University of Bologna - Department of Mathematics

Piazza di Porta S.Donato, 5 - 40127 - Bologna

***xccurv*: the 2D modeller User's Guide - Version 1.0**

G. Casciola

Department of Mathematics
University of Bologna

Bologna 2000

Abstract

This report describes the *xccurv* system. This is a program for modelling free form 2D curves, which is only based on NURBS (Non Uniform Rational B-Splines) mathematics primitive.

G. Casciola
Department of Mathematics, University of Bologna, P.zza di Porta S.Donato 5, Bologna, Italy. E-mail: `casciola@dm.unibo.it`.

Contents

CHAPTER 1

What is *xccurv*?

xccurv is the 2D modeller of the *xcmodel* system [XCMODEL00]. This program is self-contained and executable from the *xcmodel* console window.
It is distributed with the archive **xccurvdev.tar.gz** (version in its development phase) and **xccurvusr.tar.gz** (executable version). Downloading and installation instructions are in [XCMODEL00].
xccurv is a system for modelling 2D curves, only based on NURBS mathematics primitive. This interactive graphics system was designed bearing in mind two main objectives:

- to provide a development environment for the experimentation of new techniques and algorithms in the sector of geometric modelling;
- to provide a learning environment and a practical application of the theories of geometric modelling presented in many books and papers (see references).

For these reasons, *xccurv* is very different from other 2D CAD systems on the market. Apart from being based on NURBS, it has general characteristics, not belonging to CAD systems, allowing the user to provide all the data needed to define a curve (such as knot partition, weights, type of parametrization), while CAD systems remain more restrictive, imposing methods that are considered efficient, but are invariable.
For example, *xccurv* allows the user to choose the NURBS degree, and this degree can have up to a maximum value of 10. In CAD systems, much lower degrees of spline are used (at most cubic).
As a result, the algorithms implemented in *xccurv* are as general as possible, in order to be applied in all the cases that the user may require. This could limit the efficiency of the system. For example, the calculation of the B-spline functions (the basis of the space of the polynomial spline) exploits the recurring formula that defines them. In CAD systems, on the other hand, algorithms are used that make the calculation of the B-splines, as long as they are of low degree, through the conversion of each span into the

power basis and the use of the Horner method.
Additionally, while in a CAD system 'circle curve' or 'circle arc curve', etc., are defined as being primitive, *xccurv* provides the instruments for the creation of these curves, while waiting for the users to have the basic knowledge that will allow them to be constructed.

The code is written in ANSI C language and can be executed on different workstations. In order to make the system portable, it was decided not to use graphics development tools. The graphic user interface only uses routines from the Xlib library, provided with the Xwindow system. This helps flexibility, since the Xwindow system is a standard for workstations. The X programming is therefore complex, in order to manage the user interface by means of the creation of windows, pop-up menus, buttons, events management, etc.

CHAPTER 2

How to work with *xccurv*

When *xccurv* is set up, it opens five windows (see figure 2.1):

- *Menu*: in which five option buttons and the button to quit the program are viewed.
- *Colors*: presents the color palette, if colors are available on your computer. Otherwise there is a palette of greys or only black and white.
- *Curves*: is used to design curves. Every point on the window has real coordinates belonging to the interval $[-1, 1] \times [-1, 1]$. One can imagine that it contains a system of cartesian axes from the centre of the window, whose ascisse axis is horizontal and whose versus is to the right, while the ordinates axis is vertical and its versus is to the top.
- *Text*: is used to input from the keyboard or to view texts (data relating to the error curves and messages).
- *Functions*: is used to view the knot partition, to graphically represent the spline basis functions, the component functions of the curves, the speed and curvature functions and finally the $\|C'(t)\|$ function that estimates the accuracy of the parametrization of the curve.

The user can vary the position and dimensions of some of the windows, within the limits defined by the program, in order to view all the information required. For example, it is not possible to change the dimensions of the **menus colors** and **texts** windows. It is possible, on the other hand, to change the dimensions of the curves and functions windows.

As in any interactive graphics system, *xccurv* uses the keyboard and the mouse. In the text we use the expression "click on something with the LMB" (Left Mouse Button) or "the CMB" (Centre Mouse Button) or "the RMB" (Right Mouse Button), when the user places the mouse pointer on something on the screen and presses and releases (to click) a mouse button. In order

to carry out any operation allowed by the system, the user has to select an option from the main menu. To select, click on a menu button with the LMB. A pop-up menu will then open, indicating the possible choices. For example, clicking on the **Curve** button, a window appears showing the operations that can be made on a curve (such as create, delete, reparametrize, etc.).
If the user chooses the **New** button, a pop-up menu will appear, allowing the user to choose the type of curve (for example, shape approximation or least square approximation or an interpolation curve).
All the *xccurv* menus have a **Close** button to exit from the menu without making a choice. Figure 2.1 shows the work environment and a pop-up menu.

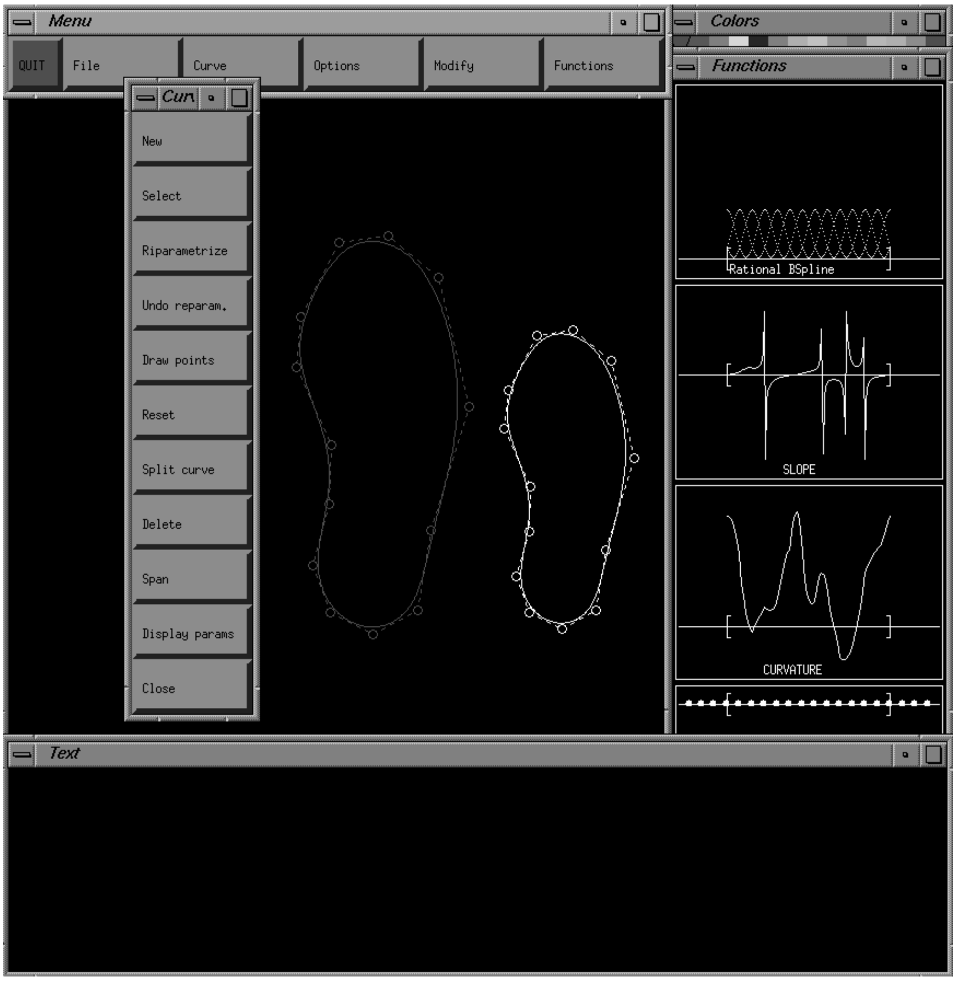

Figure 2.1: Work environment

Figure 2.2 shows all the *xccurv* menus and how they are linked, starting from the main menu buttons. This manual is organised to explain all the

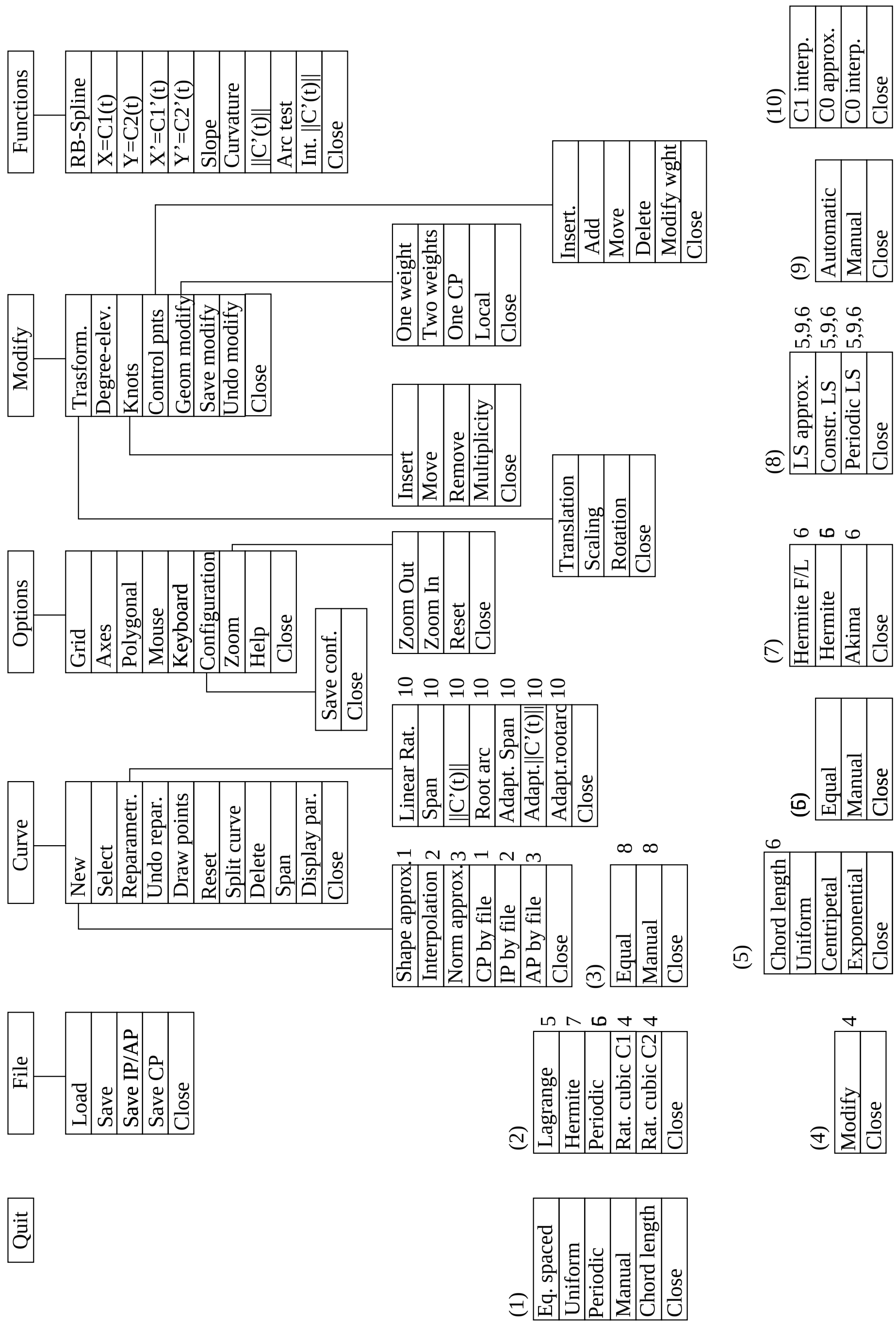

Figure 2.2: Menu diagram

menus and their buttons and how *xccurv* works, following a top-down menu order.
Note that there is an on-line **help**, that can be activated in the Option menu, and that guides the user, step by step, in his/her actions.
The detailed description of the methods, only described here from a functional point of view, can be found in [CASC98]. The mathematical notation used in this manual is the same as that used in the instructions leaflet mentioned above.

2.1 What to do with *xccurv*

xccurv is used to model and examine traditional and free-form 2D curves. It is possible to work simultaneously on several curves. The selected (active) curve is visualised in white and it can be modelled or examined. The non-active curves are shown in the selected color from the Colors menu in which they were created. There are several techniques for creating curves, from interactive to automatic by interpolation and approximation methods (see **New** button). *xccurv* allows the user to display curve parameters and some test functions, such as slope, curvature etc. (see Functions menu). *xccurv* allows the user to apply and analyze some modelling tools, such as knot-insertion, knot-removal, degree-elevation, etc. (see Modify menu). The active curve can be modified by working directly on its parameters or aided by the system giving some geometric constraints (see Modify menu). The active curve can be transformed by geometric transformation, such as translation, scaling and rotation. *xccurv* allows the user to save the active curve in a file (.db extension), such as save its control points and interpolation/approximation points only (.cp and .ip extensions). It is also possible to load a curve or other file previously saved (see File menu and Data file formats section).

CHAPTER 3

Options button

This button opens the following menu from which it is possible to choose the functions of *xccurv*.

- Grid
- Axes
- Polygonal
- Mouse
- Keyboard
- Configuration
- Zoom
- Help
- Close

Some buttons in this menu are check-boxes indicating the selection of a yes/no option.

3.1 Grid button

This button opens a menu with the following options:

- $\times 2$
- /2
- able/unable
- Close

The third button allows you to able/unable a reference grid in the *Curves* window. If this is abled, every operation following it will be carried out in a system of discrete coordinates, that is, every point is approximated to the nearest point on the grid.
The $\times$**2** and /**2** buttons, if the grid is active, allow it to be doubled or halved.

3.2 Axes button

This button draws the coordinate axes in the *Curves* window.

3.3 Polygonal button

This button ables/unables the viewing of the control polygon of the active curve.

3.4 Mouse and Keyboard buttons

The **Mouse** and **Keyboard** buttons are mutually exclusive check-boxes. If **Mouse** is chosen, almost all the operations for data insertion will be carried out using the mouse, whereas, by choosing **Keyboard**, these will be made using the keyboard.

3.5 Configuration button

This button allows the user to modify the position of all the *xccurv* windows and the dimensions of the *Curves* and *Functions* windows (the *Menus*, *Texts* and *Colors* windows have a fixed size). It is possible to save this configuration in the .xccurvrc file. The system will use this file to set the configuration windows in following work sessions.
Click on **Configuration** button with LMB, and the following pop-up menu will appear:

- Save conf.
- Close

Now you can modify the positions and dimensions of *xccurv* windows as usual with your Window Manager; when you have finished click on **Save conf.** button with LMB to save the new configuration. If you don't click on the **Save conf.** button, but on **Close**, your configuration will be active only for the present work session.

3.6 Zoom button

This button opens the following menu:

- Zoom out
- Zoom in
- Reset
- Close

Buttons **Zoom in** and **Zoom out** respectively allow you to magnify and reduce the image curves in the *Curves* window. Button **Reset**, resets the *Curves* window to $[-1, 1] \times [-1, 1]$.

3.7 Help button

This button is used to able/unable the on-line help of **xccurv**. For every selected operation, this help guides the user on how to carry out the operation correctly.

CHAPTER 4

File button

This button opens the following menu from which it is possible to choose to read/write from/on the data files for the NURBS curves.

- Load
- Save
- Save IP/AP
- Save CP
- Close

4.1 Load button

This button allows a file, with the extension .db, to be loaded. The files with this extension contain all the data needed to define a 2D NURBS curve in a standard format (see Data file formats section) The curve defined by this type of data file will be viewed and will be active. A mask will appear in the *Texts* window (see fig.4.1) through which it is possible to:

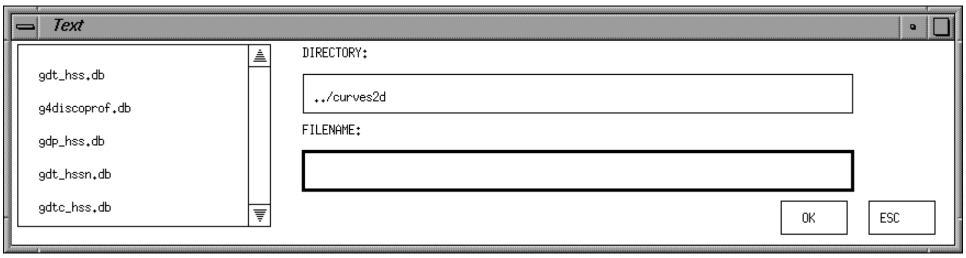

Figure 4.1: Load and Save mask

- select the search directory. Click on DIRECTORY box with the LMB. Then use Backspace to delete the old name and digit the new one. Press Enter on the keyboard to view the directory list;

- choose the file to be loaded, by giving it a name. Click on the FILE NAME box with the LMB. Use Backspace to delete the old name and digit the new one. Press Enter on the keyboard to load the file;

- select a file name from those on the list. Click on the filename with the LMB. Press Enter on the keyboard to load the file; to scroll the file names, click on the arrows with the LMB.

- Press OK to exit;

- Press ESC to dismiss.

If it is impossible to load the file, or if the file or the chosen directory does not exist, an error message will appear.

4.2 Save button

This button allows all the definition data for a 2D NURBS curve active in a file, with a .db extension, to be saved (see db format in section 9.). In the *Texts* window, the mask of fig. 4.1 appears, through which it is possible to:

- select the search directory. Click on the DIRECTORY box with the LMB then use Backspace to delete the old name and digit the new. Press Enter on the keyboard to view the directory list;

- choose a name for the file to be saved. Click on the FILE NAME box with the LMB. Use Backspace to delete the old name and digit the new one. Press Enter on the keyboard to save the file;

- select a filename from those on the list. Click on the filename with the LMB; Press Enter with the keyboard to save the file; to scroll the file names, click on the arrows with the LMB.

- Press OK to exit;

- Press ESC to dismiss.

If it is impossible to save the file, or if the chosen directory does not exist, an error message will appear.

4.3 Save IP/AP button

If the active curve has been created by a process of interpolation or approximation of data, this button allows the interpolation or approximation points used to be saved in a file with an .ip extension. In the *Texts* window the mask of fig.4.1 appears, through which it is possible to:

- select the search directory. Click on the DIRECTORY box with the LMB then use Backspace to delete the old name and digit the new. Press Enter on the keyboard to view the directory list;
- choose a name for the file to be saved. Click on the FILE NAME box with the LMB. Use Backspace to delete the old name and digit the new one. Press Enter on the keyboard to save the file;
- select a filename from those on the list. Click on the filename with the LMB; Press Enter with the keyboard to save the file; to scroll the file names, click on the arrows with the LMB.
- Press OK to exit;
- Press ESC to dismiss.

If it is impossible to save the file, or if the chosen directory does not exist, an error message will appear.

4.4 Save CP button

This button allows the user to save the control points of the active curve in a file with the .cp extension. The mask of fig.4.1 appears in the texts window, which can be used to:

- select the search directory. Click on the DIRECTORY box with the LMB then use Backspace to delete the old name and digit the new. Press Enter on the keyboard to view the directory list;
- choose a name for the the file to be saved. Click on the FILE NAME box with the LMB. Use Backspace to delete the old name and digit the new one. Press Enter on the keyboard to save the file;
- select a filename from those on the list. Click on the filename with the LMB; Press Enter on the keyboard to save the file; To scroll the file names click on the arrows with the LMB.
- Press OK to exit;
- Press ESC to dismiss.

If it is impossible to save the file, or if the chosen directory does not exist, an error message will appear.

CHAPTER 5

Curve button

This button opens the following menu from which it is possible to choose different generic functions for curves (such as create a new curve, reparametrize or subdivide a curve), but also different management functions (such as select a curve to become active, delete one or all curves, view the definition parameters of a curve, etc.).

- New
- Select
- Reparametrize
- Undo reparametrization
- Draw points
- Reset
- Split curve
- Delete
- Span
- Display parameters
- Close

5.1 New button

This button opens the following menu which presents the different curve creation modes provided by *xccurv* (see also fig.2.1).

- Shape approx.

- Interpolation
- Norm approx.
- CP file
- IP file
- AP file
- Close

A curve can be created by:

- shape approximation from a set of 2D points (control points (CP)),
- interpolation from a set of 2D points (interpolation points (IP)),
- approximation in norm from a set of 2D points (approximation points (AP)).

For each of these modes, the user can input the 2D points using the mouse or keyboard, or read these points from a data file.
The next steps, after the input of the 2D points, that is, the input of the remaining data (degree, knot partition, weights) to define the curve, are the same.
Therefore we can describe these cases together.

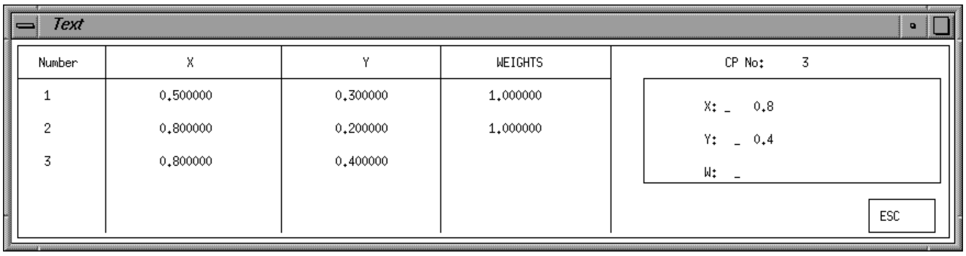

Figure 5.1: CP setting, using the keyboard

5.1.1 Shape Approx. and CP file buttons

These buttons provide the creation of a shape, while preserving the approximation curve, or rather, a variation diminishing curve respect to a polygonal of vertices called control points:

1. Control Points setting:

   Mouse → click on curves window with LMB to select CPs;
   → click with CMB to select a CP coinciding with the previous one;
   → click with RMB to stop

   Keyboard → digit CP coordinates and weights in the text-box mask of fig.5.1;
   → press Enter with keyboard to confirm;
   → click on ESC with LMB to stop.

   File → choose the file (.cp) to load;
   → press Enter with keyboard to confirm;
   → click on OK with LMB to stop.

2. Set the degree of the curve: digit the degree in the text-box; press Enter with keyboard to confirm.

3. Choose extended partition Δ^*. A menu with the following choices will appear:

   - Equally spaced knots
   - Uniform knots
   - Periodic knots
   - Manual knots
   - Chord lenght parametrization knots
   - Close

We describe the possible choices for extended partitions.

- **Equally spaced and Periodic buttons**

  – interior knots are set equally spaced in the [0,1] parametric domain. Let K be the number of interior knots (it holds $K = ncp - m$ where ncp is the CP number and m is the curve order), then t_i are given by:

  $$t_i = \frac{i-m}{K+1} \qquad i = m, \ldots, m+K+1$$

  – exterior knots are defined by

  $$\begin{array}{ll} t_i = a - (b - t_{i+K+1}) & i = 1, \ldots, m-1 \\ t_i = b + (t_{i-K-1} - a) & i = K+m+2, \ldots, K+2m \end{array}$$

where $[a, b]$ is the parametric domain. Since interior knots are equally spaced, we have the same extended partition both in Equally spaced and Periodic selections.

- **Uniform button**
  - interior knots are equally spaced;
  - exterior knots are coincident with the end interval points:

$$\begin{array}{ll} t_i = 0.0 & i = 1, \ldots, m-1 \\ t_i = 1.0 & i = K+m+2, \ldots, 2m+K \end{array}$$

- **Manual button**
  The extended partition can be setted by the user. With the mouse he/she can set only the interior knots, while the exterior are automatically set to be coincident with the end interval points; with the keyboard he/she can set all the knots.
  - Mouse → click K times on interval domain with LMB (*Functions* window)
  - Keyboard → digit all knot values, that is $K + 2m$ values, in textbox in *Texts* window.

- **Chord length parametrization button**
  - interior knots are chosen so that:

$$\frac{t_{i+1} - t_i}{t_{i+2} - t_{i+1}} = \frac{\sum_{j=i-m+1}^{i-1} \|P_j - P_{j-1}\|_2}{\sum_{j=i-m+2}^{i} \|P_j - P_{j-1}\|_2}$$

    where P_j are the curve CPs.
  - exterior knots are coincident with the end interval points:

  Note that all automatic extended partitions have interior knots with a multiplicity to 1. It is only possible to set knots manually with a multiplicity greater than 1.

The last parameters to set are the NURBS weights. If you have selected the control points using the keyboard, then you are required to give them weights; if you have used the mouse, then the weights are set automatically to 1. To change their values, you must use the Options menu. Then choose the **Control point** button and, finally the **Modify weights** button.
To summarise, once we have given all the following information:

- degree g (or order $m = g + 1$);

- knot partition (Δ^*) and number of knots ($K + 2m$);
- control points (CPs) and number ($K + m$);
- weights;

xccurv can proceed to compute and graphically represent the curve.

5.1.2 Interpolation and IP file buttons

These buttons provide the creation of an interpolation curve starting from some given 2D points (IP).

1. Interpolation Points setting:

Mouse	→	click on *Curves* window with LMB to select interpolation points;
	→	click with RMB to stop.
Keyboard	→	digit IP coordinates in the mask text-box as for fig.5.1;
	→	press Enter with keyboard to confirm;
	→	click on ESC with LMB to stop.
File	→	choose the file (.ip) to load;
	→	press Enter with keyboard to confirm;
	→	click on OK with LMB to stop.

2. Choose an interpolation method:
   - Lagrange interpolation
   - Hermite interpolation
   - Periodic interpolation
   - Rational cubic interpolation C^1 with tension
   - Rational cubic interpolation C^2 with tension
   - Close

3. Choose a parametrization strategy:
   - Uniform
   - Chord length
   - Centripetal
   - Exponential
   - Close

4. Choose NURBS weights:
   - Non rational
   - Manual
   - Close

We describe the available interpolation methods:

- **Lagrange interpolation**
  Let $\{Q_i\}$ $i = 1, \ldots, n$, be the interpolation points. This method computes the NURBS curve $\underline{C}(t)$, so that:

  $$\underline{C}(\tau_i) = Q_i \qquad i = 1, \ldots, n$$

  The degree of the curve must be given. The knot partition will be computed automatically so that the Schoenberg-Whitney conditions are satisfied. Exterior knots are coincident with the end interval points.

- **Hermite interpolation**
  Let $\{Q_i^\ell\}$ $i = 1, \ldots, n$, and $\ell = 0, \ldots, \ell_i$ be the interpolation points. This method computes the NURBS curve $\underline{C}(t)$ so that:

  $$\underline{C}^\ell(\tau_i) = Q_i^\ell \qquad i = 1, \ldots, n \quad \ell = 0, \ldots, \ell_i$$

  This option opens the following menu:

  – **Hermite F/L**
  This method, in addition to the interpolation points $\{Q_i^0\}$, $i = 1, \ldots, n$, interpolates the first derivative at the end interval points; that is Q_1^1 and Q_n^1 ($\ell_1 = \ell_n = 1$ and $\ell_i = 0$ for $i = 2, \ldots, n-1$) must be given. The interpolation curve is of degree 3. The interior knots are chosen to coincide with the interpolation parameter values τ_i; the exterior knots are coincident with the end interval points.

  – **Hermite**
  In addition to the interpolation points $\{Q_i^0\}$, $i = 1, \ldots, n$, $d \in \{1, 2, 3, 4\}$ must be given and then the information $\{Q_i^\ell\}$, $i = 1, \ldots, n$, and $\ell = 1, \ldots, d$. The interpolation curve is of degree $2d + 1$. The interior knots are chosen to coincide with the interpolation parameter values τ_i and are of multiplicity $d + 1$. The exterior knots are coincident with the end interval points.

  – **Akima**
  Start from the given points $\{Q_i^0\}$, $i = 1, \ldots, n$. This option computes, the points $\{Q_i^1\}$, $i = 1, \ldots, n$ with Akima technique. Then

an interpolation of all this information, with a curve of degree 3, is performed. The interior knots are chosen to coincide with the interpolation parameter values τ_i and have multiplicity 2. The exterior knots are coincident with the end interval points.

- **Periodic interpolation**
  Let $\{Q_i\}_{i=1,\ldots,n}$, be the interpolation points. This method computes the NURBS curve $\underline{C}(t)$ so that:

$$\begin{cases} \underline{C}(\tau_i) = Q_i \quad\quad i = 0, \ldots, n \\ \underline{C}^1(\tau_1) = \underline{C}^1(\tau_n) \\ \underline{C}^2(\tau_1) = \underline{C}^2(\tau_n) \end{cases}$$

  The interior knots are chosen to coincide with the interpolation parameter values $i\tau_i$. The exterior knots are chosen to satisfy the periodic conditions automatically.

- **Rational cubic C^1 interpolation**
  Let $\{Q_i^\ell\}_{i=1,\ldots,n}$, and $\ell = 0, 1$ be the interpolation points and w_i $i = 1, \ldots, n-1$, be tension parameters associated with each pair $\{Q_i^\ell\}$ $\{Q_{i+1}^\ell\}$ of points. This method computes the interpolation NURBS curve (cubic over quadratic) with the property of being globally C^1 and for $w_i \to \infty \ \ \forall i$ to converge to the polygonal defined by the interpolation points. If $w_i = 1 \quad \forall i$, it is the cubic Hermite interpolation curve.

- **Rational cubic C^2 interpolation**
  Let $\{Q_i^\ell\}_{i=1,\ldots,n}$, and $\ell = 0, \cdots, \ell_i$ where $\ell_1 = \ell_n = 1$ and $\ell_i = 0 \quad i = 2, \cdots, n-1$ be the interpolation points and w_i $i = 1, \ldots, n-1$, be tension parameters associated with each pair $\{Q_i^\ell\}$ $\{Q_{i+1}^\ell\}$ of points. This method computes the interpolation NURBS curve (cubic over quadratic) with the property of being globally C^2. If $w_i = 1 \ \ \forall i$, it is the cubic Hermite interpolation curve that we called Hermite F/L.

After the choice of an interpolation method, the user must choose the parametrization strategy, that is, the rule to set the interpolation parameter values τ_i. *xccurv* provides four strategies:

- **Chord length parametrization**
  Sets the τ_i parameters in the interval $[0, 1]$, so that:

$$\frac{\tau_{i+1} - \tau_i}{\tau_{i+2} - \tau_{i+1}} = \frac{\|Q_{i+1} - Q_i\|_2}{\|Q_{i+2} - Q_{i+1}\|_2}$$

- **Uniform parametrization**
  Sets the τ_i parameters equally spaced in the interval $[0, 1]$.

- **Centripetal parametrization**
  Sets the τ_i parameters in the interval $[0,1]$, so that:

$$\frac{\tau_{i+1}-\tau_i}{\tau_{i+2}-\tau_{i+1}} = \left(\frac{\|Q_{i+1}-Q_i\|_2}{\|Q_{i+2}-Q_{i+1}\|_2}\right)^{1/2}$$

- **Exponential parametrization**
  Sets the τ_i parameters in the interval $[0,1]$, so that:

$$\frac{\tau_{i+1}-\tau_i}{\tau_{i+2}-\tau_{i+1}} = \left(\frac{\|Q_{i+1}-Q_i\|_2}{\|Q_{i+2}-Q_{i+1}\|_2}\right)^{\alpha} \quad \text{with} \quad \alpha > 0$$

Finally, the user must set the NURBS weights to define the NURBS space.

- **Non Rational**
  Each weight is set to 1. Thus the NURBS space is a non-rational spline space.

- **Manual**
  For each weight:

keyboard	→	digit its value
	→	Press Enter to confirm
	→	click on Esc with LMB to exit.

  Remember that the same number of weights must be set as the number of the interpolation points.

5.1.3 Norm approximation and AP file buttons

These buttons allow the computation of a least square weighted approximation curve starting from some given 2D points (AP).

1. Approximation Points setting:

Mouse	→	click on curves window with LMB to select approximation points;
	→	click with RMB to stop.
Keyboard	→	digit AP coordinates in the text-box mask as for fig.5.1;
	→	press Enter with keyboard to confirm;
	→	click on ESC with LMB to stop.
File	→	choose the file (.ip) to load;
	→	press Enter with keyboard to confirm;
	→	click on OK with LMB to stop.

2. Choose the values to weight the approximation points:
    - Equal
    - Manual
    - Close
3. Choose an approximation method:
    - Least square
    - Constrained least square
    - Periodic least square
    - Close
4. Choose curve degree:

    Keyboard $\rightarrow$ digit an integer value ≤ 10

5. Choose a parametrization strategy:
    - Uniform
    - Chord length
    - Centripetal
    - Exponential
    - Close
6. Choose extended partition:
    - Automatic
    - Manual
    - Close
7. Choose the number of knots:

    Keyboard $\rightarrow$ digit an integer value

8. Choose NURBS weights:
    - Non rational
    - Manual
    - Close

The weights to associate to the approximation points can be:

- **Equal**
  This option sets all the weights to 1; non-weighted approximation.

- **Manual**
  This option allows to input the weights by keyboard.

Approximation methods:

- **Least square approximation method**
  Let $\{Q_i\}_{i=1,\ldots,n}$, be the approximation points. This method computes the NURBS curve $\underline{C}(t)$, so that the following expression is the minimum value:
  $$\sum_{i=1}^{n} \|\underline{C}(\tau_i) - Q_i\|_2^2 = \sum_{i=1}^{n}((C_1(\tau_i) - x_i)^2 + (C_2(\tau_i) - y_i)^2)$$

- **Constrained least square approximation method**
  In addition to satisfying the least square approximation, this method constrains the curve to pass from the first and last given points, that is:
  $$\underline{C}(\tau_1) = Q_1 \quad e \quad \underline{C}(\tau_n) = Q_n$$

- **Periodic least square approximation method**
  In addition to satisfying the least square approximation, this method constrains the curve to be periodic, that is:
  $$\begin{cases} \underline{C}(\tau_1) = \underline{C}(\tau_n) \\ \underline{C'}(\tau_1) = \underline{C'}(\tau_n) \\ \vdots \\ \underline{C}^{(m-2)}(\tau_1) = \underline{C}^{(m-2)}(\tau_n) \end{cases}$$

Then the user must give the degree curve and choose the parametrization strategy, that is, the rule to set the interpolation parameter values τ_i. *xccurv* provides four strategies:

- **Chord length parametrization**
  This option sets the τ_i parameters in the interval $[0, 1]$, so that:
  $$\frac{\tau_{i+1} - \tau_i}{\tau_{i+2} - \tau_{i+1}} = \frac{\|Q_{i+1} - Q_i\|_2}{\|Q_{i+2} - Q_{i+1}\|_2}$$

- **Uniform parametrization**
  This option sets the τ_i parameters equally spaced within the interval $[0, 1]$.

- **Centripetal parametrization**
  This option sets the τ_i parameters in the interval $[0, 1]$, so that:
  $$\frac{\tau_{i+1} - \tau_i}{\tau_{i+2} - \tau_{i+1}} = \left(\frac{\|Q_{i+1} - Q_i\|_2}{\|Q_{i+2} - Q_{i+1}\|_2}\right)^{1/2}$$

- **Exponential parametrization**
  This option sets the τ_i parameters in the interval $[0,1]$, so that:

$$\frac{\tau_{i+1} - \tau_i}{\tau_{i+2} - \tau_{i+1}} = \left(\frac{\|Q_{i+1} - Q_i\|_2}{\|Q_{i+2} - Q_{i+1}\|_2} \right)^{\alpha} \quad \text{with} \quad \alpha > 0$$

The next steps are the choice of the number of knots and their position. The latter can be:

- **Manual**
  For each knot:

  Keyboard → digit its value
  → Press Enter to confirm
  → click on Esc with LMB to exit.

- **Automatic**
  xccurv sets the interior knots so that the least square approximation curve will be unique. The exterior knots are set to coincide with the end interval points.

The final step is to define the NURBS space by giving the NURBS weights:

- **Non Rational**
  Each weight is set to 1. Thus the NURBS space is a non-rational spline space.

- **Manual**
  For each weight:

  → digit its value
  → Press Enter to confirm
  → click on Esc with LMB to exit.

  Remember that the same number of weights must be set as the number of the approximation points.

5.2 Select curve button

xccurv allows the user to model a maximum number of nine curves at the same time. One of these must be selected as active (appearing in white). All the operations that *xccurv* provides are only able for the active curve. To select a curve as being active, click on one CP of a curve with LMB and click with RMB to confirm. Note that the last curve created or loaded by a file is set active.

5.3 Reparametrization button

This button opens the following pop-up menu, from which it is possible to choose different reparametrization techniques:

- Linear rational
- Span
- $\|C'(t)\|$
- Root arc
- Adaptive span
- Adaptive $\|C'(t)\|$
- Adaptive root arc
- Close

These consist in changing the current parameter of the active curve with another parameter using a linear rational reparametrization function that keeps the curve a NURBS too.
All the proposed techniques use a linear rational or a piecewise linear rational function to approximate the arc length parametrization function

$$\varphi(t) = \int_0^t \|C'(u)\|_2 \, du$$

by interpolation or uniform approximation.
For each chosen technique, the following menu is presented, from which it is possible to choose the approximation mode:

- C^1 interpolation
- C^0 uniform approximation
- C^0 interpolation
- Close

The proposed techniques partition the parametric interval in different ways and consequently use a different piecewise reparametrization function.

5.3.1 Linear rational button

This option reparametrizes the whole active curve using a single linear rational function. In this case the choise of C^1 or C^0 interpolation gives the same approximation function for $\varphi(t)$.

5.3.2 Span button

This option performs piecewise reparametrization of the active curve after splitting the parametric domain at the knots.

5.3.3 $\|C'(t)\|$ button

This option performs a piecewise reparametrization of the active curve after splitting the parametric domain at the points where the $\varphi(t)$ function

$$\varphi(t) = \int_0^t \|C'(u)\|_2 \, du$$

changes convexity (concavity).
Because a linear rational function is always convex (concave), this option allows the user to approximate the $\varphi(t)$ shape.

5.3.4 Root arc button

This option performs a piecewise reparametrization of the active curve after splitting the parametric domain at the zeroes of the arc test function $a(t)$

$$a(t) = \frac{\int_0^t \|C'(u)\| \, du}{L} - t \quad t \in [0, 1]$$

where L is the curve length ($L = \int_0^1 \|C'(u)\| \, du$).
A well-parametrized curve, in the arc length sense, results in $a(t) = 0 \ \ \forall t$. Splitting the reparametrization at the $a(t)$ zeroes means reparametrizing the curve only where necessary.

The following three techniques, called Adaptive, approximate the arc length parametrization function $\varphi(t)$ adaptively, up to a given tolerance.
The first step is to compute an approximate linear rational function over the whole parametric domain. If tolerance is not reached, the parametric interval is divided into two intervals, and so on:

5.3.5 Adaptive span button

This option divides the interval in correspondence with a knot. Otherwise, it splits the interval in half, if it does not contain knots.

5.3.6 Adaptive $\|C'(t)\|$ button

This option divides the interval into two intervals to respect the concave/convex behaviours of the $\|C'(t)\|$ function. Otherwise, it splits the interval in half, if the $\|C'(t)\|$ function is already concave or convex, but tolerance has not been reached.

5.3.7 Adaptive root arc button

This option divides the interval into two intervals to respect the roots of the $a(t)$ function. Otherwise, it splits the interval in half, if it does not contain other $a(t)$ roots.

5.4 Undo reparametrization button

This option allows the user to undo the last reparametrization of the active curve. If the curve has been parametrized more than once, it is possible to undo every reparametrization.

5.5 Draw points button

xccurv graphically represents the active curve using dots. These dots correspond to curve points with equally spaced parameter values.
This option is useful to test curve parametrization. In fact, if the curve is well-parametrized, these curve points will be equally spaced, even on the curve. To continue and redisplay the curve in the standard mode, click with LMB.

5.6 Reset button

This button resets *xccurv*. If the curves have not been saved they will be lost.

5.7 Subdivide button

This option allows the user to split the active curve into two curves.

- To subdivide, click on the chosen splitting point with LMB;
- Click with RMB to confirm;

5.8 Delete button

This option allows the user to delete the active curve. After this operation, no curve will be active. Proceed immediatly to select a curve to be active or create a new curve.

5.9 Span button

This button allows the user to visualize and match a knot interval with a curve span.

- To select a knot interval, click on it with LMB; repeat for other knot intervals.
- Click on the *Functions* window with RMB to stop.

5.10 Display parameters button

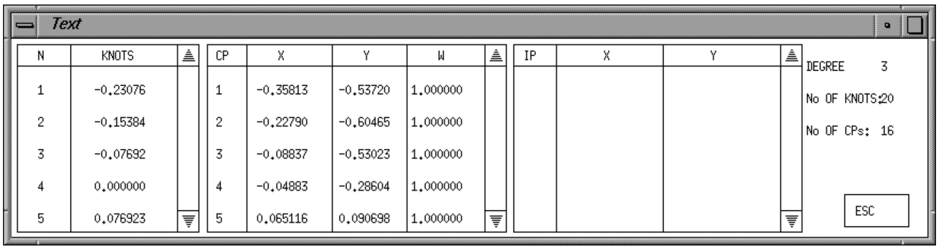

Figure 5.2: Display parameters mask

This button displays the definition parameters of the active curve in the mask shown in fig.5.2 on the *Texts* window.

- Click on arrows with LMB to scroll the knots, CPs and IPs/APs tables.
- Click on Esc with RMB to exit.

CHAPTER 6

Modify button

This button opens the following pop-up menu from which it is possible to modify the active NURBS curve:

- Tranformation
- Degree-elevation
- Knots
- Control points
- Geometric modify
- Save modify
- Undo modify
- Close

By **Modify** we mean variation of the curve parameters. Sometimes this involves a shape curve variation, and sometimes only a different analytical representation.
Every variation is not saved, unless you choose **Save modify**. Every unsaved variation can be deleted by choosing **Undo modify**.

6.1 Transformation button

This button opens the following menu with the available geometric transformations:

- Translation
- Scaling
- Rotation
- Close

6.1.1 Translation button

This button allows the user to translate the active curve from its position to a new one given by the user.

- translation center:

Mouse	→ click on chosen point with LMB;
	→ click with RMB to confirm
Keyboard	→ digit its coordinates;
	→ Press Enter to confirm

- translation vector defined with respect to the given center:

Mouse	→ click on chosen point with LMB
	→ click with RMB to confirm
Keyboard	→ digit its coordinates;
	→ Press Enter to confirm;

6.1.2 Scale button

This button allows the user to scale the active curve.

- scaling center:

Mouse	→ click on chosen point with LMB;
	→ click with RMB to confirm.
Keyboard	→ digit its coordinates;
	→ Press Enter to confirm;

- scaling factors:

Keyboard	→ digit their positive values;
	→ Press Enter to confirm;

6.1.3 Rotation button

This button allows the user to rotate the active curve.

- rotation center:

Mouse	→ click on chosen point with LMB;
	→ click with RMB to confirm
Keyboard	→ digit its coordinates;
	→ Press Enter to confirm.

- rotation angle in degrees:

Keyboard	→	digit its value;
	→	Press Enter to confirm.

6.2 Degree-elevation button

This button allows the user to make a degree elevation of the curve by 1 degree. This procedure does not modify the curve shape, but only its analytical representation.
To make a degree elevation greater than 1, click on the button more than once.

6.3 Knots button

This button opens the following menu, from which it is possible to choose between different variations of the current knot partition. Sometimes this involves a shape curve variation, and sometimes only a curve representation variation.

- Insert
- Move
- Remove
- Multiplicity
- Close

6.3.1 Insert button

This button allows the user to make a knot insertion of 1 or more knots in the actual knot partition. This procedure does not modify the shape of the curve, but only its analytical representation.

Mouse	→	click on chosen position inside parametric interval with LMB;
	→	click with RMB to stop.
Keyboard	→	digit new knot value;
	→	Press Enter to confirm;
	→	click on Esc with LMB to exit.

6.3.2 Move button

This button allows the user to move one or more interior knots on the actual partition from their initial position to a new one. This procedure modifies the curve shape.

Mouse	→	press on a knot with LMB. Drag to new position, then release;
	→	click with RMB to confirm.
Keyboard	→	select a knot by clicking on it with LMB in the knot table;
	→	digit its new value (use Backspace to delete the old value);
	→	Press Enter to confirm;
	→	click on Esc with LMB to exit.

6.3.3 Remove button

This option tries to remove a knot from the actual knot partition without modifying the curve shape. If this procedure is successful, the effect will be a curve with a different analytical representation.

Mouse	→	click on a knot with LMB;
	→	click with RMB to confirm.
Keyboard	→	select a knot by clicking on it with LMB in the knot table;
	→	Press Enter to confirm;
	→	click on Esc to exit.

6.3.4 Multiplicity button

This button allows the user to identify the multiplicity of the knots in the actual partition.

Mouse	→ click on a knot with LMB;
	→ click with RMB to stop.

An integer value (the knot multiplicity) will be displayed near the chosen knot.

6.4 Control points button

This button opens the following menu from which it is possible to modify the control points and weights.

- Insert
- Add
- Move
- Delete
- Modify weights
- Close

6.4.1 Insert button

This button allows the user to perform the operation known as Inverse Knot-Insertion. This consists in the specification of a point on the control polygon. *xccurv* identifies a point on the parametric domain from which, by knot insertion, a new polygonal can be obtained having the chosen point as one of the control points.
This procedure does not modify the curve shape, but only its analytical representation.

Mouse	→	click on chosen polygonal point with LMB;
	→	click with RMB to confirm.
Keyboard	→	click on CP table with LMB to choose the insertion position;
	→	digit its coordinates and weight values;
	→	press Enter to confirm each value;
	→	click on Esc with LMB to exit.

Note that, in the Keyboard case, if the new coordinate point does not belong to the polygonal, a message appear and the user must digit others coordinate points.

6.4.2 Add button

This button allows the user to add new control points. This procedure modifies the curve shape.

Mouse	→	to select where to add a CP, click on a polygonal segment with LMB or click on the first or last CP with CMB;
	→	click with RMB to confirm;
	→	click on new position with LMB;
	→	click with RMB to confirm;
	→	click on graded scale with LMB to give weight value;
	→	click with RMB to confirm;

Keyboard	→	to select the position at which to add a CP, click on CP table with LMB;
	→	digit its coordinates and weight values;
	→	press Enter to confirm each value;
	→	click on Esc with LMB to exit.

Note that the first step is to choose where to insert the new CP, then to give its position (Mouse) or coordinates (Keyboard).

6.4.3 Move button

This button allows the user to move a control point. This procedure modifies the control polygon and therefore the curve shape.

Mouse	→	click on chosen CP with LMB;
	→	click with RMB to confirm;
	→	click on new position with LMB;
	→	click with RMB to confirm.

Note that, at any new position, before confirming, *xccurv* shows the new polygonal and curve shape.

Keyboard	→	click on a CP with LMB in CP table;
	→	digit its coordinate values;
	→	press Enter to confirm;
	→	click on Esc with LMB to exit.

Note that, using the Keyboard, it is possible to move a lot of CPs with a single **Move** operation, but you can only view the new curve shape at the end.

6.4.4 Delete button

This button allows the user to delete a control point. This procedure modifies the control polygon and therefore the curve shape.

Mouse	→	click on chosen CP with LMB;
	→	click with RMB to confirm.
Keyboard	→	click on a CP with LMB in CP table;
	→	click on Enter with LMB to confirm;
	→	click on Esc with LMB to exit.

Note that, using the Keyboard, it is possible to delete a lot of CPs before quitting.

6.4.5 Modify weight button

This button allows the user to modify a weight value associated with a control point. This procedure modifies the curve shape.

Mouse	→	click on chosen CP with LMB;
	→	click with RMB to confirm;
	→	click on graded scale with LMB;
	→	click with RMB to confirm.

Note that, at any new weight value, before confirming, **xccurv** shows the new curve shape.

Keyboard	→	click on chosen CP with LMB in CP table;
	→	digit new weight value;
	→	press Enter to confirm;
	→	click on Esc with LMB to exit.

Note that, using the Keyboard, it is possible to modify a lot of weight values, before quitting.

6.5 Geometric modify button

This button opens the following menu, from which it is possible to make some geometric modifications to the shape of the curve. By **Geometric modify** we mean that it is possible to impose some geometric constraints, such as, passing over a given point.

- One weight
- Two weights
- One CP
- Local
- Close

6.5.1 One weight button

This button allows the user to modify the active curve so that it passes over a point S^* by only changing a weight. More precisely, the user must select a curve point (S) and a point on the control polygon (P). Then he/she must choose S^* on the straight segment MP (see fig.6.1), where M is the analog of S, when the weight associated to the control point P is zero.

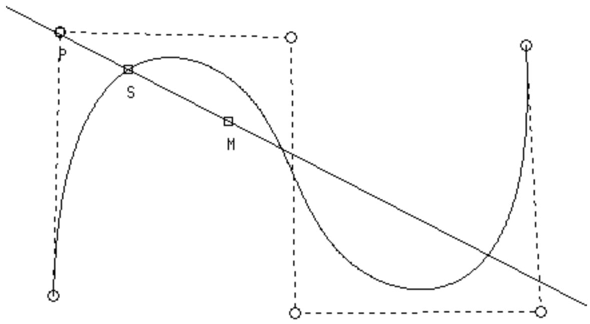

Figure 6.1: One weight modify

Mouse → click on a curve point (S) with LMB;
→ click with RMB to confirm;
→ click on a polygonal point (P);

The line defined by S and P and the segment MP on this line will be shown.

→ click on MP segment with LMB to choose S^*;
→ click with RMB to confirm.

Note that, at any chosen S^*, before confirming, *xccurv* shows the new curve.

6.5.2 Two weights button

This button allows the user to modify the active curve so that it passes over a point S^* by changing only two weights. More precisely, the user must select two control points; *xccurv* shows a curve point S and a point M. S is the curve point that will be moved to S^*, while M represents the curve point S, if the weights associated with the two selected control points were zeroes. Now the user can set S^* to be in the triangle with the two control points and M as the verices (see fig. 6.2).
xccurv also shows the line through S and M. If the user chooses S^* in the triangle, but also on SM, the curve will be modified to be simmetry preserving.

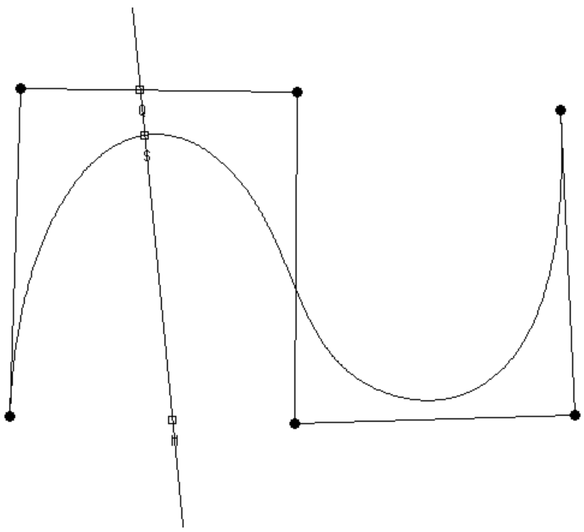

Figure 6.2: Two weights modify

Mouse $\rightarrow$ click on a CP (P_r) with LMB;
$\rightarrow$ click with RMB to confirm;
$\rightarrow$ click on a CP (P_s) with LMB;
$\rightarrow$ click with RMB to confirm;
$\rightarrow$ click on a point (S^*) inside $P_r P_s M$ triangle with LMB;
$\rightarrow$ click with RMB to confirm.

Note that at any chosen S^*, before confirming, *xccurv* shows the new curve.

6.5.3 One CP button

This button allows the user to modify the active curve so that it passes over a point S^* by changing only one control point. More precisely, the user must select a curve point (S); *xccurv* finds the control point nearest to S and shows the line through S and the control point. The user must set S^* to be on this line.

Mouse $\rightarrow$ click on a curve point (S) with LMB;
$\rightarrow$ click with RMB to confirm;

Now *xccurv* shows the line defined by S and the control point nearest to S.

$\rightarrow$ click on this line with LMB to choose S^*;
$\rightarrow$ click with RMB to confirm.

Note that, at any chosen S^*, before confirming, *xccurv* shows the new curve.

6.5.4 Local button

This button allows the user to select a curve segment and modify only the shape of this curve segment only. This procedure consists in local knot refinement and in a control point movement.

Mouse	$\rightarrow$	click on a curve point with LMB to select the first segment end point;
	$\rightarrow$	click on a curve point with LMB to select the last segment end point;
	$\rightarrow$	click with RMB to confirm.

xccurv computes the parametric interval associated with this curve segment and inserts $m + 1$ knots (with m as the order curve) inside this interval. Now there is a basis function with this interval as support. By moving the control point associated with this basis function, only the curve segment will be modified.

Mouse	$\rightarrow$ click on a new position with LMB to move the control point;
	$\rightarrow$ click with RMB to confirm.

Note that *xccurv* shows the new curve at any new control point position.

6.6 Save modify and Undo modify buttons

These buttons allow the user respectively, to save the modification made to the active curve (both shape and representation modification) and to undo the previous curve.

CHAPTER 7

Functions button

This button opens the following menu which provides the visualization of 10 different function graphs. These are very useful when you need to inspect the analytic and shape characteristics of the active curve.
xccurv is able to show up to 3 function graphs at the same time. At the bottom of the *Functions* window, the parametric interval and the actual knot partition are always shown.

- RB-Spline
- $X = C_1(t)$
- $Y = C_2(t)$
- $X = C_1'(t)$
- $Y = C_2'(t)$
- Slope
- Curvature
- $\|C'\|$
- Arc Test
- Int ($\|C'\|$)
- Close

7.1 RB-Spline button

This option shows the graphs of the basis function for the NURBS space $R(P_m, M, \Delta, W)$ to which the active curve belongs.

7.2 $X = C_1(t)$ and $Y = C_2(t)$ buttons.

These options show the graphs of the components of the active curve. The control polygons of these functions that have vertices $(\xi_i, x_i)_{i=1,\ldots,m+K}$ (or $(\xi_i, y_i)_{i=1,\ldots,m+K}$) are also shown. The ξ_i are the nodes and (x_i, y_i) are the control point coordinates.
If the active curve has been created by interpolation or approximation, the points (t_i, x_i) (o (t_i, y_i)) where t_i are the parameter values associated to the interpolation or approximation points (x_i, y_i) will be shown.

7.3 $X = C_1'(t)$ and $Y = C_2'(t)$ buttons

These options show the graphs of the first derivative functions of the curve component functions.

7.4 Slope button

This option shows the graph of the slope function $v(t)$, defined by:

$$v(t) = \frac{C_2'(t)}{C_1'(t)}$$

7.5 Curvature button

This option shows the graph of the curvature function $k(t)$, defined by:

$$k(t) = \frac{C_1'(t)C_2''(t) - C_2'(t)C_1''(t)}{([C_1'(t)]^2 + [C_2'(t)]^2)^{3/2}}$$

7.6 $\|C'(t)\|$ button

This option shows the graph of the function $\|C'(t)\|_2 - L$, where L is the curve length.
This graph is useful to show the user whether the active curve is well-parametrized or not. In fact, if a curve is well-parametrized, in the arc length sense, then

$$\|C'(t)\| \longrightarrow L$$

and therefore $\|C'(t)\| - L$ must zero.
In addition, the minimum and maximum function values are shown.

7.7 Arc test button

This option shows the graph of the function $a(t)$, defined by:

$$a(t) = \frac{\int_0^t \|C'(u)\|\, du}{L} - t \quad t \in [0,1],$$

where L is the curve length ($L = \int_0^1 \|C'(u)\|\, du$). This graph is useful to show the user whether the active curve is well-parametrized or not. In fact, if a curve is well-parametrized, in the arc length sense, then $a(t)$ must be zero.
In addition the minimum and maximum function values are shown.

7.8 Int $\|C'(t)\|$ button

This option shows the graph of the function $\varphi(t)$, defined by:

$$\varphi(t) = \int_0^t \|C'(u)\|_2\, du$$

This graph is useful to show the user whether the active curve is well-parametrized or not. In fact, if a curve is well parametrized, in the arc length sense, then $\varphi(t)$ approximates a linear function; $\varphi(t)$ is the analytic reparametrization function. When you choose to reparametrize the active curve, you can compare the $\varphi(t)$ function with the reparametrization function that *xccurv* uses.

CHAPTER 8

Data file formats

In this section the sintax of each file format used by *xccurv* is given and explained. The data files created or used by *xccurv* are stored in the directory `xcmodel/curves2d` as default. The # character in the following is a comment to the data in the file.

8.1 NURBS curve 2D

The following example file is `xcmodel/curves2d/c9p.db`. The .db extension identifies a NURBS entity.

```
FILENAME:c9p.db                                   # the curve file name
DEGREE                                    # introduces the curve degree
2                                                       # curve degree
N.C.P.                       # introduces the Number of Control Points
9                                            # number of control poins
N.KNOTS                             # introduces the Number of Knots
12                                                   # number of knots
COORD.C.P.(X,Y,W)         # introduces the CPs coord. and weight
2.775558e-17 3.000000e-01 1.000000e+00                # X Y W values
3.000000e-01 3.000000e-01 7.071070e-01
....
-3.000000e-01 -2.775558e-17 1.000000e+00
-3.000000e-01 3.000000e-01 7.071070e-01
2.775558e-17 3.000000e-01 1.000000e+00
KNOTS                                     # introduces the knot vector
0.000000e+00              # knot values in not decrescent order
0.000000e+00
....
1.000000e+00
1.000000e+00
```

8.2 Interpolation/Approximation points

The following example file is `xcmodel/curves2d/fgo.ip`. The .ip extension identifies a list of points.

```
FILENAME fgo.ip                                     # the file name
N.P. 18         # introduces the Number of Points and the value
-0.030516 -0.373239                         # X Y point coordinates
-0.098592 -0.295775
....
0.105634 -0.316901
0.037559 -0.373239
-0.021127 -0.377934
```

8.3 Control Points

The following example file is `xcmodel/curves2d/spiral.cp`. The .cp extension identifies a list of control points.

```
FILENAME spiral.cp                                  # the file name
N.C.P. 10          # introduces the Number of CPs and the value
0.000000 0.000000 1.000000    # X Y W 2d coordinates and weight
0.000000 -0.100000 1.105168
....
0.400000 -0.300000 7.178994
0.400000 0.400000 13.317328
-0.400000 0.400000 20.908569
-0.400000 -0.500000 26.867794
```

List of Figures

Bibliography

[BBB87] R.H.Bartels, J.C.Beatty, B.A.Barsky, An introduction to splines for use in computer graphics and geometric modelling, Morgan Kaufman publishers (1987).

[DEB78] C.deBoor, A practical guide to splines, Springer Verlag (1978).

[CASC98] G.Casciola, Metodi Numerici per la Grafica, Dispense C.d.L. Informatica, Università degli Studi di Bologna, (1998).

[FAR93] G.Farin, Curves and surfaces for CAGD: a practical guide, III Edition, Academic press (1993).

[HOLA93] J.Hoschek, D.Lasser, Fundamentals of Computer Aided Geometric Design, A.K.Peters (1993).

[PITI95] L.Piegl, W.Tiller, The NURBS book, Springer Verlag (1995).

[ROAD90] D.F.Rogers, J.A.Adams, Mathematical elements for computer graphics II, McGraw-Hill (1990).

[YAM88] F.Yamaguchi, Curves and Surfaces in Computer Aided Geometric Design, Springer (1988).

[XCMODEL00] G.Casciola, *xcmodel*: a system to model and render NURBS curves and surfaces; User's Guide - Version 1.0, Progetto MURST: "Analisi Numerica: Metodi e Software Matematico", Ferrara (2000), http://www.dm.unibo.it/~casciola/html/xcmodel.html