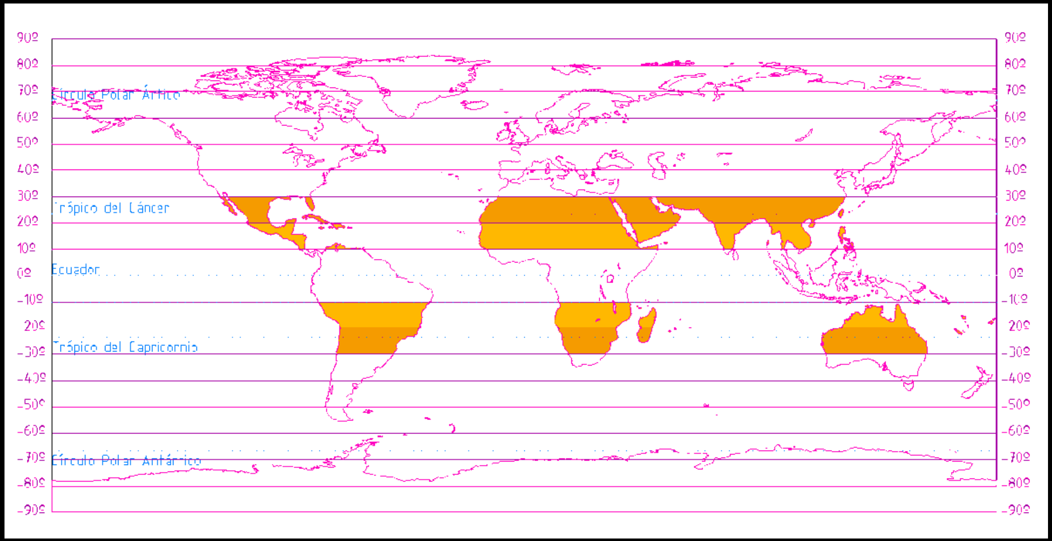
Île de Falik 7°15'N

Ambarita, Indonésie 2°40'N

South Sulawesi, Indonésie. 4°20'S

Kasubi, Ouganda0°20'N

Zone tropicale humide. Une architecture de parapluies et de saillants contre la pluie et le soleil




Isla Waya, Fiji 17°17'S


Bonampak, Mexique 16°42'N


Valle de Viñales, Cuba 22°37'N

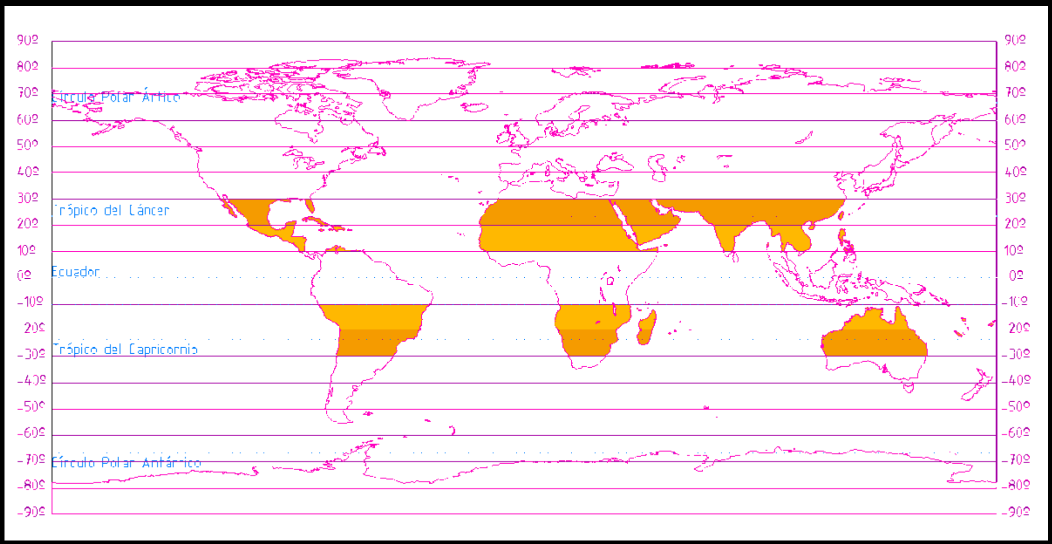

Sahebganj, Inde


Diamantina, Brésil 18°12'S


Hoi An, Vietnam 17°20'N


Lijiang, Chine 26°51'N

Zone tropicale aride. Architecture de forteresses, tentes et carapaces contre le soleil



E

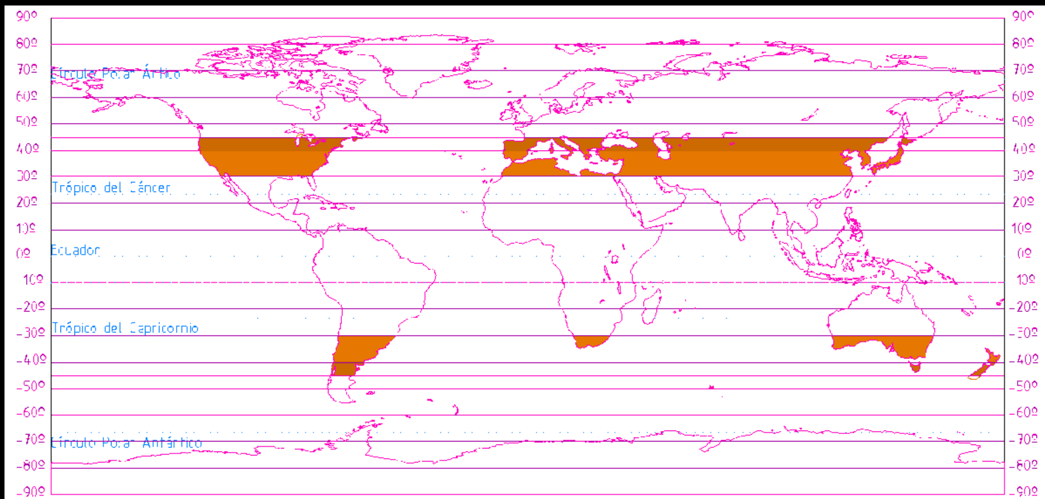
14°20'N

Shibam, Yemen 15°55'N

Djenné, Mali 13°54'N

Jaipur, Inde 27°N

Zona tempérée chaude. Architecture d’ombres et inertie thermique



30°22'N

37°16'N

Fez, Maroc 34°03'

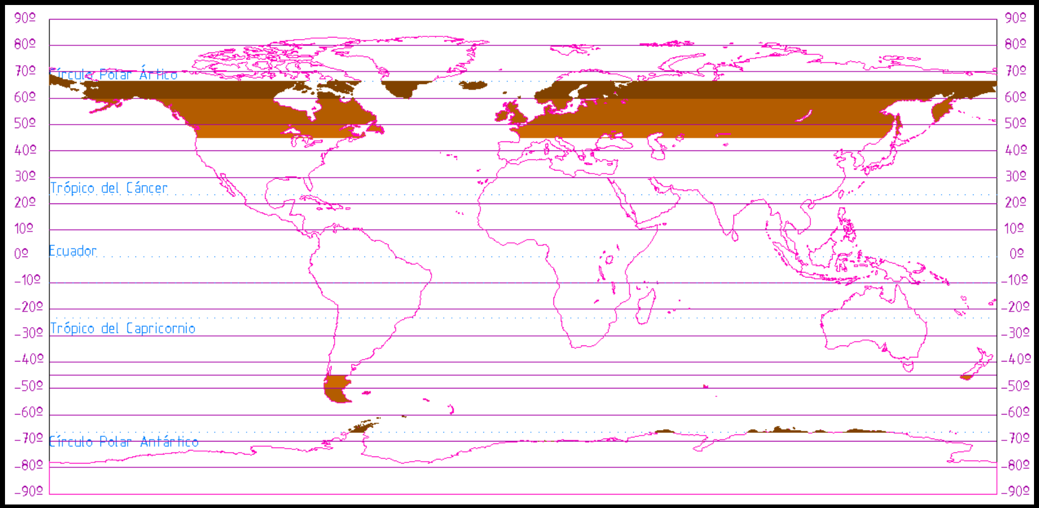
Damas, Syrie

33°31'N

Córdoba, Espagne 37°53'N

Esfahan, Iran 32°39'N

Zone tempérée froide. Architecture où le soleil est bienvenu, mais où la priorité est de conserver la chaleur.



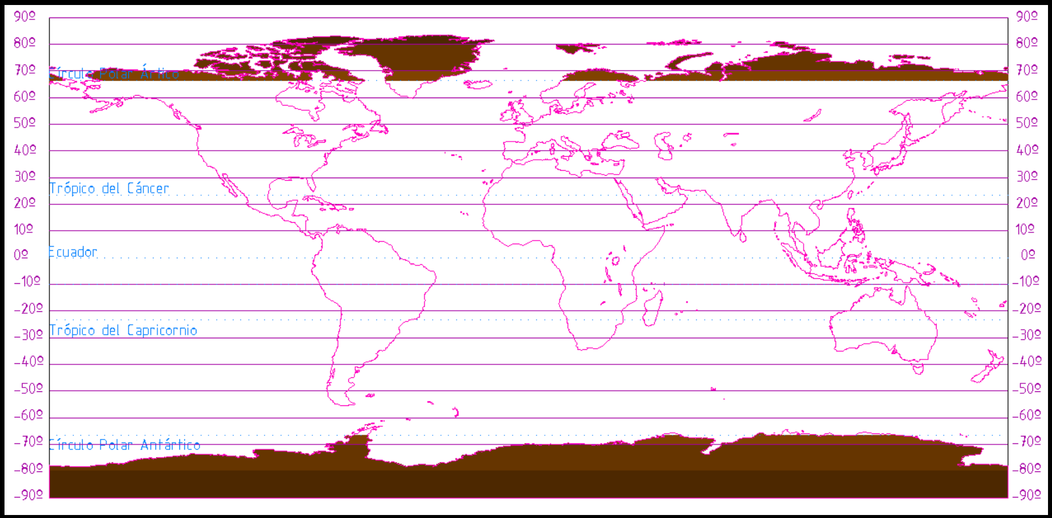
Reykiavik, Islande 64°08'N

Ushuaia, Argentine 54°47'S

Massachusetts, EUA 42°14'N

Visby, Suède 56°31'N

Zone froide. Architecture contre le froid, les maîtres de l' isolation

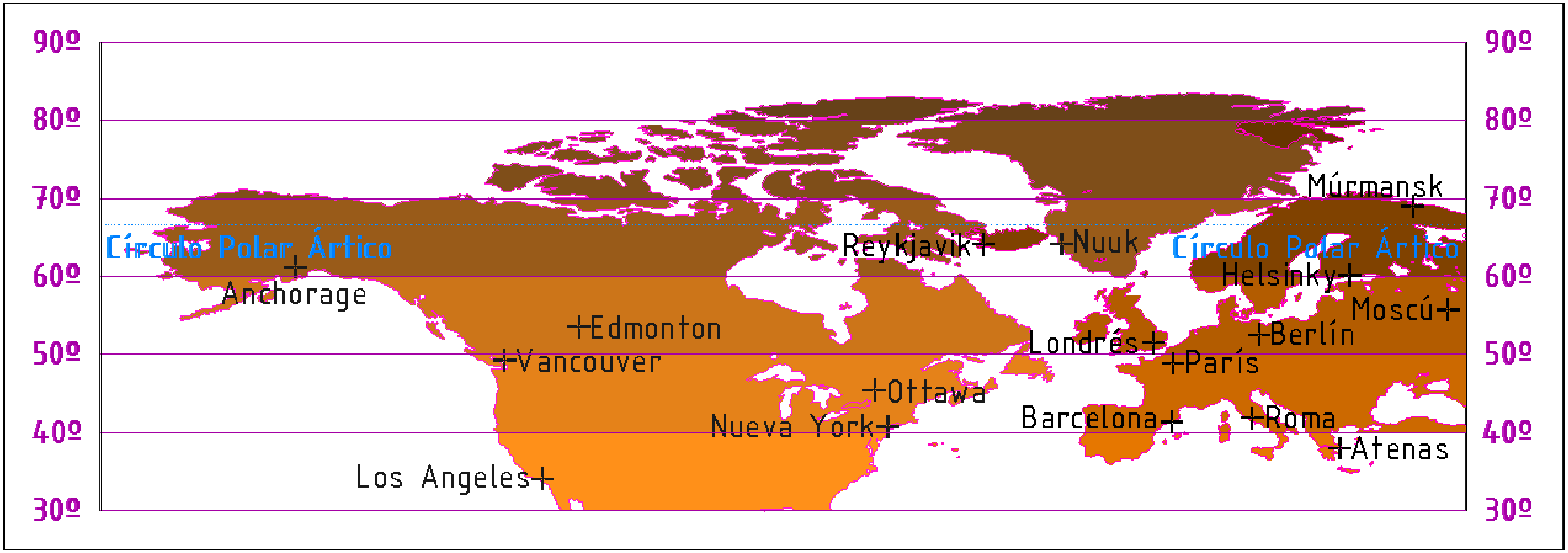


Norilsk, Russie 69°20'N

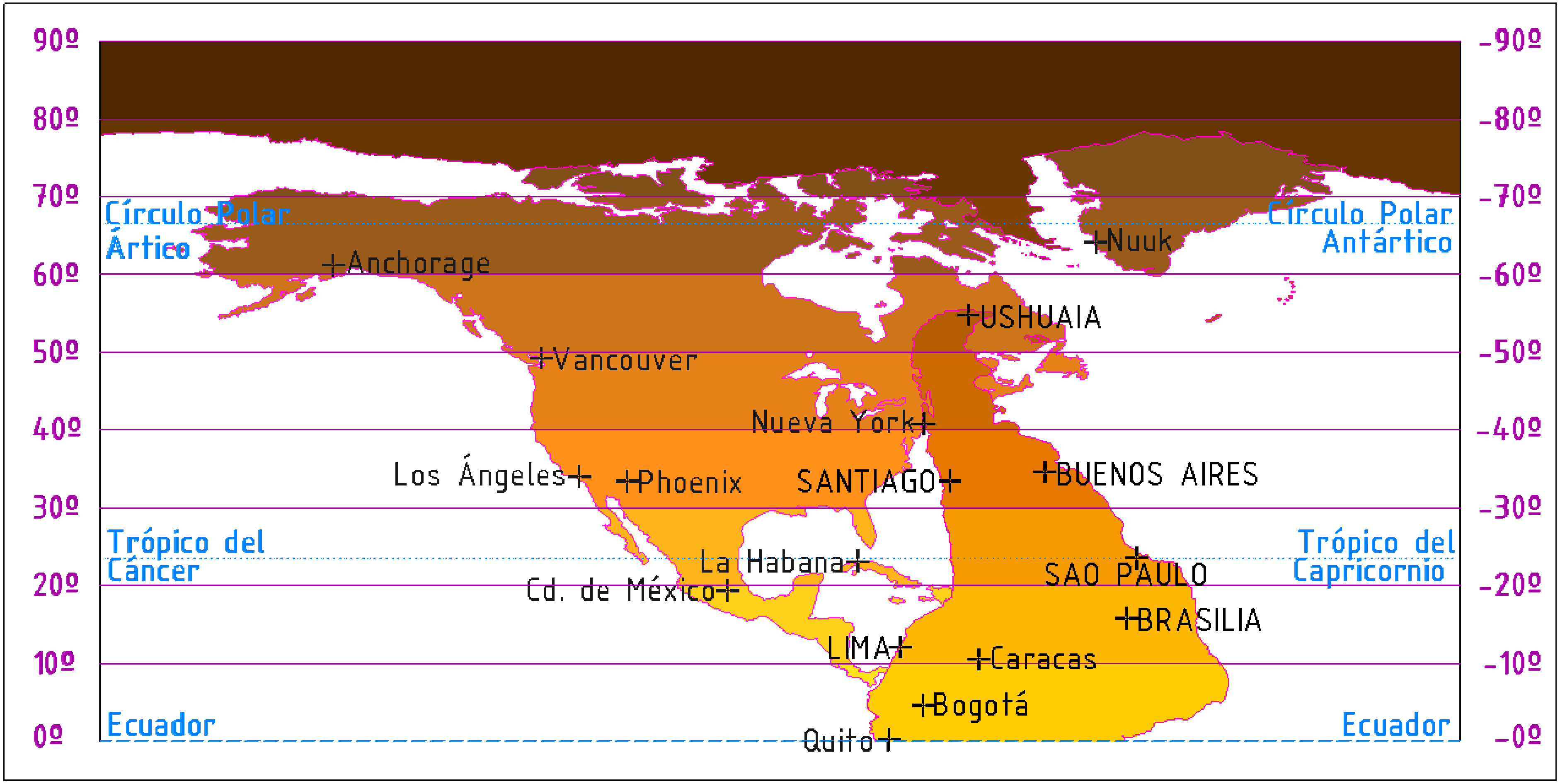
Campement d'été chez les Inuits 67°51'N

Igloo chez les Inuits (Groenland, Alaska et Nord du Canada)

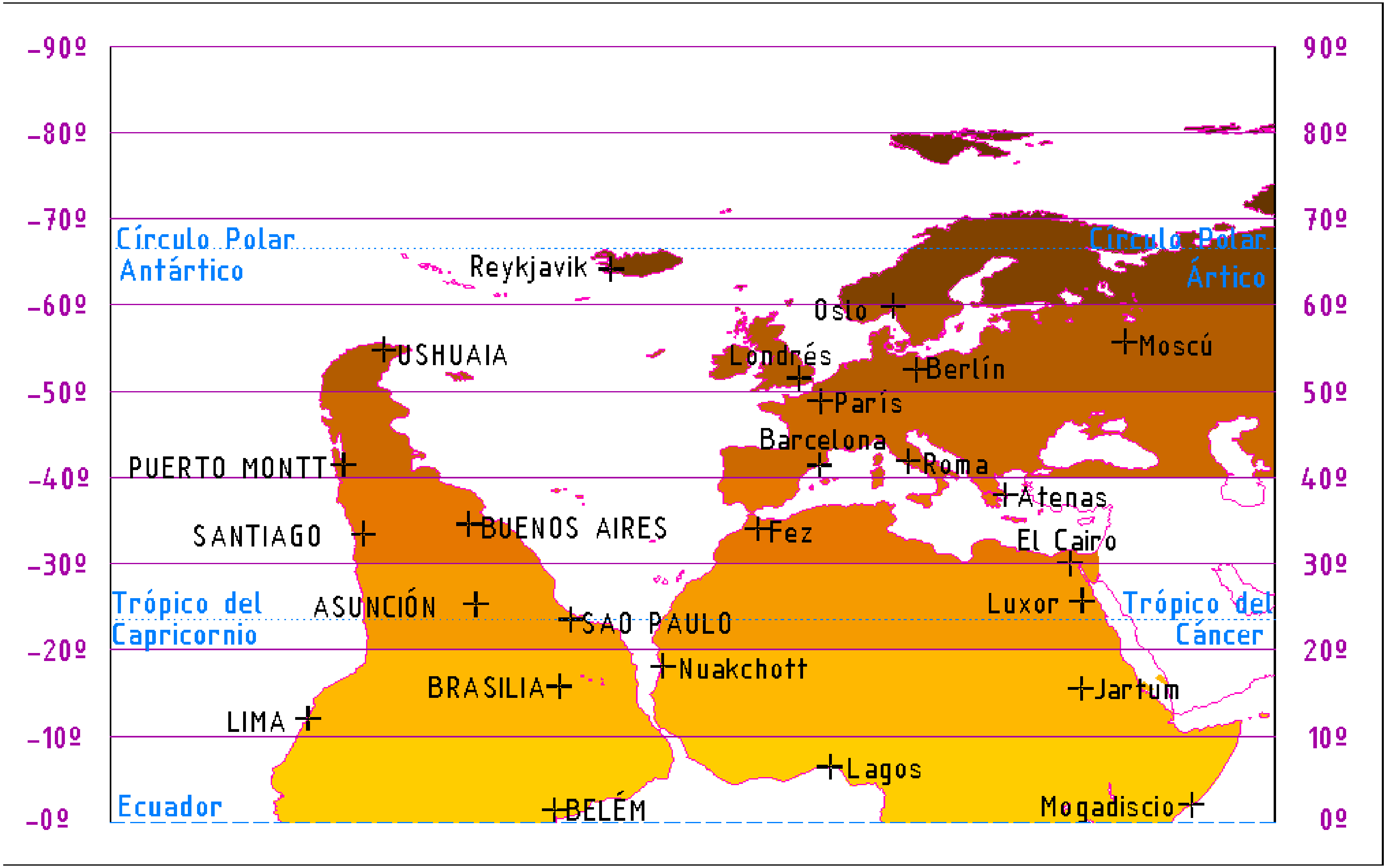
Hotel de Glace, Kiruna, Suède

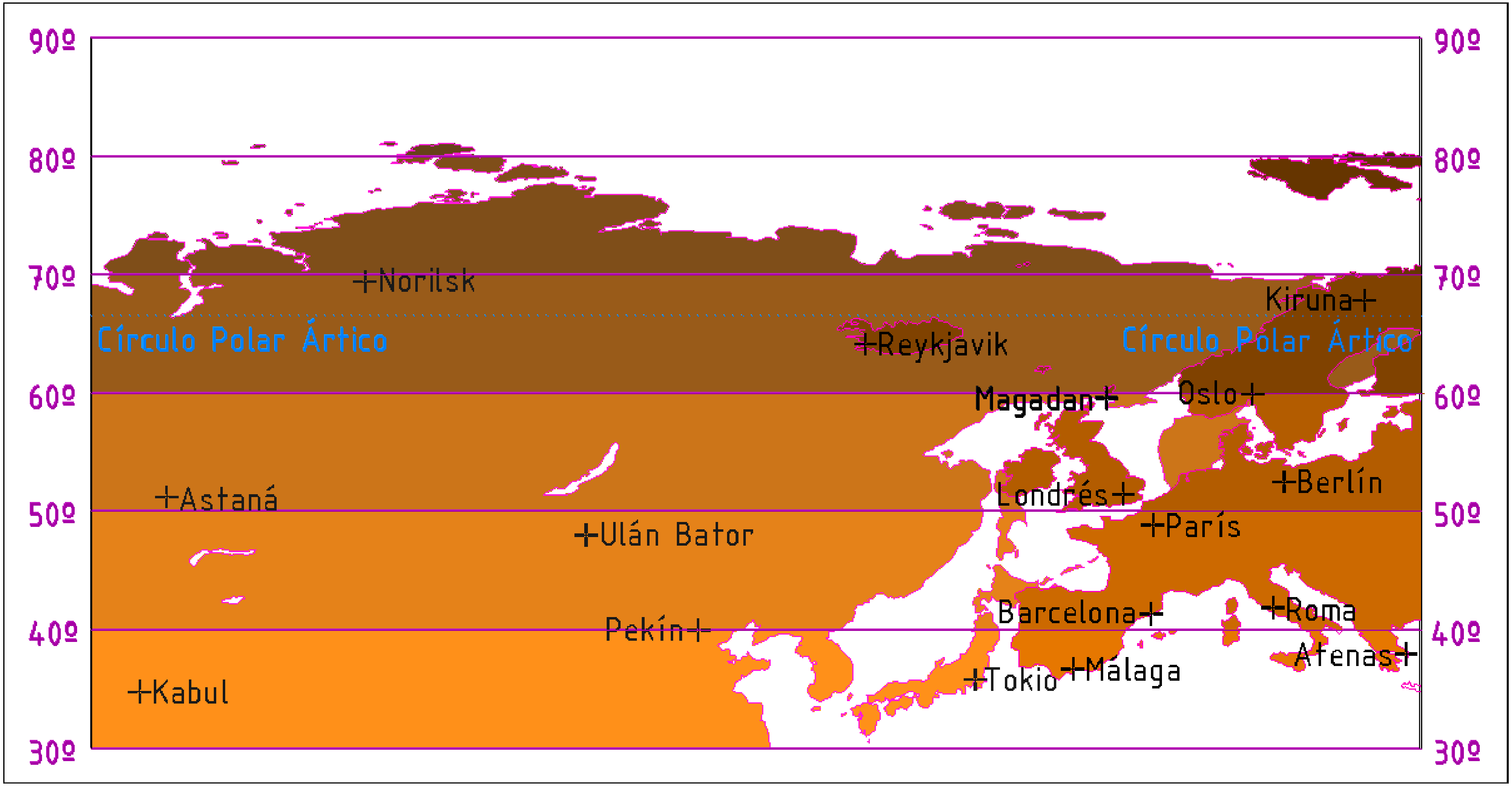


90º
80º
70º
60º
50º
40º
30º
Círculo Polar Ártico
Anchorage
Edmonton
Vancouver
Los Angeles
Nueva York
Ottawa
Reykjavik
Nuuk
Múrmansk
Helsinky
Moscú
Londrés
Berlín
París
Barcelona
Roma
Atenas

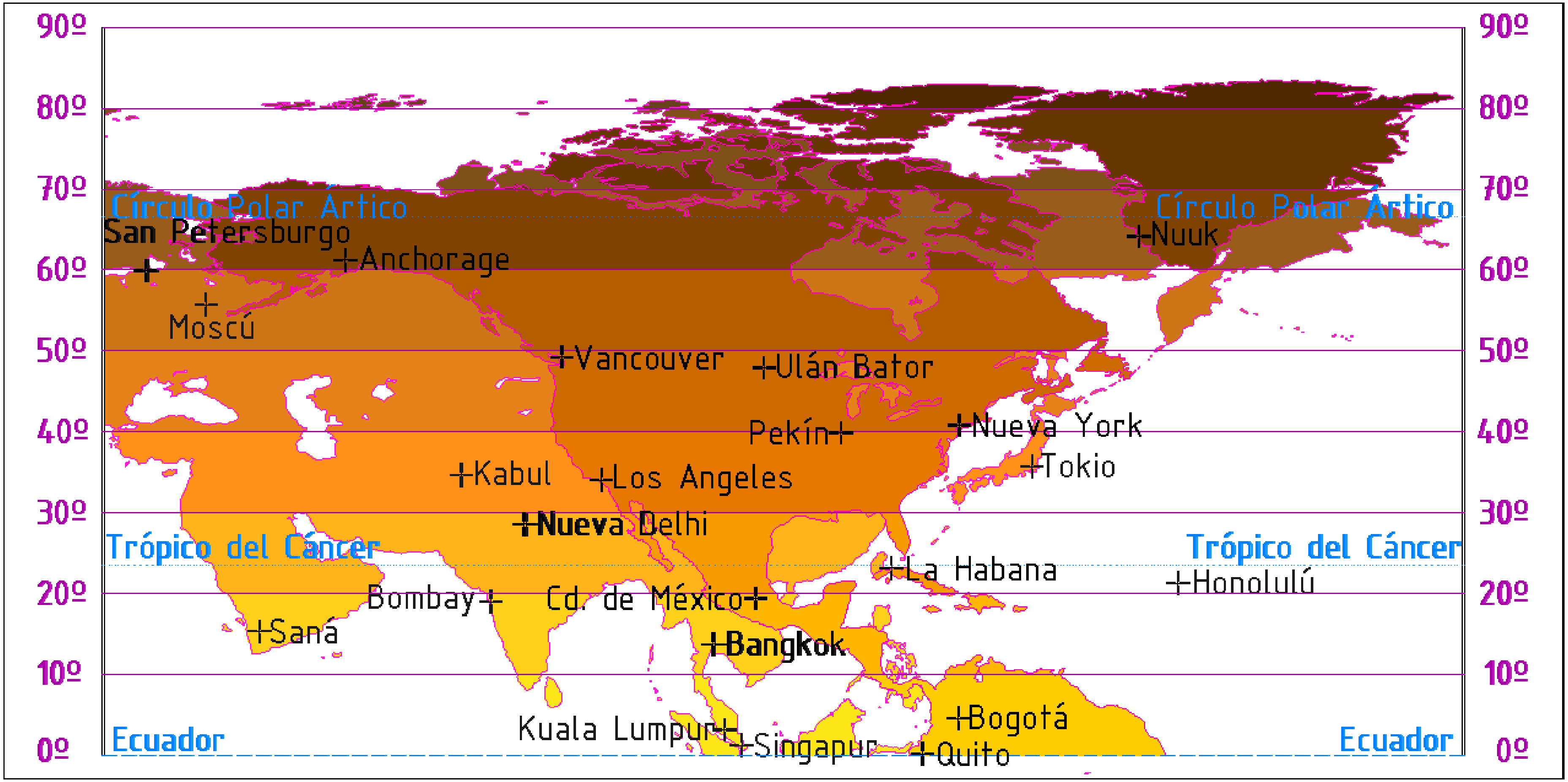


90º
80º
70º
60º
50º
40º
30º
20º
10º
0º
-90º
-80º
-70º
-60º
-50º
-40º
-30º
-20º
-10º
-0º
Círculo Polar Ártico
Círculo Polar Antártico
Trópico del Cáncer
Trópico del Capricornio
Ecuador
Ecuador
Anchorage
Nuuk
USHUAIA
Vancouver
Nueva York
Los Ángeles
Phoenix
SANTIAGO
BUENOS AIRES
La Habana
SAO PAULO
Cd. de México
BRASILIA
LIMA
Caracas
Bogotá
Quito


-90º
-80º
-70º
-60º
-50º
-40º
-30º
-20º
-10º
-0º
90º
80º
70º
60º
50º
40º
30º
20º
10º
0º
Círculo Polar Antártico
Círculo Polar Ártico
Trópico del Capricornio
Trópico del Cáncer
Ecuador
Reykjavik
Oslo
Londrés
Berlín
Moscú
París
Barcelona
Roma
Atenas
Fez
El Cairo
Luxor
Nuakchott
Jartum
Lagos
Mogadiscio
USHUAIA
PUERTO MONTT
SANTIAGO
BUENOS AIRES
ASUNCIÓN
SAO PAULO
BRASILIA
LIMA
BELÉM



90º
80º
70º
60º
50º
40º
30º
Norilsk
Círculo Polar Ártico
Reykjavik
Kiruna
Círculo Polar Ártico
Magadan
Oslo
Astaná
Berlín
Londrés
Ulán Bator
París
Barcelona
Roma
Pekín
Atenas
Kabul
Tokio
Málaga

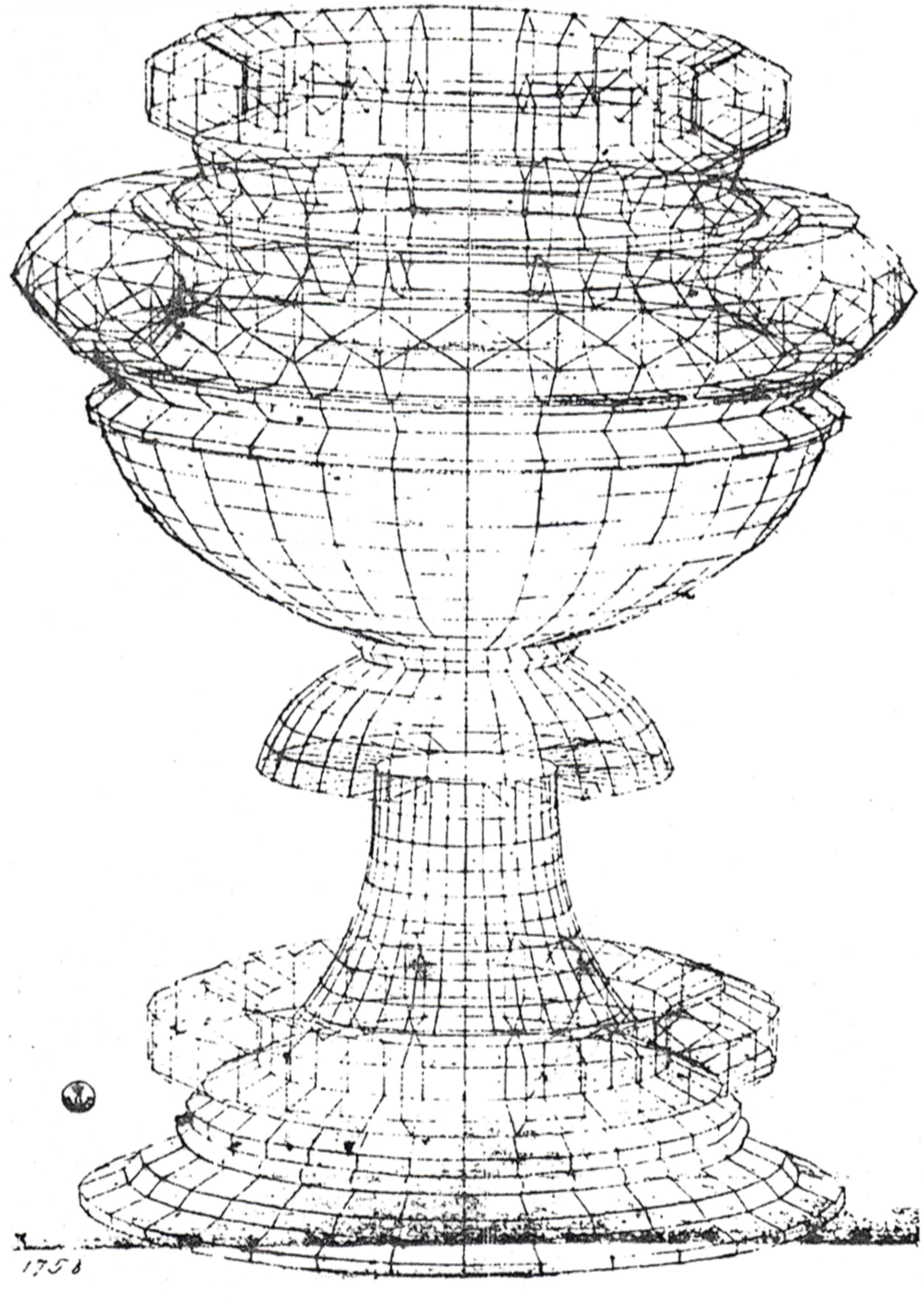


90º
80º
70º
60º
50º
40º
30º
20º
10º
0º
Círculo Polar Ártico
Trópico del Cáncer
Ecuador
San Petersburgo
Anchorage
Moscú
Nuuk
Vancouver
Ulán Bator
Pekín
Nueva York
Tokio
Kabul
Los Angeles
Nueva Delhi
La Habana
Honolulú
Bombay
Cd. de México
Saná
Bangkok
Kuala Lumpur
Singapur
Bogotá
Quito


DIÁRIO DO PARÁ
Banca
Tamandaré

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

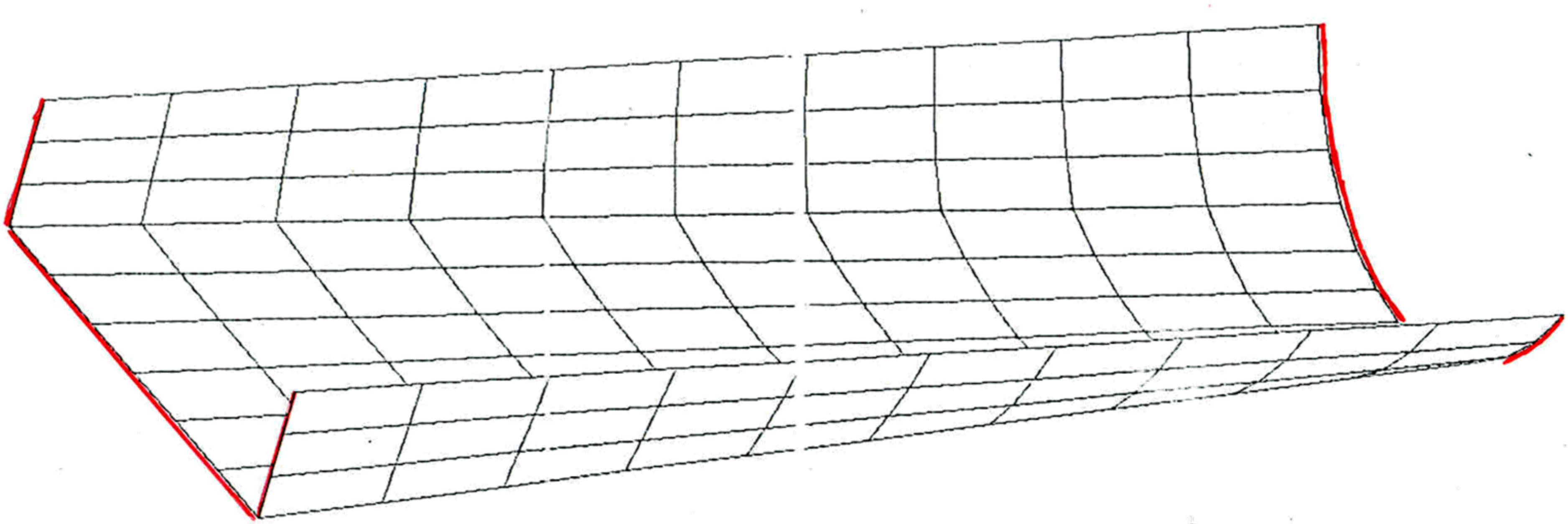


Uccello's chalice.

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

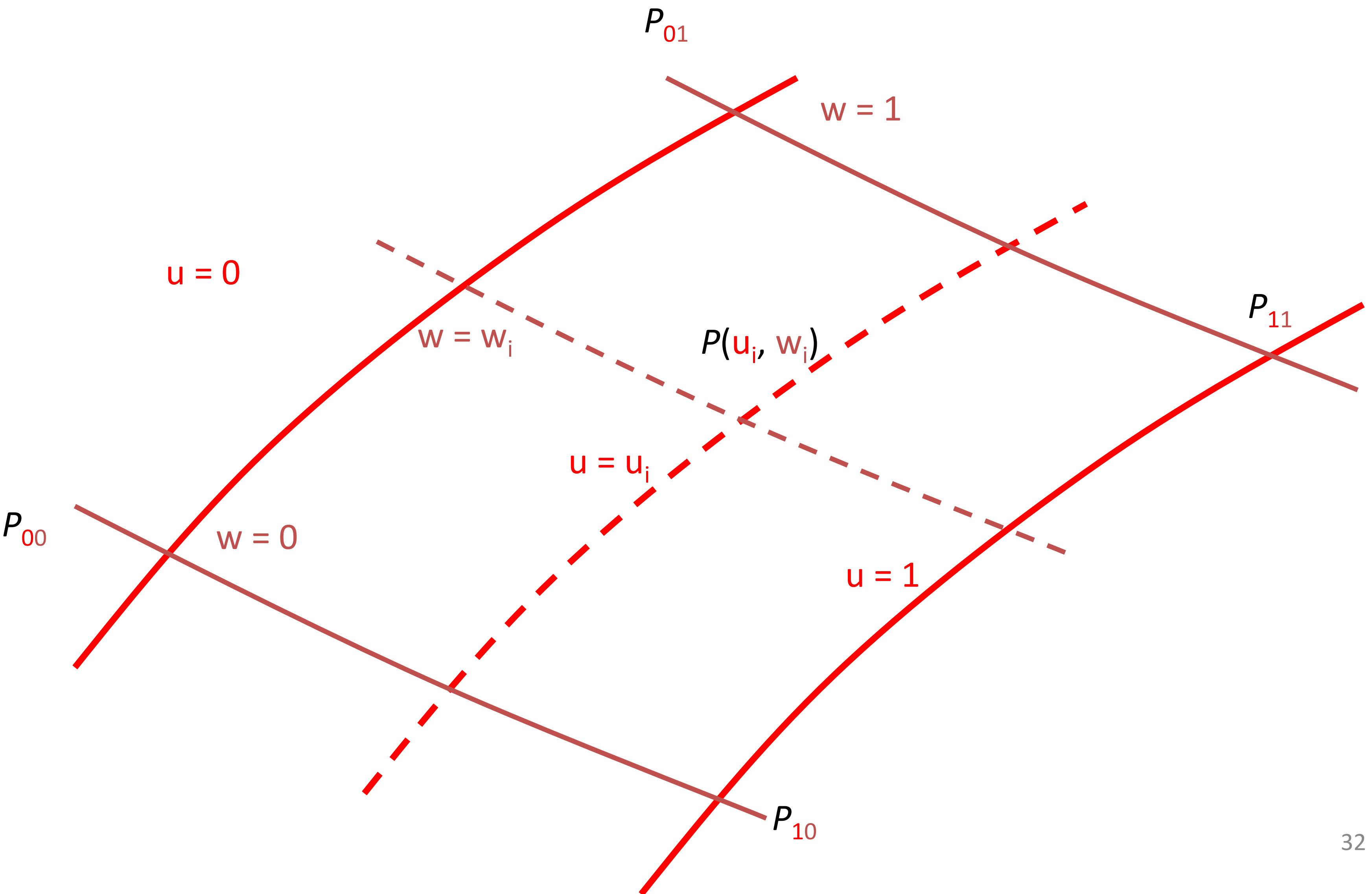
Lofting or « gabariage »



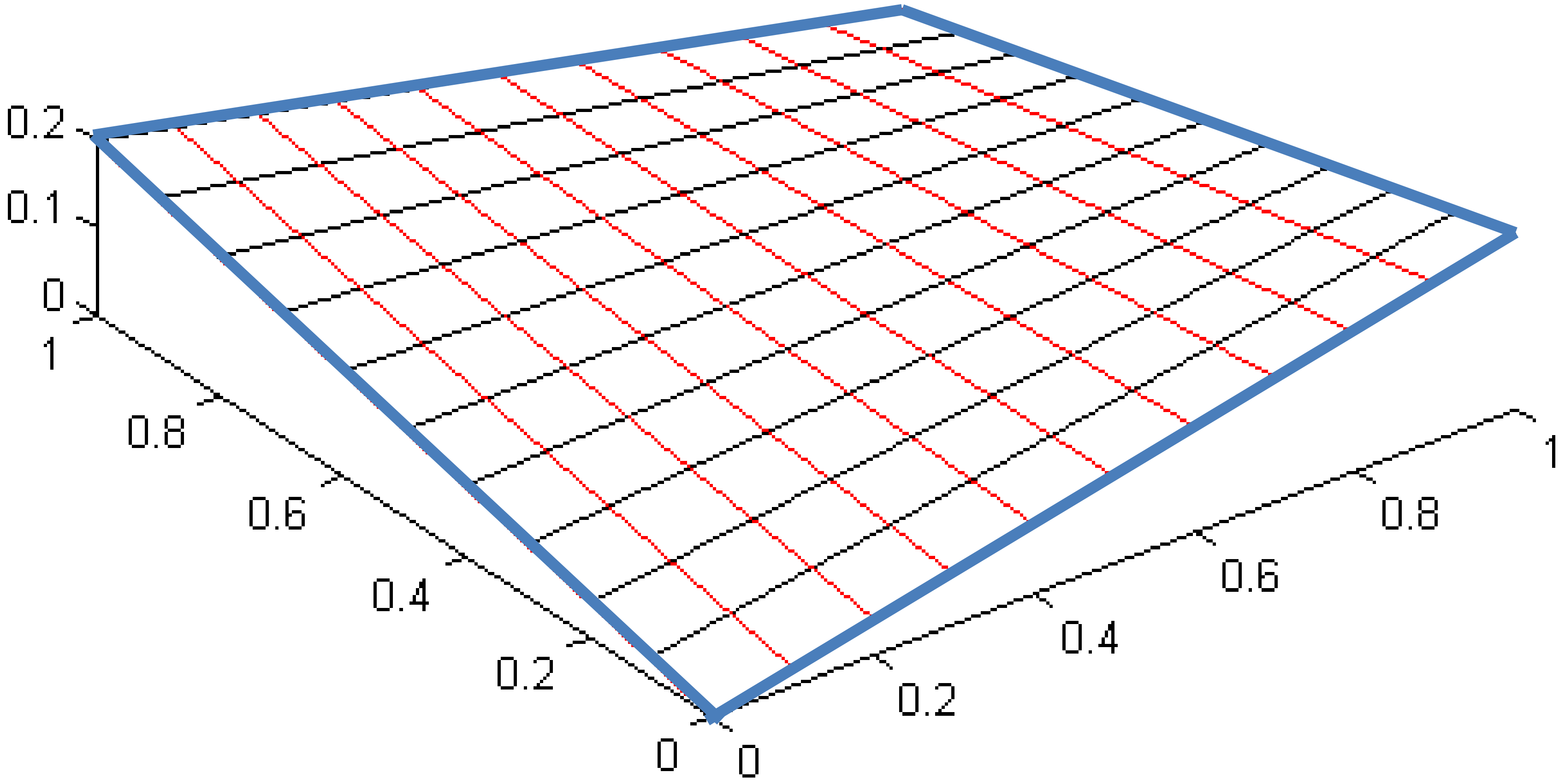
A particularly well accepted application of Coons patches is the so-called "lofting“ technique. It consists in generating a surface by linearly evolving a profile from an initial shape (polyline in red on the left) to another shape (arc of circle, in red on the right).

This has been a very widespread technique in defining aircraft wing profiles.

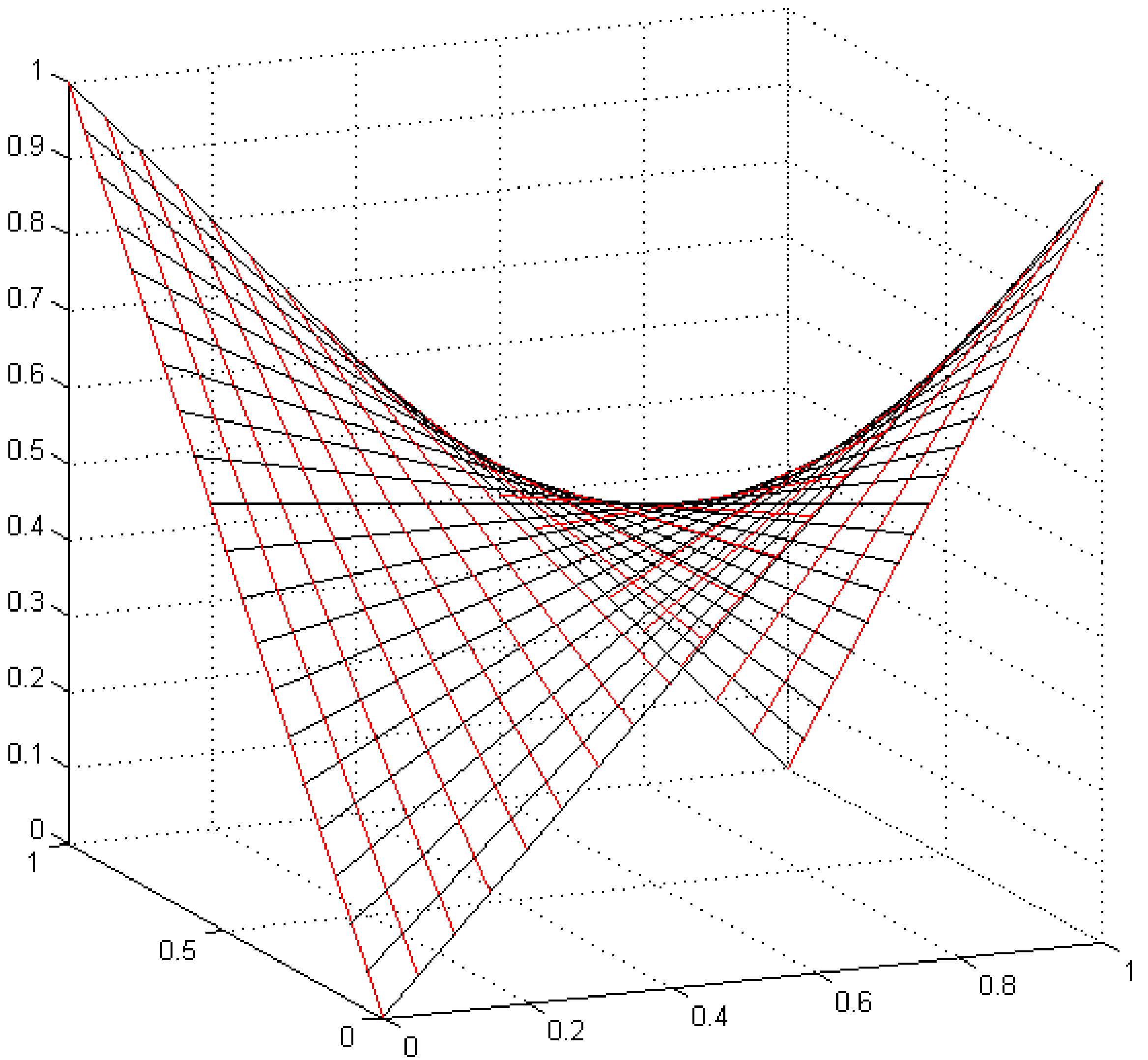
Principle of generating a patch (parameters *u* and *w*).
The patch is supported on two networks of curves.
When the curves are reduced to straight lines, we get the original **Coons patch**.



Coons patch: the four sides are straight lines. In this example, starting with a quadrilateral, which is a polygon (hence a planar object), we simply added a twist to transform it into a hyperbolic paraboloid.

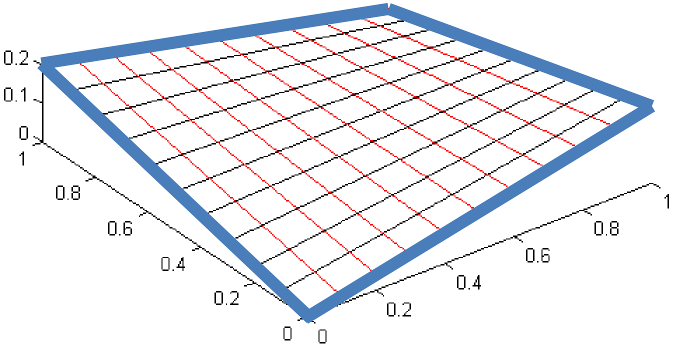


The representation of a Coons patch is completed very easily by drawing the network of lines based on the two pairs of opposite sides.



Isoparametric element

Coons patch



$$[Q] = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & \ldots & \ldots \\ x_3 & \ldots & \ldots \\ x_4 & \ldots & z_4 \end{bmatrix}$$

$[Q]$ is a matrix with 4 rows and 3 columns containing the Cartesian coordinates of the 4 points or nodes defining the patch. The blending functions of the 4 vertices are some functions equal to 1 at one of the vertices and 0 at the others:

$$[F] = [f_1 \quad f_2 \quad f_3 \quad f_4] = [(1-s)(1-t) \quad s(1-t) \quad st \quad (1-s)t]$$

The points of the patch are defined by their two parametric coordinates s and *t*.

$$P^T = [F][Q]$$

The set of functions f_i represents a barycentric combination fulfilling the partition of unity condition.

$$f_1 + f_2 + f_3 + f_4 = 1$$

More explicitly, the transpose of the point vector is a line matrix which is a function of two parameters

$$P(s,t)^T = \begin{bmatrix} x(s,t) & y(s,t) & z(s,t) \end{bmatrix} = [F]Q$$

To simplify the subsequent development, we limit ourselves to **two dimensions** by modeling patches and fields in the plane. We rewrite the nodes definition in 2D:

$$[Q] = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{bmatrix} = \begin{bmatrix} X & Y \end{bmatrix}$$

$$\begin{bmatrix} x(s,t) & y(s,t) \end{bmatrix} = [F][Q] = \begin{bmatrix} (1-s)(1-t) & s(1-t) & st & (1-s)t \end{bmatrix}[Q]$$

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric or intrinsic coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

In the following developments we show how to realize the coordinates transformations from cartesian to parametric coordinates

We first establish the relations between Cartesian and parametric coordinates.

Cartesian space

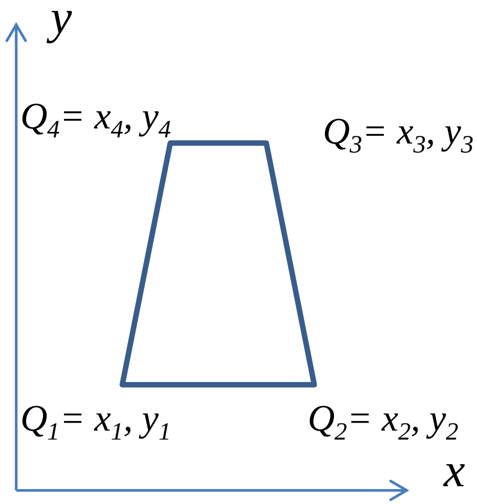
x, y

$dS = dx\,dy$

Parametric space

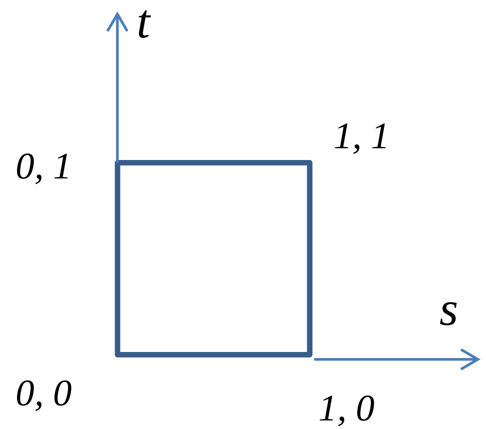
s, t

$dS = ???$



$$[X] = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

$$[Y] = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$



$$x(s,t) = [F][X] = \left[(1-s)(1-t) \quad s(1-t) \quad st \quad (1-s)t\right][X]$$

$$y(s,t) = [F][Y] = \left[(1-s)(1-t) \quad s(1-t) \quad st \quad (1-s)t\right][Y]$$

$[J]$ is the Jacobian matrix. For the bilinear element it is equal to:

$$[J] = \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} -(1-t) & (1-t) & t & -t \end{bmatrix}[X] & \begin{bmatrix} -(1-t) & (1-t) & t & -t \end{bmatrix}[Y] \\ \begin{bmatrix} -(1-s) & -s & s & (1-s) \end{bmatrix}[X] & \begin{bmatrix} -(1-s) & -s & s & (1-s) \end{bmatrix}[Y] \end{bmatrix}$$

We formulate how to transform an integration in Cartesian space into the equivalent one in parametric space.

For this purpose we use the Jacobian J which is the determinant of the Jacobian matrix $[J]$.

$$J(s,t) = \det\left(\left[J(s,t)\right]\right) = \frac{\partial x}{\partial s}\frac{\partial y}{\partial t} - \frac{\partial x}{\partial t}\frac{\partial y}{\partial s}$$

$$\iint_{\Omega_{Coons}} dxdy = \int_0^1 \int_0^1 J(s,t)\; dsdt$$

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

Gradient of a scalar function, for instance the temperature $\boldsymbol{\tau}(s, t)$.

$$\begin{bmatrix} \dfrac{\partial \tau}{\partial s} \\ \dfrac{\partial \tau}{\partial t} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial x}{\partial s} & \dfrac{\partial y}{\partial s} \\ \dfrac{\partial x}{\partial t} & \dfrac{\partial y}{\partial t} \end{bmatrix} \begin{bmatrix} \dfrac{\partial \tau}{\partial x} \\ \dfrac{\partial \tau}{\partial y} \end{bmatrix} = [J] \begin{bmatrix} \dfrac{\partial \tau}{\partial x} \\ \dfrac{\partial \tau}{\partial y} \end{bmatrix}$$

The gradients are easily computed in parametric coordinates, but we need them in Cartesian ones (the real world).

$$\begin{bmatrix} \dfrac{\partial \tau}{\partial x} \\ \dfrac{\partial \tau}{\partial y} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \dfrac{\partial \tau}{\partial s} \\ \dfrac{\partial \tau}{\partial t} \end{bmatrix}$$

The inverse of the Jacobian matrix $[J]$ has to be known everywhere in the integration domain.

The temperature field is expressed in parametric coordinates as a function of the nodal temperatures:

$$\tau = \left[(1-s)(1-t) \quad s(1-t) \quad st \quad (1-s)t \right][T] = [F][T]$$

It is easy to compute the gradient of τ in parametric coordinates:

$$\begin{bmatrix} \dfrac{\partial \tau}{\partial s} \\ \dfrac{\partial \tau}{\partial t} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial F}{\partial s} \\ \dfrac{\partial F}{\partial t} \end{bmatrix} [T] = \begin{bmatrix} -(1-t) & (1-t) & t & -t \\ -(1-s) & -s & s & (1-s) \end{bmatrix} [T]$$

However, the conductivity matrix uses the gradient of the temperature expressed in Cartesian coordinates, So, we have to express it in terms of parametric coordinates than to the invers of the Jacobian.

$$\begin{bmatrix} \dfrac{\partial \tau}{\partial x} \\ \dfrac{\partial \tau}{\partial y} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \dfrac{\partial \tau}{\partial s} \\ \dfrac{\partial \tau}{\partial t} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \dfrac{\partial F}{\partial s} \\ \dfrac{\partial F}{\partial t} \end{bmatrix} [T] = [J]^{-1} \begin{bmatrix} -(1-t) & (1-t) & t & -t \\ -(1-s) & -s & s & (1-s) \end{bmatrix} [T]$$

In the isoparametric elements the blending functions of the geometry: *x* and *y* are the same as that of the temperature field τ:

$$\begin{aligned} \tau(s,t) &= [F][T] \\ x(s,t) &= [F][X] \\ y(s,t) &= [F][Y] \end{aligned}$$

[T] = nodal temperatures,
[X] = nodal x coordinates
[Y] = nodal y coordinates

If the patch is defined in the **3D space** to represent a surface geometry, the definitions seen above have to be slightly modified:

$$[Q]=\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix}=\begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & \dots & \dots \\ x_3 & \dots & \dots \\ x_4 & \dots & z_4 \end{bmatrix}$$

$$P(s,t)^T=\left[x(s,t) \quad y(s,t) \quad z(s,t)\right]=[F]Q$$

Jacobian matrix:

$$[J]=\left[\frac{\partial P}{\partial s}\times\frac{\partial P}{\partial t}\right]=\left[\frac{\partial y}{\partial s}\frac{\partial z}{\partial t}-\frac{\partial y}{\partial t}\frac{\partial z}{\partial s} \quad \frac{\partial z}{\partial s}\frac{\partial x}{\partial t}-\frac{\partial x}{\partial s}\frac{\partial z}{\partial t} \quad \frac{\partial x}{\partial s}\frac{\partial y}{\partial t}-\frac{\partial x}{\partial t}\frac{\partial y}{\partial s}\right]$$

Determinant of the Jacobian:

$$\det\left(\left[J(s,t)\right]\right)=\sqrt{\left(\frac{\partial y}{\partial s}\frac{\partial z}{\partial t}-\frac{\partial y}{\partial t}\frac{\partial z}{\partial s}\right)^2+\left(\frac{\partial z}{\partial s}\frac{\partial x}{\partial t}-\frac{\partial x}{\partial s}\frac{\partial z}{\partial t}\right)^2+\left(\frac{\partial x}{\partial s}\frac{\partial y}{\partial t}-\frac{\partial x}{\partial t}\frac{\partial y}{\partial s}\right)^2}$$

Same result as in slide 14: $\det\left(\left[J(s,t)\right]\right)=\frac{\partial x}{\partial s}\frac{\partial y}{\partial t}-\frac{\partial x}{\partial t}\frac{\partial y}{\partial s}$

Integral on the patch of a function *g* depending of the two parameters *s* and *t* :

$$\iint\limits_{\Omega_{Coons}} g^*(x,y)\ dxdy = \int_0^1\int_0^1 g\left(s,t\right) J\left(s,t\right)\ dsdt$$

In particular situations it is possible to express g(s, t) in cartesian coordinates

g (s; t) → g*(x, y)

The above integral must often be computed numerically, because its Jacobian cannot easily be calculated analytically.

Integration to perform for obtaining the conductivity matrix:

$$\iint_{\Omega_{Coons}} \begin{bmatrix} \frac{\partial \tau}{\partial x} \\ \frac{\partial \tau}{\partial y} \end{bmatrix}^T \begin{bmatrix} \frac{\partial \tau}{\partial x} \\ \frac{\partial \tau}{\partial y} \end{bmatrix} dxdy$$

$$\begin{bmatrix} \frac{\partial \tau}{\partial x} \\ \frac{\partial \tau}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix}^{-1} \begin{bmatrix} \begin{bmatrix} -(1-t) & (1-t) & t & -t \end{bmatrix} \\ \begin{bmatrix} -(1-s) & -s & s & (1-s) \end{bmatrix} \end{bmatrix} [T]$$

Computation in parametric coordinates:

$$\int_0^1 \int_0^1 \left\{ \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix}^{-1} \begin{bmatrix} \begin{bmatrix} -(1-t) & (1-t) & t & -t \end{bmatrix} \\ \begin{bmatrix} -(1-s) & -s & s & (1-s) \end{bmatrix} \end{bmatrix} \right\}^T \left\{ \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix}^{-1} \begin{bmatrix} \begin{bmatrix} -(1-t) & (1-t) & t & -t \end{bmatrix} \\ \begin{bmatrix} -(1-s) & -s & s & (1-s) \end{bmatrix} \end{bmatrix} \right\} J(s,t)\ dsdt$$

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

Numerical integration on the Coons patch

$$\iint_{Coons} f(x,y)dxdy = \sum f(x_i, y_i)w_i$$

This technique consists in evaluating the integral *f* (*x*, *y*) in a certain number of points. It is then sufficient to calculate the sum of its weighted values by *a priori* assigning a weight w_i to each of the evaluation points. For a parametric function, we must add the Jacobian transformation.

$$\iint_{Coons} g^*(x,y)dxdy = \int_0^1\int_0^1 g(s,t)\,J(s,t)\,dsdt = \sum_i g(s_i,t_i)w_i J(s_i,t_i)$$

1D Gauss quadrature		
Number of points	*Positions* x_i *in the interval* [0 1]	*Weight* w_i
1	0.5	1
2	$0.5 \pm \sqrt{3}/6$	.5
3	0.5 , $0.5 \pm \sqrt{3/20}$	4/9, 5/18, 5/18
4	$0.5 \pm 1/70\sqrt{525 - 70\sqrt{30}}$ $0.5 \pm 1/70\sqrt{525 + 70\sqrt{30}}$	$1/4 + \sqrt{30}/72$; $1/4 + \sqrt{30}/72$ $1/4 - \sqrt{30}/72$; $1/4 - \sqrt{30}/72$

Gauss quadrature on a 1m x 1m square

Evaluated function		**Exact Solution**	Number of Gauss points			
			1	4 = 2 x 2	9 = 3 x 3	16 = 4 x 4
1.	$\sin(\pi s)$	**0.6366**	1.0	0.6162	0.6371	**0.6366**
2.	s^3	**0.25**	0.125	**0.25**	**0.25**	**0.25**
3.	s^5	**0.1667**	0.0313	0.1528	**0.1667**	**0.1667**
4.	s^7	**0.1250**	0.0078	0.0949	0.1238	**0.1250**

The above results confirm that n integration points are necessary to obtain an exact solution for polynomials of degree $(2n-1)$.

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conductivity matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

Numerical quadrature of the conductivity matrix

Matlab© function *Kelu.m* for the Gauss quadrature.

(the Matlab© instructions reproduce the formula that have just been presented.)

Matlab© function *Kelu.m* for isoparamatric evaluation of the conductivity matrix

```
function [K] = Kelu(xyz,lo)
Q  = [xyz(lo(1),1:3); xyz(lo(2),1:3); xyz(lo(3),1:3); xyz(lo(4),1:3)];
s  = [.5-sqrt(3)/6 .5+sqrt(3)/6 .5+sqrt(3)/6 .5-sqrt(3)/6];  % 4 Gauss pts
t  = [.5-sqrt(3)/6 .5-sqrt(3)/6 .5+sqrt(3)/6 .5+sqrt(3)/6];  % 4 Gauss pts
K  = zeros(4,4);area=0.;
for i=1:4                                         % Loop on the 4 Gauss points
    fs   = [-(1-t(i)) (1-t(i)) t(i) -t(i)   ];              % Derivative s
    ft   = [-(1-s(i)) -s(i)    s(i) (1-s(i))];              % Derivative t
    gra  = [fs;ft];                 % Gradient of the scalar bilinear function
    ds   = fs * Q;
    dt   = ft * Q;
    area = area         + sqrt(dot(cross(ds,dt),cross(ds,dt)))/4;
    J    = [fs*Q(:,1) ft*Q(:,1);fs*Q(:,2) ft*Q(:,2)];
    K=K+((J^(-1)*gra)'*J^(-1)*gra)*sqrt(dot(cross(ds,dt),cross(ds,dt)))/4;
end                                  % disp(['Patch area : ',num2str(area)])
end
```

This Matlab function allows computing the conductivity matrix of an isoparametric quadrilateral with a bilinear temperature field.

To obtain the conductivity matrix, this result has to be multiplied by the conductivity coefficient and the thickness.

Matlab© function *Kelu.m* for isoparamatric evaluation of the conductivity matrix

```
 1  function [K] = Kelu(xyz,lo)
 2  Q  = [xyz(lo(1),1:3); xyz(lo(2),1:3); xyz(lo(3),1:3); xyz(lo(4),1:3)];
 3  s  = [.5-sqrt(3)/6 .5+sqrt(3)/6 .5+sqrt(3)/6 .5-sqrt(3)/6];  % 4 Gauss pts
 4  t  = [.5-sqrt(3)/6 .5-sqrt(3)/6 .5+sqrt(3)/6 .5+sqrt(3)/6];  % 4 Gauss pts
 5  K  = zeros(4,4);area=0.;
 6  for i=1:4                                   % Loop on the 4 Gauss points
 7      fs   = [-(1-t(i)) (1-t(i)) t(i) -t(i)   ];              % Derivative s
 8      ft   = [-(1-s(i)) -s(i)    s(i) (1-s(i))];              % Derivative t
 9      gra  = [fs;ft];                 % Gradient of the scalar bilinear function
10      ds   = fs * Q;
11      dt   = ft * Q;
12      area = area         + sqrt(dot(cross(ds,dt),cross(ds,dt)))/4;
13      J    = [fs*Q(:,1) ft*Q(:,1);fs*Q(:,2) ft*Q(:,2)];
14      K=K+((J^(-1)*gra)'*J^(-1)*gra)*sqrt(dot(cross(ds,dt),cross(ds,dt)))/4;
15  end                                     % disp(['Patch area : ',num2str(area)])
16  end
```

Line 2 : Definition of the 4 vertices of the patch: the coordinates are stored in the *xyz* matrix of nodal coordinates and the vector *lo* is a pointer of positions in the *xyz* matrix. For instance, it is possible to define both in Matlab© notations:

```
xyz  = [0 0 0;1 0 0;1 1 0;0 1 0];lo=[1 2 3 4];
```

Lines 3 & 4 : Define the sequence of positions of Gauss points in a unit square according to the table of 1D Gauss quadrature (slide 24)

Line 5 : Initialization of the conductivity matrix.

Lines 7 to 15 : Loop on the 2 x 2 = 4 Gauss points. The weights $w_i = 1/4$

Matlab© function *Kelu.m* for isoparamatric evaluation of the conductivity matrix

```
 1  function [K] = Kelu(xyz,lo)
 2  Q  = [xyz(lo(1),1:3); xyz(lo(2),1:3); xyz(lo(3),1:3); xyz(lo(4),1:3)];
 3  s  = [.5-sqrt(3)/6 .5+sqrt(3)/6 .5+sqrt(3)/6 .5-sqrt(3)/6];  % 4 Gauss pts
 4  t  = [.5-sqrt(3)/6 .5-sqrt(3)/6 .5+sqrt(3)/6 .5+sqrt(3)/6];  % 4 Gauss pts
 5  K  = zeros(4,4);area=0.;
 6  for i=1:4                                              % Loop on the 4 Gauss points
 7      fs   = [-(1-t(i)) (1-t(i)) t(i) -t(i)   ];                    % Derivative s
 8      ft   = [-(1-s(i)) -s(i)    s(i) (1-s(i))];                    % Derivative t
 9      gra  = [fs;ft];                      % Gradient of the scalar bilinear function
10      ds   = fs * Q;
11      dt   = ft * Q;
12      area = area         + sqrt(dot(cross(ds,dt),cross(ds,dt)))/4;
13      J    = [fs*Q(:,1) ft*Q(:,1);fs*Q(:,2) ft*Q(:,2)];
14      K=K+((J^(-1)*gra)'*J^(-1)*gra)*sqrt(dot(cross(ds,dt),cross(ds,dt)))/4;
15  end                                         % disp(['Patch area : ',num2str(area)])
16  end
```

Lines 7 & 8 : Computation of the derivatives of the blending functions (slide 14)

Line 9 : Define the gradient in parametric coordinates

Lines 10 & 11 : Compute the volume differentials in parametric coordinates (slide 14)

Lines 12 : Determinant of the Jacobian which will enable to compute the area of the patch (slides 14 or 19)

Lines 13 : Jacobian matrix (slides 14 or 19)

Lines 14 : Compute the conductivity matrix (slide 21)

The following tests corroborate the previous analytical results.

Unit square

```
xyz  = [0 0 0;1 0 0;1 1 0;0 1 0];lo=[1 2 3 4];[K]=Kelu(xyz,lo)*6
Patch area : 1, K =
                                4    -1    -2    -1
                               -1     4    -1    -2
                               -2    -1     4    -1
                               -1    -2    -1     4
```

Other size

```
xyz  = [0 0 0;4 0 0;4 4 0;0 4 0];lo=[1 2 3 4];[K]=Kelu(xyz,lo)*6
Patch area : 16, K =
                                4    -1    -2    -1
                               -1     4    -1    -2
                               -2    -1     4    -1
                               -1    -2    -1     4
```

Other orientation

```
xyz  = [1 0 0;0 1 0;-1 0 0;0 -1 0];lo=[1 2 3 4];[K]=Kelu(xyz,lo)*6
Patch area : 2, K =
                                4    -1    -2    -1
                               -1     4    -1    -2
                               -2    -1     4    -1
                               -1    -2    -1     4
```

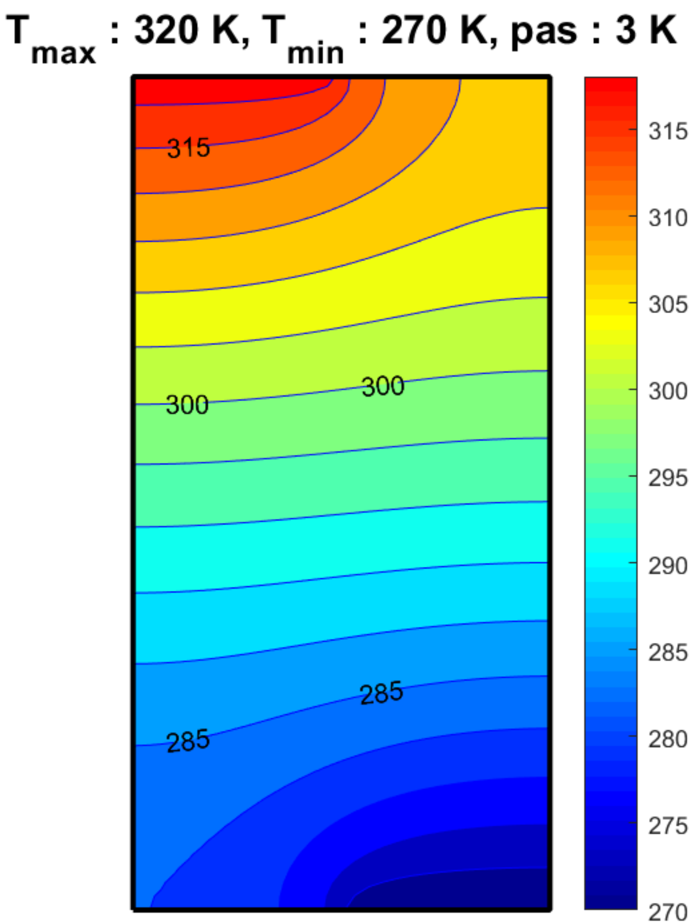
Rectangle

```
xyz  = [0 0 0;2 0 0;2 1 0;0 1 0];lo=[1 2 3 4];[K]=Kelu(xyz,lo)*6
Patch area : 2, K =
                         10.     2.    -5.    -7.
                          2.    10.    -7.    -5.
                         -5.    -7.    10.     2.
                         -7.    -5.     2.    10.
```

Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

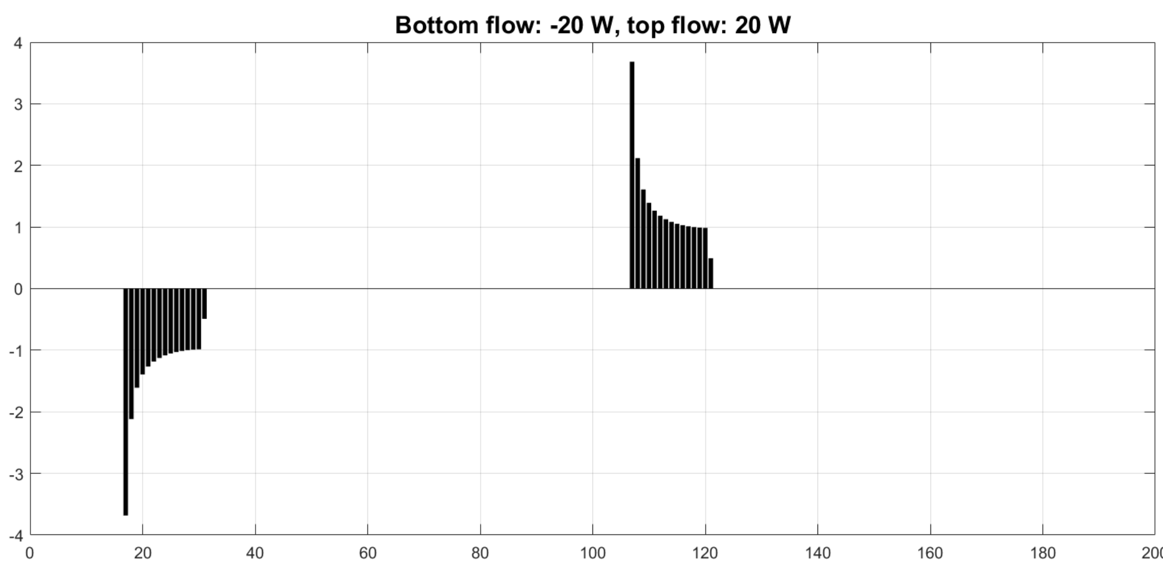
```
conde.m conductivity: 1 W/(mK)
Coordinates  P1, P2 : 0  0  1  0 m
Coordinates  P3, P4 : 1  2  0  2 m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 30 x 60
Fix. nod. 2 hor. fa.: 30
Diss tcaT*(K*tca)/2 : 500 WK
```



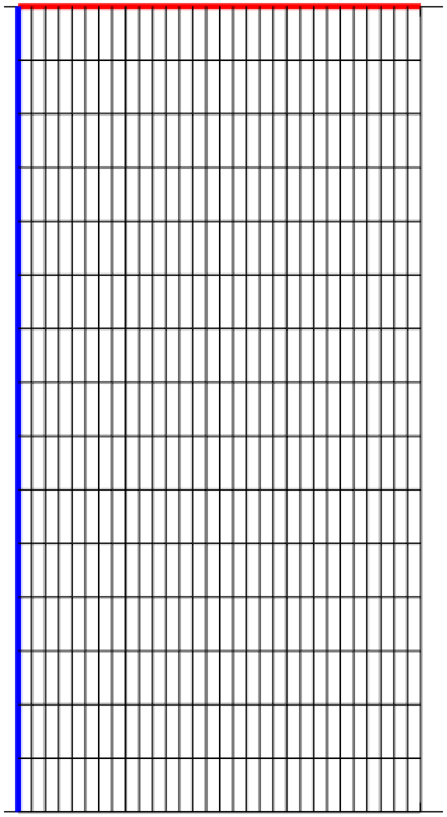
In the following example, temperatures are imposed on one segment of the upper face and on another of the lower base.

In this case 320 K on the upper face and 270 K on the base. The mesh is 30 x 60. The number of *DOF* is equal to 1891.

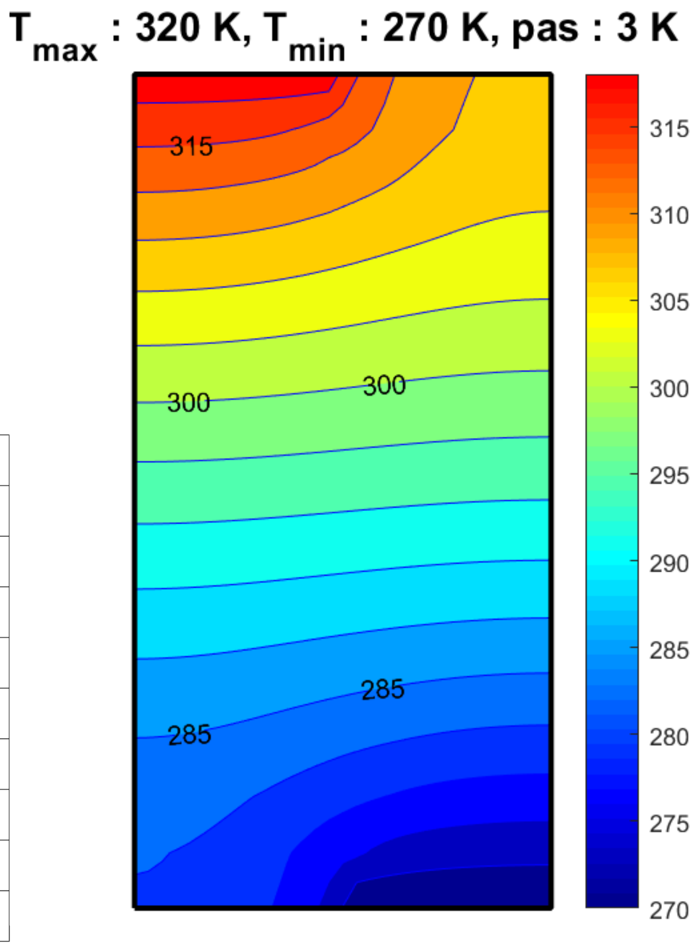
As expected, the isotherms are orthogonal to the walls, except in the two zones where the temperatures are imposed. In this example, the temperatures were imposed on 15 nodes of the upper and lower faces. The fluxes on both horizontal faces are ± 20. W; they are highly concentrated on the corners.

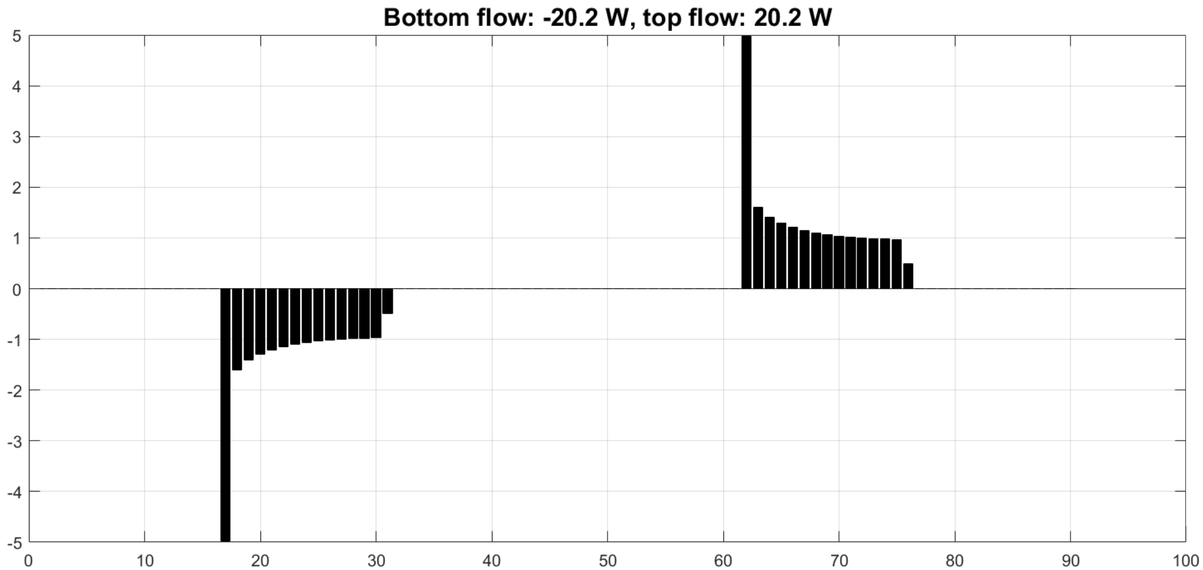


Decreasing significantly (coefficient 4) the number of elements in the *y* direction yields to aspect ratios of four for the elements but does not influence so much the solution.



```
conde.m conductivity: 1 W/(mK)
Coordinates  P1, P2 : 0  0  1  0 m
Coordinates  P3, P4 : 1  2  0  2 m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 30 x 15
Fix. nod. 2 hor. fa.: 30
Diss tcaT*(K*tca)/2 : 506 WK
```

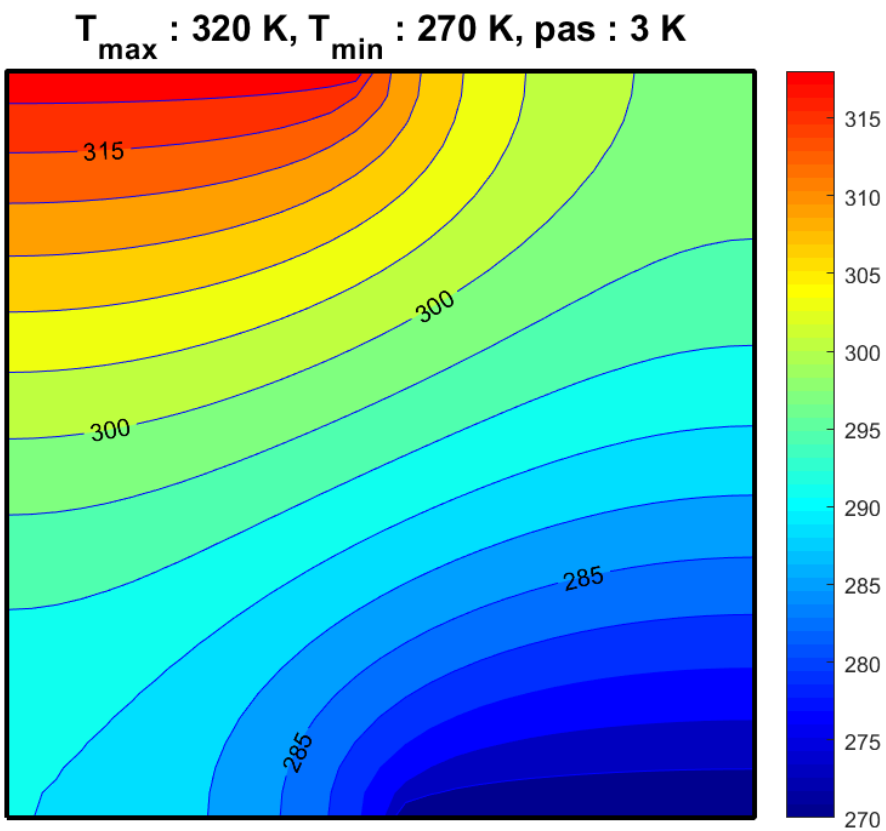


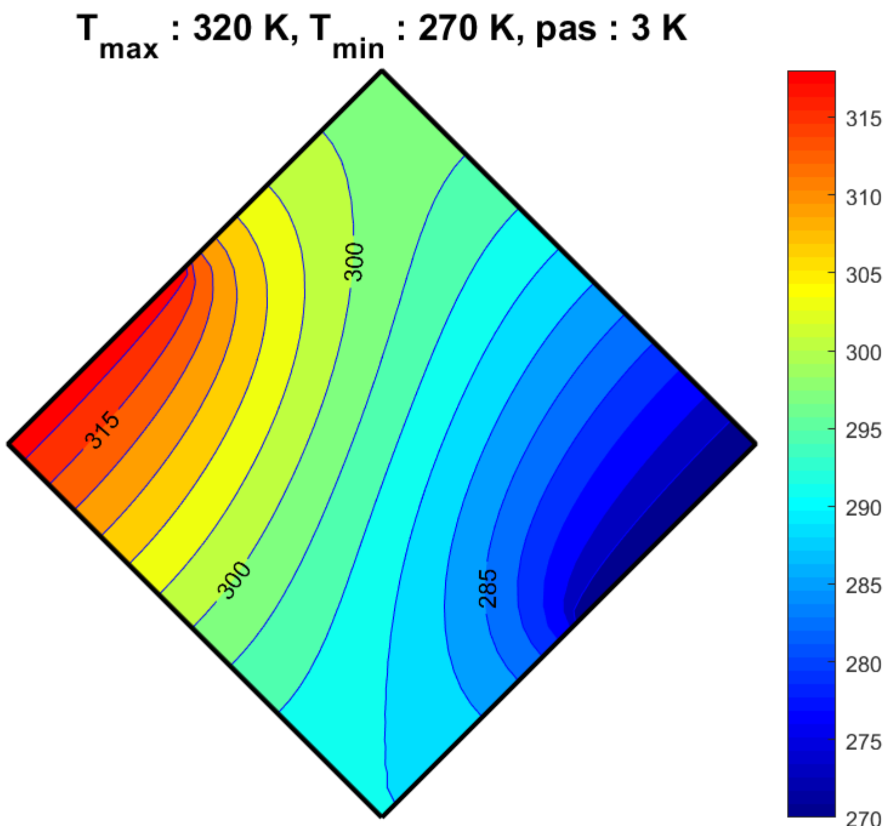


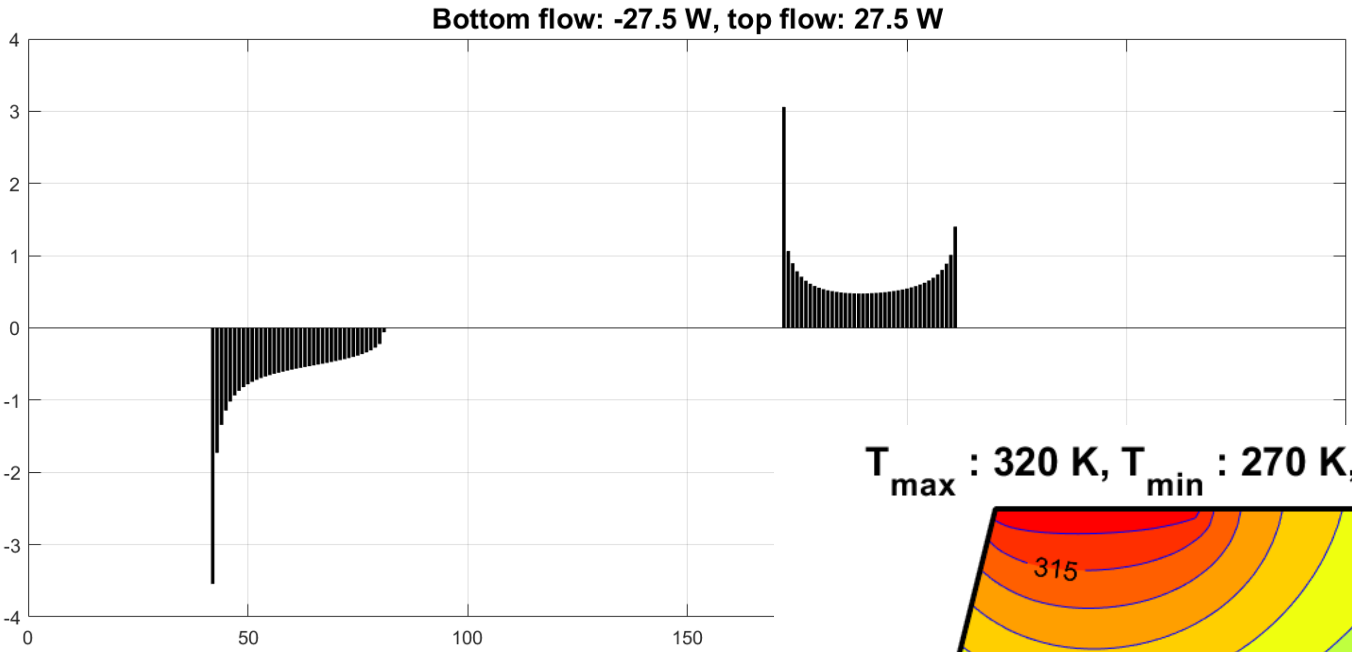
With the same procedure, we can easily handle other shapes and change their orientation. As expected, the orientation does not influence the result.

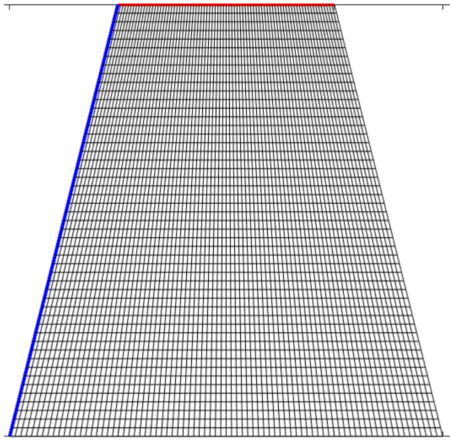
```
conde.m conductivity: 1 W/(mK)
Coordinates  P1, P2 : 0  0  2  0 m
Coordinates  P3, P4 : 2  2  0  2 m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 30 x 30
Fix. nod. 2 hor. fa.: 30
Diss tcaT*(K*tca)/2 : 817 WK
```

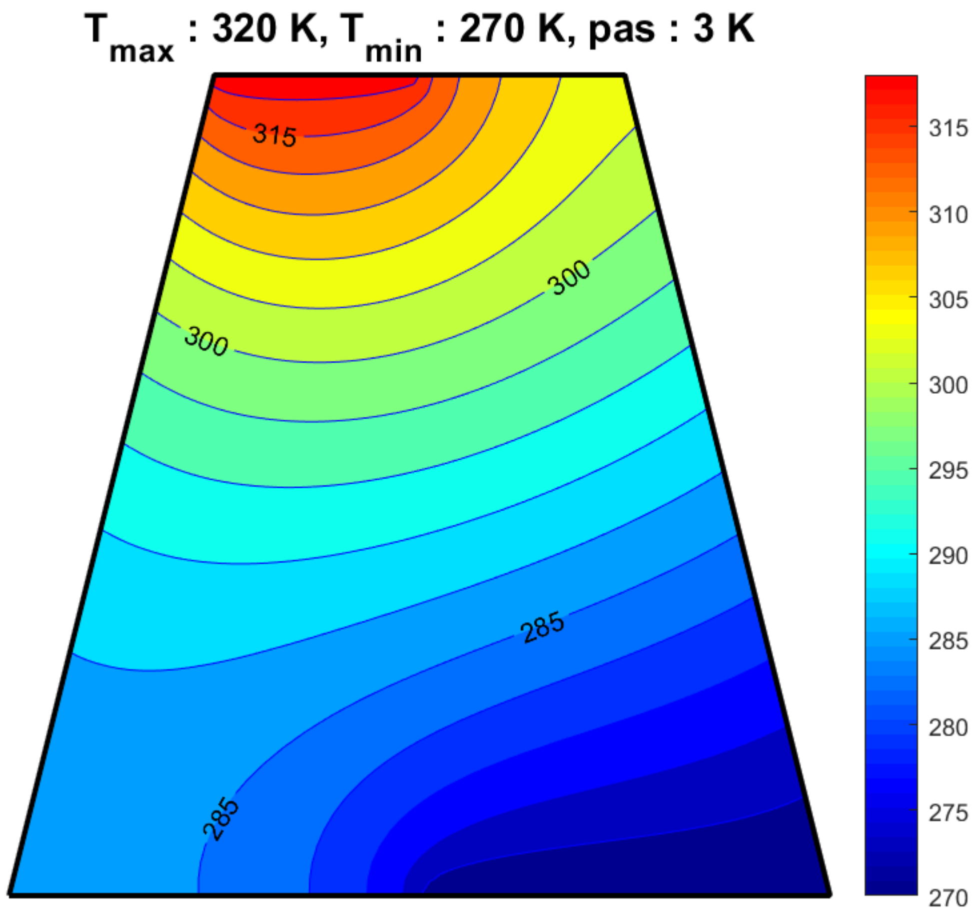
```
conde.m conductivity: 1 W/(mK)
Coordinates  P1, P2 : 0 -1  1  0 m
Coordinates  P3, P4 : 0  1 -1  0 m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 30 x 30
Fix. nod. 2 hor. fa.: 30
Diss tcaT*(K*tca)/2 : 817 WK
```







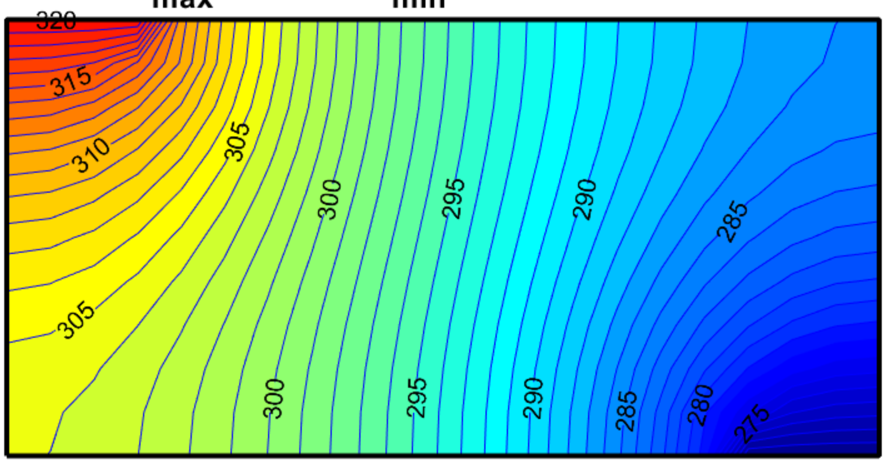




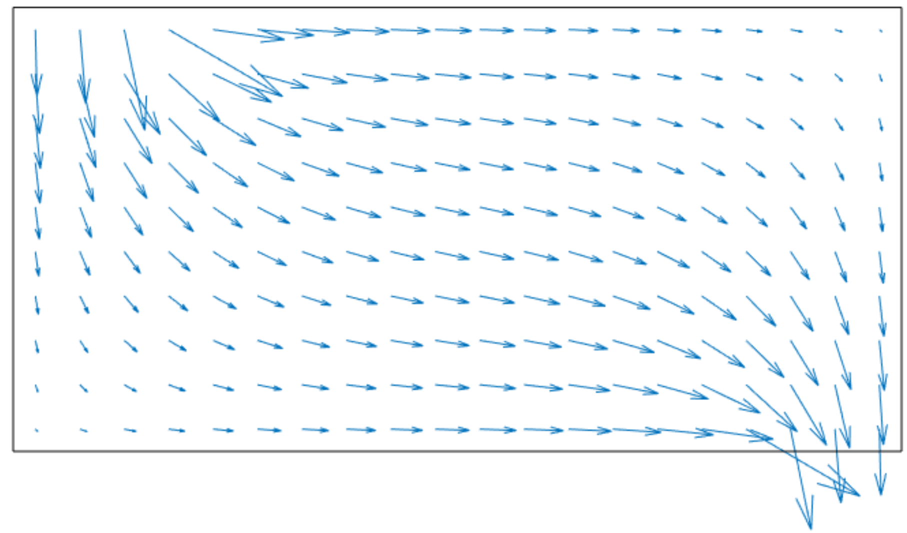

```
conde.m conductivity: 1 W/(mK)
Coordinates  P1, P2 : 0   0  2  0 m
Coordinates  P3, P4 : 1.5 2 0.5 2 m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 80 x 50
Fix. nod. 2 hor. fa.: 80
Diss tcaT*(K*tca)/2 : 687 WK
```

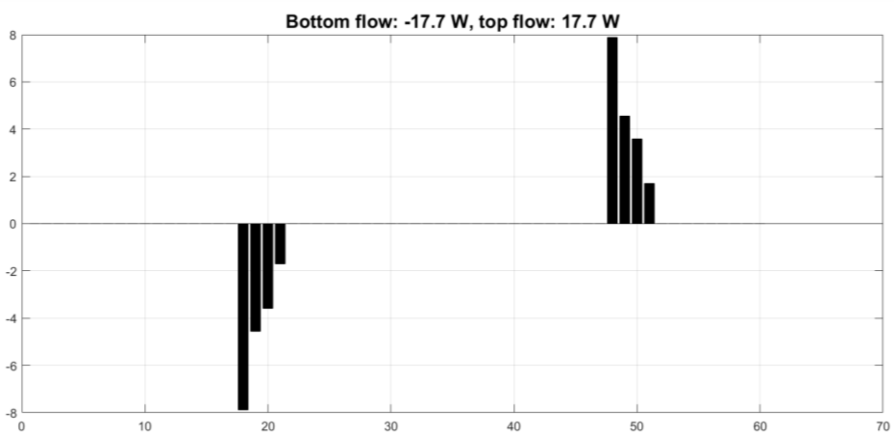
This last example shows the versatility of the isoparametric elements

T_{max} : 320 K, T_{min} : 270 K, pas : 1 K



- gradT, max: 33, mean: 9.3 K/m



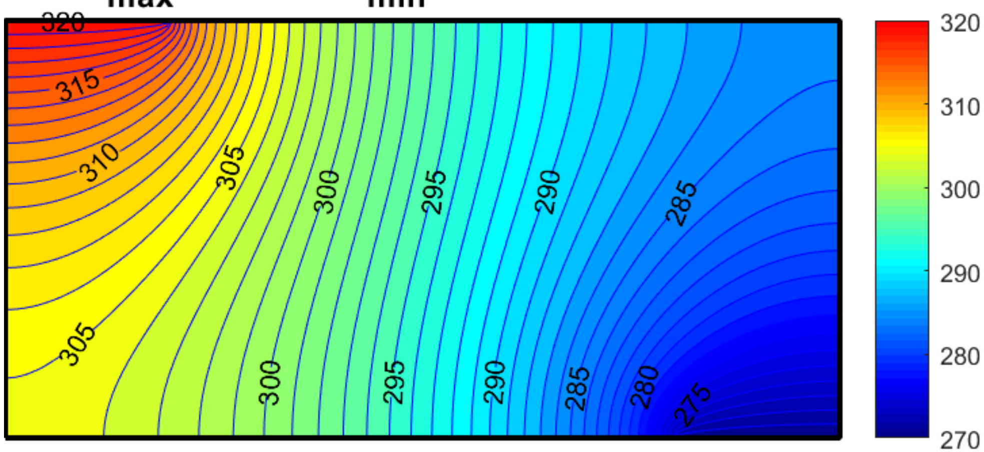


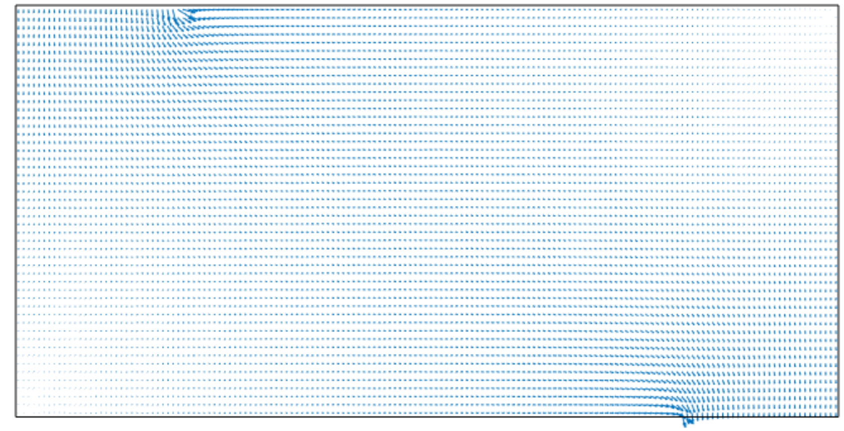

```
pp_Base_visopa_conduction
conde.m uniform k      : 1 W/(mK)
Coordinates  P1, P2 : 0 0 4 0 m
Coordinates  P3, P4 : 4 2 0 2 m
Numb. fixed nodes   : 4
- gradT, max        : 33.4266, mean: 9.33 K/m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 20 x 10
Fix. nod. 2 hor. fa.: 8
Diss tcaT*(K*tca)/2 : 442 WK
Fixed nodes hor. f. : 8
```

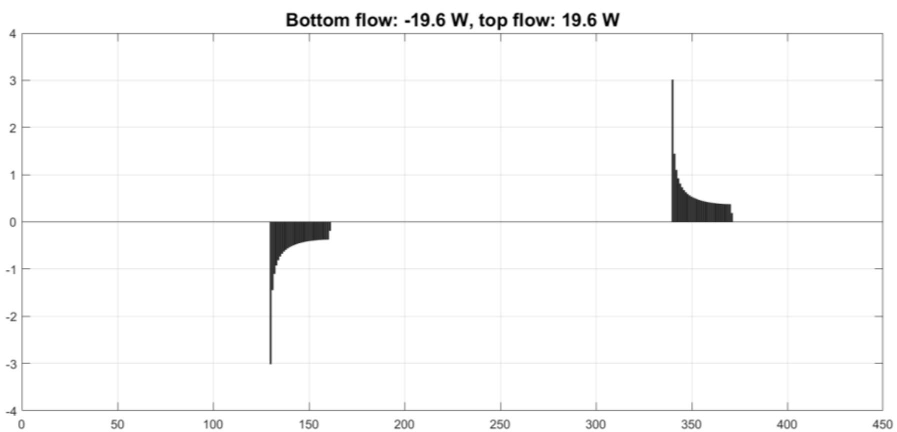
Here we consider the drawing of the heat flows shown by one arrow in each element. The arrows are oriented in the opposite direction of the gradient and their length is proportional to the intensity of the gradient (Matlab© function: *Hflos.m*).

- gradT, max: 86, mean: 9.9 K/m

T_{max} : 320 K, T_{min} : 270 K, pas : 1 K

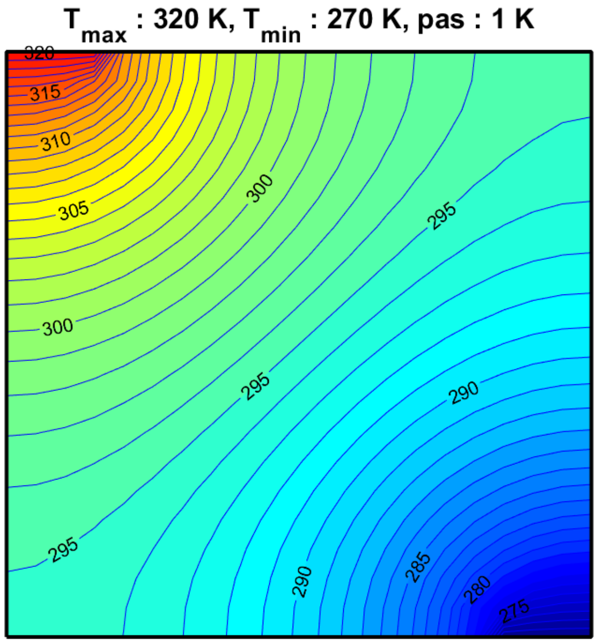


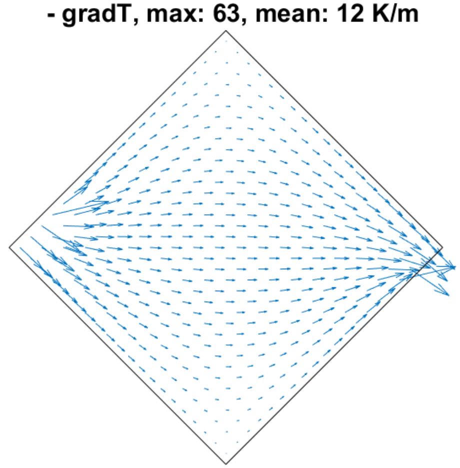


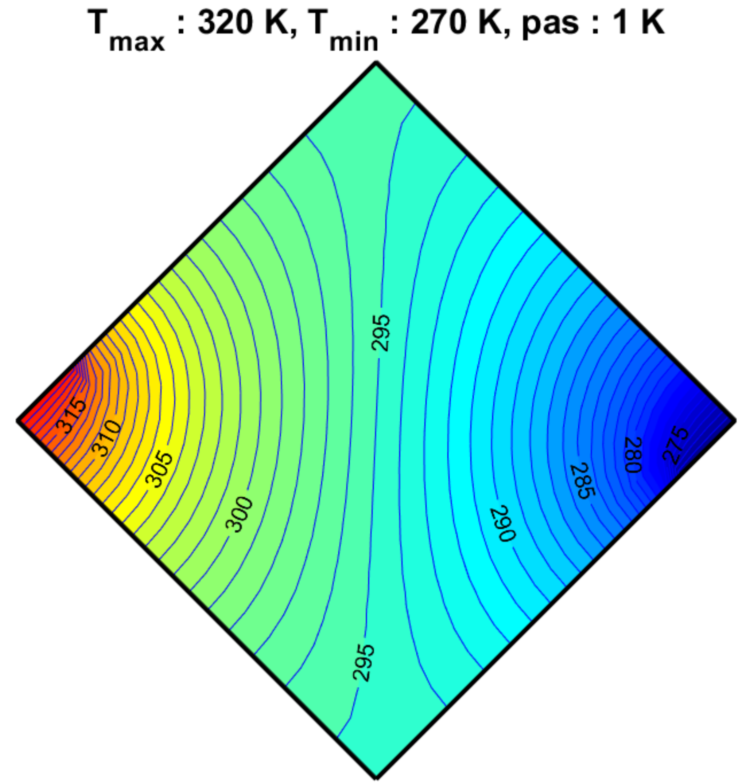


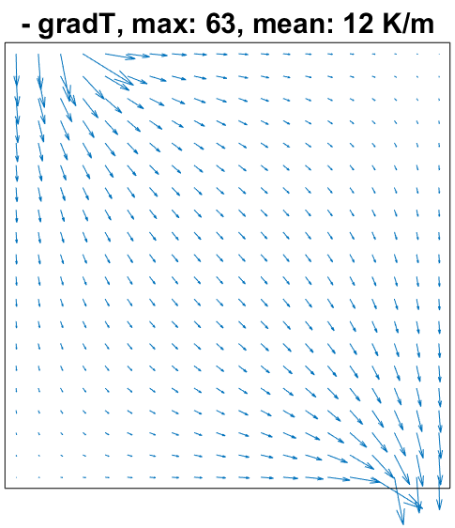

```
pp_Base_visopa_conduction
conde.m uniform k     : 1 W/(mK)
Coordinates  P1, P2 : 0 0 4 0 m
Coordinates  P3, P4 : 4 2 0 2 m
Numb. fixed nodes   : 32
- gradT, max        : 85.7574, mean: 9.9219 K/m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 160 x 50
Fix. nod. 2 hor. fa.: 64
Diss tcaT*(K*tca)/2 : 490 WK
Fixed nodes hor. f. : 64
```

Now, the mesh is very fine, the dissipation energy seems to converge to the value of 490 WK, the total flows on the top and bottom sides tends to 19.6 W and the mean temperature gradient is equal to 9.9 Km^{-1} .





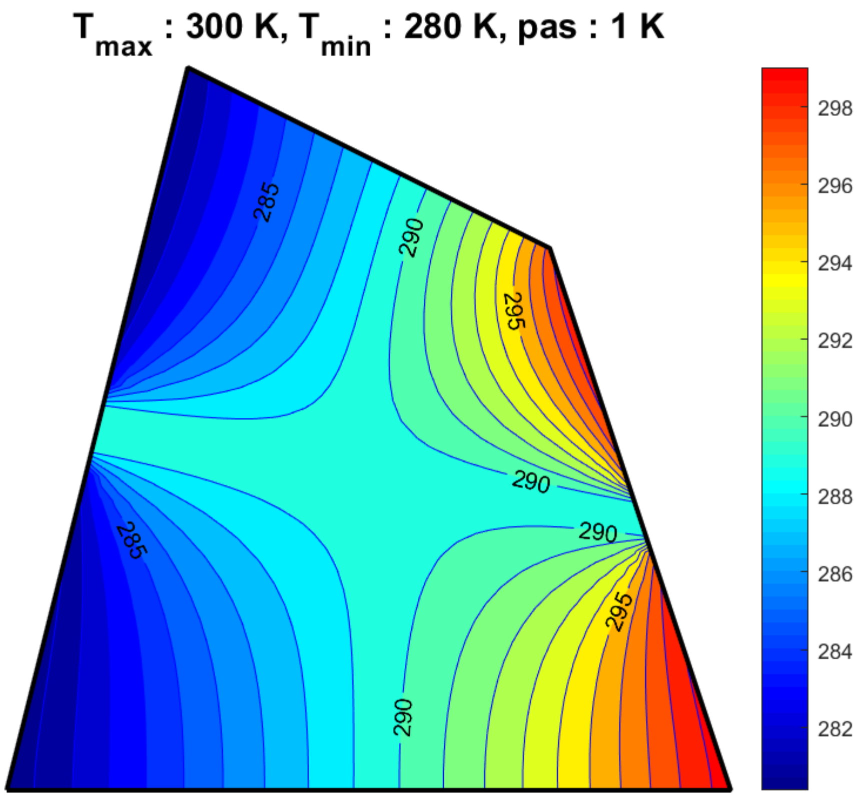




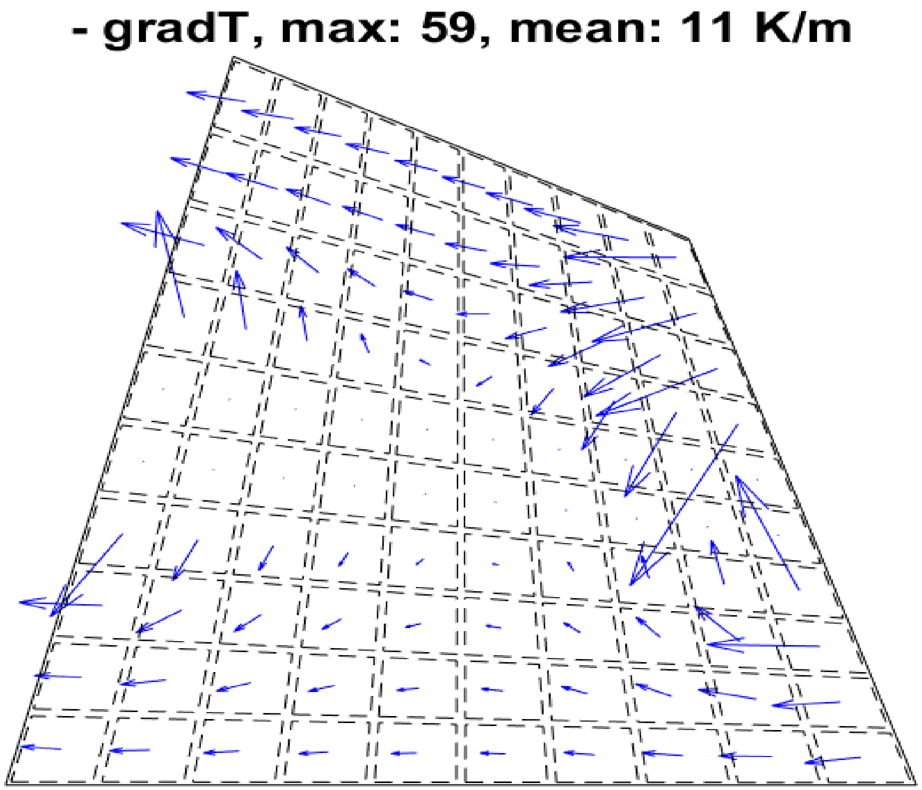

```
pp_Base_visopa_conduction
conde.m uniform k     : 1 W/(mK)
Coordinates  P1, P2 : 0 0 2 0 m
Coordinates  P3, P4 : 2 2 0 2 m
Numb. fixed nodes   : 4
- gradT, max        : 62.7574, mean: 12.1515 K/m
Base temperature    : 270 K
Top  temperature    : 320 K
Mesh size           : 20 x 20
Fix. nod. 2 hor. fa.: 8
Diss tcaT*(K*tca)/2 : 436 WK
Fixed nodes hor. f. : 8
```

The objective of this example is to show that the results are not sensitive to the orientation of the domain. The areas of both domains being the same, we obtain exactly the same results

The temperature field and its gradient



Visualization of a scalar field with isocurves showing the levels

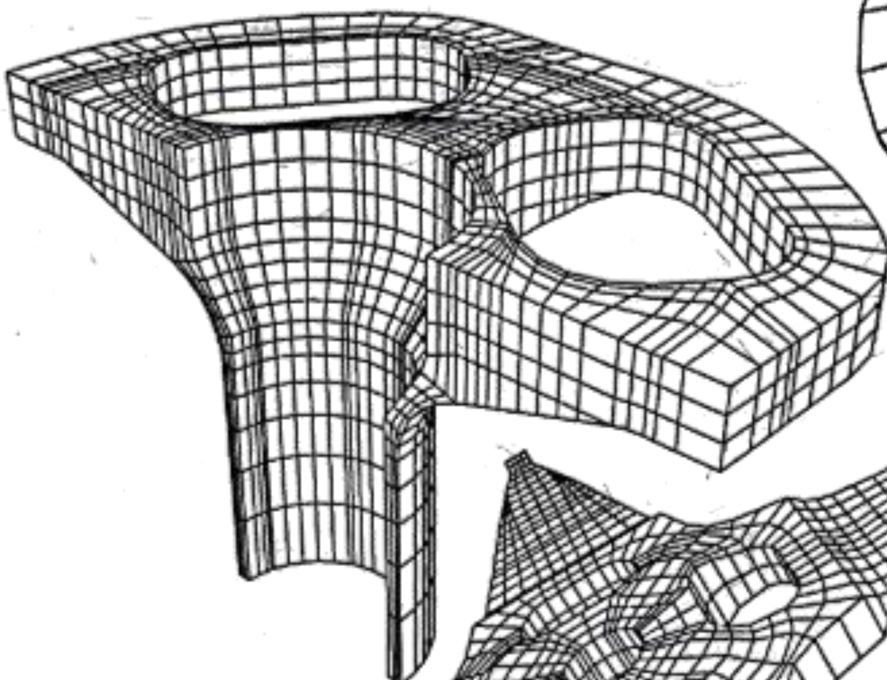


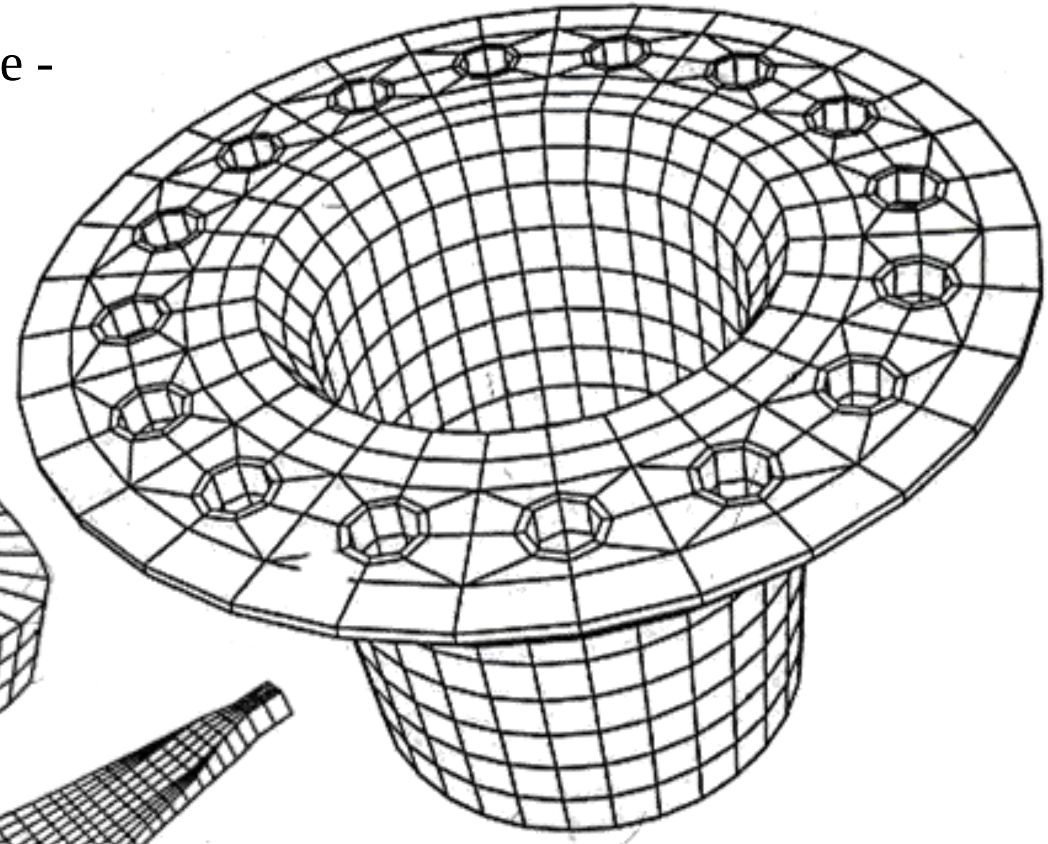
Visualization of a vector field with one arrow in each element

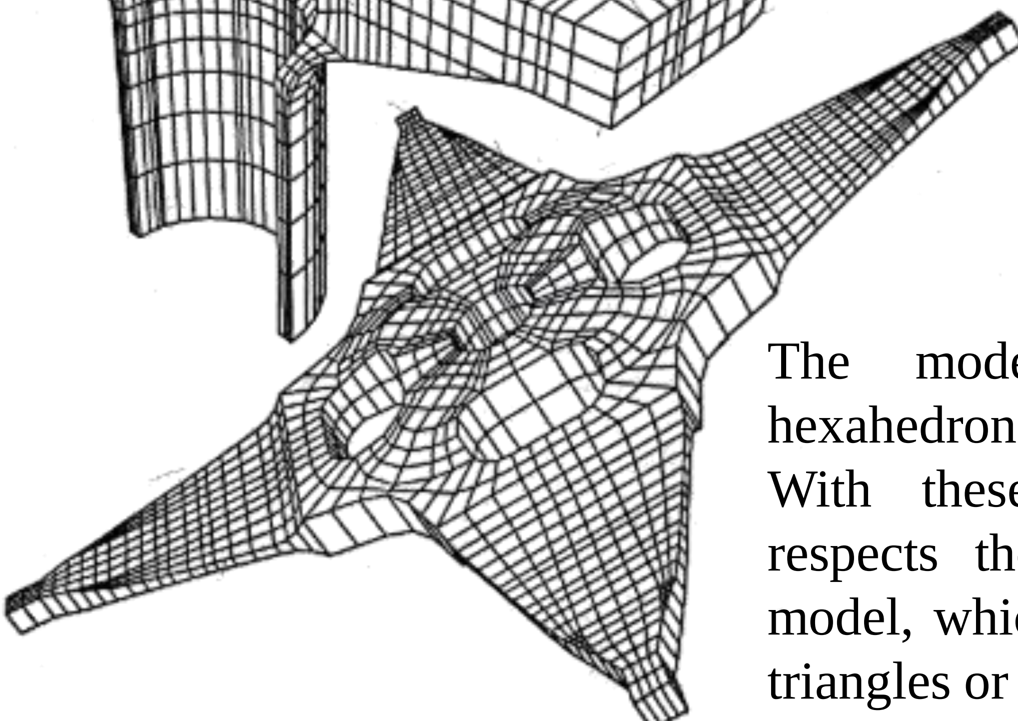
Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

Finite element model from Aérospatiale - Eurocopter Marignane





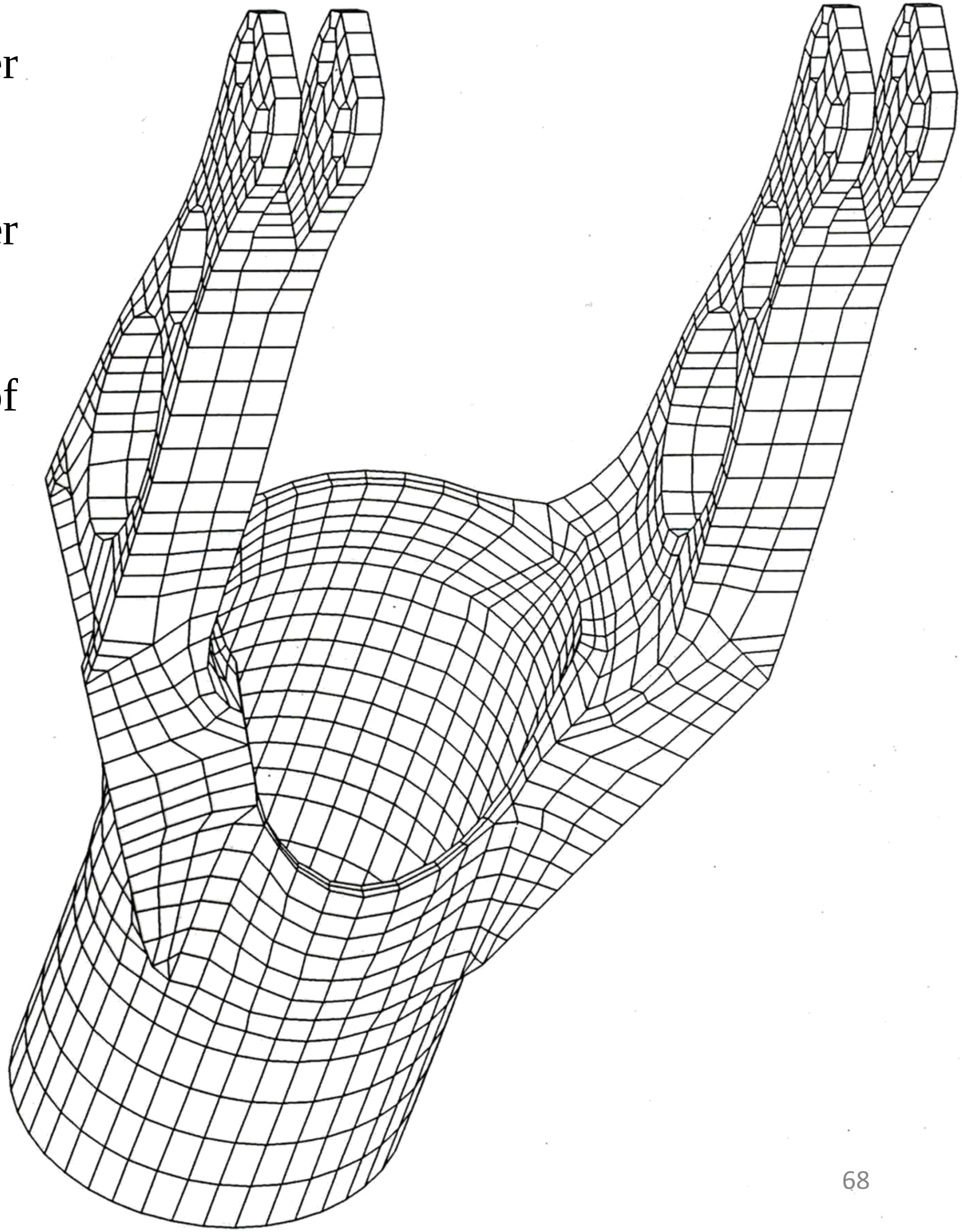


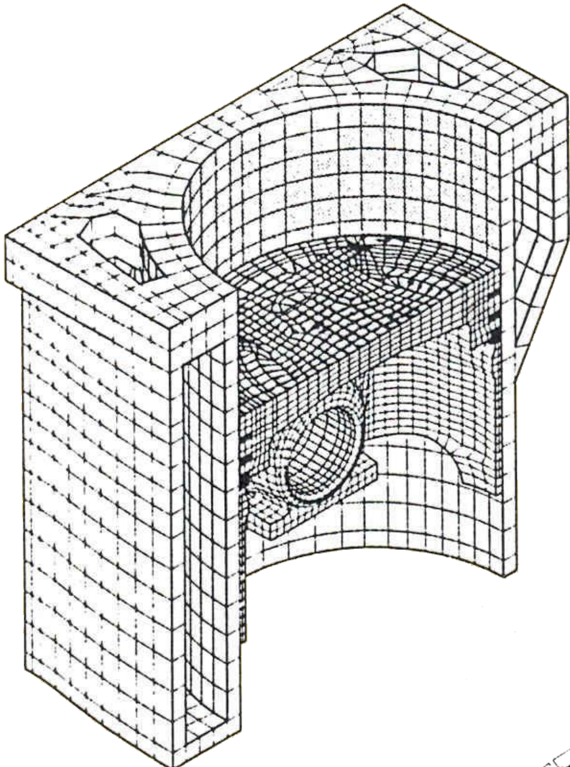
The model is mainly composed of hexahedrons.
With these models the mesh perfectly respects the symmetries of the geometric model, which is very difficult to obtain with triangles or tetrahedrons.

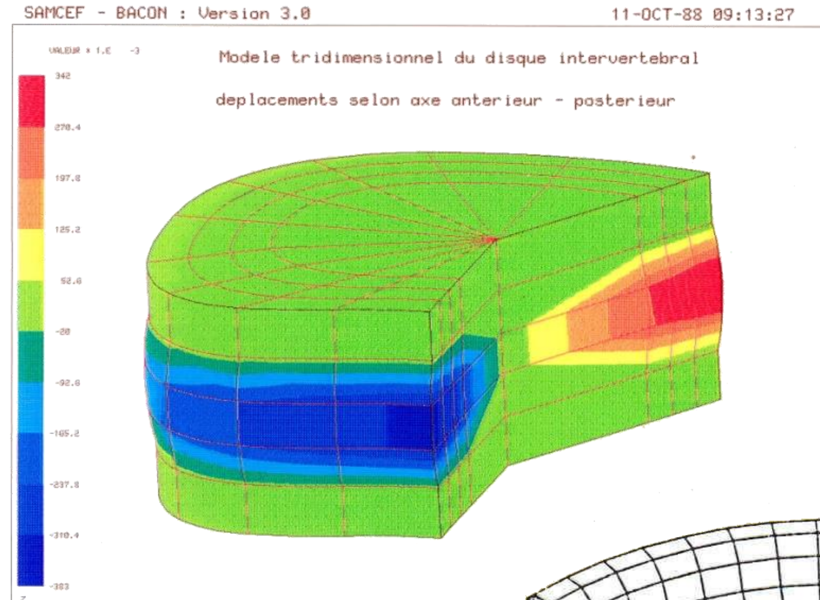
Helicopter model from Eurocopter manufacturer.

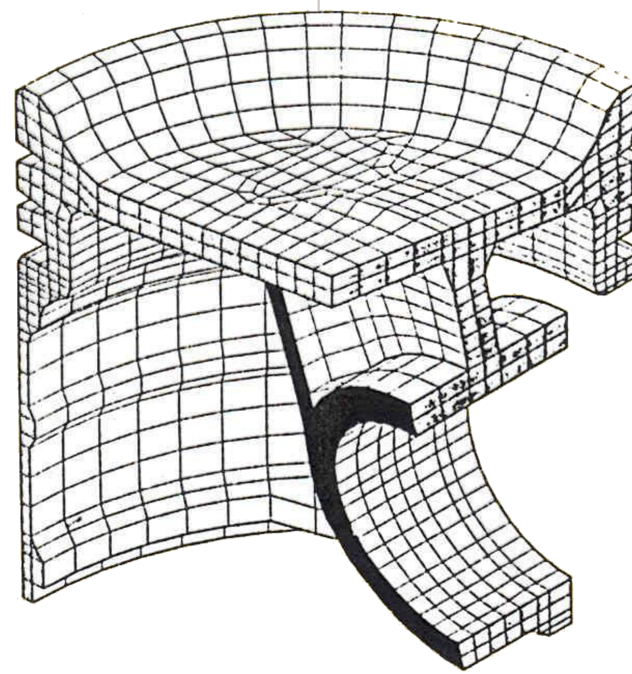
The quadrilaterals of the outer surface are not planar.

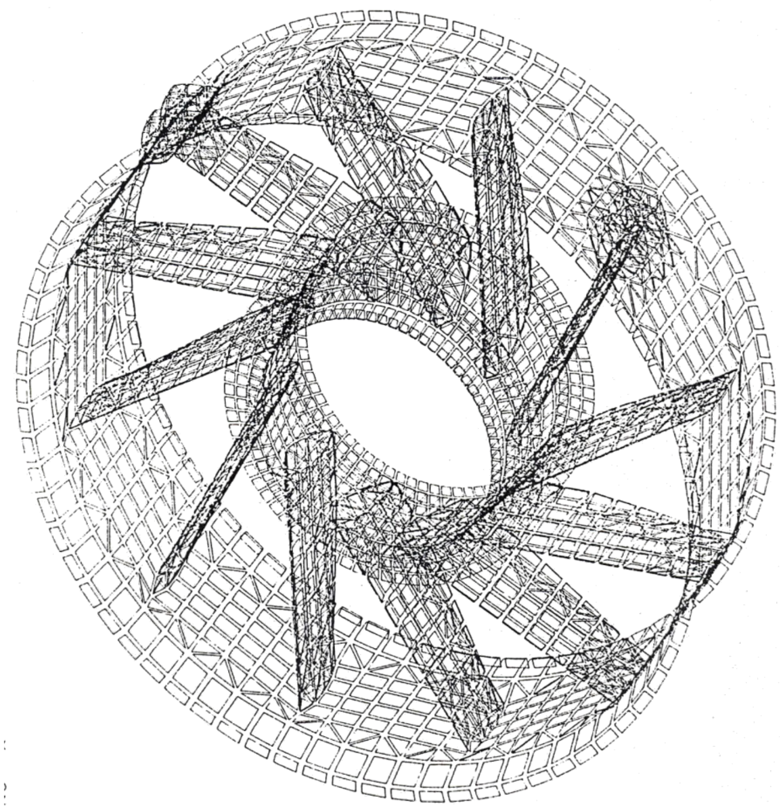
The model is mainly composed of hexahedrons



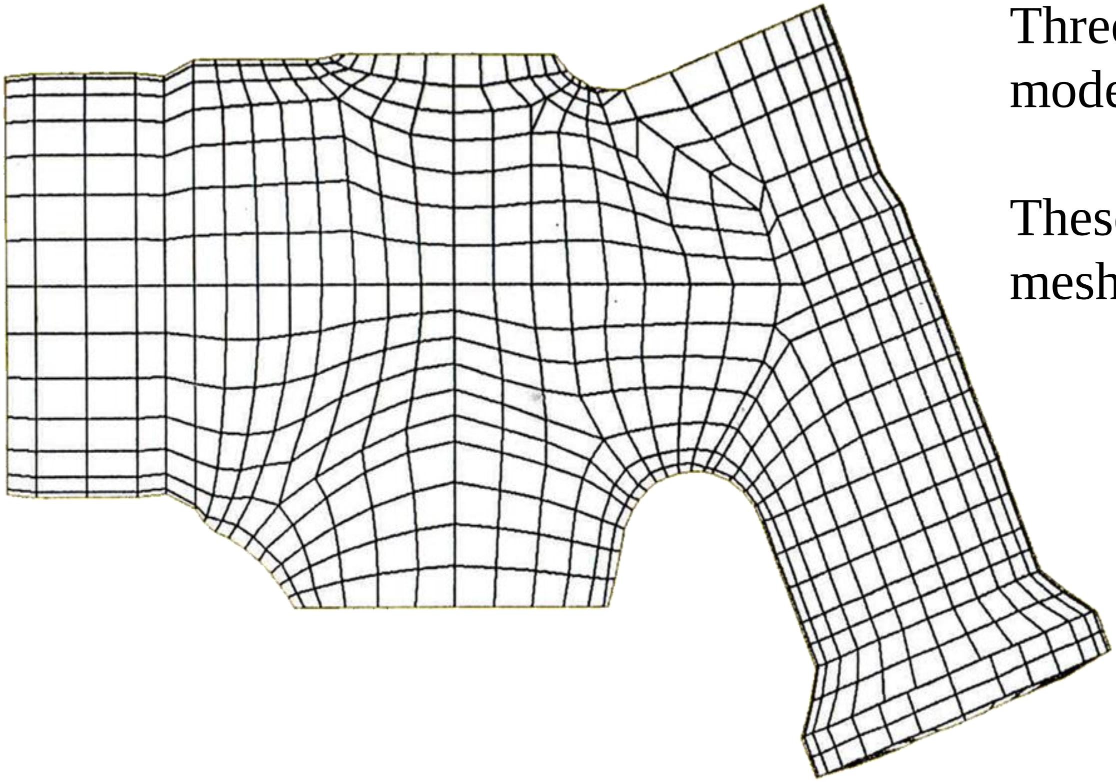






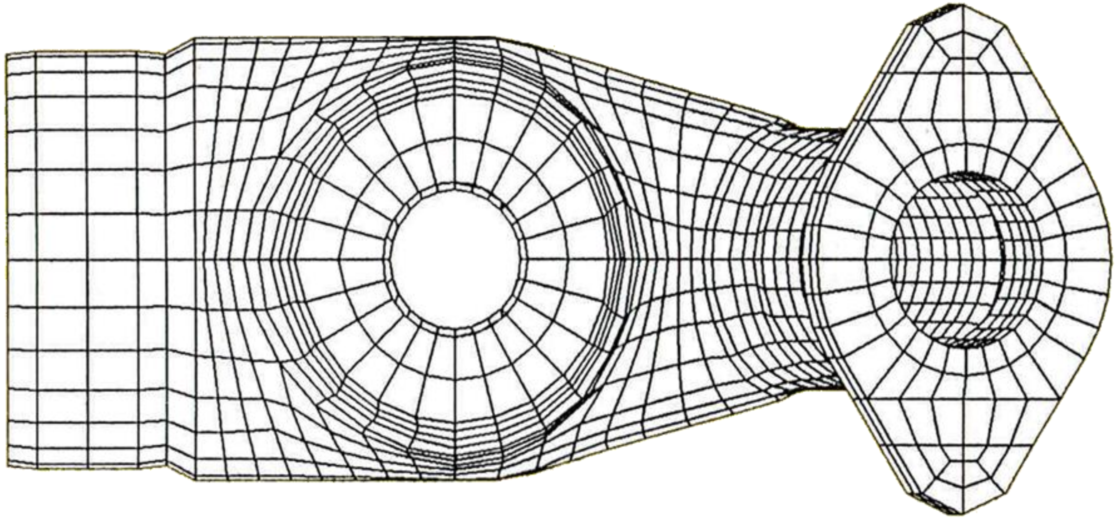


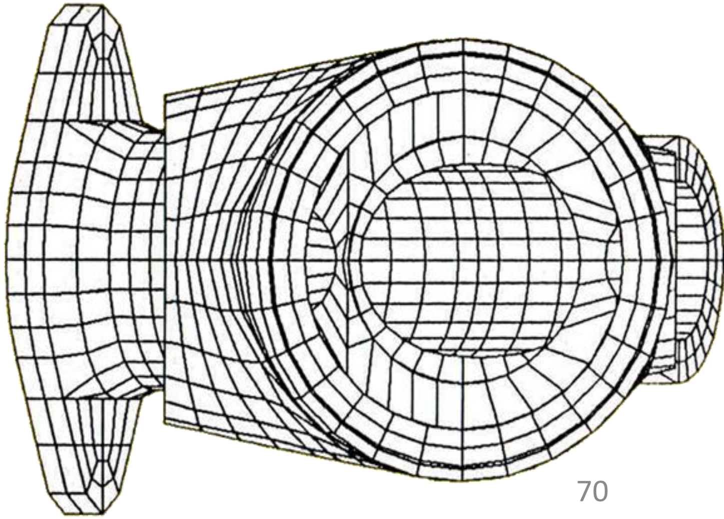
Finite element models, 1970-80…



Three views of a mechanical part modeled with hexahedrons.

These elements allow keeping in the mesh the symmetry of the geometry.





Isoparametric element

1. Early background: geometric mesh
2. CAD background
3. Mathematical formulation of the Coons patch
4. Formulation of the integrals in parametric coordinates
5. Computation of Cartesian gradients in intrinsic or parametric coordinates
6. Gauss quadrature
7. Numerical integration of the conduction matrix
8. Examples
9. Extension of Coons patch to 3D hexahedrons in elasticity
10. Conclusion

For the Coons patches and the similar elements, the basic feature is the integration (or quadrature). It is using the Jacobian of the transformation allowing to pass from parametric coordinates to Cartesian coordinates or *vice versa*.

The price to pay for using high degree elements and therefore curved edges is twofold

1. it is necessary to go through a numerical integration of the quantities to be evaluated on the domain of the element.

2. It is necessary to use a parametric formulation for the control of the geometry.

Higher degree elements: bi-cubic Béziers patch including the drawing of the interior lines network *s* = constant and *t* = constant. This patch is based on 16 nodes.