

UNIVERSITAT POLITECNICA DE CATALUNYA

Departament de Llenguatges i Sistemes Informàtics

Tesis Doctoral

**SISTEMA DE DISEÑO
DE LENTES PROGRESIVAS
ASISTIDO POR ORDENADOR**

Barcelona, Septiembre 1991

Presentada por:
Juan Carlos Dürsteler

Dirigida por:
Lluís Pérez Vidal

A mis padres, sin quienes esto no hubiera sido posible.

Agradecimientos

Sería difícil de concebir la realización del presente trabajo sin el concurso y la ayuda de innumerables personas que han contribuido al mismo, bien con el aporte de su experiencia personal o bien con el aporte del material de base sobre el que se fundamenta el presente trabajo. Enumerar todas las contribuciones sería arduo y, sin duda, olvidaríamos aportaciones importantes.

No obstante, deseo agradecer especial y sinceramente la colaboración de una serie de personas que me han prestado un inestimable soporte, y sin las cuales este trabajo no hubiera sido factible.

A Lluís Perez Vidal, que aceptó ser el director de la presente tesis, agradezco su constante apoyo y orientación, así como sus consejos, sugerencias y buen humor.

A Pere Brunet le debo la inspiración inicial que hizo viable la conversión de mi trabajo en una tesis y el agradecimiento del interés y ánimo que me ha ofrecido durante el desarrollo de la misma.

Agradezco a Patrick Mills y Mark Powell, de Quintek Ltd. en Inglaterra, la discusión y experto consejo de los aspectos más intrincados de la elaboración del sistema de redirección de mensajes y la configuración de la red de procesadores en paralelo que se ha utilizado como hardware.

Salvador Bosch, del Departamento de Óptica de la Universidad de Barcelona y Alvar Vinacua del Departamento de L.S.I. han tenido la amabilidad de leer el manuscrito y contribuir con su opinión a mejorar la claridad y comprensibilidad del mismo.

Finalmente agradezco especialmente a Ramón Soliva, director del Departamento de Investigación de Industrias de Óptica S.A. su comprensión y constante impulso de esta tesis, así como sus valiosas contribuciones al desarrollo y consecución de la misma.

Este trabajo se ha gestado a caballo entre el Departament de Llenguatges i Sistemes Informàtics de la U.P.C. (L.S.I.) y el Departamento de Investigación

iv

de la empresa Industrias de Óptica S.A. (INDO), en el seno del cual se han realizado materialmente los trabajos.

Juan Carlos Dürsteler

Barcelona, Septiembre de 1991

SISTEMA DE DISEÑO DE LENTES PROGRESIVAS ASISTIDO POR ORDENADOR

FE DE ERRATAS

- Capítulo 2, página 14, ecuaciones 2.10 y 2.11, deben leerse

$$D_m = \frac{D_1' + D_2'}{2} = (n' - n) \frac{k_1 + k_2}{2}$$

y

$$A_S = |D_1' - D_2'| = |n' - n| |k_1 - k_2|$$

- Capítulo 3, página 23, línea 16, donde dice "Davis[Dav81] patenta" debe decir "Davenport[Dav81] patenta".
- Capítulo 6, página 68, línea 1, donde dice "consistente" sustituir por "consiste".
- Capítulo 6, página 68, línea 5 empezando por el final, omitir GET_PSUM.
- Capítulo 6, página 73, línea 10, donde dice "se tarda la mismo" sustituir por "se tarda lo mismo".

Contenido

1	Introducción	1
1.1	El ordenador y el diseño de lentes	1
1.2	Objetivos y motivación	2
1.3	Organización	2
2	Fundamentos	5
2.1	Introducción	5
2.2	La Presbicia	5
2.3	Acomodación y Convergencia	7
2.4	Las Lentes Progresivas	9
2.5	Zonas principales	10
2.5.1	El meridiano principal	10
2.5.2	Teorema de Minkwitz	13
2.5.3	Zonas marginales	17
2.5.4	Zona de visión lejana	17
2.5.5	Zona de visión cercana	18
3	Estado del Arte	19
3.1	Antecedentes históricos	19
3.2	Desarrollos recientes	21
3.2.1	Visión estática vs. visión dinámica	21
3.2.2	La <i>segunda generación</i>	23
3.2.3	El progresivo de Kitchen y Rupp	23
3.2.4	El progresivo de Guilino y Barth	25
3.2.5	El progresivo de Fürter y Lahres	25
3.2.6	El progresivo de Winthrop	26
4	El Diseño de la Superficie	27
4.1	Introducción	27
4.1.1	Interpolación	27

4.1.2	Forma polinómica:	28
4.1.3	Forma de Lagrange	29
4.1.4	Polinomios de Bernstein	30
4.1.5	B-Splines	33
4.1.6	Curvas Racionales	35
4.1.7	B-Splines racionales	38
4.2	Superficies por producto tensorial	39
4.2.1	Superficies de B-Splines	41
4.2.2	B-Splines racionales	42
4.3	Geometría diferencial de la superficie	45
4.3.1	La primera forma fundamental	46
4.3.2	La segunda forma fundamental	47
4.3.3	La curvatura normal	47
4.3.4	Aberraciones superficiales	49
4.4	Diseño y control local	50
5	Metodología de Diseño	53
5.1	Introducción	53
5.1.1	Optimización y función de mérito.	54
5.2	Especificaciones ópticas	55
5.2.1	Variación de potencia	55
5.2.2	Umbilicalidad de la progresión	57
5.3	Variables de optimización	57
5.3.1	Distribución de aberraciones y pesos	59
5.3.2	Tratamiento gráfico de datos	62
6	Implementación en Paralelo	65
6.1	El algoritmo de Optimización	66
6.1.1	El método Simplex	66
6.2	Paralelización	70
6.2.1	Paradigmas de paralelización	70
6.2.2	Identificación de código paralelizable	72
6.2.3	Selección del nivel	72
6.3	Implementación	75
6.3.1	Topología de la red	75
6.3.2	Parte secuencial (Master)	76
6.3.3	Parte paralela (Workers)	77
6.3.4	Subsistemas de redirección (Routers)	78

7	Comparación de Interpolaciones	83
7.1	Contenido de las pruebas	83
7.2	Variantes de Optimización	88
7.3	Función de mérito	88
7.4	Realización	89
7.5	Resultados	91
7.6	Conclusiones	103
8	Conclusiones y Futuros Trabajos	105
8.1	Aportaciones	105
8.2	Conclusiones	106
8.3	Futuras líneas de trabajo	106
A	El Ojo Humano	109
A.1	Anatomía del ojo	109
A.2	El ojo teórico	112
B	Sistemas de Transputers	115
B.1	Configuración estática y dinámica	117
B.2	Programación y depuración	118
C	Obtención de Vértices de Control	121
C.1	Introducción	121
C.2	Un algoritmo	122

Lista de Figuras

2.1	Convergencia de la mirada	8
2.2	Lente Bifocal	9
2.3	Trayectoria del meridiano principal.	11
2.4	Variación de la potencia a lo largo del meridiano principal.	12
2.5	Estructura del haz emergente	13
2.6	Representación de la Superficie	16
2.7	Zonas del progresivo.	18
3.1	El progresivo de Aves.	20
3.2	El progresivo de Poullain–Cornet.	20
3.3	Distribución de conos y bastones en la retina.	22
3.4	Progresivo de Davenport.	24
3.5	Progresivo de Winthrop.	26
4.1	Curva y vector de nodos.	28
4.2	Oscilaciones con polinomios de Lagrange	30
4.3	Curva de Bézier	32
4.4	Bases Polinómicas	34
4.5	Curva B-Spline y ptos. de Bézier	35
4.6	Interpolación de cónicas mediante Bézier racional	37
4.7	B-Splines racionales	40
4.8	Superficies B-Spline	43
4.9	Representación de la esfera mediante NURBS	44
4.10	Indicatriz de Dupin.	49
5.1	Diseño del meridiano principal	56
5.2	Trayectoria del meridiano	56
5.3	Selección de nodos modificables	58
5.4	Función de pesos.	60
6.1	Operación de los routers	80

7.1	Diseño del pasillo	84
7.2	Distribución de zonas	85
7.3	Punto de partida ad. 1.	86
7.4	Punto de partida ad. 2.	86
7.5	Punto de partida ad. 3.	87
7.6	Variante 1. Astigmatismo adición 3.	95
7.7	Variante 1. Potencia adición 3.	96
7.8	Variante 1. Ejes adición 3.	96
7.9	Variante 2 Astigmatismo adición 3.	97
7.10	Variante 2. Potencia adición 3.	98
7.11	Variante 2. Ejes adición 3.	98
7.12	Variante 3 Astigmatismo adición 3.	99
7.13	Variante 3. Potencia adición 3.	100
7.14	Variante 3. Ejes adición 3.	100
7.15	Variante 4 Astigmatismo adición 3.	101
7.16	Variante 4. Potencia adición 3.	102
7.17	Variante 4. Ejes adición 3.	102
A.1	Sensibilidad espectral del ojo.	110
A.2	Anatomía del Ojo	111
B.1	Algunas topologías posibles con 9 transputers	116
B.2	Comunicación entre <i>routers</i> y aplicaciones	118
B.3	Velocidad de proceso vs. numero de transputers	119

Lista de Tablas

4.1	Interpolación de Curvas y Superficies	51
6.1	Efecto de la paralelización	74
7.1	Resultados	94
7.2	Tiempo de C.P.U.	94
7.3	Conclusiones	103
A.1	El ojo esquemático de Gullstrand	113
A.2	El ojo de Gullstrand (cont.)	114

Capítulo 1

Introducción

1.1 El ordenador y el diseño de lentes

El advenimiento del ordenador ha supuesto una revolución en muchos campos. En el caso de la óptica oftálmica esta revolución ha significado la ruptura de una constricción que la limitaba : la imposibilidad de diseñar y evaluar lentes que no fueran de geometría esférica y que, sin embargo, tuvieran propiedades ópticas adecuadas a la fisiología visual humana.

La aparición del Diseño Asistido por Ordenador (*C.A.D.*), y de la Fabricación Asistida por Ordenador (*C.A.M.*), han propiciado la automatización de muchos procesos industriales de diseño y fabricación, incluyendo la óptica de precisión. A diferencia de ésta, la oftálmica ha de corregir aberraciones con una lente simple, no con un sistema de lentes, lo que limita los grados de libertad disponibles.

La solución de esta dificultad consiste en modificar una o ambas superficies de la lente, haciéndolas *asféricas* y añadiéndoles así más grados de libertad.

En el transcurso de las tres últimas décadas han surgido diversos métodos de interpolación de superficies que han dado gran impulso al diseño de elementos tan diversos como carrocerías de automóviles, fuselajes y alas de aviones y cascos de barcos.

La dificultad y especificidad de este planteamiento aplicado a la óptica hacen que no exista ningún paquete comercial de diseño y fabricación en este área.

1.2 Objetivos y motivación

Dentro del campo de la óptica oftálmica, el desarrollo de *lentes esféricas*, y en particular de *multifocales progresivos*, ha resultado un campo particularmente activo en la década de los 80.

El advenimiento comercial de las *lentes esféricas* para corrección de defectos visuales data de 1984, aunque hace ya mucho tiempo que se han venido usando en óptica de precisión.

Por otra parte, el *multifocal progresivo*, cuyo desarrollo básico data de finales de la década de los 50, ha evolucionado enormemente y ha producido en los años ochenta una *tercera generación*, casi siempre basada en una re-formulación de la superficie con más grados de libertad.

La falta de herramientas de software adecuadas al *C.A.D.—C.A.M.* de este tipo de lentes y el avance en los métodos de interpolación de superficies propiciaron la realización de este trabajo, en el marco de un proyecto de investigación cuyo objetivo final era triple:

- Fundamentar y formular desde el punto de vista teórico el procedimiento y el tipo de superficie necesarios para poder diseñar lentes multifocales progresivas.
- Evaluar y comparar distintas formas de interpolar la superficie a fin de encontrar un método de interpolación eficiente y apropiado al diseño de este tipo de lentes.
- Desarrollar un sistema real de diseño y manufactura computerizada para la fabricación industrial de las mismas.

Cabe decir que este proyecto ha sido totalmente completado y que existen ya prototipos funcionales de las lentes, para diversas graduaciones y adiciones y se está preparando la fabricación y comercialización masiva de las mismas.

1.3 Organización

La estructura del presente trabajo es, obviando el presente capítulo, la siguiente:

- El **Capítulo 2** contiene una revisión de los conceptos básicos y fundamentos teóricos concernientes a las lentes progresivas y a la comprensión de sus características.
- El **Capítulo 3** se refiere a los antecedentes históricos y los desarrollos recientes en este campo, constituyendo, en conjunto, una revisión del *estado del arte* del mismo.

- El **Capítulo 4** pasa revista a los distintos tipos de superficie adecuados a los requerimientos de representación e interpolación de las magnitudes de interés en óptica oftálmica, y contiene un recordatorio de las magnitudes de la geometría diferencial de las superficies que son de interés para el cálculo de las características ópticas de las mismas.
- En el **Capítulo 5** se presenta la implementación de los algoritmos de optimización y cálculo de la función de mérito en paralelo, utilizando un sistema multiprocesador basado en Transputers. Se hace también aquí una discusión sobre la topología más adecuada al algoritmo y se identifica el paradigma de paralelismo al que más se aproxima.
- El **Capítulo 6** se refiere a la metodología original de diseño implementada en el sistema, entendida como el conjunto de facilidades interactivas que permiten al usuario realizar un diseño concreto partiendo simplemente de una idea previa, modificable sobre la marcha.
- En el **Capítulo 7** se realiza una comparación de los resultados de aplicar los mismos procesos de diseño y optimización a distintos formalismos de superficies, evaluando cual es el que mejor se adapta al diseño de lentes progresivas. En este capítulo, igual que en los dos anteriores, se describen las principales aportaciones de este trabajo.
- En el **Capítulo 8** se hace una síntesis de los trabajos realizados y se esbozan posibles líneas de trabajo futuro.
- El **Apéndice A** es un pequeño resumen del ojo humano y sus valores teóricos, incluido aquí por mor de completitud.
- El **Apéndice B** trata de las especificaciones del hardware (procesadores y tarjetas) y software (compilador y depurador simbólico de lenguaje C) que se han empleado para la programación en paralelo. Igualmente se tratan los aspectos concretos de la configuración de la red en el caso de los sistemas multiprocesador basados en Transputers que se han utilizado.
- El **apéndice C** resume la forma de calcular los *vértices de control* de una superficie B-Spline, que interpola una serie de puntos conocidos de una superficie.

Capítulo 2

Fundamentos

2.1 Introducción

Es inevitable, a la hora de abordar la situación con respecto a las lentes progresivas, fijar unos elementos mínimos de óptica, que, aunque conocidos, son de importancia central en este campo. El lector interesado en ampliar los rudimentos que aquí se repasan encontrará en [Cas85,Ros78,BW86], bibliografía abundante sobre la óptica en general.

En cuanto a la óptica fisiológica, cabe reseñar la obra de Legrand [Leg45], [Leg46], Obstfeld [Obs78] y Duke—Elder [Duk61], entre otros. Por otra parte, la bibliografía referente a lentes progresivas es relativamente escasa y muy dispersa, no existiendo ningún libro dedicado en concreto al tema. Como introducción a este campo, desde el punto de vista histórico, merece la pena mencionar los artículos de Sullivan [SF88] y Bennet [Ben61].

2.2 La Presbicia

El factor responsable del mecanismo de enfoque, o *acomodación*, en el ojo es el cristalino, pequeña lente deformable situada dentro del globo ocular (ver apéndice A). Con el transcurso de los años, las propiedades características del cristalino van degradándose. Aunque este fenómeno no está aún plenamente comprendido, parece que es debido principalmente [Leg45] a la progresiva pérdida de flexibilidad del mismo.

El cristalino permite el enfoque del sistema óptico del ojo, variando la curvatura de sus caras anterior y posterior según las necesidades de enfoque del sistema. De esta manera modifica su potencia en función de la distancia a que

se halle del *punto objeto* (punto al que se está mirando y cuya imagen se quiere hacer incidir en la retina). En *visión lejana*, mirando al infinito, el cristalino se halla en su posición de reposo. En *visión cercana*, mirando a un objeto situado en las proximidades, el cristalino se halla sometido a la tensión de los músculos ciliares [Leg45] que le obligan a aumentar su curvatura en función de la relación *acomodación convergencia* (ver sección 2.3).

Si consideramos un sistema óptico sobre el que incide un haz de rayos proveniente del infinito, llamaremos *foco* al lugar donde se reúnen los rayos una vez atravesado el sistema. Se llaman *planos principales* a dos planos *conjugados* — uno imagen del otro — cuyo *aumento lateral*, o relación entre los tamaños del objeto y su imagen por el sistema, es $\beta = 1$. Denominamos *punto principal* al punto de intersección de un plano principal con el eje del sistema. La *distancia focal* es, entonces, la distancia entre el punto principal y el foco.

Se define la *potencia* de una lente como el inverso de la distancia focal. Es corriente trabajar con *longitudes reducidas*, dividiendo la longitud por el índice de refracción n del substrato. Así, si f es una distancia focal, la focal reducida será $\frac{f}{n}$. Su inversa se denomina *poder refractor*, aunque algunos autores no hacen distinción entre éste y la potencia, definiéndola en términos reducidos.

Para una superficie S que separa medios de índices n y n' la potencia es [Cas85]

$$D = \frac{n'}{f'} = \frac{n' - n}{r} \quad (2.1)$$

Siendo r el radio de curvatura de la superficie refractante.

El convenio de signos para estas magnitudes es el siguiente: para las distancias en el eje se considera positivo el sentido de la luz incidente, que es de izquierda a derecha. Así, si denominamos *vértice* v a la intersección de la superficie con el eje óptico, el radio de curvatura r es positivo si el centro de curvatura de la superficie está a la *derecha* de v , ya que su origen se toma en el propio v .

Se llama lente *monofocal* a aquella cuya potencia óptica es única y uniforme en toda la lente. *Multifocal* es aquella lente cuya potencia no es igual en toda su superficie, pudiendo tener dos o más zonas discretas de potencia uniforme, o bien una zona de variación continua de la misma, generalmente entre dos localizaciones concretas.

El punto más alejado que es capaz de enfocar el sistema ocular en reposo se llama [Cas85] *punto remoto*, y el punto más cercano *punto próximo*, ambos medidos con respecto al ojo. Para un ojo normal el punto remoto está en el infinito y el punto próximo varía con la edad, en virtud de la pérdida de facultades del cristalino. La expresión:

$$A = \frac{1}{d_{pr}} - \frac{1}{d_{pp}} \quad (2.2)$$

se llama amplitud de acomodación, siendo d_{pr} la distancia del punto remoto al ojo y d_{pp} la distancia del punto próximo al ojo. Si d_{pr} y d_{pp} se expresan en metros A queda expresada en dioptrías. Se toma como promedio [Cas85] para el ojo normal de una persona adulta -250 mm. para la distancia de visión próxima, a la que se denomina *distancia mínima de visión distinta*.

La consecuencia de la inevitable pérdida de flexibilidad del cristalino es la reducción de la capacidad de enfoque del ojo y el subsiguiente alejamiento del punto próximo. Por tanto, se experimenta una dificultad creciente, a partir de alrededor de los 40 años, de enfocar los objetos a distancias próximas. Con el tiempo, la lectura y los trabajos que requieren el uso de la visión de cerca, resultan de difícil realización.

Este efecto, denominado *pérdida de acomodación* o *presbicia*, se puede corregir mediante el uso, en visión de cerca, de una lente convergente monofocal con la potencia necesaria para llevar el punto próximo a la distancia mínima de visión distinta. Esta potencia se denomina *adición*. Con esta solución, el usuario *amétrope*, necesitado de otra corrección en visión de lejos, requiere de dos pares de gafas, unas para visión de cerca y otras para visión de lejos.

2.3 Acomodación y Convergencia

Denominaremos [Leg45] *punto de fijación* a aquel punto del espacio objeto cuya imagen se forma sobre la fovea; *línea de visión* o *línea de fijación* a la recta que une el punto de fijación con el *centro de rotación* del ojo. Éste último es aquel punto fijo con respecto a la cabeza del observador, respecto del cual todos los movimientos del ojo se pueden considerar como rotaciones puras.

En visión binocular, la línea de visión de los ojos está determinada por la siguiente ley [Leg45]:

- *El punto de fijación es común a ambos ojos.*

Sea M el punto de fijación (Veáse la figura 2.1). Las dos líneas principales de visión se cortan en él y, por lo tanto, la imagen de M se forma en el centro de cada fovea.

Sean Q_1 y Q_2 los centros de rotación de ambos ojos. Se denomina *línea de base* a la recta que une Q_1 y Q_2 y *distancia interpupilar* P_1P_2 a la distancia entre los centros de las pupilas de entrada de cada uno de los dos ojos. Debido a que la fovea no está situada en el eje óptico del ojo [Duk61], la línea de visión

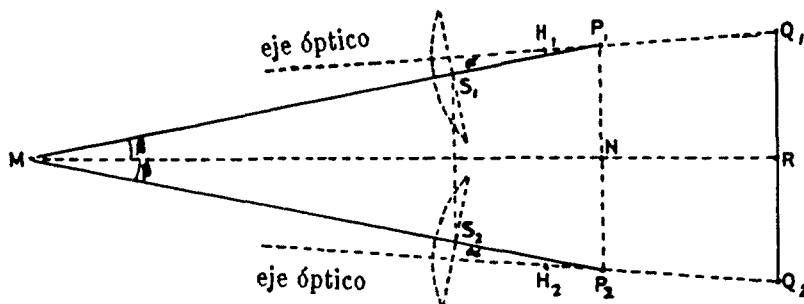


Figura 2.1: Convergencia de la mirada

y el eje óptico forman un ángulo α . La línea de base y el punto M forman el *plano de visión*. Supongamos que M se encuentra a una distancia \overline{MR} de la base y que R se encuentra en el punto medio de la misma. La *convergencia* se define como el inverso de \overline{MR} expresado en metros. La distancia interpupilar se puede calcular como

$$\overline{P_1P_2} = \overline{Q_1Q_2} - 2\overline{P_1Q_1} \sin(\beta - \alpha) \quad (2.3)$$

donde β es el *ángulo de excursión* que forma la línea de visión con \overline{MR} . La tangente de este ángulo es entonces

$$\tan(\beta) = \frac{\overline{NP_1}}{\overline{MN}} = \frac{\overline{P_1P_2}}{2\overline{MN}} = \frac{\overline{P_1P_2}}{2\overline{MR} - \overline{P_1Q_1} \cos(\beta - \alpha)} \quad (2.4)$$

De esta forma podemos conocer la distancia interpupilar en función de la convergencia expresada como $\frac{1}{\overline{MR}}$.

La convergencia, con ser necesaria para que la imagen caiga sobre la fovea, no es suficiente para tener una visión nítida. Se requiere, además, que el cristalino haga un esfuerzo de acomodación que enfoque la imagen sobre la fovea. Ello supone la existencia de una relación acomodación-convergencia. El valor de la acomodación se puede calcular como el inverso del segmento $\overline{MH_1}$, siendo H_1 el punto principal objeto del ojo. Fácilmente vemos que

$$\overline{MH_1} = \overline{MP_1} - \overline{H_1P_1} = \frac{\overline{P_1P_2}}{2 \sin \beta - \overline{H_1P_1}} \quad (2.5)$$

Sustituyendo $\overline{H_1P_1}$ por su valor teórico (1.44 mm sin acomodación y 0.87 mm. para una acomodación de 7 D.) y dando valores a la acomodación podremos encontrar que, para el ojo emétrepe, así como para el amétrepe corregido, la relación acomodación convergencia vale aproximadamente 1, esto es, para el ojo teórico, la acomodación es sensiblemente igual a la convergencia.

Para más información sobre el ojo veáse el apéndice A.

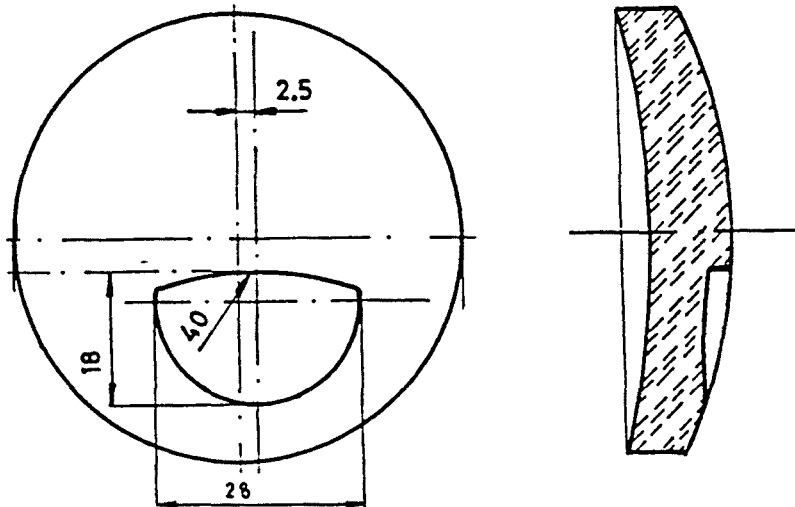


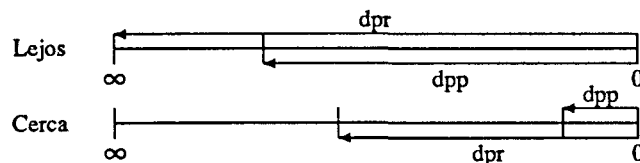
Figura 2.2: Lente Bifocal

2.4 Las Lentes Progresivas

El descubrimiento, en el siglo pasado [Leg45], de la relación existente entre la acomodación y la convergencia de los ojos hacia el objeto de la visión permitió el advenimiento de la lente *bifocal* (figura 2.2). Ésta consiste en una lente monofocal que lleva incorporada una lentilla de índice de refracción superior, delante de la posición que adoptan los ojos en visión cercana, dejando libre toda la parte restante de la lente para la visión lejana. Ello permite cambiar la potencia de la lente con solo converger la mirada hacia el objeto, eliminando así la necesidad del cambio de gafas, que se produce cuando se tiene un monofocal para visión de cerca y otro para la visión de lejos.

Sin embargo, el bifocal, ampliamente usado por la mayoría de la población presbita, adolece principalmente de tres defectos:

1. *Salto de imagen*, que se manifiesta en la transición entre la lentilla y el resto de la lente, al haber una discontinuidad en la potencia de la lente.
2. *Distancias que no cubre la adición*. Debido a que una lente convergente aproxima el punto remoto, entre éste y el infinito, el ojo no podrá ver nítidamente a través de la lentilla [Obs78].



Si la distancia d_{pr} al punto remoto de la lentilla es menor que la distancia d_{pp} al punto próximo del ojo en visión de lejos, entre éste y áquel quedará una zona en la que el usuario no será capaz de enfocar ni mirando por fuera ni por dentro de la lentilla. – Recordemos que no hay visión nítida más cerca del punto próximo ni más lejos del punto remoto –.

Este problema se soluciona parcialmente añadiendo una segunda lentilla de adición intermedia, incrementando así en uno los saltos de imagen. Este tipo de lente se denomina trifocal.

3. *Imagen de vejez.* El usuario es claramente identificable como presbita debido a la lentilla que porta en sus gafas.

La superación del bifocal como lente de prescripción para la presbicia constituye la lente progresiva. La idea es conceptualmente sencilla; se trata de construir una o ambas superficies de la lente con arreglo a la relación acomodación–convergencia.

A diferencia del bifocal, esto se consigue mediante una formulación adecuada de, habitualmente la superficie anterior de la lente. En ella los radios de curvatura locales de cada punto de la lente se escogen de manera que la potencia resultante sea, en cada punto de la misma, la necesaria para suplir la acomodación del cristalino cuando la línea de visión atraviesa ese punto de la lente.

Como veremos, esto no es rigurosamente factible en todos los puntos de la lente, si queremos evitar los efectos indeseables de las aberraciones ópticas que inevitablemente se producen en este tipo de lente. Por ello se establece un compromiso de diseño entre los requisitos ópticos ideales y la realidad de las geometrías posibles.

2.5 Zonas principales

2.5.1 El meridiano principal

Se llama *campo de visión* a la porción que es visible de una esfera centrada en el centro de rotación del ojo, de radio igual a la distancia al punto de fijación, con

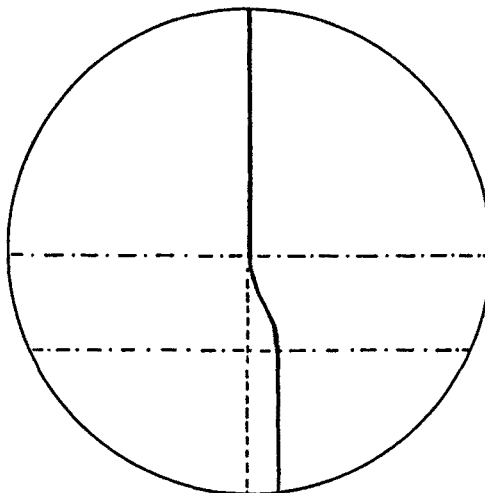


Figura 2.3: Trayectoria del meridiano principal.

la cabeza inmóvil (veáse [Leg45] [BZ46]). Dado que el ser humano es un animal terrestre, existe una notable disimetría en cuanto a la utilización de la visión de lejos y de la de cerca. La visión de lejos utiliza primordialmente el hemicampo visual superior, mientras que el hemicampo inferior se utiliza primordialmente en visión intermedia y próxima.

Esto supone que en la mitad superior de la superficie de la lente, usada para visión de lejos, necesitamos una potencia estable igual a la prescripción de lejos, que va aumentando *progresivamente* a medida que los ojos convergen y el punto de fijación se acerca al observador. Una vez se ha conseguido la máxima convergencia, la potencia ha de mantenerse en el valor de la adición. Puesto que un paciente emétrope no necesita corrección en visión de lejos, en este caso la potencia en la mitad superior será nula.

Denominaremos *meridiano principal* al lugar geométrico de las intersecciones sucesivas de la línea de visión con un plano imaginario situado en la posición de uso de la lente (*plano de la lente*), cuando el ojo pasa de mirar al infinito a mirar a un objeto situado a 30 cm. del individuo, en un plano vertical ortogonal al anterior. La trayectoria del meridiano principal, de capital importancia para las lentes progresivas, tiene la forma de la figura 2.3 adjunta, en virtud de la convergencia de los ojos.

Podemos ver que los ojos, a la par que descienden, convergen en sentido nasal. Los valores típicos para individuos normales indican que para un descenso de 10 mm. a partir de la posición de reposo del ojo, la convergencia sobre el plano de la lente es de 2.5 mm.

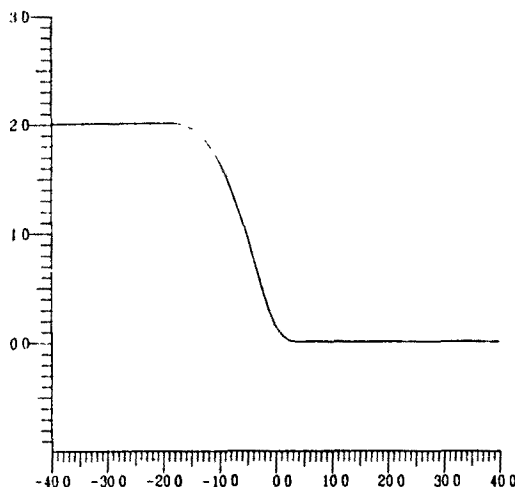


Figura 2.4: Variación de la potencia a lo largo del meridiano principal.

De lo dicho se desprende que, al menos a lo largo del meridiano principal, deberemos proporcionar una sucesión continua de potencias de manera que el ojo, en su posición normal de convergencia, atraviere una zona del meridiano principal dotada de la potencia requerida por la acomodación correspondiente a esa convergencia.

Igualmente nos interesa que el meridiano principal sea una zona de gran calidad de imagen, lo que supone limitar al máximo las aberraciones. Por ello, es una condición deseable que el meridiano principal esté constituido por puntos umbilicales, esto es, por puntos en los que los dos radios de curvatura principales de la superficie son iguales.

En la figura 2.4 podemos ver, en coordenadas cartesianas, una posible distribución de las potencias en función del espacio recorrido a lo largo del meridiano principal.

Sea $p(s)$ esta función y $f(s)$ la función que nos da el propio meridiano en función de s . Sea $r(s)$ el radio de curvatura de la curva en cada punto. Teniendo en cuenta la conocida fórmula

$$\frac{1}{r(s)} = \frac{f''(s)}{(1 + f'^2(s))^{3/2}} \quad (2.6)$$

en un punto dado pero $r(s) = (n' - n)/p(s)$ por lo que

$$\frac{p(s)}{n' - n} = \frac{f''(s)}{(1 + f'^2(s))^{3/2}} \quad (2.7)$$

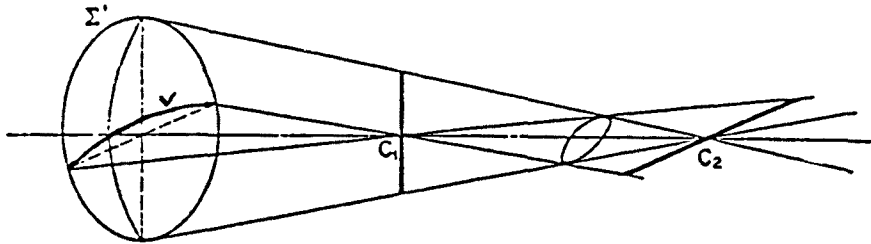


Figura 2.5: Estructura del haz emergente

Resolviendo la ecuación diferencial (típicamente mediante un método numérico como Runge–Kutta), obtenemos los valores de $f(s)$, que constituyen el meridiano principal.

2.5.2 Teorema de Minkwitz

Dada una superficie esférica en el sentido más amplio del término (i.e. no esférica), provista de *al menos un* plano de simetría, llamaremos *línea de progresión* (*Scheitellinie*) [Min63] a la intersección entre el plano de simetría y la propia superficie. En el caso de una lente progresiva, la línea de progresión coincide con el meridiano principal. Nos proponemos estudiar el comportamiento de la potencia y el astigmatismo en los alrededores del plano de simetría.

Consideremos un punto P sobre la superficie refractante S en el que incide un frente de onda Σ cuyo haz principal, proveniente de un punto situado en el infinito, es perpendicular a la superficie. En general [Cas85] después de la refracción de un frente de onda incidente esférico, el frente emergente Σ' no es esférico, lo que implica que posee dos radios de curvatura principales (ver Figura 2.5). Es decir hallaremos dos imágenes extremas del punto emisor y todas las intermedias, en vez de una.

Ello se traduce en que la imagen de un punto no es otro punto y tenemos un cierto espacio entre ambas focales donde obtendremos diferentes imágenes del punto. En el punto medio entre ambas encontramos el *círculo de menor confusión*, que proporciona la imagen más nítida de entre todas las consideradas.

Idealmente habría que calcular todas las magnitudes desde el punto de vista del trazado de rayos a través de la superficie, conociendo la aberración de onda, esto es, lo que se deforma el frente de onda una vez refractado. Ello, sin embargo, es notablemente costoso en tiempo de cálculo, por lo que se impone una simplificación.

La simplificación que se hace a la hora de evaluar las lentes progresivas consiste en trabajar con las curvaturas, no del frente de onda en cada punto, sino de la propia superficie. Esto, que en principio no es evidente, queda refrendado por el hecho de que los resultados de realizar una medición de las magnitudes superficiales y de las ópticas en una lente progresiva son comparables cualitativa y cuantitativamente dentro de las tolerancias de los instrumentos empleados (menores de 0.1 D.).

Para ello consideremos lo siguiente. Sea u la línea de curvatura correspondiente a la sección *meridiana* (también llamada *tangencial*) y v la correspondiente a su perpendicular, que llamaremos sección *sagital*. Los centros de curvatura correspondientes a los radios de curvatura principales se denominan *foco tangencial* y *foco sagital*, en razón a la línea de curvatura a la que pertenezcan. Denominaremos r_1 y r_2 a los radios de curvatura tangencial y sagital respectivamente y k_1 y k_2 a sus inversos (las curvaturas).

Sea f'_1 la focal sagital y f'_2 la focal tangencial. Sus poderes refractores serán

$$F'_1 = \frac{n}{f'_1}, \quad F'_2 = \frac{n}{f'_2} \quad (2.8)$$

Para una superficie que separa medios de índices n y n' la potencias son [Cas85]

$$D'_1 = \frac{n'}{f'_1} = \frac{n' - n}{r_1}, \quad D'_2 = \frac{n'}{f'_2} = \frac{n' - n}{r_2} \quad (2.9)$$

Denominaremos *potencia media* al promedio de las dos potencias antedichas

$$D_m = \frac{D'_1 + D'_2}{2} = (n - n') \frac{k_1 + k_2}{2} \quad (2.10)$$

que, como vemos, no es más que la *curvatura media* afectada del índice de refracción. Denominaremos *astigmatismo superficial* a la diferencia

$$A_S = |D'_1 - D'_2| = |n - n'| |k_1 - k_2| \quad (2.11)$$

Esta magnitud da cuenta de en que medida la imagen de un punto es otro punto, condición indispensable, dentro de la tolerancia del ojo humano, para ver con *nitidez*.

Para una superficie que separa medios de índices diferentes, sólo cuando $k_1 = k_2$, el astigmatismo A_S es nulo y por tanto la imagen es absolutamente

nítida a través del punto P . En estas circunstancias tenemos un punto umbilical en la superficie. Hay que reseñar no obstante que el ojo humano soporta, en promedio, astigmatismos de hasta 0,25 D. sin ninguna incomodidad.

Por otra parte, una esfera esta constituida íntegramente por puntos umbilicales y, sin embargo, no produce imágenes libres de astigmatismo *óptico*, aunque éste, en general no sea grande. La simplificación a que aludíamos antes, es tanto mejor cuanto más se aparta la superficie de una esfera. Es obvio que cuanto más astigmática sea la superficie tanto más astigmatismo tendrá el frente de onda que la atraviesa.

En estas condiciones [Min63] podemos enunciar el siguiente

Teorema 2.1 (Minkwitz) *En una superficie simétrica refractante S , cuya línea de progresión sea compuesta por puntos umbilicales se cumple:*

$$\lim_{\xi \rightarrow \pm 0} \frac{\partial A_S(s, \xi)}{\partial \xi} = \pm 2 \left| \frac{dD_m(s)}{ds} \right| \quad (2.12)$$

esto es, el gradiente de astigmatismo en dirección perpendicular a la línea de progresión es el doble del gradiente de potencia media a lo largo de la misma.

Supongamos que la línea de progresión es una línea de puntos umbilicales. Es fácil ver que en dirección perpendicular a la misma sólo pueden existir puntos de este tipo si la superficie es una esfera, cuyos puntos se caracterizan precisamente por ser todos umbilicales. Estudiemos el astigmatismo a lo largo de una sección perpendicular a la línea de progresión.

Expresemos la línea de progresión

$$\mathbf{x} = \mathbf{x}(s) \quad (2.13)$$

en función de la longitud de arco s . Sea el vector tangente \mathbf{t} , la normal principal \mathbf{n} y la binormal $\mathbf{b} = \mathbf{t} \times \mathbf{n}$. Podemos representar la superficie (Figura 2.6) que contiene a la línea de progresión como

$$\mathbf{F}(s, \xi) = \mathbf{x}(s) + \xi \mathbf{b} + \eta(s, \xi^2) \mathbf{n}(s) \quad , \quad [\eta(s, 0) = 0] \quad (2.14)$$

Dado $k = k(s)$, siendo una curva plana, según las fórmulas de Frenet-Serret (ver p.ej. [Pog87])

$$\dot{\mathbf{t}} = k(s) \mathbf{n} \quad \text{y} \quad \dot{\mathbf{n}} = -k(s) \mathbf{t} \quad (2.15)$$

Puesto que la línea de progresión ha de ser umbilical podemos desarrollar

$$\eta(s, \xi^2) = \frac{k(s)}{2} \xi^2 + a_4(s) \xi^4 + \dots \quad (2.16)$$

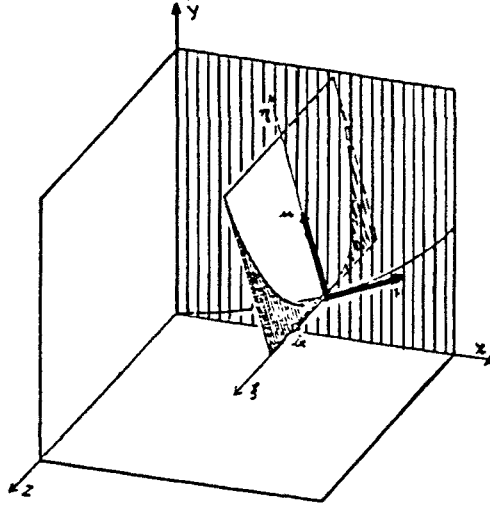


Figura 2.6: Representación de la Superficie

Dado que

$$|k_1 - k_2| = 2\sqrt{H^2 - K} \quad (2.17)$$

y desarrollando teniendo en cuenta las 2.15

$$H(s, \xi) = -\dot{k} + \left[\frac{3}{4}\dot{k}^3 - \frac{1}{4}\ddot{k} - 6a_4\right]\xi^2 + \dots \quad (2.18)$$

$$K(s, \xi) = k^2 + \left[\frac{1}{2}k\ddot{k} - \dot{k}^2 - \frac{3}{4}k^4 + 12a_4k\right]\xi^2 + \dots \quad (2.19)$$

de donde

$$H^2 - K = \dot{k}^2\xi^2 + \dots \quad (2.20)$$

y recordando 2.17

$$\lim_{\xi \rightarrow \pm 0} \frac{\partial |k_1 - k_2|}{\partial \xi} = \pm 2|\dot{k}| \quad (2.21)$$

Si multiplicamos 2.21 por $|n' - n|$

$$\lim_{\xi \rightarrow \pm 0} \frac{\partial |n' - n| |k_1 - k_2|}{\partial \xi} = \pm 2|n' - n| |\dot{k}| \quad (2.22)$$

y podemos escribir el contenido de la proposición

$$\lim_{\xi \rightarrow \pm 0} \frac{\partial A_S(s, \xi)}{\partial \xi} = \pm 2 \left| \frac{dD_m(s)}{ds} \right| \quad (2.23)$$

que podemos expresar finalmente como

$$A_S \approx 2|D_m(s)||\xi| \quad (2.24)$$

El teorema de Minkwitz es de gran importancia en el diseño de lentes progresivas, puesto que relaciona el gradiente de potencia en el meridiano principal con el gradiente de astigmatismo en dirección ortogonal al mismo. Ello se traduce en la aparición en la zona de progresión de un *pasillo* libre de astigmatismo, rodeado por zonas de astigmatismo creciente.

La anchura del pasillo, entendida como la de la zona por debajo de un cierto nivel de astigmatismo, es tanto menor cuanto mayor es el gradiente de potencia. Esto reduce el campo de visión cuando se está mirando a distancias intermedias.

Una consecuencia evidente de esta proposición es que resulta imposible construir una lente progresiva totalmente libre de aberraciones fuera del meridiano principal.

2.5.3 Zonas marginales

En las zonas colindantes con la de progresión se producen, debido al teorema de Minkwitz, distribuciones importantes de aberraciones, principalmente astigmatismo y error en potencia. Denominaremos *zonas marginales* (fig. 2.7) a aquellas en que el astigmatismo supere un umbral que representa el valor máximo tolerable por el ser humano.

El valor de este umbral varía con el individuo. Por ello Maitenaz [Mai74] adopta 0.3 D. para este valor mientras que Davis [Dav78] lo toma como 0.5 D. y Shinohara y Okazaki [SO85] aceptan hasta 1 D. (ver [SF88]). Nosotros utilizaremos 0.5 D. como el valor más probable, ya que es el estándar *de facto* en la representación de curvas isoastigmáticas entre diversos fabricantes y, por lo tanto, establece una buena referencia de delimitación de la zona marginal.

Debido a que el teorema de Minkwitz establece su validez en función de un límite cuando la distancia al meridiano principal tiende a 0, a suficiente distancia encontramos cantidades importantes de astigmatismo cuya cuantía y distribución ya no dependen exclusivamente de los gradientes en el meridiano principal, sino que están en función del área sobre la que se distribuyen estas aberraciones. Cuanto menor es el área de distribución tanto mayor es la magnitud de la aberración.

2.5.4 Zona de visión lejana

Se denomina *zona de visión lejana* (fig 2.7) a aquella zona, generalmente situada en la mitad superior de la lente, cuya potencia corresponde con la prescripción

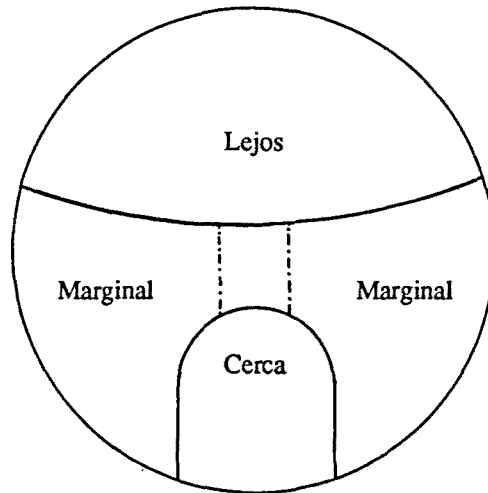


Figura 2.7: Zonas del progresivo.

para lejos del paciente y cuyo valor máximo de error en astigmatismo está por debajo del valor umbral ($\pm 0.5D$).

Esta zona puede ser esférica o asférica. Los diseños más recientes suelen ser asféricos, en razón de que ello les permite utilizar superficies de mayor radio de curvatura, más planas y más estéticas, sin comprometer las cualidades ópticas (ver [SA83] y [Atc84]), aunque son de mayor complejidad de realización.

La extensión de la misma varía sensiblemente de unos diseños a otros. Estudios recientes [Har88] establecen $\pm 40^\circ$ en ángulo visual como una amplitud necesaria para el confort, con $\pm 15^\circ$ para la zona de progresión y $\pm 20^\circ$ para la zona de cerca.

2.5.5 Zona de visión cercana

La *zona de visión cercana* o *zona de cerca* admite la misma definición que la de visión lejana salvo porque en ésta la potencia es sensiblemente la de la adición y porque suele ocupar un área en la mitad inferior de la lente. Se puede decir lo mismo acerca de su asfericidad. Su amplitud es variable según los diseños y suele obedecer al compromiso que establece el diseñador entre ésta y la extensión de las zonas marginales.

La potencia de la adición puede ser estable o bien puede aumentar aún, en función de la profundidad de campo que se quiera proporcionar al diseño y de la capacidad de acomodación residual del paciente.

Capítulo 3

Estado del Arte

3.1 Antecedentes históricos

La primera patente [SF88] de una lente progresiva fue concedida a Aves [Ave07] en 1907. Constaba (ver figura 3.1) de una superficie anterior cilíndrica cuya potencia aumentaba continuamente de A hasta B , y una superficie posterior que era una porción de cono orientada verticalmente, con la punta del cono hacia arriba. La lente resultante era biconvexa, sólo tenía utilidad para los presbíteros hipermétropes y fue de un uso muy limitado.

En 1914, Gowlland [Gow14] propone un progresivo combinando una superficie anterior toroidal y una superficie posterior paraboloïdal. Según [SF88] esta fue, años más tarde en 1928, la primera lente progresiva comercializada. Al estar constituida por superficies de revolución resultaba relativamente sencilla de fabricar.

Poullain y Cornet [PC20] patentaron en 1920 una lente progresiva con una sola superficie, dotada de una línea umbilical. Bennet [Ben61] acuñó para esta superficie el término “Trompa de elefante” por su forma característica. Esta lente era simétrica y la potencia no se estabilizaba en ningún momento. Por tanto, en toda su extensión presentaba una astigmatismo concordante con el teorema de Minkwitz.

La mejora respecto del de Aves consistía en que, al ser una sola superficie la progresiva, en la otra se podía tallar la esfera requerida por la prescripción del paciente, incluyendo un cilindro si fuese necesario. Así se podían beneficiar miopes, hipermétropes y astigmatas. Su realización, no obstante, era muy difícil para la época.

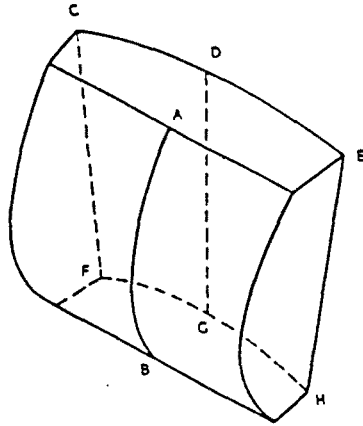


Figura 3.1: El progresivo de Aves.

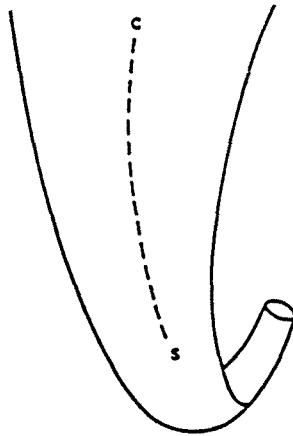


Figura 3.2: El progresivo de Poullain-Cornet.

Entre 1907 y 1953, en que B. Maitenaz patentó su primer progresivo, aún en estudio, se sucedieron las patentes y experimentos en este campo. Ninguna de ellas devino en una lente comercial con éxito. Los artículos de Bennet [Ben61] contienen amplia información sobre este período, que omitiremos para no extendernos en exceso. [SF88] es también una fuente histórica de interés.

Los trabajos de Bernard Maitenaz ([Cre53], [Mai62], [Mai66]) en la década de los 50 sentaron las bases del primer progresivo que tendría éxito comercial, bajo el nombre de "Varilux".

La superficie progresiva se presentaba en la cara anterior de la lente y, a diferencia de sus predecesores, constaba de tres zonas bien diferenciadas. Una zona de lejos y una zona de cerca totalmente esféricas y de gran amplitud, y una tercera zona de transición con un corredor umbilical, cada una de cuyas secciones ortogonales era un arco de circunferencia de radio igual al de la línea de progresión en el punto de intersección de ambas. La zona de visión lejana era en esta lente sensiblemente la mayor de las tres.

La disposición geométrica de la lente y su formulación matemática producían en esta lente zonas de visión estable y nítida en lejos y cerca pero a costa de un pasillo estrecho y con grandes magnitudes de aberraciones ópticas en sus zonas marginales.

3.2 Desarrollos recientes

El intento de resolver los problemas de que adolecía Varilux condujo a la aparición de una segunda generación de lentes progresivas, comprometida principalmente en la reducción de las magnitudes y gradientes de astigmatismo y distorsión en las áreas marginales de la lente. Ello se conseguía mediante la distribución de las aberraciones sobre zonas más amplias de la lente, esto es, limitando las zonas de lejos y de cerca al introducir en ellas cantidades controladas de aberraciones.

3.2.1 Visión estática vs. visión dinámica

Maitenaz [SF88] introduce en 1972 el concepto de *visión estática* y *visión dinámica*. La visión dinámica ocurriría cuando el objeto al que se mira se mueve en relación a la lente progresiva, atravesando porciones de diferente aumento de la lente y produciendo una sensación de ondulación en el usuario. Por contra, la visión estática se produce cuando se mira un objeto sin alterar su posición relativa respecto de la lente.

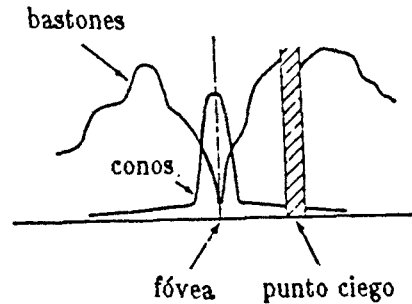


Figura 3.3: Distribución de conos y bastones en la retina.

Resultaba entonces que el primer diseño de Maitenaz favorecía la visión estática, con sus grandes zonas esféricas, en detrimento de la visión dinámica, muy empobrecida en las zonas marginales.

En realidad éstos no son conceptos nuevos. La *visión foveal* utiliza la información que proporcionan los aproximadamente 34.000 conos [Leg46] que se distribuyen en un círculo de 250μ de radio. La visión foveal subtiende un ángulo de $1^{\circ}40'$. La concentración de conos en la fovea es la más alta de la retina [Cas85] y disminuye rápidamente hacia su periferia. Igualmente (ver figura 3.3) los bastones, ausentes por completo en la fovea, crecen en número a partir de ella para alcanzar un máximo de $160.000mm^{-1}$ a unos $6mm$, de la misma, disminuyendo hasta unos $50.000mm^{-1}$ en el borde de la retina [Cas85].

La visión foveal suministra información de gran calidad, pero en un campo restringido. Podemos relacionar fácilmente ésta con la visión estática.

La visión periférica, fuera de la fovea abarca todo el campo de visión y, si bien no proporciona imagen nítida del objeto al que miramos, suministra información del entorno y de nuestra posición respecto a él, con lo que sería importante en visión dinámica.

3.2.2 La segunda generación

La siguiente realización de Bernard Maitenaz consistió en añadir un grado más de libertad a la formulación de su superficie. Esto se consiguió convirtiendo las secciones ortogonales al meridiano principal en secciones cónicas en vez de circulares. La excentricidad variaba continuamente desde la zona de lejos a la de cerca.

Para una misma línea de progresión, distintas formulaciones de la evolución de la excentricidad producían distintas distribuciones de aberraciones. La optimización de este parámetro extra mejoró notablemente el comportamiento de este tipo de lente que se denominó comercialmente "Varilux II".

A raíz del éxito comercial de Varilux, otras empresas desarrollaron sus propios diseños de lente progresiva diferenciándose básicamente por su formulación de la superficie y por la forma de entender el equilibrio entre visión dinámica y estática.

De este modo aparecen progresivos para visión foveal con amplias zonas de cerca y de lejos, concentrando altos valores de aberraciones en las zonas marginales. Davis [Dav81] patenta en 1981 una lente con zonas marginales muy restringidas. En las zonas de lejos y de cerca las características ópticas son aceptables. Sin embargo la concentración de aberraciones en las zonas marginales, particularmente la distorsión (figura 3.4), se elevan a valores muy altos, inaceptables [SF88] para una buena percepción visual. La formulación matemática se basa en consideraciones geométricas e involucra funciones sinusoidales y cosinusoidales.

3.2.3 El progresivo de Kitchen y Rupp

Kitchen y Rupp [KR81] patentan el mismo año otra lente de similares características pero de mayor suavidad en la transición de la zona de cerca a las marginales. Su formulación matemática describe la superficie como una función

$$z_m(x, y) = z_{m_0}(x) + \frac{1}{B(x)} \left(\sqrt{\rho^2(x) + B(x)y^2} - \rho(x) \right) - \frac{1}{n(x)} C(x)y^n(x) \quad (3.1)$$

y

$$z_n(x, y) = z_{n_0}(x) + R_p(x) - \sqrt{R_p(x)^2 - y^2} - \frac{1}{n(x)} C(x)y^n(x) \quad (3.2)$$

donde $z_{m_0}(x)$ es la función que define la geometría del meridiano principal. $B(x)$ es la función de evolución de la excentricidad de la sección cónica de la función, cambia continuamente ésta de modo que al principio del corredor es un círculo, pasa por diversas formas elípticas y termina de nuevo en excentricidad 0 en la mitad de la zona de cerca. $\rho(x)$ es la componente sobre el plano $y-z$

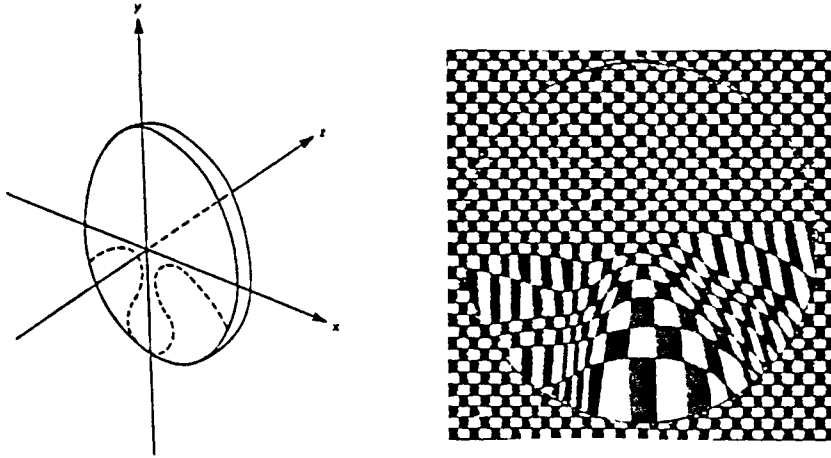


Figura 3.4: Progresivo de Davenport. Esquema de la distribución de zonas libres de aberraciones (izquierda) e imagen de una malla de cuadrados vista a través de la lente, mostrando la distorsión causada (derecha). (Dibujo tomado de la patente)

del radio de curvatura local $R_m(x)$ de la línea umbilical $z_{m0}(x)$, $C(x)$ es un parámetro que suprime la contribución del término $\frac{1}{n(x)}C(x)y^n(x)$ en la “parte central de la curva”, $n(x)$ es el parámetro que permite enlazar con regularidad $z_m(x, y)$ con $z_n(x, y)$, hallándose su valor mediante la resolución de

$$\frac{dz_m(x, y)}{dy} = \frac{dz_n(x, y)}{dy} \quad (3.3)$$

$$z_{n0}(x) = \frac{1}{B_L} \left(\sqrt{R_L^2 + B_L(x)^2} - R_L \right) \quad (3.4)$$

$$R_p(x) = R(x)(1 - A(x)) \quad (3.5)$$

Este tipo de progresivos son más adecuados a usuarios tolerantes a anomalías severas en la visión periférica. Por ello su éxito comercial ha sido relativo y la población potencial de usuarios es comparativamente menor que la de los posibles portadores de progresivos *multipropósito*, en los que el compromiso entre visión periférica y visión foveal es más equilibrado.

Entre estos últimos se cuenta, claramente Varilux II, pero hay una gran cantidad de patentes dentro de este campo. Sin embargo el número de fabricantes es mucho menor que el de patentes y, en éstas, es rara la que no se pierde en generalidades evitando revelar detalles clave para la comprensión de su realización.

Por ello mencionaremos aquí solamente aquellas que aportan información verificable o bien las que han dado lugar a productos que hemos podido medir en el laboratorio.

3.2.4 El progresivo de Guilino y Barth

Destaca por su claridad la patente de Guilino y Barth [Rod79] para Rodenstock, en la que explican detalladamente el concepto de la lente que se comercializó con el nombre de "Progressiv R". La superficie de esta lente se describe mediante funciones periódicas, que, en coordenadas cilíndricas (y, ρ, φ) se expresan

$$\rho(\varphi, y) = \sum_{n=0}^{\infty} a_n(y) \cos[nk(y)\varphi] \quad (3.6)$$

donde $\rho(0, y) = f(y)$ es el meridiano principal umbilical y $a_n(y) = 0$ para $n \geq 2$ valiendo

$$\left. \begin{aligned} a_0(y) &= f(y) + \frac{g(y)}{k^2(y)} \\ a_1(y) &= -\frac{g(y)}{k^2(y)} \\ g(y) &= f^2(y) + \frac{f^2(y)f''(y)}{1+f'^2(y)} \end{aligned} \right\} \quad (3.7)$$

$k(y)$ es una función que modula la longitud de onda y tiene la siguiente forma [Rod79]

$$k(y) = 3 + \frac{7}{1 + e^{-3(y+1.8)}} \quad (3.8)$$

cuando y está expresada en cm., $f(y)$ se calcula resolviendo la fórmula 2.6 usando como valor de la progresión de la potencia en el meridiano la siguiente ecuación

$$D = A - \frac{A}{1 + e^{-c(y+d)^{30}}} \quad (3.9)$$

donde D es la potencia, A es la adición (ambas en dioptrías) y c y d son dos parámetros que se hallan tabulados en la patente .

3.2.5 El progresivo de Fürter y Lahres

Fürter y Lahres [FL81], [FL86], a la sazón trabajando para Zeiss, producen varias patentes entre 1980 y 1986 en las que exponen variantes de lentes progresivas, que se aceptan como las de una familia de lentes cuyo primogénito ha sido comercializada como "Gradal HS". Las patentes son notablemente vagas y es difícil asegurar que definan en realidad este producto. De entre todas las lentes que hemos podido medir, ésta destaca por el bajo nivel de astigmatismo máximo en las zonas marginales, que no es superado por ningún otro progresivo que

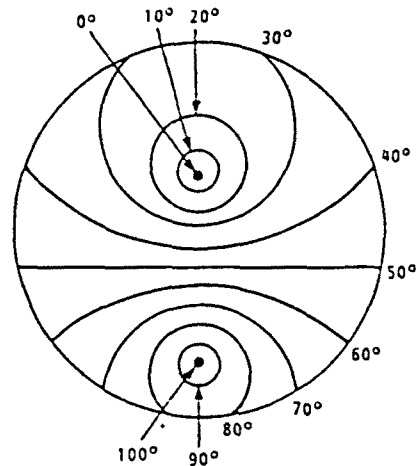


Figura 3.5: Progresivo de Winthrop. Distribuye la potencia por analogía con el principio de Dirichlet. Cada curva representa un mismo nivel de temperatura en una placa de cobre de igual diámetro que la lente a diseñar. La potencia se distribuye de manera análoga.

distribuya las aberraciones en una área comparable. Según [Fur85] Gradal HS posee una superficie diseñada mediante Splines, sin embargo en las patentes no se hace mención explícita a ninguna formulación, ni se menciona qué tipo de Splines se utilizan.

3.2.6 El progresivo de Winthrop

En 1985 Winthrop [Win85] patenta una lente cuya distribución de potencias se rige por el principio de Dirichlet. Para ello establece una similitud entre la distribución de temperaturas en un disco de cobre del diámetro de la lente, con dos puntos, coincidentes con el centro de visión de lejos y de cerca, sometidos a una diferencia de temperaturas de 100 grados. Winthrop posiciona las líneas de igual potencia análogamente a las isoterma.

Esta disposición efectivamente produce una distribución muy suave de aberraciones con valores absolutos muy por debajo de los de sus competidores. Sin embargo, estas aberraciones se halla distribuidas por virtualmente toda la lente, dejando unas zonas "limpias" de astigmatismo muy exiguas. Esta lente está en el otro extremo del espectro: es apropiada para individuos con visión predominante periférica, a los que no incomoda tener baja calidad en visión foveal.

Capítulo 4

El Diseño de la Superficie

4.1 Introducción

Las cualidades ópticas de la superficie a diseñar imponen ciertas restricciones al formalismo matemático que la haya de representar. Concretamente:

1. *Hay que evitar saltos de imagen.* Estos son producidos por discontinuidades en la potencia de la lente que, como vimos, está relacionada con la curvatura local de la superficie. Así, pues, habremos de imponer continuidad en las curvaturas y, por ende, en las segundas derivadas de la superficie con las cuales se calculan las formas fundamentales que nos proporcionan las curvaturas y ejes principales.

Esto impone que la superficie habrá de ser *como mínimo* de clase C^2 .

2. *Suavidad.* Una superficie ondulada, con cambios de potencia notables entre puntos cercanos es muy molesta, puesto que obliga al ojo a acomodar continuamente al pasar de un punto a otro. Ello, pues, desaconseja la utilización de formulaciones que sean susceptibles de producir oscilaciones indeseadas en la superficie a diseñar.

4.1.1 Interpolación

Para elegir un formalismo de superficie adecuado a nuestras necesidades, vale la pena pasar brevemente revista a las posibilidades existentes. Por mor de la claridad veremos primeramente los esquemas de interpolación para curvas y, después, su extensión a superficies. Seguiremos, en cuanto a estructura, las líneas maestras del excelente *survey* sobre curvas y superficies de Böhm, Farin y Kahmann [BFK84].

El problema es el siguiente : dados $n + 1$ puntos $p_0(u), p_1(u), \dots, p_n(u)$ encontrar una curva que los interpole.

Una idea central del diseño asistido por ordenador es la subdivisión de curvas y/o superficies en trozos más pequeños, susceptibles de ser tratados individualmente, pero que enlazan unos con otros para formar un todo coherente.

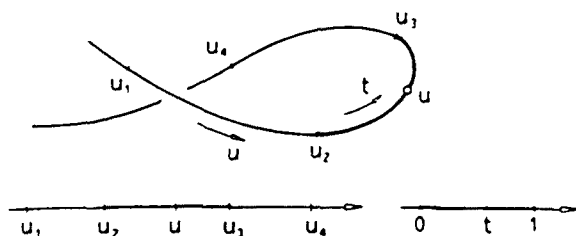


Figura 4.1: Curva y vector de nodos.

Sea una curva $x(u)$ interpretada como la imagen de una parte del eje u (su dominio) mediante una transformación en 2D ó 3D. Una segmentación [BFK84] de una curva en trozos (*patches*) corresponde a una partición de su dominio mediante *nodos* $u_0 < u_1 < \dots < u_n$, el conjunto de los cuales denominamos *vector de nodos* $U\{u_0 \dots u_n\}$ (ver figura 4.1). Es usual la introducción de un parámetro local t que en el intervalo $[u_k, u_{k+1}]$ permite escribir [BFK84]

$$u = u_k(1 - t) + u_{k+1}t \quad (4.1)$$

de forma que este segmento es la imagen del intervalo $[0, 1]$ del eje t .

4.1.2 Forma polinómica:

Podemos representar una curva mediante un polinomio de grado n de la forma

$$x(u) = a_0 + a_1 t + \dots + a_n t^n \quad (4.2)$$

Para determinar los coeficientes a_i necesitamos $n + 1$ puntos.

Propiedades:

- La evaluación de este tipo de interpolación es directa y computacionalmente eficiente. No obstante,
- los coeficientes a_i carecen de interpretación geométrica .

- El principal problema de este tipo de interpolación [dB78] es que si la función no es bien comportada en algún lugar del intervalo, la aproximación es pobre en todas partes. Esta dependencia global en propiedades locales se puede evitar segmentando el dominio.
- Igualmente un espaciado uniforme de los datos puede tener efectos negativos, que se subsanan si podemos escoger los puntos distribuidos como puntos de Chebyshev (veáse [dB78]).

4.1.3 Forma de Lagrange

Podemos expresar la curva $x(u)$ de la forma

$$x(u) = p_0 L_0^n + p_1 L_1^n + \dots + p_n L_n^n \quad (4.3)$$

donde L_i^n son los polinomios de Lagrange. Éstos se construyen de la siguiente forma:

$$L_i^n(u_k) = \frac{F_i^n(u)}{F_i^n(u_i)} \quad (4.4)$$

donde

$$F_i^n(u) = \frac{\prod_{j=1}^n (u - u_j)}{u - u_i} \quad (4.5)$$

y la interpretación de los mismos es la siguiente:

$$L_i^n(u_k) = \delta_{i,k} = \begin{cases} 1 & \text{si } i = k \\ 0 & \text{si } i \neq k \end{cases} \quad (4.6)$$

siendo δ la delta de Kronecker.

Propiedades:

- p_i tienen interpretación geométrica, ya que son los propios puntos a interpolar.
- $\sum_i L_i^n(u) \equiv 1$, esto es, constuyen una partición de la unidad.
- Como contrapartida, es fácil producir oscilaciones con polinomios de grado elevado [dB78]

Podemos ver esta última propiedad en la figura 4.2. Los asteriscos son los puntos a interpolar. Se aprecia claramente las oscilaciones, que para este caso son máximas en los puntos extremos. La interpolación en este caso presenta oscilaciones muy violentas que hacen al método inválido para esta distribución de puntos.

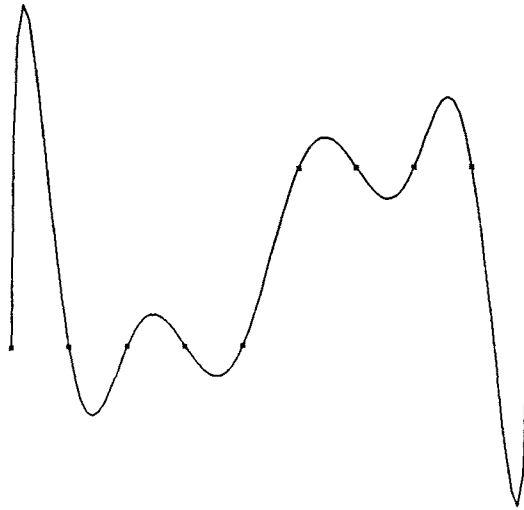


Figura 4.2: Oscilaciones con polinomios de Lagrange

4.1.4 Polinomios de Bernstein

Las dificultades antes mencionadas para las formulaciones de Lagrange y polinómica elemental se pueden evitar mediante el uso de los denominados *polinomios de Bernstein*, cuya forma es la siguiente

$$x(u) = b_0 B_0^n(t) + b_1 B_1^n(t) + \dots + b_n B_n^n(t)$$

más abreviadamente

$$x(u) = \sum_{i=0}^n b_i B_i^n(t) \quad (4.7)$$

que se pueden calcular, para grado n como

$$B_i^n(t) = \binom{n}{i} (1-t)^{n-i} t^i, \quad i = 0, 1, \dots, n \quad (4.8)$$

b_0, b_1, \dots, b_n son los llamados *puntos de Bézier*. Estos puntos *no* son puntos por los que pasa la curva sino que, modificando su posición, se controla la forma que ofrece la misma.

Propiedades:

1. $\sum_{i=0}^n B_i^n(t) \equiv 1$

2. $B_i^n(t) \geq 0$ para $t \in [0, 1]$.
3. $B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t)$
4. $B_i^n(t) = 0$ para $i < 0, i > n$
5. Ningún plano tiene más intersecciones con el polígono de control que con la propia curva. Propiedad conocida como *variation diminishing property* que podemos traducir por *variación disminuida*, haciendo referencia a que, en virtud de esta propiedad la curva nunca oscila violentamente fuera de los puntos de control que la definen. [NR83]

La propiedad 1 indica que los polinomios de Bernstein son una partición de la unidad, lo que implica que la relación entre la curva y sus puntos de Bézier es invariante bajo transformación afín.

Según 3, los polinomios se pueden calcular recursivamente. (Ver algoritmo de deCasteljau en [BFK84]), y, finalmente, 4 nos indica que los polinomios de Bernstein son nulos fuera del dominio.

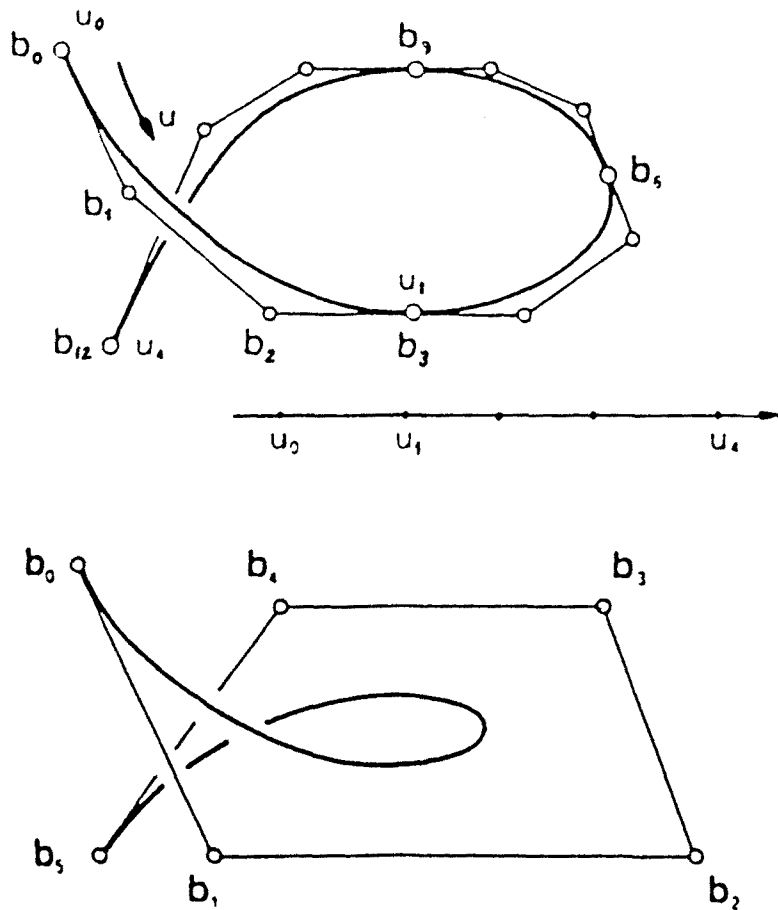


Figura 4.3: Curva de Bézier (de [BFK84])

4.1.5 B-Splines

La interpolación mediante el esquema de Bézier tiene una característica importante consistente en que la modificación de un punto de Bézier influye en toda la extensión a la curva interpolada. A esta propiedad se denomina comúnmente *control global*.

Muchas situaciones aconsejan, sin embargo que la modificación de los parámetros de control de la curva tenga *control local*, esto es que su cambio afecte sólo a un entorno limitado alrededor del mismo.

Dentro de esta categoría se incluyen los B-Splines, que podemos representar como

$$\mathbf{x}(u) = \sum_i \mathbf{d}_i N_i^n(u) \quad (4.9)$$

donde \mathbf{d}_i son los puntos de control o de *de Boor* y $N_i^n(u)$ son B-Splines normalizados, definidos en el vector de nodos $U\{u_0, \dots, u_n\}$ con $\dots < u_0 < u_1 < u_2 < \dots$

Propiedades:

1. $\sum_i N_i^n(u) \equiv 1$
2. $N_i^n(u) \geq 0$
3. $N_i^n(u) = 0$ si $u \notin [u_i, u_{i+n+1}]$
4. $N_i^n(u)$ es $(n - 1)$ veces continuamente diferenciable

$$N_i^n(u) = (u - u_i) \frac{N_i^{n-1}(u)}{u_{i+n} - u_i} + (u_{i+n+1} - u) \frac{N_{i+1}^{n-1}(u)}{u_{i+n+1} - u_{i+1}},$$

5. donde $N_i^0(u) = \begin{cases} 1, & \text{si } u \in [u_i, u_{i+1}], \\ 0, & \text{si } u \notin [u_i, u_{i+1}] \end{cases}$

6. B-Splines también tiene la propiedad de variación disminuida.

Las bases de los B-Splines constituyen una partición de la unidad cuyo significado en cuanto a invariancia bajo transformaciones afines es análogo al caso de Bézier. Igualmente son definidas positivas, esto es, el segmento de curva para $u \in [u_i, u_{i+1}]$ está dentro del *convex hull* de los puntos \mathbf{b}_i .

El apartado 3 hace referencia al control local. Puesto que la base de polinomios $N_i^n(u)$ es nula fuera del intervalo $[u_i, u_{i+n+1}]$, la influencia del punto \mathbf{d}_i abarcará solo desde i hasta $i + n + 1$, esto es solo alcanza a $n + 1$ intervalos, siendo n el grado de los polinomios de la base.

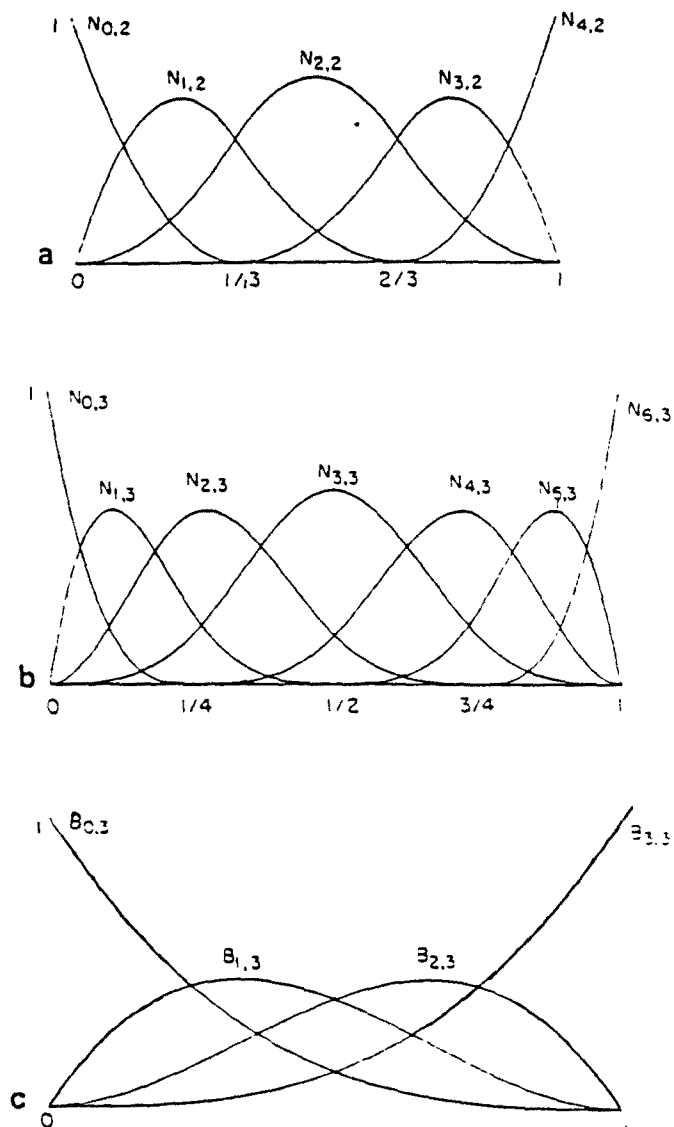


Figura 4.4: (a) Base B-Spline cuadrático con vector de nodos $\{0, 0, 0, 1/3, 2/3, 1, 1, 1\}$ (b) Base B-Spline cúbico sobre $\{0, 0, 0, 0, 1/4, 1/2, 3/4, 1, 1, 1, 1\}$. (c) Polinomios de Bernstein cúbicos sobre $\{0, 0, 0, 0, 1, 1, 1, 1\}$ ([PT87].)

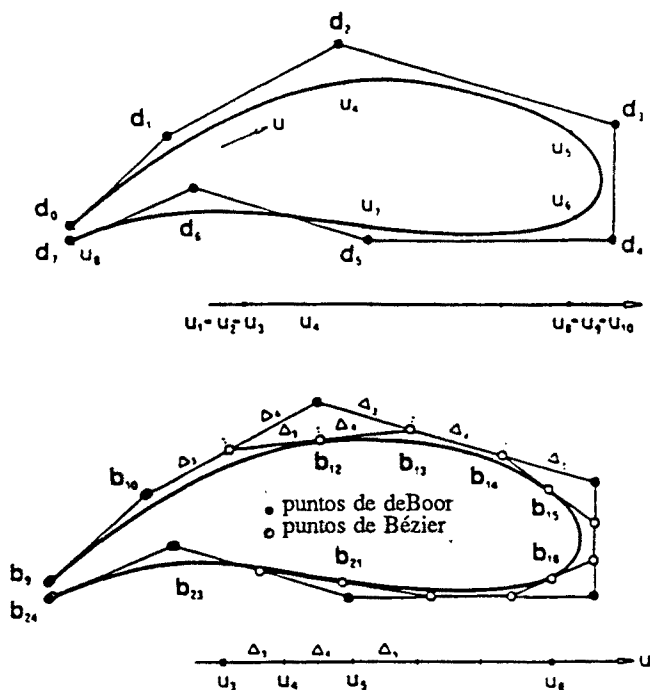


Figura 4.5: Curva B-Spline y pto. de Bézier (de [BFK84]).

Los B-Splines se pueden calcular también recursivamente, según el apartado 5. Si coinciden k nodos $u_j = \dots = u_{j+k-1}$ la curva resulta solo C^{n-k} continuamente diferenciable en u_j , lo que implica que cabe asegurar en todo momento que $k \leq n + 1$ para que la fórmula de recursión siga siendo válida.

Se puede ver [BFK84] que, escogiendo adecuadamente los puntos de deBoor, se puede interpolar con B-Splines una curva de Bézier, lo que implica que éstas son un caso particular de aquellas.

4.1.6 Curvas Racionales

Es usual la utilización de coordenadas homogéneas para representar puntos en el espacio tridimensional en términos de puntos en el espacio tetradimensional [FvD78] [RA76] [NR83]. Cualquier punto *no infinito* del espacio 4D se puede representar mediante la tétada $(\beta x, \beta y, \beta z, \beta)$ siendo $\beta > 0$.

Un punto cartesiano representado por la tríada (x, y, z) se puede representar como un punto en 4D con β normalizado a 1, esto es, por $(x, y, z, 1)$. Esto se

puede interpretar [PT87] como la aplicación de una proyección perspectiva de la forma

$$\mathbf{H}\{(\beta x, \beta y, \beta z, \beta)\} = \begin{cases} (\beta x/\beta, \beta y/\beta, \beta z/\beta) & \text{si } \beta \neq 0 \\ \text{Punto en} & \\ \text{el infinito} & \text{si } \beta = 0 \end{cases}$$

Siguiendo a [PT87] denotaremos los puntos en el espacio 4D como P^β y, por tanto, $P = \mathbf{H}\{P^\beta\}$

Una curva racional, en términos de Polinomios de una cierta base, es la proyección sobre el espacio 3D de una curva no racional en 4D. En función de, por ejemplo, polinomios de Bernstein se expresa de la forma siguiente:

$$\mathbf{x}(u) = \frac{\beta \mathbf{x}(t)}{\beta(t)} = \frac{\beta_0 \mathbf{b}_0 B_0^n(t) + \dots + \beta_n \mathbf{b}_n B_n^n(t)}{\beta_0 B_0^n(t) + \dots + \beta_n B_n^n(t)}$$

que podemos escribir como

$$\mathbf{x}(u) = \frac{\sum_{i=0}^n \beta_i \mathbf{b}_i B_i^n(t)}{\sum_{i=0}^n \beta_i B_i^n(t)} \quad (4.10)$$

donde β_i es el peso correspondiente al punto de Bézier \mathbf{b}_i , $t \in [0, 1]$ y $B_i^n(t)$ son los polinomios de Bernstein de grado n .

Podemos reescribir [Pie87] la ecuación anterior de la forma

$$\mathbf{x}(u) = \sum_{i=0}^n \beta_i \mathbf{b}_i \tilde{B}_i^n(t) \quad (4.11)$$

en la que $\tilde{B}_i^n(t)$ son las funciones de la base racional

$$\tilde{B}_i^n(t) = \frac{\beta_i B_i^n(t)}{\sum_{i=0}^n \beta_i B_i^n(t)} \quad (4.12)$$

Propiedades de la curvas de Bézier racionales :

Son válidas todas las propiedades ya vistas de las curvas no-rationales [Pie87], añadiendo:

- Proporcionan mayor número de grados de libertad, debido a la adición de los pesos, admitiendo un dominio mayor de representación de superficies
- Son generalizaciones apropiadas de las curvas de Bézier y, por lo tanto, los interpolan. Para ello basta ver que si $\forall \mathbf{b}_i$ hacemos $\beta_i = 1$ entonces $\tilde{B}_i^n(t) = B_i^n(t)$

- Las curvas de Bézier tienen la importante propiedad de interpolar exactamente [Pie87] segmentos de cónica mediante la elección de pesos adecuada. De esta manera, se puede interpolar un círculo o una elipse usando varios trozos de Bézier. Esto se puede apreciar en la figura 4.6, escogiendo adecuadamente el parámetro s se puede elegir la cónica interpolada.

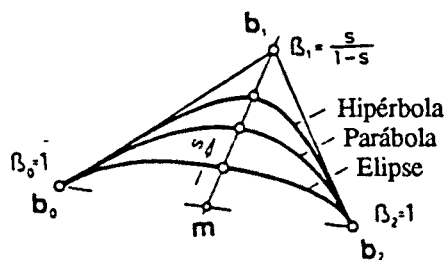


Figura 4.6: Interpolación de cónicas mediante Bézier racional (de [Pie87]).

La interpretación del impacto que tiene cambiar el valor del peso en un punto de Bézier dado es eminentemente geométrica.

1. Dado que

$$\lim_{\beta_i \rightarrow +\infty} \tilde{B}_i^n(t) = 1$$

y que

$$\lim_{\beta_i \rightarrow +\infty} \tilde{B}_j^n(t) = 0 \text{ con } j \neq i$$

vemos que a medida que aumentamos el peso i hacia +infinito la curva tiende a acercarse al punto de Bézier b_i .

2. Por otra parte $\tilde{B}_i^n(\beta_i = 0) = 0$, en particular [Pie87], si $\beta_1 = \beta_2 = \dots = \beta_n = 0$, el grafo de la curva es una recta que une b_0 con b_n .

4.1.7 B-Splines racionales

Los mismos conceptos anteriormente expresados, aplicados a las bases de los B-Splines nos permiten escribir

$$\mathbf{x}(u) = \frac{\beta_0 \mathbf{d}_0 N_0^n(u) + \dots + \beta_n \mathbf{d}_n N_n^n(u)}{\beta_0 N_0^n(u) + \dots + \beta_n N_n^n(u)} \quad (4.13)$$

El conjunto de puntos de control en 4D

$$P_i^\beta = (\beta_i x_i, \beta_i y_i, \beta_i z_i, \beta_i) \text{ para } i = 0, \dots, n$$

define una curva no-racional en 4D, cuya proyección en 3D se llama *B-Spline racional*.

$$\begin{aligned} \mathbf{x}(u) &= \mathbf{H}\{\mathbf{x}^\beta(u)\} = \mathbf{H}\left\{\sum_{i=0}^n N_i^p(u) P_i^\beta\right\} = \\ &= \frac{\sum_{i=0}^n N_i^p(u) \beta_i P_i}{\sum_{i=0}^n N_i^p(u) \beta_i} = \sum_{i=0}^n R_i^p(u) P_i \end{aligned} \quad (4.14)$$

donde

$$P_i = \mathbf{H}\{P_i^\beta\} = (x_i, y_i, z_i)$$

y

$$R_i^p(u) = \frac{N_i^p(u) \beta_i}{\sum_{j=0}^n N_j^p(u) \beta_j}$$

que son las bases de los B-Splines racionales, también conocidos como NURBS (Non Uniform Rational B-Splines).

Propiedades de los NURBS

1. $R_i^p(u) \geq 0, \forall i, p, u$
2. $\sum_{i=0}^n R_i^p(u) \equiv 1, \forall u \in [u_0, u_m]$
3. $R_i^p(u) = 0$ si $u \notin [u_i, u_{i+p+1}]$ lo que conduce al control local, al igual que en el caso de los B-Splines no racionales.
4. En un nodo $R_i^p(u)$ es $p - k$ veces diferenciable siendo k la multiplicidad del nodo.
5. $R_i^p(u) = N_i^p(u)$ si $\beta_i = 1, \forall i$ Así pues, NURBS es una generalización apropiada de B-Splines, a los que interpola cuando todos los pesos son 1, e igualmente de Bézier.

6. Excepto en el caso en que $p = 0$, $R_i^p(u)$ tiene exactamente un máximo.
7. Son *proyectivamente invariantes*, es decir para aplicar una transformación proyectiva a los puntos de la curva basta con aplicarla sobre los puntos de control.
8. Ningún plano tiene más intersecciones con el polígono de control que con la propia curva. Propiedad de variación disminuida.
9. Los NURBS son capaces de interpolar exactamente cónicas [PT87], mediante una elección adecuada de los pesos.

Como vemos se mantienen las propiedades de no negatividad, control local y variación disminuida, entre otras. Respecto a las propiedades de los pesos cabe decir lo siguiente:

- El significado geométrico de variar los valores de los pesos es totalmente análogo al caso de Bézier. En la figura 4.7 podemos apreciar las variaciones de la curva según las del peso. Concretamente para $w_i = 1$ obtenemos exactamente los B-Splines no racionales, como ya hemos visto.
- β_i afecta a la curva localmente, concretamente, si β_j está fijado para todo $j \neq i$, entonces $x(u)$ solo cambia para $u \in [u_i, u_{i+p+1}]$

4.2 Superficies por producto tensorial

Sea una curva

$$x(u) = \sum_{i=0}^n c_i F_i(u)$$

expresada en términos de unas bases F_i .

Si permitimos que cada c_i trace a su vez en el espacio una curva $c_i(v)$ expresada en los mismos términos

$$c_i(u) = \sum_{k=0}^m a_{i,k} G_k(v) \quad (4.15)$$

la superficie resultante $x(u, v)$ se llama *superficie producto tensorial*:

$$x(u, v) = \sum_{i=0}^n \sum_{k=0}^m a_{i,k} F_i(u) G_k(v) \quad (4.16)$$

Como ejemplos inmediatos podemos considerar las superficies producto tensorial de la forma monomial y la de Lagrange

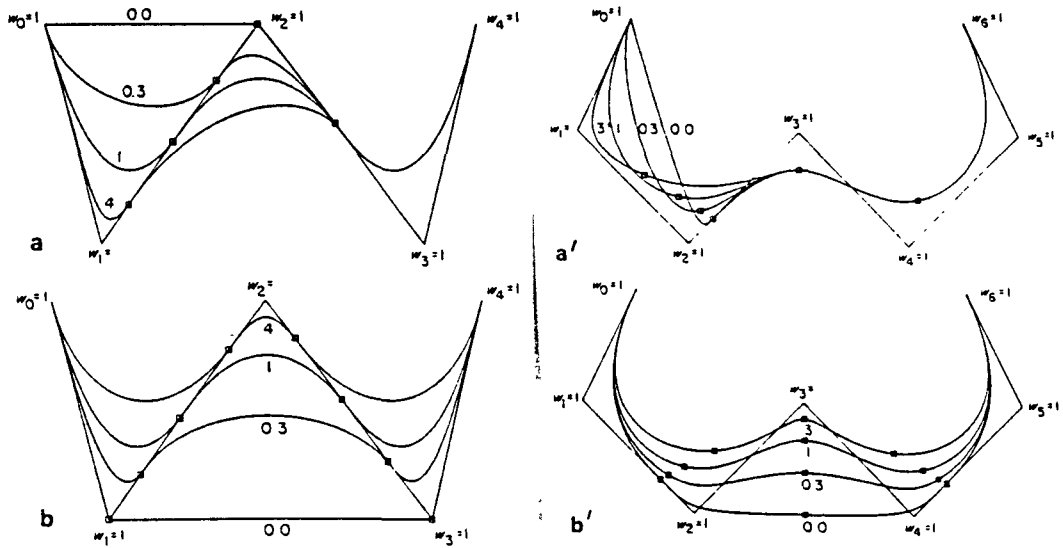


Figura 4.7: B-Splines racionales con distintos valores de (a y a') w_1 , (b) w_2 y (b') w_3 . (de [PT87]).

- Forma monomial

$$\mathbf{x}(u, v) = \sum_{i=0}^n \sum_{k=0}^m a_{i,k} u^i v^k \quad (4.17)$$

Aquí u_i es una v -línea de parámetro u constante.

- Forma de Lagrange

$$\mathbf{x}(u, v) = \sum_{i=0}^n \sum_{k=0}^m p_{i,k} L_i^n(u) L_k^m(v) \quad (4.18)$$

$L_i^n(u)$ y $L_k^m(v)$ son los polinomios de Lagrange correspondientes.

- Trozos de Bézier

$$\mathbf{x}(u, v) = \sum_{i=0}^n \sum_{k=0}^m b_{i,k} B_k^m(s) B_i^n(r) \quad (4.19)$$

donde r y s son los parámetros locales, de forma que, para $u \in [u_i, u_{i+1}]$ y $v \in [v_k, v_{k+1}]$ tenemos (ver ecuación 4.1)

$$\begin{aligned} u &= u_i(1 - r) + u_{i+1}r, \\ v &= v_k(1 - s) + v_{k+1}s \end{aligned} \quad (4.20)$$

En pro de la brevedad, y dado su relativo interés para nuestro trabajo no profundizaremos más sobre las superficies de Bézier. Más información se puede hallar en [BFK84], [Far90]

4.2.1 Superficies de B-Splines

La superficie producto de las bases $N_k^m(v)$ y $N_i^n(u)$, para unos nodos dados u_i , v_k representada por

$$\mathbf{x}(u, v) = \sum_{i=0}^n \sum_{k=0}^m d_{i,k} N_k^m(v) N_i^n(u) \quad (4.21)$$

es una representación única de la superficie, que llamaremos B-spline, en la cual los puntos $d_{i,k}$ son los llamados de *de Boor* o de control.

Propiedades (ver [BFK84], [PT87]):

- $N_i^n(u) \geq 0$
- $N_k^m(v) \geq 0$

- $\sum_{i=0}^n \sum_{k=0}^m N_k^m(v) N_i^n(u) \equiv 1, \forall (u, v) \in [0, 1] \times [0, 1]$
- Las u -líneas (de v constante) son curvas B-Spline de grado m con nodos:

$$d_i(v) = \sum_{k=0}^m d_{i,k} N_k^m(v)$$

- $N_i^n(u) = 0$ si $u \notin [u_i, u_{i+n+1}]$
- $N_k^m(v) = 0$ si $v \notin [v_k, v_{k+m+1}]$
- El trozo $u \in [u_j, u_{j+1}], v \in [v_l, v_{l+1}]$ solo es influido por los puntos de de Boor $d_{j-n, l-m}, \dots, d_{j, l}$; esto es, tenemos control local también para superficies B-Spline
- Proyectivamente invariante, es decir, la aplicación de una transformación proyectiva a los puntos de control produce una superficie que se relaciona con la anterior, mediante la misma transformación proyectiva.
- Análogamente al caso de curvas, las superficies B-Spline son una generalización apropiada de las superficies de Bézier.

4.2.2 B-Splines racionales

El B-Spline racional (en 3D) se define, al igual que en casos anteriores, como la proyección en 3D de un B-Spline no racional en 4D.

$$\begin{aligned} \mathbf{x}(u, v) &= H\{\mathbf{x}^\beta(u, v)\} = H\left\{\sum_{i=0}^m \sum_{j=0}^n N_i^p(u) N_j^q(v) \mathbf{P}_{ij}^\beta\right\} \\ &= \frac{\sum_{i=0}^m \sum_{j=0}^n N_i^p(u) N_j^q(v) \beta_{i,j} \mathbf{P}_{i,j}}{\sum_{i=0}^m \sum_{j=0}^n N_i^p(u) N_j^q(v) \beta_{i,j}} \end{aligned} \quad (4.22)$$

y, análogamente a 4.11, podemos escribirlo como

$$\mathbf{x}(u, v) = \sum_{i=0}^m \sum_{j=0}^n R_{ij}^{p,q}(u, v) \mathbf{P}_{ij} \quad (4.23)$$

donde \mathbf{P}_{ij} son los puntos de control 3D y

$$R_{ij}^{p,q}(u, v) = \frac{N_i^p(u) N_j^q(v) \beta_{i,j}}{\sum_{r=0}^m \sum_{s=0}^n N_r^p(u) N_s^q(v) \beta_{r,s}}$$

Las funciones $R_{ij}^{p,q}(u, v)$ constituyen la base bivariada de los B-Splines racionales.

Propiedades

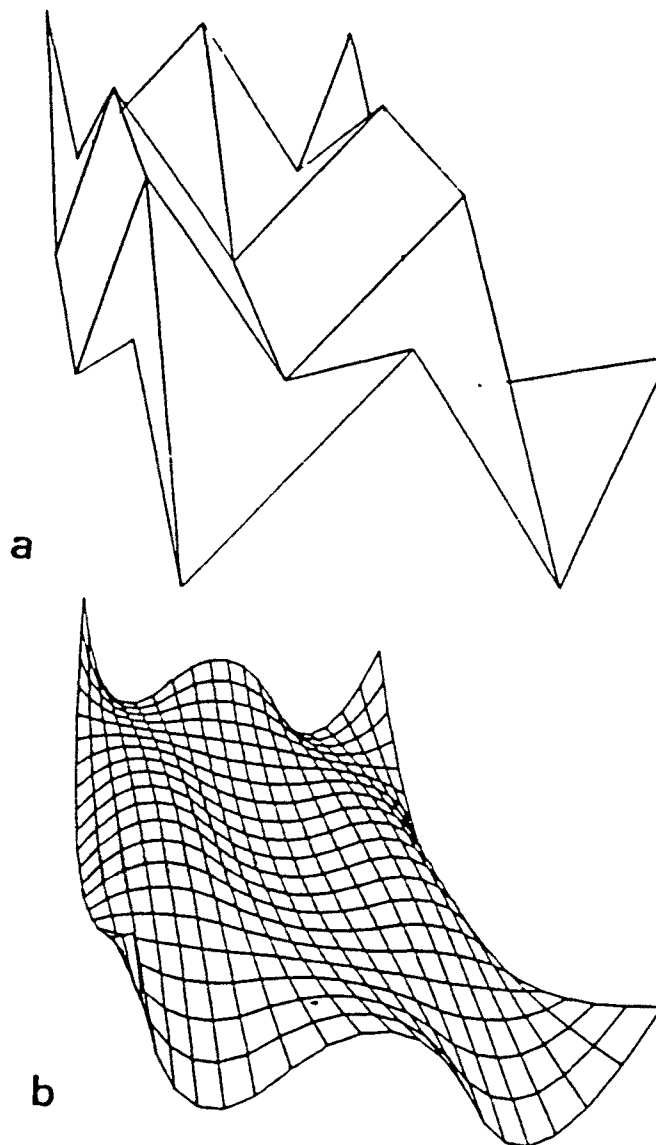


Figura 4.8: (a) Malla de pts. de control para B-Spline, (b) Superficie B-Spline de grado (2,3). Vectores de nodos $U=\{0,0,0,1/3,2/3,1,1,1\}$ y $V=\{0,0,0,0,1,1,1,1\}$ (de [PT87]).

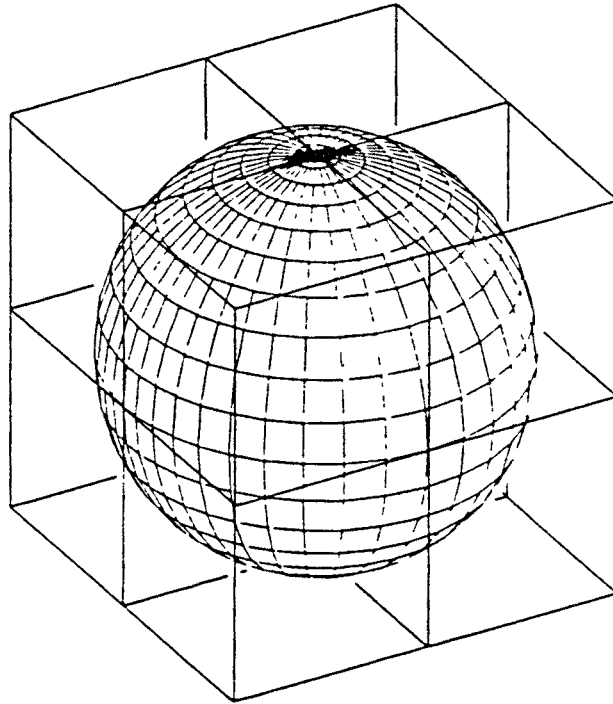


Figura 4.9: Representación de la esfera mediante B-Splines racionales ([PT87].)

- $R_{ij}^{pq}(u, v) \geq 0, \forall i, j, u, v$
- $\sum_{i=0}^m \sum_{j=0}^n R_{ij}^{pq}(u, v) \equiv 1 \forall (u, v) \in [0, 1] \times [0, 1]$
- $R_{ij}^{pq}(u, v) = 0$ si $\begin{cases} u \notin [u_i, u_{i+n+1}] \\ v \notin [v_j, v_{j+m+1}] \end{cases}$
- En un nodo de multiplicidad k , $R_{ij}^{pq}(u, v)$ es $p - k$ veces diferenciable en la dirección u y $q - k$ en dirección v .
- $R_{ij}^{pq}(u, v) = N_i^p(u)N_j^q(v)$, si todo $\beta_{ij} = 1$
- Proyectivamente invariante.
- La superficie B-Spline racional es una generalización apropiada de los B-Splines no racionales y de las superficies de Bézier.
- β_{ij} sólo afecta a $R_{ij}^{pq}(u, v)$ para $\begin{cases} u \in [u_i, u_{i+n+1}] \\ v \in [v_j, v_{j+m+1}] \end{cases}$
- Los pesos tienen una clara interpretación geométrica : si el peso crece la curva se acerca al punto de control correspondiente y viceversa si el peso decrece.
- Los NURBS son capaces de interpolar exactamente superficies conicoideas como esferas, elipsoides, paraboloides e hiperboloides, mediante la elección adecuada de los pesos y las situaciones de los nodos de control. Una muestra de ello es la Figura 4.9.

4.3 Geometría diferencial de la superficie

Sea E^3 el espacio euclídeo tridimensional. Una *representación paramétrica regular* [Lip85] de clase C^m con $m \geq 1$ de un conjunto de puntos S de E^3 es una aplicación $\mathbf{x} = \mathbf{f}(u, v)$ de un conjunto abierto U del plano uv sobre S , tal que:

1. \mathbf{f} es de clase C^m en U .
2. Si $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ es una base de E^3 y $\mathbf{f}(u, v) = f_1(u, v)\mathbf{e}_1 + f_2(u, v)\mathbf{e}_2 + f_3(u, v)\mathbf{e}_3$, entonces $\forall (u, v)$ de U , el

$$\text{rango} \begin{pmatrix} \frac{\partial f_1}{\partial u} & \frac{\partial f_1}{\partial v} \\ \frac{\partial f_2}{\partial u} & \frac{\partial f_2}{\partial v} \\ \frac{\partial f_3}{\partial u} & \frac{\partial f_3}{\partial v} \end{pmatrix} = 2$$

Denominamos *carta local* de clase C^m con $m \geq 1$ en S , a una aplicación de un conjunto abierto U en S que cumple:

1. \mathbf{x} es de clase C^m en U .
2. $\mathbf{x}_u \times \mathbf{x}_v \neq 0 \forall (u, v) \in U$.
3. \mathbf{x} es inyectiva y bicontinua sobre U .

Así, una carta local es una representación paramétrica regular de una parte de S , que es inyectiva y bicontinua [Lip85].

Así como una curva de E^3 queda determinada unívocamente por dos cantidades que son curvatura y torsión [Lip85], una superficie resulta unívocamente definida por las llamadas *primera y segunda formas fundamentales*.

4.3.1 La primera forma fundamental

Sea una carta local $\mathbf{x} = \mathbf{x}(u, v)$ de clase ≥ 1 obtenida mediante una transformación topológica de un dominio \mathcal{G} y sea $\mathbf{x} = \mathbf{x}(u(t), v(t))$ una curva en \mathcal{G} . Una transformación de \mathcal{G} en $\mathbf{x}(u, v)$ transforma la curva en otra curva γ sobre la carta local. Su longitud [Pog87] viene dada por

$$\begin{aligned} s &= \int \sqrt{x_u^2 \left(\frac{du}{dt}\right)^2 + 2x_u x_v \left(\frac{du}{dt}\right)\left(\frac{dv}{dt}\right) + x_v^2 \left(\frac{dv}{dt}\right)^2} dt \\ &= \int_{\gamma} \sqrt{x_u^2 du^2 + 2x_u x_v dudv + x_v^2 dv^2} \end{aligned} \quad (4.24)$$

La *primera forma fundamental* es la forma cuadrática definida sobre los vectores (du, dv) del plano uv como:

$$I(du, dv) = ds^2 = Edu^2 + 2Fdudv + Gdv^2 \quad (4.25)$$

siendo

$$E = x_u^2, \quad F = x_u x_v, \quad G = x_v^2 \quad (4.26)$$

De las fórmulas 4.24 y de la definición de la primera forma fundamental, se sigue que para conocer la longitud de una curva basta con el conocimiento de la primera forma fundamental, por lo cual se dice de ésta que *determina una métrica* en la superficie (ver [Pog87]). Igualmente se puede ver que el área de una superficie y el ángulo que subtenden dos curvas sobre una superficie quedan determinados también por la primera forma fundamental:

$$S = \int_{\mathbf{x}} \int \sqrt{EG - F^2} dudv$$

$$\cos \theta = \frac{Edu\delta u + F(du\delta v + dv\delta u) + Gdv\delta v}{(Edu^2 + 2Fdudv + Gdv^2)^{1/2}(E\delta u^2 + 2F\delta u\delta v + G\delta v^2)^{1/2}}$$

4.3.2 La segunda forma fundamental

Sea una carta local de clase ≥ 2 . En cada punto de la carta [Lip85] hay una normal unitaria $N = \frac{\mathbf{x}_u \times \mathbf{x}_v}{|\mathbf{x}_u \times \mathbf{x}_v|}$. La *segunda forma fundamental* es la forma cuadrática definida por

$$II(du, dv) = -dx dN = L du^2 + 2M du dv + N dv^2 \quad (4.27)$$

siendo

$$L = -\mathbf{x}_u N_u, \quad M = -\frac{1}{2}(\mathbf{x}_u N_v + \mathbf{x}_v N_u), \quad N = -\mathbf{x}_v N_v \quad (4.28)$$

Dado que \mathbf{x}_u y \mathbf{x}_v son perpendiculares a N , $\forall(u, v)$ tenemos

$$\begin{aligned} (\mathbf{x}_u N)_u &= \mathbf{x}_{uu} N + \mathbf{x}_u N_u = 0 & (\mathbf{x}_u N)_v &= \mathbf{x}_{uv} N + \mathbf{x}_u N_v = 0 \\ (\mathbf{x}_v N)_u &= \mathbf{x}_{vu} N + \mathbf{x}_v N_u = 0 & (\mathbf{x}_v N)_v &= \mathbf{x}_{vv} N + \mathbf{x}_v N_v = 0 \end{aligned}$$

y entonces

$$\mathbf{x}_{uu} N = -\mathbf{x}_u N_u, \quad \mathbf{x}_{uv} N = -\mathbf{x}_u N_v, \quad \mathbf{x}_{vv} N = -\mathbf{x}_v N_v$$

y de aquí podemos obtener

$$L = \mathbf{x}_{uu} N = \mathbf{x}_{uu} \frac{\mathbf{x}_u \times \mathbf{x}_v}{|\mathbf{x}_u \times \mathbf{x}_v|} = \frac{(\mathbf{x}_{uu} \mathbf{x}_u \mathbf{x}_v)}{|\mathbf{x}_u \times \mathbf{x}_v|} = \frac{\begin{vmatrix} x_{uu} & y_{uu} & z_{uu} \\ x_u & y_u & z_u \\ x_v & y_v & z_v \end{vmatrix}}{\sqrt{EG - F^2}}$$

y análogamente podemos escribir

$$M = \frac{\begin{vmatrix} x_{uv} & y_{uv} & z_{uv} \\ x_u & y_u & z_u \\ x_v & y_v & z_v \end{vmatrix}}{\sqrt{EG - F^2}}, \quad N = \frac{\begin{vmatrix} x_{vv} & y_{vv} & z_{vv} \\ x_u & y_u & z_u \\ x_v & y_v & z_v \end{vmatrix}}{\sqrt{EG - F^2}}$$

4.3.3 La curvatura normal

Sea $\mathbf{x} = \mathbf{x}(u, v)$ una carta local en una superficie de clase ≥ 2 que contiene un punto P y sea $\mathbf{x} = \mathbf{x}(u(t), v(t))$ una curva regular C de clase C^2 que pasa por P . Se llama *vector curvatura normal* \mathbf{k}_n , al vector proyección del vector curvatura \mathbf{k} de C en P sobre la normal N en P .

$$\mathbf{k}_n = (\mathbf{k}N)N \quad (4.29)$$

(Veáse [Pog87], etc. para una discusión sobre el vector curvatura y torsión). Al número

$$k_n = \mathbf{k}N \quad (4.30)$$

lo llamaremos *curvatura normal* de C en P . Supongamos que C tiene normal principal \mathbf{n} continua y que el sentido de \mathbf{n} es tal que $0 \leq \angle(\mathbf{n}, \mathbf{N}) \leq \pi/2$. Podemos escribir

$$k_{\mathbf{n}} = k\mathbf{N} = \dot{\mathbf{t}} = k(\mathbf{n}, \mathbf{N}) = k \cos \alpha \quad (4.31)$$

siendo $\alpha = \angle(\mathbf{n}, \mathbf{N})$. Se puede ver que

$$k_{\mathbf{n}} = \frac{II}{I} = \frac{Ldu^2 + 2Mdudv + Ndv^2}{Edu^2 + 2Fdudv + Gdv^2} \quad (4.32)$$

y teniendo en cuenta 4.31

$$k \cos \alpha = \frac{Ldu^2 + 2Mdudv + Ndv^2}{Edu^2 + 2Fdudv + Gdv^2} \quad (4.33)$$

De aquí se desprende que $k \cos \alpha$ depende sólo de la dirección de C , es decir del valor du/dv que reciben el nombre de *números directores* de la recta en el plano tangente paralelo al vector $\mathbf{x}_u du + \mathbf{x}_v dv$. Si tomamos como curva C la sección de la superficie por un plano perpendicular al plano tangente (*sección normal*), entonces $|\cos \alpha| = 1$ y entonces

$$k |\cos \alpha| = k_0$$

donde k_0 es la curvatura de la sección normal.

Sea un punto O sobre una superficie, al que consideraremos como origen de coordenadas y consideremos el plano tangente en este punto como el plano xy . Supongamos que la superficie está dada por una ecuación tal como $z = f(u, v)$, es decir $\mathbf{x} = ue_1 + ve_2 + f(u, v)e_3$. En este caso podemos escribir

$$E = 1 + p^2, \quad F = pq, \quad G = 1 + q^2$$

$$L = \frac{r}{\sqrt{1 + p^2 + q^2}}, \quad M = \frac{s}{\sqrt{1 + p^2 + q^2}}, \quad N = \frac{t}{\sqrt{1 + p^2 + q^2}}$$

siendo

$$p = z_u, \quad q = z_v, \quad r = z_{uu}, \quad s = z_{uv}, \quad t = z_{vv}$$

y por la elección del sistema coordenado que hemos hecho obtenemos $E = 1$, $F = 0$, $G = 1$, $L = r$, $M = s$, $N = t$. Entonces

$$k_{\mathbf{n}} = \frac{rdu^2 + 2sdudv + t dv^2}{du^2 + dv^2} \quad (4.34)$$

Puesto que $k_{\mathbf{n}}$ sólo depende de du/dv , podemos suponer que $du^2 + dv^2 = 1$ y que $du = \cos \theta$ y $dv = \sin \theta$, entonces

$$k_{\mathbf{n}} = r \cos^2 \theta + 2s \cos \theta \sin \theta + t \sin^2 \theta$$

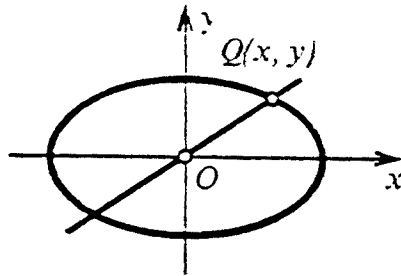


Figura 4.10: Indicatriz de Dupin.

y si suponemos que $|k_n| = \frac{1}{R^2}$ y $x = R \cos \theta$ y $y = R \sin \theta$ obtenemos la ecuación llamada *indicatriz de Dupin*

$$rx^2 + 2sxy + ty^2 = \pm 1 \quad (4.35)$$

Si tenemos en cuenta la figura 4.10, y expresamos la curvatura normal en una dirección OQ en términos de las coordenadas x e y del punto Q sobre la indicatriz, tenemos que $dx/dy = x/y$, así pues

$$k_n = \frac{rx^2 + 2sxy + ty^2}{x^2 + y^2}$$

Dado que Q está sobre la indicatriz, el numerador vale ± 1 y el denominador vale OQ^2 , de forma que

$$k_n = \frac{\pm 1}{OQ^2} \quad (4.36)$$

que nos relaciona la indicatriz de Dupin con la curvatura normal. De esta fórmula se deduce

1. La curvatura normal en una dirección asintótica es cero.
2. La curvatura normal de una superficie adopta valores extremos en dos direcciones perpendiculares entre sí que se denominan *direcciones principales*, que son los ejes de la indicatriz. Las correspondientes curvaturas se llaman *curvaturas principales*.

4.3.4 Aberraciones superficiales

Se puede demostrar que

Teorema 4.1 *Un número real k es curvatura principal si y sólo si k es solución de*

$$(EG - F^2)k^2 - (EN + GL - 2FM)k + LN - M^2 = 0 \quad (4.37)$$

y por otra parte,

Teorema 4.2 *Una dirección du/dv es dirección principal si y sólo si du y dv cumplen*

$$(EM - LF)du^2 - (EN - LG)dudv + (FN - MG)dv^2 = 0 \quad (4.38)$$

A partir del conocimiento de las curvaturas y direcciones principales y teniendo en cuenta las definiciones de la potencia media 2.10 y el astigmatismo superficial 2.11 podemos obtener, para una superficie dada los valores de estas magnitudes. Para ello necesitaremos conocer las derivadas parciales de primer y segundo orden de nuestra superficie.

Para el caso de superficies por producto tensorial cuyas bases son polinomios resulta fácil hallar las derivadas parciales. Veamos el caso de los B-Splines cúbicos, por ejemplo.

Recordando la ecuación 4.21

$$\mathbf{x}(u, v) = \sum_{i=0}^n \sum_{k=0}^m \mathbf{d}_{i,k} N_k^m(v) N_i^n(u)$$

y transformándola a su forma matricial

$$\mathbf{x}(u, v) = [\mathbf{u}]^T [\mathbf{N}] [\mathbf{D}] [\mathbf{N}]^T [\mathbf{v}] \quad (4.39)$$

donde $[\mathbf{u}] = (u^3, u^2, u, 1)$, $[\mathbf{v}] = (v^3, v^2, v, 1)$, $[\mathbf{N}]$ es la matriz de la base de B-Splines para el trozo bicúbico en cuestión y $[\mathbf{D}]$ es la submatriz de 16 vértices de control del mismo. Las derivadas parciales se calculan entonces de la forma:

$$\begin{aligned} \mathbf{x}(u, v)_u &= [3u^2, 2u, 1, 0]^T [\mathbf{N}] [\mathbf{D}] [\mathbf{N}]^T [v^3, v^2, v, 1] \\ \mathbf{x}(u, v)_v &= [u^3, u^2, u, 1]^T [\mathbf{N}] [\mathbf{D}] [\mathbf{N}]^T [3v^2, 2v, 1, 0] \\ \mathbf{x}(u, v)_{uu} &= [6u, 2, 0, 0]^T [\mathbf{N}] [\mathbf{D}] [\mathbf{N}]^T [v^3, v^2, v, 1] \\ \mathbf{x}(u, v)_{uv} &= [3u^2, 2u, 1, 0]^T [\mathbf{N}] [\mathbf{D}] [\mathbf{N}]^T [3v^2, 2v, 1, 0] \\ \mathbf{x}(u, v)_{vv} &= [u^3, u^2, u, 1]^T [\mathbf{N}] [\mathbf{D}] [\mathbf{N}]^T [6v, 2, 0, 0] \end{aligned}$$

A partir de las derivadas parciales se pueden calcular los coeficientes de las formas fundamentales y, con éstas, hallar las curvaturas y ejes principales, y, de aquí, los valores de potencia media y astigmatismo.

4.4 Diseño y control local

Hemos visto en las secciones precedentes las principales formulaciones de interpolación de curvas y superficies en uso en los sistemas de diseño asistido por

ordenador, y los elementos de geometría diferencial necesarios para aplicarlos a nuestras necesidades.

Se nos plantea ahora la decisión de cuál o cuales de estas formulaciones usar en nuestro sistema de diseño y cómo integrarlas en el mismo. La tabla 4.1 nos da una idea comparativa de las características de los distintos tipos de interpolación.

Tabla 4.1: Interpolación de Curvas y Superficies

Características					
Tipo	Grado	Control	Continuidad	Interpola	Suavidad
Lagrange	n	Global	C^∞	Si	Oscila
Polinomios	n	Global	C^{n-1}	Si	Oscila
Bézier	n	Global	C^∞	No	Variación disminuída
B-Splines	n	Local	hasta C^{n-1}	No	Variación disminuída
NURBS	n	Local	hasta C^{n-1}	No	Variación disminuída

El control local es una propiedad muy interesante en nuestro caso, pues nos ofrece la posibilidad de optimizar solamente ciertas zonas de la lente, dejando invariantes las demás. Para el progresivo resulta de gran interés diseñar independientemente la zona de lejos, el pasillo y la zona de cerca, y dejarlas "congeladas", optimizando entonces las zonas marginales aisladamente.

Igualmente, la suavidad es una premisa muy importante, ya que oscilaciones indeseadas en la superficie conducen a efectos ópticos en el astigmatismo y variaciones en la potencia que alcanzan rápidamente valores inaceptables en el ámbito de la visión.

En vista de estos condicionantes, los tipos de interpolación de máximo interés para nosotros son B-Splines y NURBS, que ofrecen ambas condiciones a la vez.

Capítulo 5

Metodología de Diseño

5.1 Introducción

A la hora de diseñar interactivamente una lente progresiva, se plantean una serie de requerimientos de orden óptico, que ésta ha de cumplir. Estos requerimientos hacen referencia, en su mayoría, a la distribución de potencias y astigmatismos sobre la lente, magnitudes relacionadas con las derivadas segundas de la superficie.

La necesidad de especificar la distribución de potencias es evidente dada la naturaleza multifocal de este tipo de lente. La importancia de especificar el astigmatismo es debida a que, entre todas las características ópticas de una lente ésta es, con mucho, la que más incide en la nitidez de visión. Ello no excluye la consideración de otras características como pueden ser la orientación de los ejes principales de curvatura o los desequilibrios prismáticos entre un ojo y otro.

Es importante destacar aquí que, a pesar de que en muchas ocasiones tiene más sentido hablar de la agudeza visual, el índice de borrosidad, etc; la mayoría de estas magnitudes son combinaciones del astigmatismo y la potencia, en ocasiones junto con los ejes principales de las curvaturas. Por simplicidad, pues, nos referiremos a las magnitudes básicas, en vez de a las compuestas.

Por otra parte, en todos los trabajos y patentes consultadas, se parte de la hipótesis implícita o explícita de que lo que se pretende distribuir según unas ciertas especificaciones no es la aberración de onda de la luz que atraviesa el sistema, sino el astigmatismo y la potencia *superficiales*. Esto es, no se hace trazado de rayos (lo que resultaría prohibitivo en tiempo de cálculo), sino que se supone que si se distribuyen de una cierta forma las aberraciones superficiales, las aberraciones de onda se distribuirán de forma análoga.

La introducción de datos interactiva se hace particularmente difícil dada la cantidad de valores a especificar sobre la superficie y a la posibilidad de escoger distribuciones de potencia y astigmatismo incompatibles con la geometría diferencial de una superficie real.

Por estas razones la metodología de diseño ha de estar relacionada con la estrategia de consecución de la superficie final a partir de las especificaciones iniciales. Más concretamente, dado que no está asegurado que un diseño dado sea compatible con la geometría diferencial de la superficie, habremos de realizar una optimización tendente a conseguir la superficie que más se acerque a las especificaciones dadas.

5.1.1 Optimización y función de mérito.

Para ello definiremos una *función de mérito* f , que representa, en cada momento, un valor relacionado con lo que se ajusta nuestro diseño a las especificaciones ópticas deseadas. Muchas de las funciones de mérito usadas habitualmente se pueden englobar en la forma:

$$f = \left[\frac{1}{n} \sum_{i=1}^n P_i D_i^k \right]^{1/k} \quad (5.1)$$

donde $1 \leq i \leq n$, siendo N el número de puntos sobre los que se calcula f . D_i es la diferencia entre los valores real y deseado de una característica particular de diseño, expresable como

$$D_i = V_i' - V_i, \quad (1 \leq i \leq n) \quad (5.2)$$

en la que V_i' representa el valor a conseguir y V_i el valor real en un momento dado de la optimización. k es un entero, habitualmente 2, con lo que resulta la minimización de la raíz de la media cuadrática de las diferencias D_i . P_i es un *factor de ponderación* que permite que el término D_i reciba diferente atención o importancia en función del valor de este factor.

Obviamente, necesitamos de un punto de partida para la optimización. En puridad podríamos utilizar una superficie cualquiera como tal, pero criterios de eficiencia y rapidez nos conducen a buscar una superficie de partida lo más cercana posible a aquella que deseamos. Esta puede ser una superficie producto de un diseño anterior, o bien una superficie analítica que no se aparte demasiado de las líneas maestras del diseño.

Provistos de esta herramienta y de un sistema numérico de optimización, podemos, especificada una superficie ideal en cuanto a sus atributos ópticos,

obtener una función de mérito que la represente y, optimizando, hallar la superficie real que más se le acerca.

Nuestro problema ahora es especificar de forma interactiva y lo más sencilla (en cuanto a operatoria) posible las magnitudes que forman parte de la función de mérito, con la resolución espacial suficiente como para que el proceso de optimización se acerque a lo deseado.

5.2 Especificaciones ópticas

5.2.1 Variación de potencia

La variación de la potencia sobre la superficie de la lente cobra capital importancia en el caso del meridiano principal.

Una cuestión que diferencia profundamente un diseño de otro es la evolución de la potencia de su meridiano. Esta forma de progresión igualmente afecta al rango de utilización de la lente y a las zonas de la misma que servirán para enfocar a una distancia determinada, así como, en virtud del teorema de Minkwitz, a la anchura del pasillo.

Por ello, en el marco de este proyecto, se considera importante disponer de un sistema interactivo de diseño de la evolución de la potencia del meridiano.

Para editar la progresión de potencia sobre el meridiano se presenta al usuario una imagen en pantalla similar a la Figura 5.1 en la que un menú le permite *aceptar* el meridiano propuesto o *modificarlo*. La opción de modificación permite seleccionar uno cualquiera de los puntos señalados mediante un símbolo \square y una vez hecho esto cambiar a voluntad sus coordenadas, con la única restricción por parte del sistema, de que la progresión sea una curva univaluada.

Esto se consigue imponiendo que la abscisa del punto i a modificar cumpla

$$x_{i-1} < x_i < x_{i+1}, \forall i \in \{0 \dots n\}$$

El número n de puntos a escoger es seleccionable, según la resolución con que se quiera manejar el meridiano.

Una vez modificada la posición en ordenadas de un punto cualquiera se calculan los nodos de control de la curva B-Spline $p(s)$ que pasa por todos los puntos (ver apéndice C) y se dibuja para que el usuario pueda evaluar cómo varía la curvatura del mismo, y decidir, en su caso, una nueva modificación.

Mediante un procedimiento análogo se puede diseñar la trayectoria del meridiano principal (ver Figura 5.2), entendiéndose por tal la trayectoria de convergencia de los ojos proyectada sobre la lente. Ello, juntamente con el conocimiento

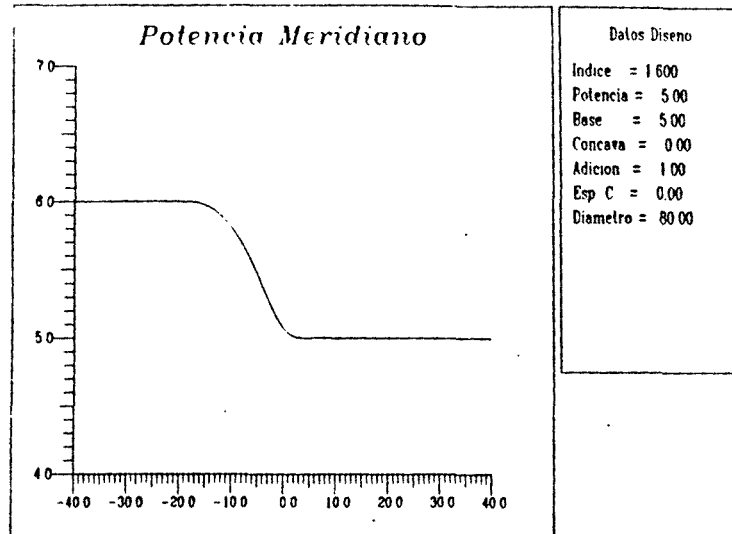


Figura 5.1: Diseño del meridiano principal

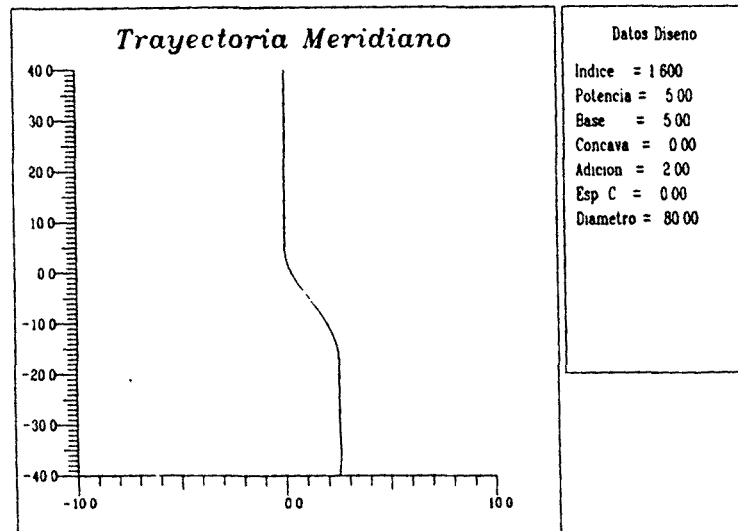


Figura 5.2: Diseño de la trayectoria del meridiano principal.

de la evolución de la potencia $p(s)$ sobre el mismo nos proporciona la posibilidad de generar una superficie inicial, como punto de partida de la optimización.

Para ello, basta con encontrar la curva del meridiano $f(s)$ a partir de la de sus potencias $p(s)$. Resolviendo la ecuación 2.6 mediante un método numérico hallamos $f(s) \forall s$, y, conociendo las coordenadas de cada punto del meridiano y su curvatura podemos generar una superficie inicial a base de imponer que las secciones ortogonales al meridiano principal sean umbilicales.

Estas secciones pueden ser arcos de círculo, o bien elipses, o cualquier otro tipo de sección que proporcione una superficie generable analíticamente. A partir del conocimiento de esta superficie se pueden dar valores de la misma dispuestos en una malla regular y extraer los nodos de control que los interpolan, con lo que ya tenemos la superficie inicial expresada como B-Splines, o como NURBS, si ponemos todos los pesos con valor 1.

5.2.2 Umbilicalidad de la progresión

Una de las razones por la que hemos elegido B-Splines como formalismo para la optimización de la superficie, es su control local. Ello implica que podemos optimizar localmente sólo algunas partes de la superficie dejando el resto intacto. Concretamente, el pasillo es uno de los candidatos a no ser modificado. Por ello, la progresión será la de la superficie inicial en la optimización.

En el apartado anterior se propone la construcción de ésta a partir de secciones ortogonales umbilicales al meridiano principal. No obstante, se puede añadir flexibilidad al diseño permitiendo la relajación hasta cierto punto de esta umbilicalidad. Propuestas en este sentido se pueden encontrar en [Jef57] [Wil81]. Igualmente, realizar un meridiano con trayectoria no rectilínea produce un efecto de pérdida de la umbilicalidad [FL86].

La visión humana tolera sin problemas de 0.25 a 0.50 D. de astigmatismo [Mai74] [Dav78], lo que permite construir un pasillo confortable aunque contenga una cierta cantidad de astigmatismo. Permitiendo esta falta de umbilicalidad relajamos la validez del teorema de Minkwitz y podemos ensanchar ligeramente el pasillo [BS87]. Ello se especifica en el diseño como un factor de relajación que el usuario especifica en la entrada de datos.

5.3 Variables de optimización

Las variables que intervienen en nuestra optimización han de ser, obviamente, los parámetros que gobiernan la forma de la superficie. Esto es, los nodos de control y/o los pesos, en el caso de los NURBS.

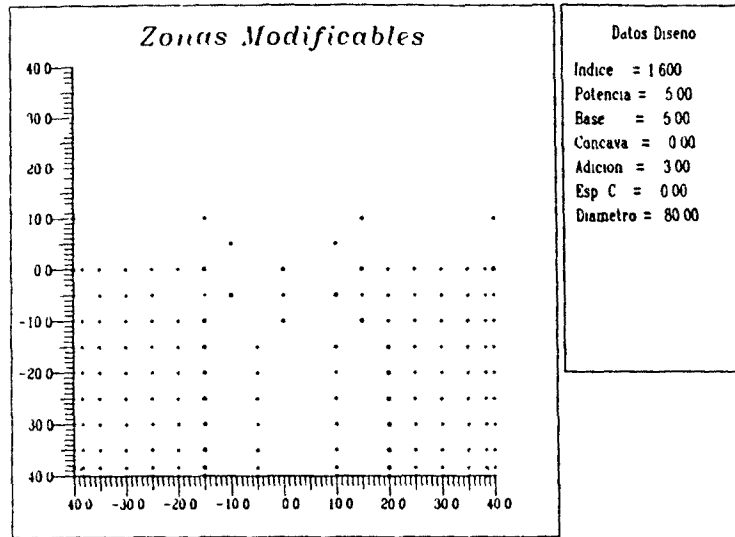


Figura 5.3: Selección de nodos modificables

Dado el carácter local de las superficies a emplear, podemos dejar fijos una serie de nodos, las propiedades de cuyos trozos de superficie sean ya satisfactorias al principio de la optimización. Esto es aplicable, por ejemplo, a las zonas de visión lejana y cercana, así como al pasillo de progresión. Por otra parte los nodos que gobiernan las zonas marginales constituyen los candidatos ideales a parámetros de optimización.

Para permitir la selección flexible, interactiva y sencilla de los nodos que intervendrán activamente en la optimización disponemos en el sistema de una pantalla como la que muestra la figura 5.3. En ella aparecen todos los nodos de la superficie representados como pequeños puntos.

Los puntos señalados mediante rombos representan puntos seleccionados como nodos *activos* en la optimización. Con un cuadrado sobre el rombo figuran los nodos que forman la *frontera* que engloba a todos los demás nodos activos. Con asteriscos está señalada la frontera hasta donde llega el efecto de modificar los puntos de control activos. Más allá de esta frontera, la superficie permanece inalterada en virtud de las propiedades de control local del formalismo seleccionado.

La zona sombreada (puntos intercalados) corresponde a los puntos donde se efectúa el cálculo de la función de mérito.

El sistema nos proporciona, en función del número de perfiles horizontales y verticales especificados, y por defecto, una selección previa de puntos marcados como rombos, que podemos aceptar directamente o modificar a voluntad. La localización de un punto a añadir, excluir o modificar se realiza mediante *mouse*, señalando el punto y oprimiendo uno de los botones.

Un menú asociado a esta pantalla permite seleccionar varias opciones:

- **Anular** Excluye un punto de control de la lista de puntos modificables.
- **Añadir** Incluye un punto de control en la lista de puntos modificables.
- **Modificar** Permite la modificación de la situación de un punto de control.
- **Aceptar** Guarda la selección actual de puntos activos y sale del menú de puntos modificables.

Una vez se han seleccionado adecuadamente los puntos de control, la lista de puntos de control activos permanece en memoria para su posterior utilización en la optimización o, en su defecto para ser grabada en un fichero junto con los demás parámetros de diseño.

5.3.1 Distribución de aberraciones y pesos

Una vez realizada una propuesta de punto de partida para la optimización procede estipular los valores V_i' a conseguir en la optimización y sus pesos asociados.

Dado que interesa tener una distribución uniforme de astigmatismos, lo más bajos posible, y que el número de puntos que participan en la función de mérito puede llegar a ser elevado, conviene disponer de un sistema interactivo de modelaje de las aberraciones en las zonas marginales.

Se han propuesto dos procedimientos para especificar los valores deseados de las aberraciones que intervienen en la función de mérito:

- A Este procedimiento permite la definición de una función analítica que proporciona el peso relativo de cada punto en la función de mérito, suponiendo que los astigmatismos buscados son 0 en toda la superficie, que los ejes principales han de estar a 90 grados y las potencias han de ser iguales a la de lejos.
- B Es un procedimiento interactivo, mediante el cual se entran las curvas de nivel de una o todas las aberraciones a tener en cuenta (astigmatismo, potencia, ejes) y las curvas de igual peso relativo. Mediante una triangulación y una interpolación se obtiene una superficie de aberración (o peso), de la cual se extraen los valores en cada punto de cálculo especificado.

Ambos procedimientos dan como resultado un fichero de formato:

`n, i, j, u, v, x, y, z, ast, pot, eje, peso.`

donde n representa el número de orden, i es una componente del trozo bicúbico y j es la otra, u es el valor del parámetro en la dirección u , v es el valor del parámetro en la dirección v , x, y, z son las componentes cartesianas correspondientes a ese trozo bicúbico con esos parámetros y ast, pot y eje son el astigmatismo, potencia y orientación del eje de la indicatriz deseados en ese punto, siendo $peso$ el peso asociado en la optimización con ese punto. Nótese que este peso está relacionado con la función de mérito en la optimización y *no* es el peso asociado a un vértice de control en el caso de los NURBS.

Procedimiento A

En esta opción, el peso de un punto de índice i se calcula mediante una función bidimensional que se diseña interactivamente, con la misma operatoria que la evolución de la potencia o la trayectoria del meridiano (ver figura 5.4). En

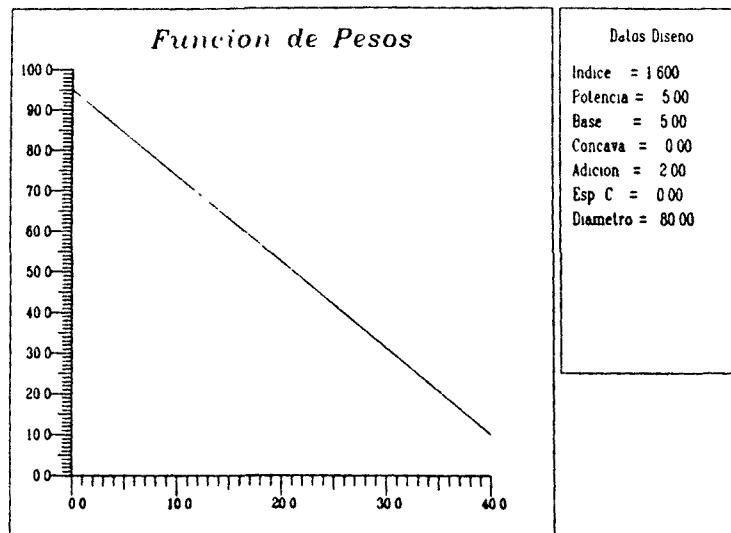


Figura 5.4: Función de pesos.

abscisas figura una distancia en unidades arbitrarias, que representa la razón entre la distancia del punto considerado a otro, especificado por el diseñador

en la entrada de datos, que llamaremos *punto de escape* y la distancia entre el punto de escape y la frontera de los puntos de cálculo de la función de mérito.

En ordenadas tenemos el peso asignado a cada una de las razones. La curva se especifica mediante una serie de puntos, de los cuales se guardan los puntos de control de un B-Spline cúbico. La conjunción de la forma de la frontera con el punto de escape produce una superficie de pesos que se puede calcular fácilmente mediante el algoritmo:

```

FUNCION Peso (xp, yp, x_esc, y_esc)
  Calcula_recta(xp, yp, x_esc, y_esc, recta)
  Calcula_interseccion(recta, frontera, p0x, p0y)
  (* Distancia hasta el punto de interseccion *)
  distancia = Raiz_2( (p0x-xp)^2 + (p0y-yp)^2 )
  distancia2 = Raiz_2( (x_esc-p0x)^2 + (y_esc-p0y)^2 )
  razon = radio*distancia/distancia2
  pes=Bspline_ld( razon, vertex)
  Peso=pes
DEVUELVE Peso
FINFUNCION

```

Donde x_p e y_p son las coordenadas cartesianas del punto en cuestión, x_{esc} e y_{esc} son las coordenadas del punto de escape, *frontera* es una estructura de datos que delimita la frontera de puntos donde se calcula la función de mérito y *vertex* contiene los vértices de control de la curva que define como varía el peso según la razón de distancias *razon*.

De esta manera, a la hora de optimizar la función de mérito se puede optar por leer las especificaciones del fichero de mérito y usarlas directamente, o bien leer los vértices de control de la curva de pesos y evaluar éstos con el algoritmo antes descrito.

Esto último es más costoso en tiempo de proceso, pero tiene la ventaja de que se puede optimizar en el espacio paramétrico, manteniendo los valores t_i , t_j , u y v , y obteniendo los pesos a partir de la x e y deducidas de aquellos.

Esto es muy útil con los N.U.R.B.S., en los cuales, variaciones de los factores de peso asociados a sus vértices de control, hacen variar la relación entre los parámetros del trozo y su expresión cartesiana, con lo que resulta más ventajoso calcular los pesos de mérito a partir de los puntos del espacio paramétrico mediante el algoritmo anterior que trabajar en el espacio cartesiano y buscar el trozo bicúbico y los parámetros u y v a partir de la x y la y (ver sección 7.4).

Procedimiento B

Esta opción permite la especificación interactiva de los valores a obtener en la función de mérito, mediante el dibujo de las curvas de nivel de una cierta *altura* de la superficie de la aberración en cuestión.

Con la misma operatoria básica ya conocida (seleccionar puntos y moverlos) se pueden representar una serie de curvas de nivel para cualquiera de las aberraciones o características que se quieran incluir en la función de mérito.

Una vez seleccionadas éstas, mediante una triangulación de Delaunay de los puntos [Slo87] y una interpolación mediante el método de Shepard (ver [She68]) se genera la superficie y una vez conocida ésta se pueden dar valores concretos a los puntos de cálculo del fichero de función de mérito.

Este método es muy versátil, pues permite especificar a “mano alzada” las curvas de nivel deseadas para cualquiera de los factores que intervienen en la optimización. Sin embargo, esta versatilidad es su principal defecto también, ya que se pueden generar condiciones a cumplir por las superficies de astigmatismos y potencias que son claramente incompatibles con la geometría diferencial de la superficie a optimizar, lo que conduce en la práctica a resultados muy alejados de lo buscado.

En este sentido, el procedimiento A, aunque menos flexible, se comporta mejor, ya que son sólo los pesos los que se varían y los valores de astigmatismos, etc tienden a los de una esfera. Esto hace que normalmente se consigan resultados más suaves y mejor distribuidos desde el punto de vista de la óptica fisiológica que con el procedimiento B, en el que todo se deja al criterio del usuario. Éste último, potencialmente más flexible, requiere del usuario una intuición y una práctica en obtener condiciones compatibles con la geometría diferencial del problema, que es costosa de conseguir, en comparación con el criterio A.

5.3.2 Tratamiento gráfico de datos

El sistema de diseño necesita de utillería gráfica que permita, no sólo la especificación interactiva de características, sino la visualización de los resultados de la optimización. Por interés de completitud explicaremos muy brevemente los conceptos básicos que subyacen a este apartado.

Para este cometido, se ha dotado al sistema de un módulo de tratamiento de datos que, a partir de un fichero, bien de diseño o bien de medida (disponemos de una mesa micrométrica de construcción propia que permite tomar datos geométricos y/u ópticos de la superficie de la lente), construye una representación

en contornos de nivel de la superficie, potencia, astigmatismo y ejes principales de la misma.

A partir de los puntos dados se genera una triangulación de Delaunay, la cual se interpola posteriormente mediante Shepard triangular [She68], en el caso de puntos irregularmente distribuidos, o mediante B-Splines en el caso regular (típicamente diseños). Los astigmatismos, etc. se representan previa resolución de la primera y segunda formas fundamentales de la superficie.

Para el dibujo de los contornos se utiliza un sencillo algoritmo de interpolación lineal entre los lados de cada triángulo (véase [Car85],[Sab86]), utilizando el método de interpolación para variar la densidad de puntos deseada y, con ello, la resolución y adaptación del contorno a la realidad.

Todos los gráficos 2D de potencia, ejes y astigmatismo que figuran en este trabajo han sido realizados con este módulo.

Capítulo 6

Implementación en Paralelo

La optimización de una superficie B-Splines o NURBS con una malla del orden de 15x15 puntos o más, aunque gracias al control local se reduzca a un área restringida de la lente, representa tener un sistema de optimización multidimensional con una dimensión por nodo modificable, lo que supone generalmente más de 100 dimensiones. Esta proliferación de grados de libertad confiere una gran flexibilidad de diseño, pero sobrecarga notablemente la CPU, elevando los tiempos de cálculo a cotas elevadas.

Por cuestiones estructurales, la implementación de los algoritmos debía realizarse sobre PC compatibles, concretamente un conjunto i386 + i387 a 25 MHz. Realizadas pruebas preliminares se vió que la potencia de cálculo era excesivamente baja para el volumen de operaciones a realizar. Por ello se decidió la adquisición de un sistema nuevo, o la ampliación del mismo.

Ello condujo a la adquisición de una placa de 9 *Transputers* (ver Apéndice B), que tenía en aquel momento (final de 1988) una relación de 5 a 1 en MFLOP/Ptas. con respecto a sistemas secuenciales de prestaciones parecidas, según pruebas realizadas por nosotros mismos con programas de prueba similares a los algoritmos a implementar.

Por todo ello los algoritmos más ávidos de tiempo de CPU (los de optimización), han sido implementados en paralelo, utilizando una versión para sistemas multiprocesador del lenguaje 'C', que contiene una librería de funciones de comunicación entre procesadores, semáforos y *threads* o tareas concurrentes en un mismo procesador.

6.1 El algoritmo de Optimización

El algoritmo general de optimización se puede esquematizar como sigue:

```

ACCION Principal
  Lee_superficie(sup_fich, superficie, limites)
  Lee_fun_merit(merit_fich)
  vector=Genera_vector_optim(superficie, limites)
  Optimiza(superficie, vector, F_merito())
FINACCION

```

en el cual, `Lee_superficie` y `Lee_fun_merit` son dos rutinas que leen los vértices de control y los puntos de función de mérito respectivamente, de sendos ficheros. Además de obtener la memoria necesaria para acomodar los datos, `Lee_superficie` proporciona, a partir del fichero de diseño, información sobre los límites de las zonas optimizables.

`Genera_vector_optim` utiliza los vértices de control y la información sobre zonas optimizables y zonas fijas para construir un vector con los vértices y/o pesos optimizables, con el que se alimenta a la función de optimización `Optimiza`.

Como algoritmo de optimización se ha escogido el método Simplex, debido a Nelder y Mead [NM65], ya que no necesita la evaluación de las derivadas de la función de mérito, solamente realiza evaluaciones del valor de la misma y, por otra parte, es un algoritmo muy robusto (ver [PFSV88]).

6.1.1 El método Simplex

Un *simplexo* en el espacio de n dimensiones es un poliedro generalizado de $n+1$ vértices. Para $n=2$ es el triángulo, para $n=3$ el tetraedro. Un simplexo es *no degenerado* si engloba un volumen interior n -dimensional finito.

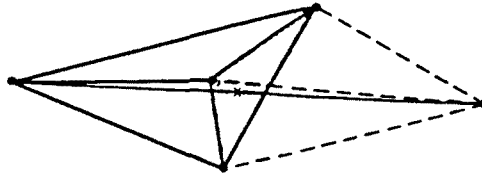
Sea P_0 un punto cualquiera de un simplexo no degenerado. En estas condiciones los n puntos restantes están cada uno sobre una de las n direcciones del espacio n -dimensional.

El método del simplexo parte de un punto de partida inicial que es una primera configuración del simplexo, esto es, $n+1$ puntos. Cualquier otro punto se puede hallar entonces como

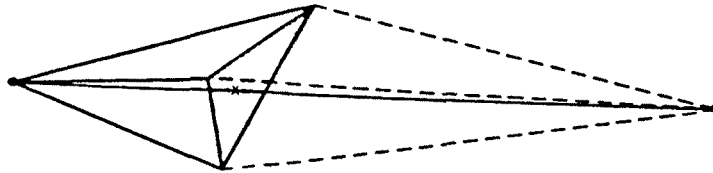
$$P_i = P_0 + \lambda_i e_i$$

siendo e_i el i -ésimo vector unitario del espacio n -dimensional, y donde λ_i es una distancia arbitraria, que en particular puede ser la misma para todos los puntos. A partir de el punto de partida se realizan una serie de movimientos de los puntos, cada uno de los cuales puede ser de una de las siguientes categorías.

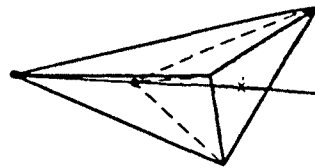
- Una *reflexión*, basada en mover un punto a su posición simétrica respecto de la cara opuesta del simplejo.



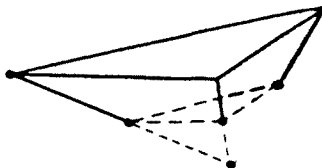
- Una reflexión afectada de un factor de expansión.



- Una contracción respecto de una dimensión, desde el punto de valor más alto.



- Una contracción respecto de todas las dimensiones hacia el punto más bajo. (Dibujos tomados de [PFSV88].)



Una combinación adecuada de las anteriores transformaciones siempre converge hacia un mínimo de la función [PFSV88]. Cuando el simplejo puede, se expande para avanzar más deprisa hacia el mínimo, mientras que cuando atraviesa un “desfiladero” se contrae correspondientemente.

El criterio de finalización consistente en terminar el proceso cuando entre una iteración y la siguiente la distancia recorrida es menor que una cierta tolerancia o bien cuando la diferencia entre dos valores consecutivos de la función está por debajo de una cierta cantidad. Igualmente se termina si el número de iteraciones supera uno prefijado.

Aunque éste es un algoritmo conocido, del que se puede encontrar una implementación explícita en [PFSV88], esbozaremos el esqueleto del mismo, ya que será necesario a la hora de distinguir la parte del algoritmo intrínsecamente secuencial de la parte susceptible de paralelización.

```

ACCION Optimiza(superficie,vector,ftol, F_merito())
  itmax = 500
  fact_a = 1.0
  fact_b = 0.5
  fact_c = 2.0
  numit=0
  Genera_partida(p_mat,y)
  MIENTRAS no salir HACER
    Busca_alto_bajo(p_mat, y, alto, bajo, segundo)
    (* Halla el punto mejor, peor y segundo mejor *)
    rtol=2.0*abs(y(alto)-y(bajo))/
      (abs(y(alto))+abs(y(bajo)))
    SI rtol < ftol o numit > itmax ENTONCES
      salir=cierto
    SINO
      ynuevo=Reflex(p_mat,y,F_merito(),
        alto,-fact_a)
      SI ynuevo <= y(bajo) ENTONCES
        ynuevo=Reflex(p_mat,y,F_merito(),
          alto,fact_c)
      SINOSI ynuevo >= y(segundo) ENTONCES
        yaux=y(alto)
        ynuevo=Reflex(p_mat,y,F_merito(),
          alto,fact_b)
      SI ynuevo >= yaux ENTONCES
        ynuevo=Contrae(p_mat,y,F_merito(),
          bajo,fact_b)
      GET_PSUM
    FINSI
  FINSI
  FINSI
  numit=numit+1

```

```

FINMIENTRAS
FINACCION

```

Aquí `p_mat` es una matriz de `ndim+1` por `ndim` elementos que contiene el punto de partida de la optimización, y `ndim+1` variaciones de sus puntos de control modificables. Está generada por `Genera_partida`. “`y`” es un vector que contiene la evaluación de la función de mérito `F_merito` en cada una de las `ndim+1` configuraciones de `p_mat`. `Reflex` es la función encargada de realizar la reflexión. Ello se realiza según un factor que, si es menor que 1 en valor absoluto, la convierte en una contracción respecto de una dirección; si es negativo produce una reflexión propiamente dicha y si es positivo la convierte en una expansión, siempre sobre la dirección del punto considerado. `Contrae` realiza una contracción omnidireccional hacia el punto más bajo.

Estas 2 funciones proporcionan los cuatro modos de transformación del simplejo antes mencionados. Ambas realizan la evaluación de la función de mérito `F_merito` en la configuración calculada, sea contracción o reflexión.

`F_merito` es una función que toma como argumento vector y que, teniendo en cuenta la matriz de magnitudes especificados para la función de mérito leídas mediante `Lee_fun_merit`, devuelve un número equivalente al valor de mérito, calculándolo de la siguiente forma:

```

FUNCION F_merito(vector)
  Reconstruye(superficie, vector)
  PARA i=0 HASTA n_merito PASO 1 HACER
    t_i = trozo_i(i)
    t_j = trozo_j(i)
    u   = par_u(i)
    v   = par_v(i)
    SI modo == bspline ENTONCES
      astig=Calc_ast_BS(superficie, t_i, t_j, u, v)
    SINOSI modo == nurbs ENTONCES
      astig=Calc_ast_NURBS(superficie, t_i, t_j, u, v)
    FINSI
    m_ast=m_ast+(astig-ast(i))^2*peso(i)
  FINPARA
  F_merito=m_ast
  DEVUELVE F_merito
FINFUNCION

```

En donde `trozo_i`, `trozo_j`, `par_u`, `par_v`, `ast` y `peso` son vectores de tamaño `n_merito` que contienen los valores del trozo bicúbico, los

parámetros u y v dentro del trozo, el astigmatismo deseado y el peso concedido, respectivamente, a los puntos donde se quiere evaluar la función de mérito. Como vemos ello se realiza para los n_merito puntos existentes en el fichero de función de mérito leído.

La función `Calc_ast_BS` calcula el astigmatismo de la superficie interpolada mediante B-Splines. Análogamente ocurre con `Calc_ast_NURBS` para la superficie interpolada mediante N.U.R.B.S.. En la implementación real se calculan a la vez el astigmatismo, la potencia y los ejes principales, lo que es más eficiente que hacerlo por separado. Por mor de la claridad se presenta aquí solo con el cálculo del astigmatismo.

6.2 Paralelización

6.2.1 Paradigmas de paralelización

Se nos plantea ahora el problema de cómo dividir nuestro algoritmo en tareas independientes realizables por separado. Esta división no es, en general, única [Mat87] y la paralelización se puede acometer a distintos niveles dentro del algoritmo.

Por otra parte, es fundamental [Atk87] desacoplar las comunicaciones del cálculo, para evitar que éste dependa de aquellas. Aún más, si el tiempo requerido para enviar la información a cada procesador es comparable con el tiempo que pasa éste calculando, puede ocurrir que la paralelización de un algoritmo sea más ineficiente que su contrapartida secuencial.

Distintos trabajos ([Pri90], [MS87]) reseñan fundamentalmente tres categorías de tipos de algoritmos susceptibles de paralelización en la práctica. Hay que notar que los algoritmos reales pueden perfectamente solapar características de varias categorías a la vez; incluso [Pri90] propone una cuarta categoría, denominada de métodos híbridos.

Tuberías

En una tubería el resultado de las operaciones realizadas en un procesador alimenta al siguiente y así sucesivamente para todos ellos, como en una línea de producción en cadena. De esta forma, una vez está llena la tubería, a la introducción de nuevos datos por un extremo de la tubería, corresponde la extracción de resultados por la otra, consumiendo en ello tan sólo un ciclo de operaciones.

En este caso existe una dependencia secuencial entre cada una de las etapas de la tubería y su rendimiento total está limitado por el rendimiento de la

más lenta de sus partes. Obviamente, el mejor caso se dará cuando todos los elementos de la tubería procesen datos al mismo ritmo [MS87].

En el caso de un algoritmo especificable de esta forma, se puede, mediante una *bufferización* [Atk87] adecuada de la entrada y la salida, eliminar cuellos de botella y mantener permanentemente ocupados al procesador y a las líneas de comunicación.

El modelo geométrico

Este modelo se basa en el hecho de que los datos (de entrada, salida o ambos) se pueden considerar distribuidos de forma geométrica. La cantidad de cálculos a realizar en cada elemento ha de ser básicamente constante. Ejemplos de ello son procesamiento de imágenes, búsqueda en bases de datos y aritmética de vectores y matrices.

La técnica a emplear en este caso consiste en dividir el trabajo en segmentos que se distribuyen equitativamente entre los procesadores disponibles. Si el tamaño de las tareas asignadas a los diversos procesadores es similar, todos ellos terminarán aproximadamente al mismo tiempo.

Si los segmentos se interrelacionan entre sí, la topología de la red cobra especial importancia [Pri90], pues de ella depende la minimización de las comunicaciones entre procesadores.

El modelo *farming*

Este modelo es similar al geométrico, pero procede cuando la cantidad de cálculo por elemento varía considerablemente de unos a otros. Ray-tracing y fotorrealismo son buenos ejemplos de este tipo de procesos.

En este caso se distribuye la carga de trabajo en gran número de pequeñas tareas que se distribuyen por la red, cada procesador con una tarea. Cuando acaban con una, envían de vuelta el resultado y reciben otra tarea.

Con una elección adecuada del tamaño de las tareas se puede solapar casi completamente la comunicación con el cálculo obteniendo incrementos lineales de rendimiento respecto del número de procesadores involucrados en la red.

Este tipo de algoritmos se benefician al máximo del paralelismo cuando la relación entre el tiempo de cómputo y el de transmisión de datos es muy superior a 1, dado que entonces los procesadores pasan un tiempo muy pequeño comparativamente comunicándose, mientras que la tarea de cálculo se reparte entre ellos, llegando a producirse una multiplicación de las prestaciones por un factor cercano al número de procesadores implicados.

En [MS87], [Pri90] existe una comparación entre las ventajas e inconvenientes de unos y otros a la hora de implementarlos en un sistema multiprocesador. Igualmente se discuten la factibilidad o imposibilidad de implementar los algoritmos de una forma u otra.

Nos proponemos, pues, identificar las secciones paralelizables de nuestro código y determinar qué tipo de paralelización y a qué nivel nos proporcionará el máximo rendimiento.

6.2.2 Identificación de código paralelizable

Dentro del algoritmo general de optimización, el bucle más externo es esencialmente secuencial. Por la naturaleza del método, una transformación del simplexo sigue a la anterior y no se puede realizar sin que aquélla haya terminado.

Cada uno de los procedimientos de reflexión o contracción son también secuenciales (se limitan a recolocar los puntos del simplexo). Dentro de ellos se realiza una evaluación de la función de mérito, como ya hemos visto. Ésta evaluación es básicamente paralelizable, puesto que se evalúa la misma sobre todos los puntos del fichero de mérito, y para cada punto se utiliza el mismo algoritmo (`Calc_ast_BS` o `Calc_ast_NURBS`). Por otra parte no tiene ninguna importancia el orden de evaluación de los mismos, puesto que al final sólo interesa la suma de todos los resultados parciales.

Si seguimos bajando de nivel, nos encontramos con que, para calcular las aberraciones que intervienen en la función de mérito es preciso calcular la primera y segunda forma fundamentales para hallar las curvaturas y direcciones principales (ver sección 4.3.3). Para ello se requieren las derivadas parciales de los B-Splines o N.U.R.B.S. en cuyo cálculo intervienen numerosos productos matriciales (sección 4.3.4). Éstos constituyen otro candidato natural a la paralelización.

6.2.3 Selección del nivel

Podemos escoger dos posibles vías de paralelización:

- **Alto nivel**, realizando la evaluación de la función de mérito con el mismo algoritmo en todos los procesadores, cada uno de ellos actuando sobre un subconjunto del número total de puntos a evaluar.
- **Bajo nivel**, paralelizando las operaciones matriciales.

Podemos evaluar la estrategia de bajo nivel de forma fácil, teniendo en cuenta que para realizar una multiplicación de dos matrices cuadradas de orden n se

requieren n^3 multiplicaciones y $n^2(n-1)$ sumas. Por otra parte, necesitamos mover $3n^2$ datos entre los procesadores. Sabemos (ver [ST87], [Inm88]) que el procesador Inmos T800 a 20 Mhz de reloj requiere 1050 ns para efectuar una multiplicación en coma flotante (64 bits) y 350 ns para una suma en las mismas condiciones. En la tarjeta de 9 transputers los enlaces funcionan a 20 Mbits/sec.

Llamando t_s al tiempo requerido para efectuar una suma, t_m al tiempo necesario para una multiplicación y t_t al necesario para transferir 64 bits (un número en doble precisión), la solución de la ecuación

$$n^3 t_m + n^2(n-1)t_s - 3n^2 t_t = 0 \quad (6.1)$$

nos da el orden de la matriz cuadrada a partir del cual se tarda la mismo en transmitir que en calcular el resultado. Descartando la solución $n = 0$ obtenemos

$$n = \frac{t_s + 3t_t}{t_m + t_s} \quad (6.2)$$

que en nuestro caso proporciona $n \approx 7$. En este caso, si pudiéramos implementar una tubería la podríamos alimentar eficientemente a partir de orden 7 sin necesidad de buffering y suponiendo que la cadena de multiplicaciones sea suficientemente larga. Una vez realizadas las primeras multiplicaciones, en cada ciclo de alimentación obtendríamos un resultado, con un tiempo total de multiplicación dividido por el número de transputers.

Si no implementamos el producto de matrices como una tubería, se habrá de cumplir que

$$\frac{n^3 t_m + n^2(n-1)t_s}{m} + 3n^2 t_t = n^3 t_m + n^2(n-1)t_s \quad (6.3)$$

para que el tiempo necesario para enviar, recibir y calcular con m procesadores sea comparable al de calcular con uno sólo. Resolviendo la ecuación anterior obtenemos

$$n = \frac{t_s(m+1) + 3mt_t}{(m-1)(tm + ts)} \quad (6.4)$$

que para 8 procesadores (hay que descontar el master) y los tiempos antedichos resulta $n \approx 8$. Por debajo de este orden será más lento el proceso en paralelo que en un sólo procesador. Hay que notar, además que para este orden sólo conseguiremos hacer las multiplicaciones en un sistema multiprocesador igual de lentas que en un monoprocesador. Ganancias sustanciales requieren que las matrices sean de mayor orden aún.

Nuestras matrices son de 4x4 en todos los casos y además, entre operaciones queda bastante código secuencial. Esto abona el terreno a la elección de la

paralelización de alto nivel, en la que el volumen de cálculo es muy grande en comparación con el número de datos a mover.

En concreto, una vez alimentados los procesadores con copias de las matrices de puntos de control y valores de la función de mérito, lo único que hay que pasar entre una iteración y la siguiente es el vector de puntos de control movibles, cuyo número de elementos es igual al número de dimensiones de la optimización (típicamente ≈ 120).

Las operaciones que hay que realizar entre dos evaluaciones de la función de mérito (parte secuencial) son comparaciones y asignaciones en número del orden de las decenas, mientras que la evaluación de la función de mérito (típicamente en ≈ 400 puntos) requiere del orden de ≈ 400 operaciones en coma flotante por punto, en un sistema optimizado ya al máximo para conseguir aumento de velocidad y minimización del número de operaciones.

Es claro que la relación cálculo/transmisión es muy superior a 1 (> 100 para los datos anteriores) y la relación entre la parte paralelizada y la puramente secuencial también es claramente favorable a la paralelizada (en un factor cercano a mil). De esta forma el sistema pasa la mayor parte del tiempo calculando en paralelo la evaluación de la función de mérito y muy poco tiempo en la parte secuencial del mismo.

El máximo rendimiento teórico con una tarjeta de m transputers sería de $m-1$ veces el rendimiento de un solo procesador, dado que dedicaremos uno de ellos (el maestro) a la parte secuencial y la gestión de entrada y salida. Habilitando procesos concurrentes (threads) en el procesador maestro para que también calculase parte de la función de mérito, se podría hacer este factor cercano a m , pero ello aumenta la complejidad del algoritmo.

En la tabla adjunta podemos ver la diferencia que hay entre la versión secuencial y la versión paralela y la razón entre uno y otro. Los resultados mostrados son promedio de los valores de las adiciones 1, 2, y 3, sobre B-Splines.

Tabla 6.1: Efecto de la paralelización

Versión (seg)		Ratio
Secuencial	Paralela	S/P
78475	12596	6.23

El ratio de 6.23 representa, para 8 transputers activos, un rendimiento en paralelo del 77.8%. Ello indica que aún se puede mejorar el rendimiento, optimizando las comunicaciones y eliminando tiempos muertos de espera entre que se envía y recibe la información (ver apartado siguiente).

6.3 Implementación

La implementación del algoritmo la realizaremos mediante una técnica similar a *processor farming*, pero con configuración estática (ver Apéndice B). Para ello dispondremos una tarea maestra (*Master*) encargada de la parte secuencial, de la entrada y salida desde y hacia el host, y de la distribución y recogida de información de y hacia la red de procesadores.

Por otro lado, dispondremos de una tarea esclava (*Worker*), con copias en cada uno de los procesadores de la red, que realizará la evaluación de la función de mérito en un subconjunto del conjunto de puntos del fichero, de forma que entre todos los *workers* se cubra el total de los puntos.

Finalmente utilizaremos unas rutinas para gestionar la transvase de información entre los *workers* y el *master* (*routers*). Éstas, en nuestro caso, son dependientes de la topología de la red.

6.3.1 Topología de la red

La topología escogida es la denominada Pipe9 (ver Apéndice B). La razón de esta elección es triple:

- Es la forma más simple de interconectar y depurar la red.
- Trabajos realizados por [PZ90] indican que, para algoritmos semejantes al nuestro, las diferencias entre árboles ternarios, anillos y *pipes* se convierten en prácticamente indistinguibles cuando el número de puntos que se envían a la vez a cada procesador para su cálculo es suficientemente grande, esto es, cuando se optimiza la influencia de las comunicaciones.
- Nuestra aplicación no necesita del envío de información referente a qué puntos calcular, puesto que como veremos el propio número i , que recibe cada transputer en el momento de la inicialización, le sirve para discriminar qué parte del total le corresponde calcular. Ello implica que el flujo de información es despreciable cuando calculamos la función de mérito, a diferencia de cuando inicializamos los procesadores con toda la información sobre vértices de control, etc.. Por ello la topología de la red, que afecta sólo a las comunicaciones, pierde relevancia y nos podemos decantar por la solución más sencilla.

Los enlaces físicos en este caso consisten simplemente en conectar la línea *Este* de uno con la *Oeste* del otro y viceversa, dejando los enlaces *Norte* y *Sur* libres.

Dentro de cada tranputer puede correr un número cualquiera de tareas concurrentemente, comunicándose entre sí mediante canales lógicos. Tanto las conexiones físicas como las lógicas se especifican en un fichero, que el programa de configuración (*configurer*) utiliza para crear el fichero ejecutable en la red concreta a que está destinado (Configuración estática).

6.3.2 Parte secuencial (Master)

El programa maestro se puede esquematizar como sigue:

```

ACCION Master
  Inicializa_com(in_ports, out_ports)
  Lee_entrada(sup_fich, log_fich, merit_fich, modo)
  Lee_superficie(sup_fich, superficie, limites)
  Lee_fun_merit(merit_fich)
  Codifica(buffer, superficie)
  PARA i=1 HASTA num_trans PASO 1 HACER
    Envia_datos(i, buffer)
    num=i-1
    longitud=1
    Envia_M(longitud, num, i)
  FINPARA
  vector=Genera_vector_optim(superficie, limites)
  Optimiza(superficie, vector, ftol, ndim, F_merito())
  Escribe_salida(log_fich, superficie)
FINACCION

```

Las cuatro primeras líneas se encargan de inicializar las comunicaciones y leer los datos de entrada, el fichero de mérito y los datos creados por el sistema interactivo de diseño. Una vez hecho esto se envían a cada uno de los procesadores de modo que haya una copia de la memoria del master en todos ellos.

Es de notar que *Envia_M* manda al procesador *i* un número *num* que le permite identificar que porción del conjunto de puntos de la función de mérito le corresponde calcular.

Codifica es un función que comprime todos los datos referentes a la superficie, límites de optimización y datos del fichero de mérito, de forma que la transmisión sea más eficiente. En conjunto se envían más de 10 Kbytes. para conseguir una imagen en los workers de la parte común del estado de memoria del master.

Acto seguido se genera el vector de puntos de control modificables y se entra en la rutina de optimización, que es la ya vista en 6.1.1. Sin embargo el cálculo de la función de mérito en este caso se realiza del modo siguiente:

```

FUNCION F_merito(vector)
  Reconstruye(superficie, vector)
  PARA i=1 HASTA num_trans PASO 1 HACER
    longitud = n_dim+1
    Envia_M(longitud,vector, i)
  FINPARA
  PARA i=1 HASTA num_trans PASO 1 HACER
    longitud = Recibe_m(merito_rec, no_procesador)
    merito = merito+merito_rec
  FINPARA
  SI iter <> 0 y cont MODULO 100 == 0 ENTONCES
    Escribe_salida(log_fich,superficie)
  FINSI
  cont=cont+1
  F_merito=merito
  DEVUELVE F_merito
FINFUNCION

```

Como vemos Reconstruye actualiza los valores de los puntos de control modificados por Optimiza y envía a cada procesador el vector de puntos de control modificados vector, después de lo cual, queda a la espera de recibir los resultados parciales de cada procesador para sumarlos a fin de conseguir el valor de la función de mérito en total.

6.3.3 Parte paralela (Workers)

Del otro lado de la red, uno en cada procesador, se encuentran los workers, con la forma:

```

ACCION Worker
  valor =1
  Inicializa_com_w(in_ports, out_ports)
  longitud=Recibe_W(buffer)
  Decodifica(buffer,superficie)
  longitud=Recibe_W(buffer)
  num =buffer
  MIENTRAS valor <> 0 HACER (* bucle infinito *)
    longitud=Recibe_W(buffer)

```

```

vector=buffer
merito=F_merito_t(vector, num)
longitud=1
Envia_W(longitud,merito)
FINMIENTRAS
FINACCION

```

Decodifica realiza la tarea inversa de Codifica, Recibe_W espera por la línea de comunicaciones, primero el número que ha enviado Envía_M desde el master y después el vector de puntos modificados por el método del simplexo. A partir de aquí F_merito_t evalúa la función de mérito, como ya hemos visto en la sección 6.1.1 salvo que sólo la calcula en uno de cada num_trans puntos, empezando por num. Con ello dividimos en num_trans subconjuntos los puntos a calcular y cada procesador calcula uno diferente.

```

FUNCION F_merito_t(vector, num)
  Reconstruye(superficie, vector)
  PARA i=num HASTA n_merito PASO num_trans HACER
    t_i = trozo_i(i)
    t_j = trozo_j(i)
    u   = par_u(i)
    v   = par_v(i)
    SI modo == bspline ENTONCES
      astig=Calc_ast_BS(superficie, t_i, t_j, u, v)
    SINOSI modo == nurbs ENTONCES
      astig=Calc_ast_NURBS(superficie, t_i, t_j, u, v)
    FINSI
    m_ast=m_ast+(astig-ast(i))^2*peso(i)
  FINPARA
  F_merito=m_ast
  DEVUELVE F_merito_t
FINFUNCION

```

Nótese que el bucle en que entra el worker es infinito. Siempre está esperando recibir y procesar la información. No hace falta comunicarle la finalización del proceso pues al acabar el master se finalizan todos los procesos pendientes en la red.

6.3.4 Subsistemas de redirección (Routers)

Cada worker es concurrente con un router que se encarga de la aceptación y/o redirección de los mensajes que recibe del master, y que envía hacia el

master los mensajes del worker. El router lanza tres *threads* diferentes, uno para comunicarse con la aplicación, otro para redirigir los mensajes línea “abajo” y otro para redirigir los mensajes línea “arriba” (ver Figura 6.1). El código de estos *threads* se puede representar esquemáticamente como sigue:

```

ACCION Aplicacion
  valor=1
  MIENTRAS valor <>0 HACER
    Lee_mensaje(cab,mensaje, in_p(CANAL_APLIC))
    fuente = procesador
    destino = raiz
    cab=construye(fuente,destino)
    Espera_semaforo (arriba)
    Manda_mensaje(cab, mensaje,out_p(CANAL_ARRIBA))
    Cambia_semaforo (arriba)
  FIN MIENTRAS
FINACCION

```

Aplicacion acepta mensajes de la aplicación y añade información para su redirección hacia el procesador raíz. El mensaje se pasa entonces línea arriba. El uso del puerto CANAL_ARRIBA esta protegido por el semáforo arriba. Ello es debido a que, en dirección al procesador raíz pueden emitirse simultáneamente varios mensajes, lo que obliga a utilizar un semáforo a efectos de sincronización.

```

ACCION Para_arriba
  valor=1
  MIENTRAS valor <>0 HACER
    (* Lee cabecera y mensaje *)
    Lee_mensaje(cab,mensaje,in_p(CANAL_ABAJO))
    Espera_semaforo (arriba)
    Manda_mensaje(cab, mensaje,out_p(CANAL_ARRIBA))
    Cambia_semaforo (arriba)
  FIN MIENTRAS
FINACCION

```

Para_arriba acepta mensajes del puerto de entrada CANAL_ABAJO (línea abajo) y los pasa entonces línea arriba. El uso del puerto CANAL_ARRIBA esta protegido por el semáforo arriba.

```

ACCION Para_abajo
  valor=1
  MIENTRAS valor <>0 HACER
    (* Lee cabecera y mensaje *)

```

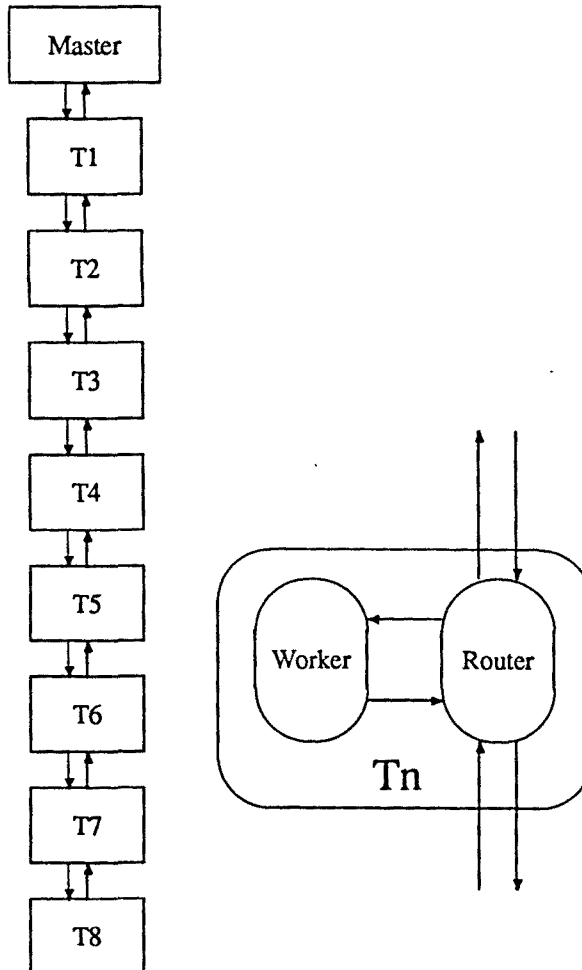


Figura 6.1: Esquema de los enlaces físicos y del flujo de mensajes entre procesadores. Éste está gobernado por los routers.

```
Lee_mensaje(cab,mensaje,in_p(CANAL_ARRIBA)
Descifra(cab,destino)
SI destino == procesador ENTONCES
    Manda_mensaje(cab, mensaje,out_p(CANAL_APLIC))
SINO
    Manda_mensaje(cab, mensaje,out_p(CANAL_ABAJO))
FINSI
FIN MIENTRAS
FINACCION
```

Para_abajo acepta mensajes del puerto de entrada CANAL_ARRIBA (línea arriba), mira el destino del mensaje y, o bien pasa el mensaje al canal de la aplicación, o si no es para ella, pasa el mensaje línea abajo. Aquí no es necesario el semáforo ya que los mensajes que van línea abajo provienen del procesador raíz en dirección al worker de algún procesador. Se emiten uno detrás de otro y por tanto no hay posibilidad de simultaneidad.

Capítulo 7

Comparación de Interpolaciones

Una vez establecida una metodología y unos algoritmos de diseño, podemos evaluar distintas formulaciones matemáticas en condiciones homogéneas. Dado que hemos descartado en principio las formulaciones que no poseen control local ni suficiente suavidad, nos proponemos evaluar B-Splines y NURBS comparativamente.

Para ello es necesario diseñar una estrategia de evaluación y comparación entre las distintas formulaciones para valores significativos de los parámetros de diseño de las lentes.

7.1 Contenido de las pruebas

En este sentido hemos preparado las siguientes pruebas: Con ayuda de nuestro sistema se han realizado 3 diseños de adiciones 1D., 2D. y 3D. respectivamente. Se han generado los puntos de partida de la optimización para los tres, en idénticas condiciones, a saber:

- Potencia de Lejos: 5 dioptrías.
- Adición : 1, 2 y 3 dioptrías.
- Diseño de pasillo codificado como "FV" (figura 7.1).
- Función de merito codificada como "FT_17X17.MER".
- Distribución de zonas optimizables codificada como "FT" (figura 7.2).

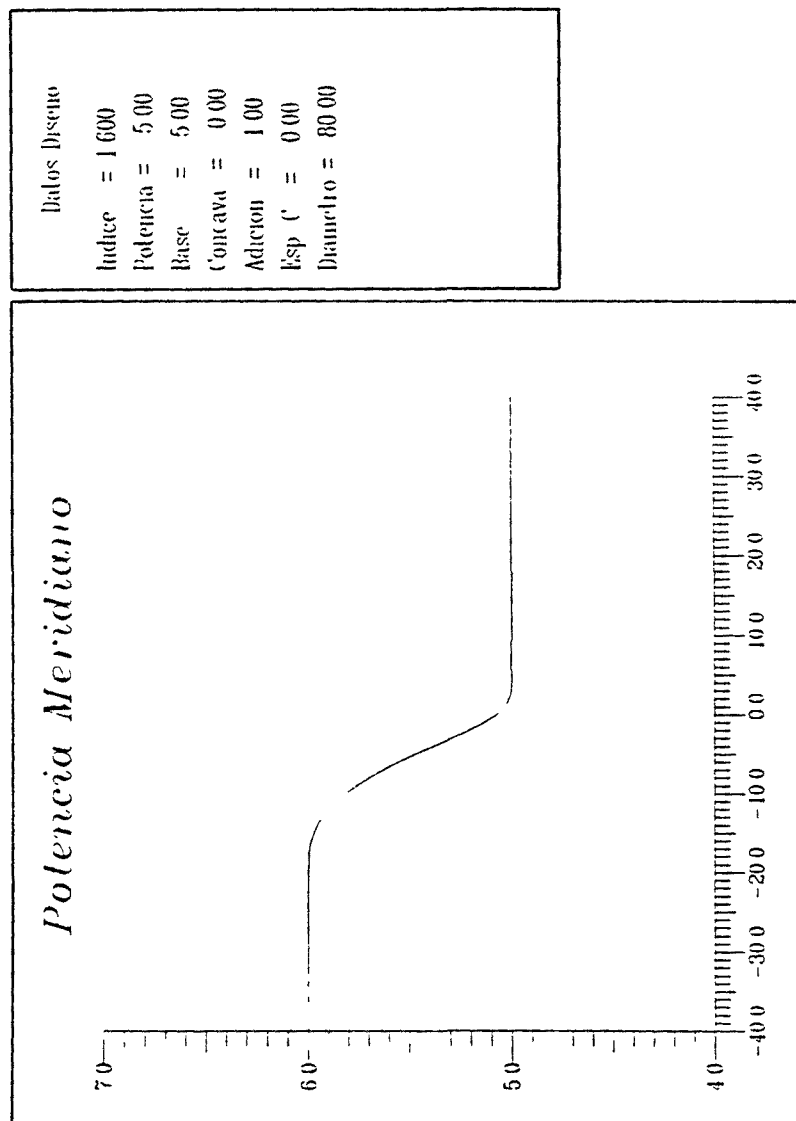


Figura 7.1: Diseño del pasillo en las adiciones de prueba. En la figura, pasillo de una adición de 1 D. Los otros dos son proporcionales a éste.

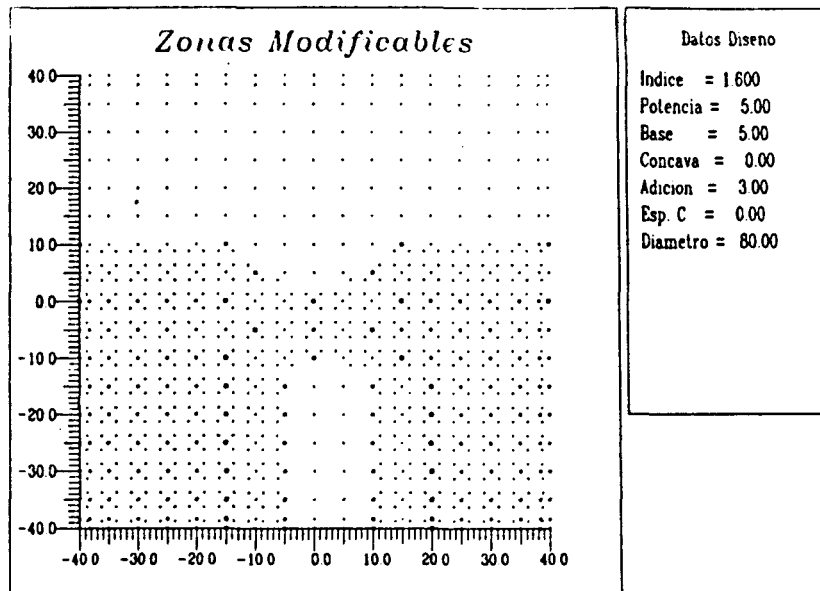


Figura 7.2: Distribución de zonas optimizables FT. (Explicación de los símbolos en sección 5.3).

Las superficies punto de partida de la optimización obtenidas a partir de estas especificaciones se pueden apreciar en las figuras 7.3, 7.4 y 7.5, que recogen la distribución de astigmatismo para las tres adiciones de referencia. Es notable la extensión de las áreas de lejos y de cerca, estando las zonas marginales confinadas en una estrecha franja a ambos lados de la progresión, con muy altos valores absolutos de astigmatismo y grandes valores del gradiente de astigmatismo en la misma.

Estos niveles de astigmatismo y de gradiente de potencia resultan difíciles de adaptar y soportar desde el punto de vista fisiológico. En las figuras se ha truncado la representación tridimensional por encima de 5 dioptrías de astigmatismo para mayor claridad, sustituyendo los valores superiores a 5 D. por valores iguales a 5 D. creando así un plano en el dibujo.

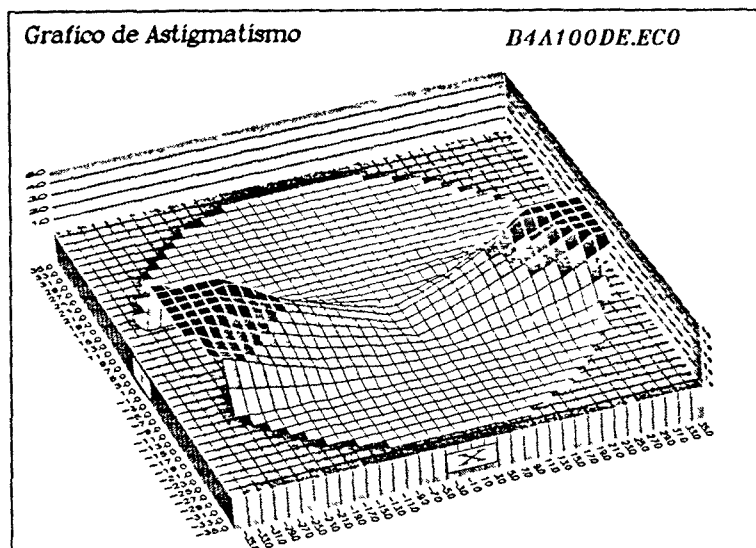


Figura 7.3: Superficie punto de partida para 1 dioptría de adición.

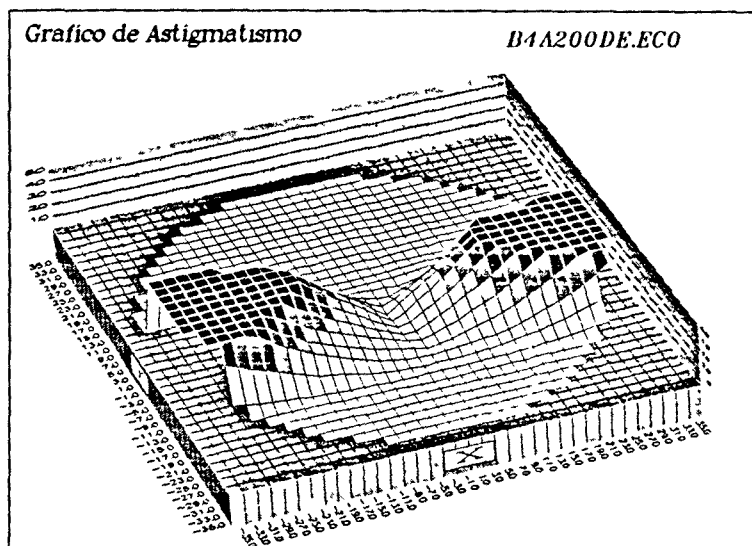


Figura 7.4: Superficie punto de partida para 2 dioptrías de adición.

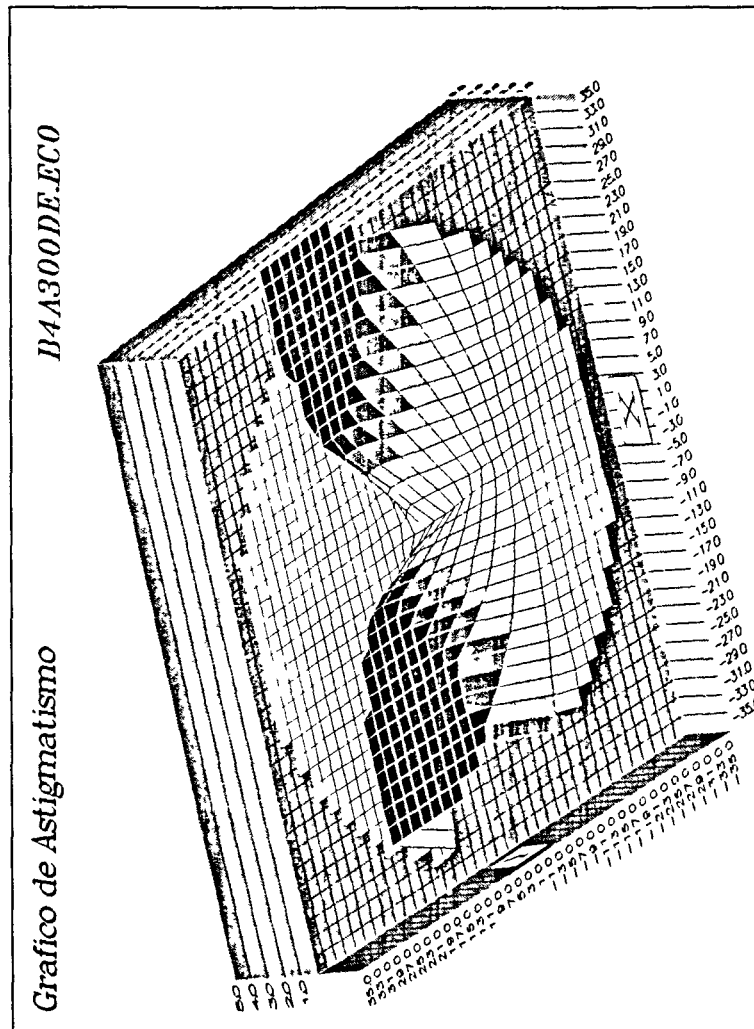


Figura 7.5: Superficie punto de partida de la optimización para 3 dioptrías de adición.

7.2 Variantes de Optimización

La función de mérito se ha planteado con el propósito exclusivo de minimizar el valor del astigmatismo en toda la superficie de las zonas marginales, sin imponer restricción alguna a la orientación de los ejes o a la distribución de potencias.

Las optimizaciones se han planteado con las variantes siguientes:

1. Optimización del punto de partida mediante B-Splines, moviendo los vértices de control de las zonas móviles.
2. A partir del resultado anterior continuar la optimización usando NURBS, pero variando sólo los pesos de las zonas modificables, sin variar los vértices de control de sus posiciones.
3. Optimización del punto de partida, mediante NURBS, modificando únicamente los pesos asociados a los vértices de control.
4. Optimización del punto de partida, mediante NURBS modificando simultáneamente los vértices de control y sus pesos asociados, siempre dentro de las zonas modificables.

7.3 Función de mérito

Ya vimos, en la ecuación 5.1, que la función de mérito se puede expresar, en general como

$$f = \left[\frac{1}{n} \sum_{i=1}^n P_i D_i^k \right]^{1/k}$$

siendo

$$D_i = V_i' - V_i, (1 \leq i \leq n)$$

donde V_i' representa el valor a conseguir y V_i el valor real en un momento dado. En nuestro caso, los valores deseados se consignan en un fichero, que denominamos *de función de mérito*.

El fichero de función de mérito "FT_17X17.MER" tiene el formato mencionado en la sección 5.3.1. Consta de 444 puntos distribuidos en 232 para el lado izquierdo y 212 para el lado derecho de la lente. Recordemos que, gracias al control local optimizamos áreas restringidas por separado.

Por otra parte optimizamos sólo el ojo derecho ya que el izquierdo se puede obtener por imagen especular de aquél. Hay más puntos en un lado de la lente que en el otro, debido a que la trayectoria convergente de la mirada hace el diseño de la lente disimétrico.

Mediante este fichero se puede modificar la cantidad y situación de los puntos en los que se evalúa la función de mérito así como los valores deseados de astigmatismo potencia y ejes, afectados del peso relativo de ese punto con respecto a los demás. De esta forma se puede evaluar la función de mérito, con gran facilidad, eliminando la necesidad de calcular las coordenadas del trozo bicúbico asociadas a un punto cartesiano determinado.

Este fichero en concreto ha sido diseñado para obtener una distribución de astigmatismos suave en toda la zona marginal, concediendo más importancia (mediante los pesos) a la zona cercana al centro de la lente que a la periferia de la misma, ya que el ojo, [GOP*87] y [POS87], normalmente no gira más allá de 10 a 15 grados. Por encima de este valor se gira la cabeza. Ello implica que la distribución correcta de los astigmatismos es más importante alrededor del centro.

No obstante, no se puede descuidar, la localización de los ejes principales en la periferia de la lente, debido a que la visión periférica utiliza esta porción de la misma y en ella la ortoscopia es muy importante para la percepción coherente de formas y movimientos (ver [Mou88]).

7.4 Realización

Inicialmente se realizó una tanda de pruebas optimizando la función de mérito en el espacio paramétrico, esto es, dados el trozo bicúbico definido por la fila i y la columna j y los parámetros u y v , se calcula siempre en estos parámetros, independientemente del punto cartesiano que definan en cada momento. En el caso de B-Splines el punto cartesiano asociado siempre es el mismo, puesto que sólo permitimos el movimiento de las componentes z de los puntos de control. Así pues, optimizar en el espacio cartesiano o en el paramétrico es equivalente, con la única salvedad de que es más rápido hacerlo en el paramétrico, puesto que no hay que encontrar el trozo bicúbico y los parámetros u y v a partir de x e y .

No ocurre lo mismo con los NURBS, en los cuales, la modificación del valor del peso afecta a las coordenadas x , y , z . Por ello, en este caso no existe equivalencia entre la optimización en el espacio paramétrico y el cartesiano, a menos que los pesos asociados a cada punto del espacio paramétrico se calculen en función del punto del espacio cartesiano correspondiente. Esto se consigue utilizando una función que represente el valor de los distintos pesos de mérito en todo el espacio cartesiano.

Para cotejar los resultados obtenidos se realizó una segunda tanda de pruebas con NURBS, en los cuales se sustituyó la optimización en el espacio paramétrico

por el espacio cartesiano. Esto obliga a buscar el trozo bicúbico y los parámetros u y v a partir de las coordenadas cartesianas. Esto, que en el caso B-Spline con distribución equiespaciada de los puntos es directo, se convierte en un problema de minimización en el caso de los NURBS.

Ello es debido a que en un caso las u -líneas y las v -líneas son rectas ortogonales y en el otro no son líneas rectas, sino que presentan ondulaciones, dependiendo de los valores de los pesos correspondientes a los vértices de control en cuestión.

Partiendo de el trozo bicúbico t_i, t_j utilizamos el método de minimización de Polack-Ribiere [Pol71], implementado según [PFSV88]. Este método hace uso del conocimiento del gradiente de la función a minimizar, que no es otra que

$$f(x', y') = (x' - x)^2 + (y' - y)^2 \quad (7.1)$$

donde x e y son las coordenadas cartesianas que queremos obtener y x' e y' son las coordenadas correspondientes a los parámetros u y v actuales en el trozo t_i, t_j .

El algoritmo de localización del trozo y los valores de los parámetros u y v , partiendo de un trozo bicúbico t_i y t_j probable, tiene la forma siguiente:

```

ACCION Localiza_x_y(x, y)
  ftol=1.0e-4
  fret=2*ftol
  avis1=0
  avis2=0
  MIENTRAS fret > ftol Y avis1 == 0 O avis2 == 0
    fret=minimiza(u,v, ftol, funcion, gradiente)
    SI conta MODULO 2 == 0 ENTONCES
      conta=conta+1
      SI u-0.0 < ftol y t_i > 0 ENTONCES
        t_i=t_i-1
        u=0.99
      SINOSI 1.0-u < ftol y t_i < nvertexh-4 ENTONCES
        t_i=t_i+1
        u=0.01
      SINO
        avis1=avis1+1
      FINSI
    SINOSI conta MODULO 2 == 1 ENTONCES
      conta=conta+1
      SI v-0.0 < ftol y t_j > 0 ENTONCES

```



```

        t_j=t_j-1
        v=0.99
    SINOSI 1.0-v < ftol y t_j < nvertexv-4 ENTONCES
        t_j=t_j+1
        v=0.01
    SINO
        avis2=avis2+1
    FINSI
FINSI
FINMIENTRAS
FINACCION

```

El contador `conta` obliga a cambiar alternativamente la secuencia de evaluación del trozo t_i y el trozo t_j para evitar que, en algunos casos, el algoritmo quede atrapado en una serie infinita de oscilaciones.

`avis1` y `avis2` sirven al propósito de evitar que el hallazgo de un mínimo local superior a la tolerancia `ftol` cuelgue el algoritmo en un bucle infinito.

Ello nos permite encontrar los parámetros u y v que producen el punto de coordenadas x e y requeridas. Puede ocurrir, sin embargo, que uno o ambos valores se hagan 0 o 1, sin que se haya encontrado el mínimo. En este caso se pasa al trozo contiguo disminuyendo en una unidad el índice t_i si es el parámetro u el que se ha hecho 0 o aumentando una unidad si se ha hecho 1. Análogamente para el índice t_j si es el parámetro v el que se ha ido a 0 o a 1.

Una vez en el otro trozo bicúbico se repite la minimización hasta hallar el punto para el cual `fret`, que es el valor de la función de mérito, sea menor que la tolerancia, o nos conduzca siempre al mismo lugar, momento en el que estaremos en el mínimo.

`nvertexh` y `nvertexv` son el número de perfiles (puntos de control en sentido horizontal) y puntos por perfil (puntos de control en sentido vertical) de la malla de vértices de control. Los trozos bicúbicos tienen índices que van desde 0 hasta `nvertexh-4` y desde 0 hasta `nvertexv-4`.

7.5 Resultados

Para evaluar comparativamente los resultados de las distintas optimizaciones hemos realizado la representación gráfica de las mismas, para las distintas variantes de optimización (ver sección 7.2). Además, para poder evaluar numéricamente las diferencias entre unos resultados y otros, hemos creado una

rutina de evaluación que asigna unos valores a cada aberración en función del criterio de mérito siguiente:

$$Ast = \sum_i (A_i)^2 p_i$$

$$Pot = \sum_i (P_i - D_n)^2 p_i$$

$$Eje = \sum_i (E_i - 90)^2 (1 - p_i)$$

donde A_i es el astigmatismo expresado en dioptrías, P_i es la potencia media en dioptrías, D_n es la potencia nominal en dioptrías de la superficie en visión de lejos y E_i es la orientación, en grados sexagesimales, del eje principal asociado al mayor radio de curvatura principal, en el punto i . p_i es el peso asociado a ese punto en la función de mérito que se usa para optimizar las superficies.

El criterio elegido evalúa cuánto se aparta nuestra superficie de una esfera (astigmatismo deseable = 0) de potencia D_n , la de visión lejana, y cuyos ejes principales se desean orientados perfectamente según la vertical (ortoscopia máxima). En el caso de los ejes, se multiplica por 1 menos el peso del punto, ya que, al revés que en el astigmatismo o la potencia, estos importan menos cuanto más en el centro de la lente se está. Esto es debido a que allí los astigmatismos vienen determinados por la forma del meridiano, en virtud del teorema de Minkwitz, en cambio en las zonas más alejadas del centro es importante mantener los ejes bien orientados para evitar los efectos indeseables sobre la visión extrafoveal.

En función de esos valores podemos establecer una tabla comparativa entre las diversas variantes (véase la tabla 7.1).

Por lo que se refiere a los tiempos invertidos en optimizar cada una de las dos tandas, la tabla 7.2 nos permite ver claramente las diferencias existentes entre ambos modos de realización, claramente desfavorables al cálculo de la función de mérito en el espacio cartesiano. La diferencia en la razón de tiempos totales entre la tanda 1 y la tanda 2 sugiere que el simplexo ha recorrido caminos diferentes por el espacio de configuración hasta llegar al mínimo. No hay que olvidar que en un caso estamos utilizando una función para calcular los pesos desde el espacio paramétrico, mientras que en la otra utilizamos directamente los valores fijos del fichero de mérito, buscando el trozo bicúbico y los parámetros en cada caso.

Estos resultados se pueden apreciar en forma gráfica en las figuras 7.6 a 7.17. Se han representado solamente los gráficos correspondientes a la adición de 3D, para evitar una proliferación excesiva de ilustraciones.

Dado que el astigmatismo es la aberración que se pretende optimizar, hemos incluido un gráfico 3D del mismo, junto con su representación en forma de curvas de nivel (Figs. 7.6, 7.9, 7.12, 7.15). Igualmente, se incluyen gráficos de curvas de nivel de la potencia media D_m (Figs. 7.7, 7.10, 7.13, 7.16) y de la orientación de los ejes principales (Figs. 7.8, 7.11, 7.14, 7.17).

En este último tipo de gráfico, la orientación de los ejes viene dada por la del segmento rectilíneo dibujado en un cierto número de posiciones. La longitud del mismo es proporcional a la magnitud del astigmatismo en ese punto. Con ello se pone de relieve el hecho de que la orientación de los ejes principales es tanto más importante desde el punto de vista fisiológico, cuanto mayor es el astigmatismo. Es evidente que si el ojo humano no percibe un astigmatismo menor de 0.5 D. , poco importa la orientación del mismo.

Si examinamos las figuras antedichas y las confrontamos con los valores de la tabla 7.1, podemos observar que la evaluación cuantitativa realizada se corresponde con las apreciaciones cualitativas que se pueden hacer de las mismas.

Efectivamente, podemos ver que el modo 3 produce las mayores y más desordenadas aglomeraciones de astigmatismo y potencia, así como los ejes mas desorientados. Entre los tres modos restantes, el modo 1 presenta mayor astigmatismo en promedio, pero menores gradientes y variaciones de potencia y los ejes claramente mejor ordenados de los cuatro modos. El modo 2 y el 4 son similares en cuanto astigmatismo, con mejores orientaciones de ejes y suavidad en la potencia para el modo 2 y menor nivel promedio de astigmatismo en el modo 4.

Tabla 7.1: Resultados

Interpolación	Modo	Astigmatismo	Potencia	Ejes
Adición : 1 D.				
B-Splines	1	222	34266	152321
N.U.R.B.S	2	122	35643	377929
	3	1449	40746	732955
	4	112	36574	401401
Adición : 2 D.				
B-Splines	1	875	36742	144822
N.U.R.B.S	2	535	39545	348751
	3	7582	52933	727395
	4	521	41586	369077
Adición : 3 D.				
B-Splines	1	1981	39362	142920
N.U.R.B.S	2	1162	43540	326957
	3	14258	67478	714516
	4	1280	48926	331040

Tabla 7.2: Tiempo de C.P.U. (en segundos)

Interpolación	Modo	Tanda 1	Tanda 2
Adición : 1 D.			
B-Splines	1	12014	11740
N.U.R.B.S	2	28006	152423
	3	23477	—
	4	280895	—
Adición : 2 D.			
B-Splines	1	15614	20074
N.U.R.B.S	2	27446	215857
	3	16788	—
	4	278566	—
Adición : 3 D.			
B-Splines	1	14371	16843
N.U.R.B.S	2	24702	171566
	3	11582	—
	4	317639	—

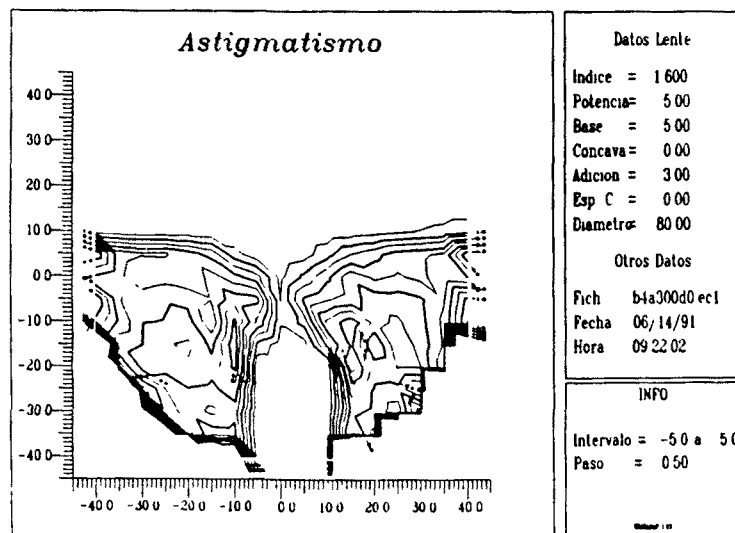
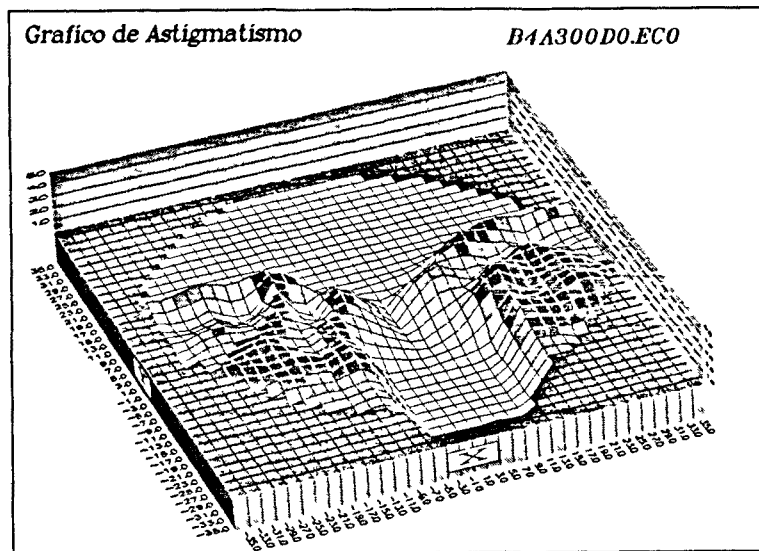


Figura 7.6: Astigmatismo de la superficie optimizada. Adición de 3 D. modo 1.

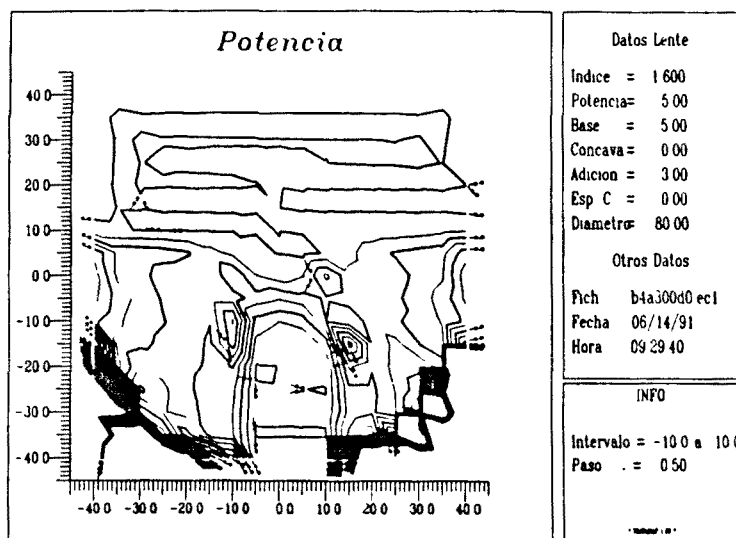


Figura 7.7: Potencia media de la superficie optimizada. Adición de 3 D. modo 1.

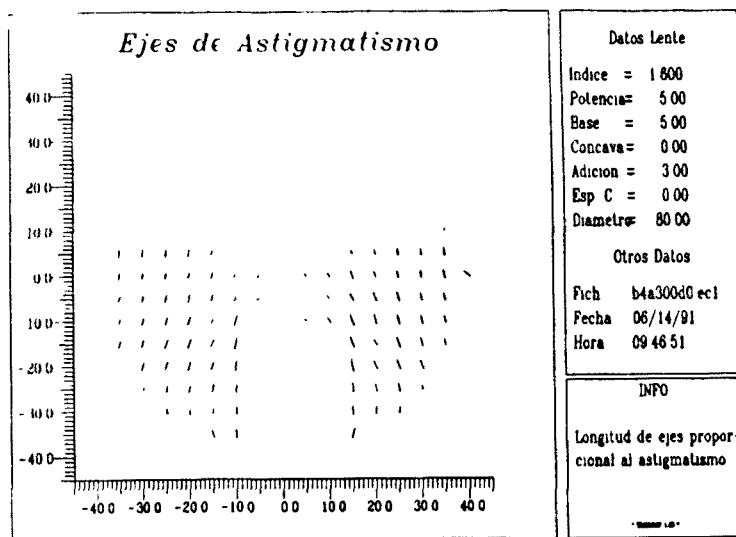


Figura 7.8: Ejes principales de la superficie optimizada. Adición de 3 D. modo 1.

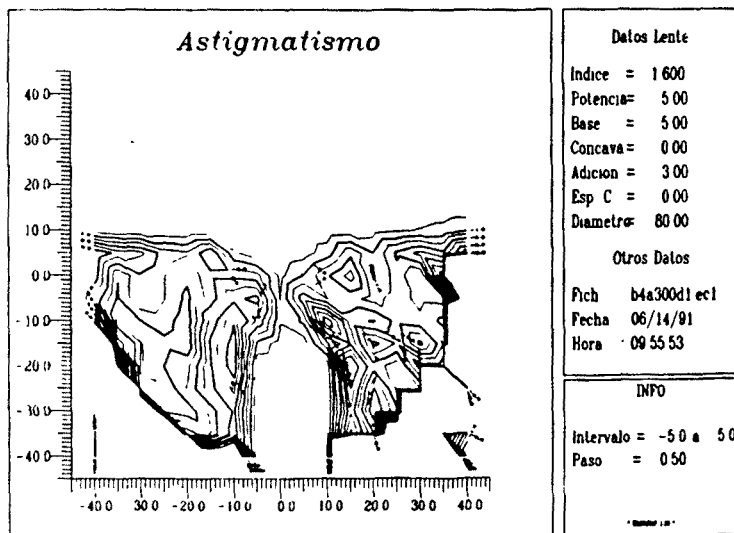
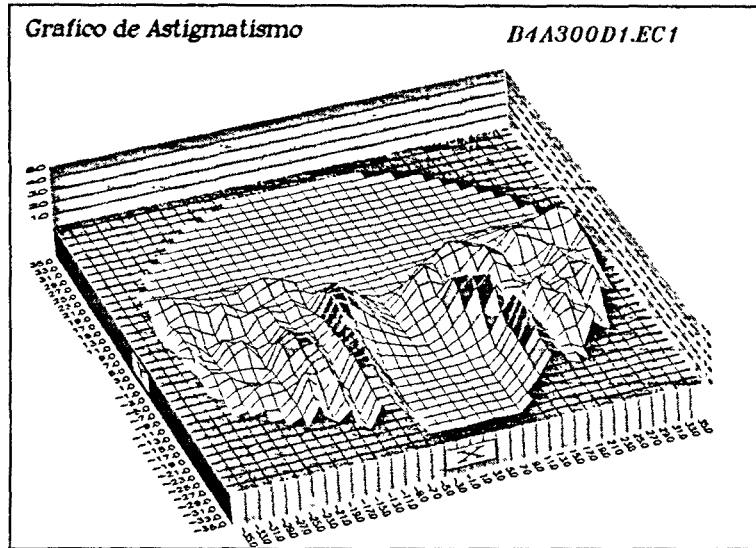


Figura 7.9: Astigmatismo de la superficie NURBS optimizada. Adición de 3 D. modo 2.

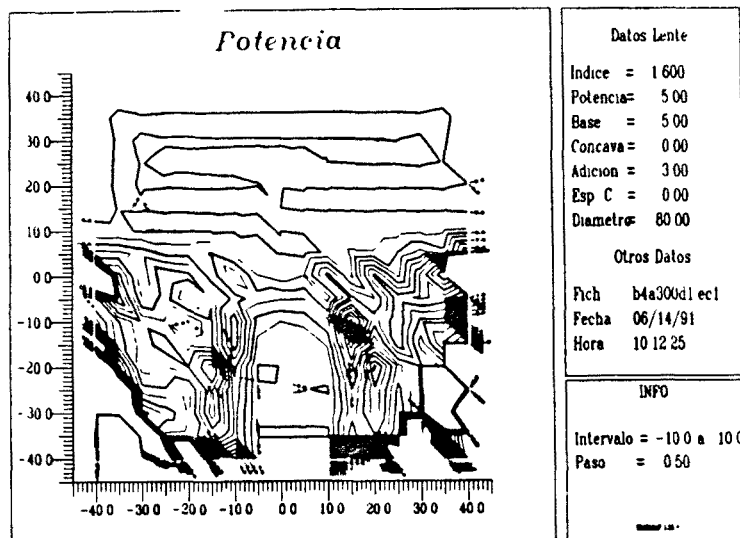


Figura 7.10: Potencia media de la superficie optimizada. Adición de 3 D. modo 2.

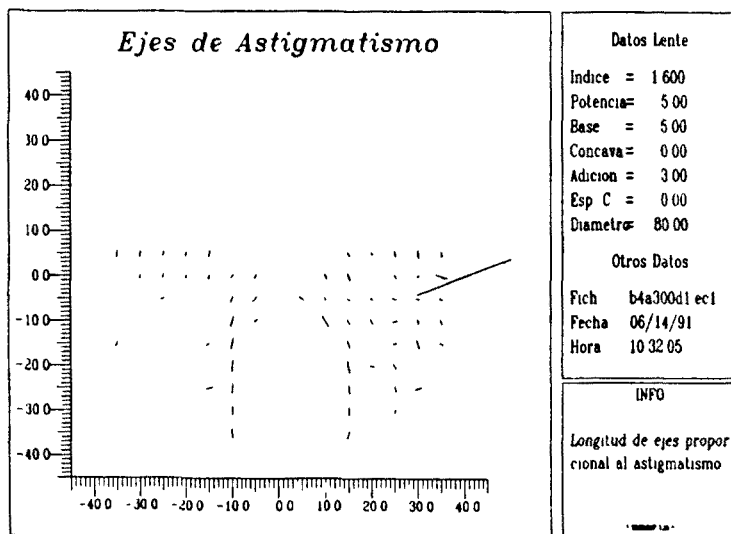


Figura 7.11: Ejes principales de la superficie optimizada. Adición de 3 D. modo 2.

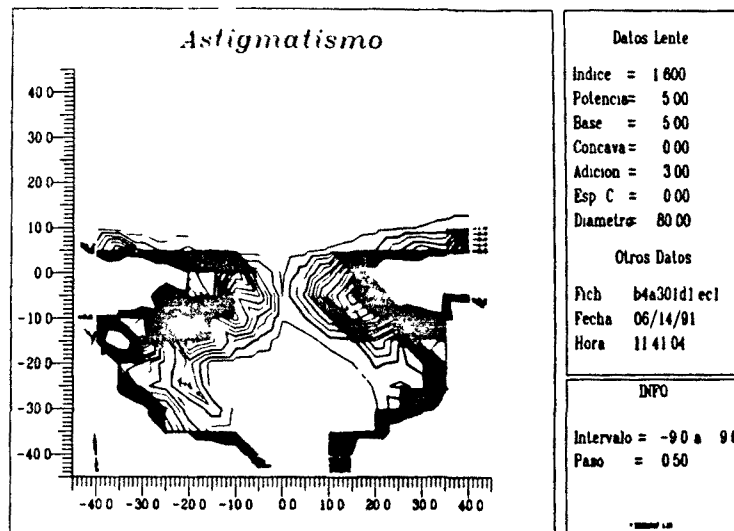
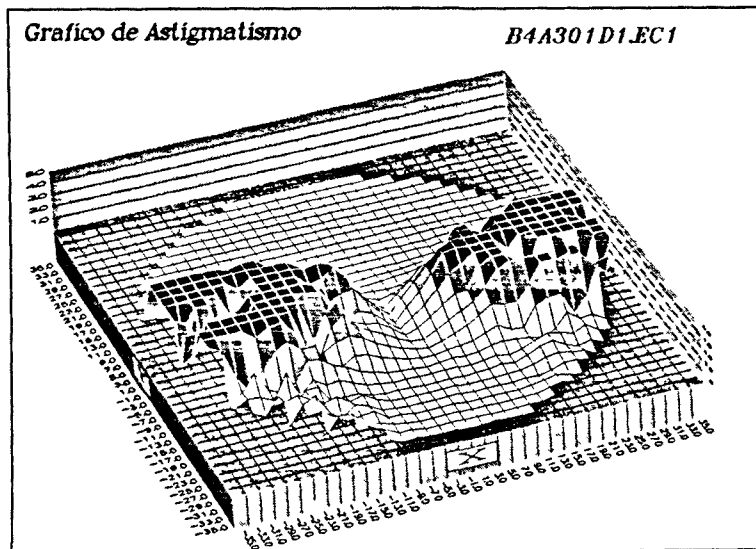


Figura 7.12: Astigmatismo de la superficie NURBS optimizada. Adición de 3 D. modo 3.

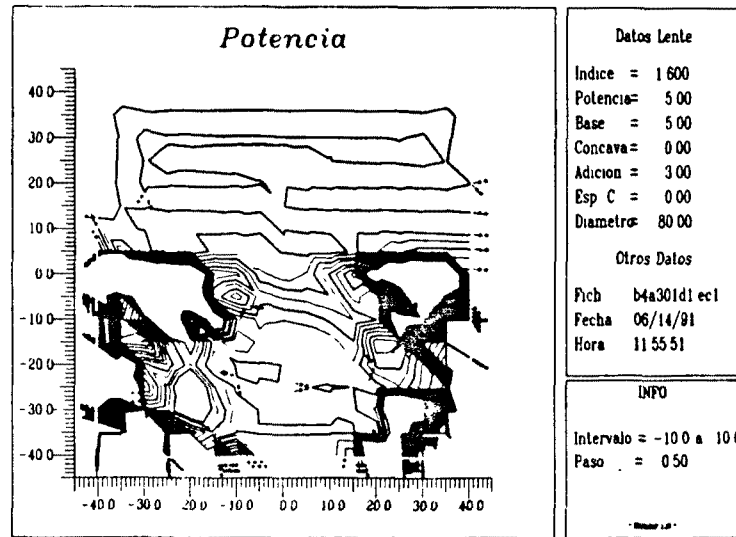


Figura 7.13: Potencia media de la superficie optimizada. Adición de 3 D. modo 3.

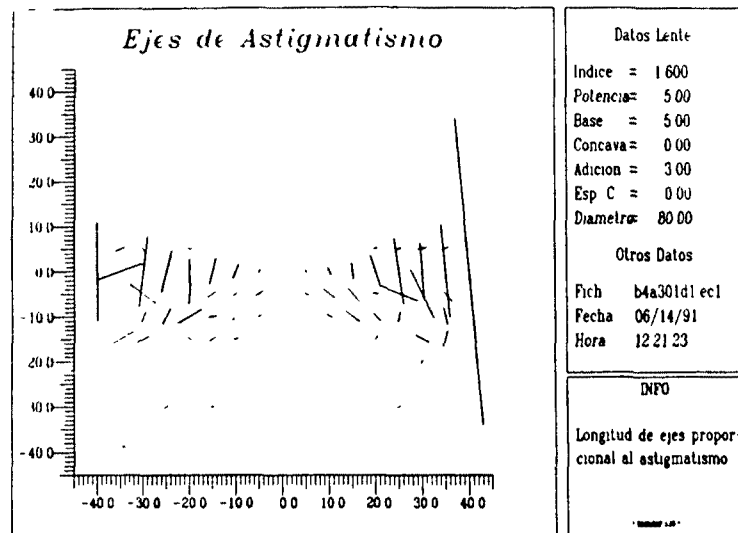


Figura 7.14: Ejes principales de la superficie optimizada. Adición de 3 D. modo 3.

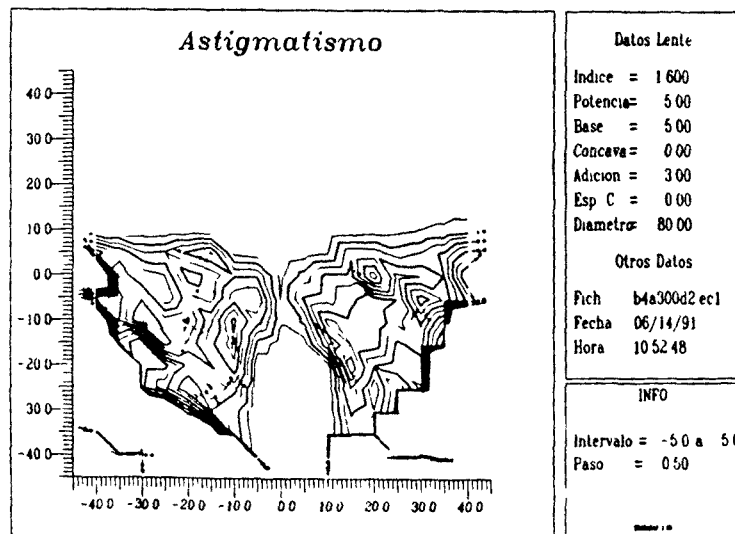
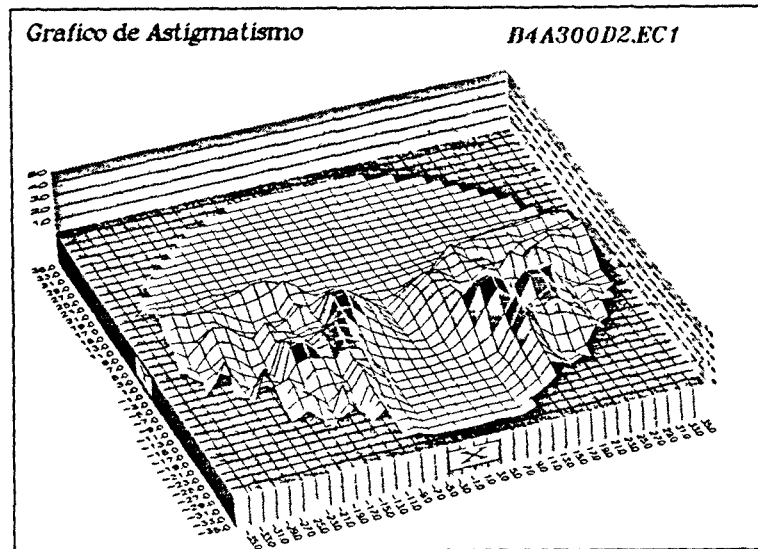


Figura 7.15: Astigmatismo de la superficie NURBS optimizada. Adición de 3 D. modo 4.

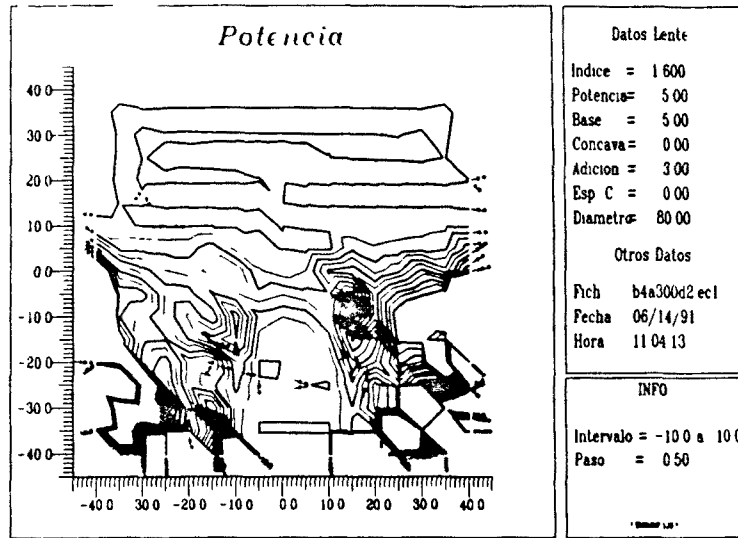


Figura 7.16: Potencia media de la superficie optimizada. Adición de 3 D. modo 4.

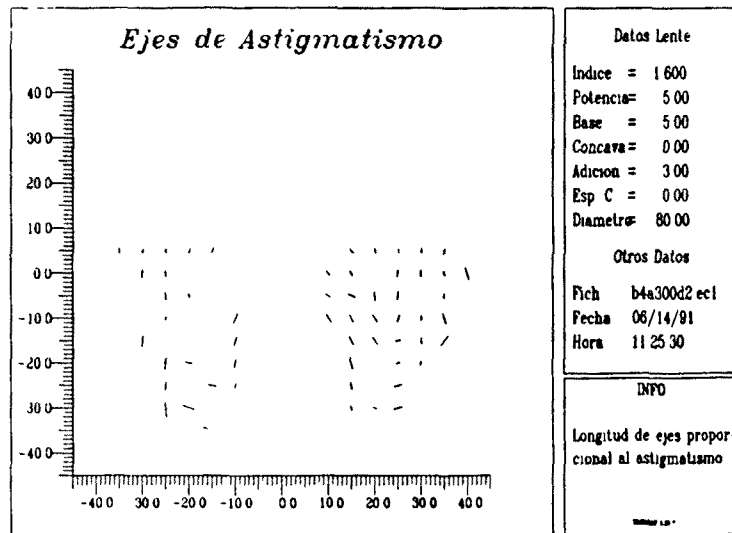


Figura 7.17: Ejes principales de la superficie optimizada. Adición de 3 D. modo 4.

7.6 Conclusiones

De los resultados anteriormente vistos podemos elaborar una tabla cualitativa referente al comportamiento de los dos tipos de interpolación estudiados. Esta es la tabla 7.3. Si tenemos en cuenta las figuras de Astigmatismo, Potencia y Ejes

Tabla 7.3: Conclusiones

Tipo de Interpolación	Modificable		Punto de Partida	Comportamiento		
	Vértices	Pesos		Astigm.	Pot.	Ejes
B-Splines	Sí	No	Original	Ref.	Ref.	Ref.
N.U.R.B.S.	No	Sí	B-Splines	Mejor	Peor	Peor
	No	Sí	Original	Peor	Peor	Peor
	Sí	Sí	Original	Mejor	Peor	Peor

de las distintas variantes optimizadas, junto con los resultados de la evaluación numérica, vemos que concuerda notablemente la apreciación sobre los gráficos con los valores dados por la rutina de evaluación.

Igualmente vemos que en las optimizaciones según los modos 2 y 4 mejoramos el astigmatismo, a costa de empeorar las variaciones de potencia y la orientación de los ejes, que no solo se apartan de sus valores ideales, sino que sufren notables variaciones locales. Esto mismo se puede decir del astigmatismo, que aunque menor en término medio que para los B-Splines, sufre en los NURBS ondulaciones importantes.

El modo 3 se ha revelado claramente descartable para la optimización de superficies progresivas dado sus bajos rendimientos. No solo produce ondulaciones en todas las superficies de aberraciones estudiadas, sino que sus valores de evaluación son claramente superiores a los de las demás variantes.

Para conseguir que los ejes y la potencia se equilibraran al modo que los hacen "espontáneamente" en los B-Splines se requeriría, para los NURBS, incluir en la función de mérito condiciones para la potencia y los ejes.

Por otra parte es destacable la diferencia en tiempo de cálculo, entre realizar la función de mérito para los NURBS en el espacio paramétrico o en el cartesiano, no encontrándose ninguna diferencia en el resultado final, si se habilita una función que nos relacione los pesos con las coordenadas x e y del punto correspondiente a la localización paramétrica preestablecida.

Podemos concluir, finalmente, que B-Splines es un tipo de interpolación adecuado al problema propuesto, que responde de una manera natural a los requerimientos y especificaciones ópticas de optimización, dejando una superficie adecuada, no sólo en cuanto a distribución y valores de astigmatismos (objeto de

la optimización), sino también en cuanto a los valores y distribución de potencias y ejes principales, que adoptan valores compatibles con los deseables desde el punto de vista óptico y fisiológico sin necesidad de plantearlo en la función de mérito.

Capítulo 8

Conclusiones y Futuros Trabajos

8.1 Aportaciones

Las aportaciones realizadas por este trabajo se pueden condensar como sigue:

- Se ha creado el dispositivo necesario para la edición interactiva de diseños y especificaciones de lentes progresivas, dentro del marco de un sistema de C.A.D. interactivo específico para Óptica Oftálmica (Capítulo 5).
- Se han utilizado B-Splines y NURBS como formalismo soporte de la representación y tratamiento de la superficie óptica de una lente oftálmica. Ello representa haber multiplicado por 20 el número de grados de libertad de los formalismos conocidos utilizados hasta el momento en óptica oftálmica para este tipo de lentes.
- Se ha establecido un procedimiento de evaluación de lentes progresivas y se ha comparado los resultados de optimizar los mismos diseños mediante ambos formalismos, a la luz del sistema de evaluación (Capítulo 7).
- Se ha puesto a punto un sistema de optimización de las superficies progresivas, utilizando un sistema multiprocesador MIMD basado en transputers. Se han creado las rutinas necesarias para la comunicación y tratamiento de los datos objeto de la optimización. (Capítulo 6).
- Igualmente se evalúa el nivel más eficiente de paralelización, la topología de la red de procesadores y el modelo de paralelismo más adecuado al algoritmo de optimización, teniendo en cuenta el hardware para el que se programa. (Secciones 6.2 y 6.3)

8.2 Conclusiones

Las conclusiones principales que se pueden extraer del trabajo realizado son las siguientes

- B-Splines se muestra como un método de interpolación idóneo para el diseño de lentes progresivas. Sus propiedades intrínsecas favorecen la distribución suave de las aberraciones y la orientación ortoscópica de los ejes principales de curvatura, de una forma natural, aún cuando sólo se optimiza el astigmatismo.
- Aún siendo NURBS un formalismo con mayor número de grados de libertad que B-Splines, la intervención de los pesos en la optimización rebaja el nivel de astigmatismo, a costa de la ortoscopia y de la regularidad de la distribución de astigmatismos y potencias medias, lo cual es negativo desde el punto de vista fisiológico, ya que se percibe como irregularidades en la visión dinámica y no favorece la visión estática.
- La implementación en paralelo del sistema de optimización está especialmente indicada en este caso, ya que, con más de 100 grados de libertad, el tiempo de cálculo es muy extenso.
- Se ha conseguido una paralelización elevada del algoritmo de optimización. Se ha ganado un factor 6.2, utilizando una tarjeta con 8 procesadores dedicados a la parte paralela del algoritmo, respecto a la velocidad de cálculo del mismo algoritmo secuencial calculado en un solo procesador. Ello representa un 77% de paralelismo conseguido.

8.3 Futuras líneas de trabajo

El sistema de diseño desarrollado es suficientemente versátil como para permitir la evaluación de otros tipos de métodos de interpolación. Aunque B-Splines se adecúa muy bien al diseño de progresivos, puede ocurrir que otro tipo de interpolación con menos grados de libertad produzca resultados equivalentes. Ello justificaría una línea de trabajo en este sentido.

La edición interactiva de funciones de mérito, elaborada hacia el final de este trabajo, se debería evaluar exhaustivamente para intentar conseguir funciones de mérito compatibles con la geometría diferencial de la superficie, conducentes a diseños con valores más bajos y equilibrados de aberraciones en las zonas periféricas de la lente.

Otro aspecto interesante es el de la optimización del factor de paralelismo conseguido, que se podría mejorar mediante una reprogramación cuidadosa del

sistema de comunicaciones entre los workers y el master. También se apunta como interesante el establecimiento de buffers que eliminasen los tiempos de espera que impone el actual sistema.

Por otra parte, sería conveniente incluir el trazado de rayos y la evaluación de las aberraciones desde el punto de vista óptico, no sólo geométrico, de cara a elaborar un criterio de calidad no basado en los promedios y diferencias de las curvaturas superficiales, sino en la aberración de la onda emergente. Esto, que hoy en día resulta aún prohibitivo, probablemente será factible, sin recurrir a supercomputadores, con la nueva serie de transputers INMOS T9000, que tendrán un rendimiento 10 veces mayor que el actual T800, siendo compatibles a nivel binario con éstos.

Apéndice A

El Ojo Humano

Para la comprensión de los fenómenos ligados a la visión es indispensable un conocimiento funcional del ojo. El ser humano es un animal predominantemente visual en lo que a proceso de la información del mundo exterior se refiere. Por otra parte, no hay que olvidar que las lentes progresivas pretenden suplementar al ojo presbita, aprovechando las cualidades fisiológicas de éste y las de adaptación y tratamiento de datos del cerebro. Conjunto sin el cual una lente progresiva deviene esencialmente inútil.

El ánimo de este apéndice no puede ser otro que el de dar unas nociones muy resumidas de la anatomía del globo ocular y de la fisiología y función del mismo.

A.1 Anatomía del ojo

El globo ocular constituye un órgano de visión cuyo cometido consiste en captar información electromagnética (comprendida entre 400 nm.–azul– y 700 nm.–rojo– de longitud de onda) del mundo exterior y enfocarla sobre un tejido fotorreceptor nervioso llamado *retina* que la traduce en impulsos nerviosos procesables por el cerebro.

La sensibilidad espectral es máxima en el ojo humano sobre los 555 nm. (ver figura A.1) que corresponde al color amarillo verdoso. La línea (1) corresponde a la sensibilidad *fotópica* (con luminosidad superior a $10^{-3} \text{cd} \cdot \text{m}^{-2}$), mientras que la (2) corresponde a la visión *escotópica* (por debajo de la citada luminosidad). Como se ve, existe un corrimiento de la sensibilidad hacia las ondas cortas en visión escotópica, denominado *efecto Purkinje*.

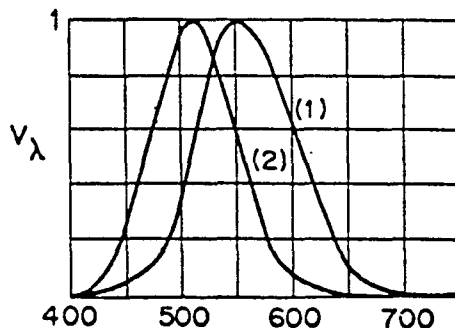


Figura A.1: Sensibilidad espectral del ojo.

No es una casualidad que el Sol emita principalmente radiación luminosa en estas longitudes de onda. El ojo humano ha sufrido una adaptación que le permite aprovechar al máximo el espectro electromagnético de la estrella que ilumina los objetos de la visión.

El globo ocular (ver figura A.2) está constituido por una envoltura aproximadamente esférica, dura y resistente denominada *esclerótica*, que le proporciona protección y donde se insertan los seis músculos que gobiernan su movimiento. En su cara anterior tiene una protuberancia en forma de casquete esférico, transparente, que se denomina *córnea*.

Dentro de la esclerótica se encuentra una membrana denominada *coroides*, grandemente vascularizada, que es la encargada de alimentar a las zonas del ojo poco o nada irrigadas. La coroides no se cierra totalmente dentro del ojo, sino que acaba, por la parte frontal, en un músculo denominado *músculo ciliar*. Este músculo tiene un orificio en su centro que se denomina *iris* y que funciona como diafragma de apertura, al ser de diámetro variable, en función de la tensión aplicada al músculo ciliar. Esta variación de apertura es automática (es un acto reflejo), provocada por la luminosidad incidente en la retina.

Detrás del iris y suspendido del músculo ciliar por un haz de fibras musculares denominadas *zónula* se halla el *crystalino*, cuerpo transparente en forma de lente biconvexa, responsable de la acomodación del ojo (su capacidad de enfoque).

El ojo internamente está dividido en dos cámaras independientes por una membrana muy fina denominada *hialoide*. La cámara anterior está llena de un líquido transparente o *humor acuoso* compuesto básicamente por albúmina y agua salada. La cámara posterior está llena del llamado *humor vítreo*, esencialmente un gel de proteínas. La presión de estos *humores* mantiene la forma del ojo y contribuye a sostener la retina unida a la coroides.

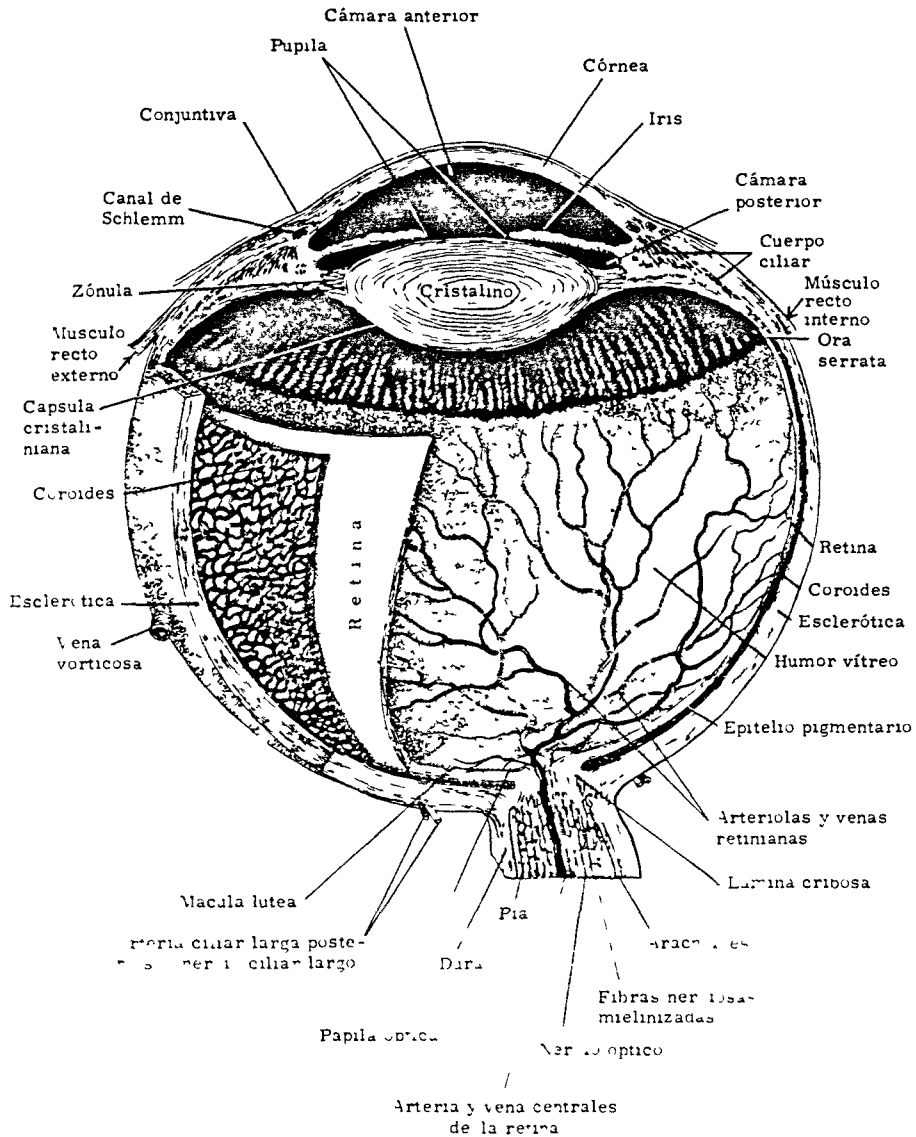


Figura A.2: Anatomía del Ojo. Corte horizontal del ojo situado en la órbita izquierda.

La *retina* es un velo de tejido nervioso situado dentro de la coroides y unido a ella sin solución de continuidad, donde se forman las imágenes. Está formada por las terminaciones nerviosas de las células terminales del *nervio óptico* que enlaza directamente con el cerebro. En su parte posterior tiene una pequeña depresión denominada *fóvea* donde se concentran el mayor número de *conos*, las células nerviosas fotorreceptoras responsables de la visión nítida en visión fotópica. En el resto de la retina predomina otro tipo de células denominadas *bastones*, de importancia en la visión escotópica y en la apreciación de movimientos y formas.

El punto por donde entra el nervio óptico se denomina *punto ciego*, en razón de que allí no existen fotorreceptores.

A.2 El ojo teórico

En 1844 Moser fue el primero en aplicar las leyes de formación de imágenes de Gauss al ojo humano. Sin embargo, una elección infortunada del índice de refracción del cristalino, le proporcionaba imágenes fuera de la retina. En 1853 Listing resolvió este problema introduciendo el llamado *ojo reducido*, cuya idea es cercana a la de Christian Huygens (1653), el cual sostenía que el ojo humano es asimilable a dos hemisferas, con un *ratio* de 1 a 3 para la razón córnea:retina, que serían sus únicas partes constituyentes.

Debemos al oftalmólogo sueco Allvar Gullstrand (1862—1930), gran parte del conocimiento moderno del sistema ocular. Partiendo de medidas propias y ajenas, construyó un modelo del ojo consistente en un sistema óptico centrado con superficies refringentes esféricas. Este modelo del ojo fue utilizado para estudiar su comportamiento óptico en la visión y en conjunción con otros sistemas ópticos, como lentes oftálmicas e instrumentos ópticos.

El ojo de Gullstrand es considerablemente menos simplista que los considerados anteriormente y su gran mérito consiste en que da una idea muy aproximada de las medidas y características promedio del ojo. No obstante, hay que resaltar que es un ojo teórico, adecuado a los propósitos de simulación óptica, no necesariamente la medida de un ojo real. En el ojo real hay gran cantidad de características que resisten a la simplificación, como es la forma claramente esférica de la córnea, la cual, según [Leg61], varía su curvatura, por ejemplo, en función del estado menstrual en la mujer, la temperatura ambiente, e incluso está sujeta a variaciones circadianas.

Lo mismo ocurre con muchos otros parámetros del ojo, que siendo intrínsecamente variables, han de ser sustituidos por valores promedio, o bien por cantidades equivalentes, dentro de una aproximación esquemática. Dentro de esta filosofía se encuentran los *ojos esquemáticos* de Gullstrand (ver Tabla A.1).

Tabla A.1: El ojo esquemático de Gullstrand

Acomodación	Ojo Exacto		Ojo Simplificado	
	Sin	Con	Sin	Con
Índice de refracción:				
Córnea	1.376	1.376		
H. acuoso y vítreo	1.336	1.336	1.336	1.336
Cristalino	1.386	1.386	1.413	1.424
Situación:				
Sup. corneal ant.	0	0	0	0
Sup. corneal post.	0.5	0.5		
Sup. ant. cristalino	3.6	3.2		
Sup. post. cristalino	0.5	0.5		
Centro óptico lente			5.85	5.2
Radios de curvatura:				
Sup. corneal ant.	7.7	7.7	0	0
Sup. corneal post.	6.8	6.8		
Sup. corneal equivalente			7.8	7.8
Sup. ant. cristalino	10.0	5.3	10.0	5.33
Sup. post. cristalino	-6.0	-5.33	-6.0	-5.33
Poder refractor:				
Sup. corneal ant.	48.83	48.83	0	0
Sup. corneal post.	-5.88	-5.88	6.8	
Sup. corneal equivalente			43.08	43.08
Sup. ant. cristalino	5.0	9.375	7.7	16.5
Sup. post. cristalino	-8.33	9.375	12.833	16.5

En el ojo esquemático *exacto* se tienen en cuenta los efectos de las aberraciones sobre el poder dióptrico del ojo.

Tabla A.2: El ojo de Gullstrand (cont.)

Acomodación	Ojo Exacto		Ojo Simplificado	
	Sin	Con	Sin	Con
Sistema corneal:				
Poder refractor (D)	43.05	43.05	43.08	43.08
Primer pto. ppal(mm.)	-0.0496	-0.0496	0	0
Segundo pto. ppal(mm.)	-0.0506	-0.0506	0	0
Primera dis. focal(mm.)	-23.227	-23.227	-23.214	-23.214
Segunda dis. focal(mm.)	31.031	31.031	31.014	31.014
Sistema cristalino:				
Poder refractor (D)	19.11	33.06	20.53	33.00
Primer pto. ppal(mm.)	5.68	5.145	5.85	5.2
Segundo pto. ppal(mm.)	5.808	5.255	5.85	5.2
Distancia focal(mm.)	69.908	40.416	65.065	40.485
Sistema completo del ojo:				
Poder refractor (D)	58.64	70.57	59.74	70.54
Primer pto. ppal(mm.)	1.348	1.772	1.505	1.821
Segundo pto. ppal(mm.)	1.602	2.086	1.631	2.025
Primera dis. focal(mm.)	-15.707	-12.397	-15.235	-12.355
Segunda dis. focal(mm.)	24.387	21.016	23.996	20.963
Posición fóvea	24.0	24.0	24.0	24.0
Posición pto. próximo		-102.3		-100.8

En vista de las dificultades que ello supone, no sólo en la elaboración de las medidas, sino en su utilización, sugirió un ojo *simplificado*, en el que el sistema óptico está libre de aberraciones, el efecto divergente de la cara posterior de la córnea es despreciado y el índice de refracción del cristalino es homogéneo, en vez de variable, como ocurre en la realidad.

Apéndice B

Sistemas de Transputers

Debido al alto número de dimensiones en que se desarrolla la optimización, resulta aconsejable realizar la misma utilizando el cálculo en paralelo. En nuestro caso particular, dado que estábamos desarrollando sobre compatibles IBM AT de altas prestaciones (iapx80386 a 25 MHz) se imponía la consideración de que el hardware de proceso en paralelo fuese compatible con estas máquinas.

El hardware sobre el que se ha implementado este proyecto, en cuanto a su programación en paralelo, ha sido un ordenador Compaq 386/25 con una tarjeta Fast9 de Quintek Ltd. (UK). En el momento de su adquisición (Sept. de 1988), este hardware presentaba la mejor relación precio/MFLOP del mercado.

La tarjeta Fast9 de Quintek constituye [Qui88] un sistema MIMD (Multiple Instruction, Multiple Data) compuesto por:

- 9 *Transputers* INMOS T800-20 de 32 bits que proporcionan cada uno 1.5 MFLOPS (4000 Whetstones/s frente a 1083 de VAX 11/780 [ST87]) de rendimiento teórico sostenido.
- Un megabyte de DRAM local por transputer.
- Una estructura de enlace reconfigurable electrónicamente mediante el chip IMS C004, que permite cambiar la topología de la red que forman los nueve transputers.
- Velocidad en las líneas de enlace configurable a $10 \text{ Mbits} \cdot \text{s}^{-1}$ o $20 \text{ Mbits} \cdot \text{s}^{-1}$
- Conectores de expansión que permiten conectar varias placas entre sí.

El Transputer INMOS T800, es un sistema VLSI [Inm88] que incorpora en un solo chip 4Kbytes de memoria, CPU de 32 bits, FPU operando concurrentemente

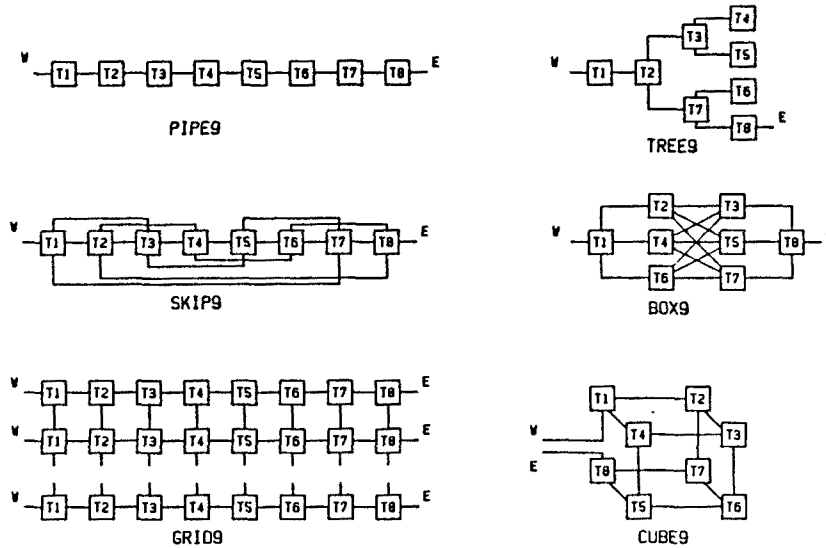


Figura B.1: Algunas topologías posibles con 9 transputers

con la CPU, y 4 enlaces físicos bidireccionales denominados respectivamente Norte, Sur, Este y Oeste. Ello permite crear, con n transputers un gran número de topologías en red. Algunos ejemplos de las que se pueden realizar sobre Fast9 se pueden ver en la Figura B.1.

La programación de los algoritmos de optimización paralelizados se ha realizado en lenguaje "Parallel C" de 3L Languages. Este es un C standard con una librería de funciones que permiten implementar semáforos, "threads", y entrada/salida mediante la comunicación serie a través de los enlaces de los transputers.

La definición topológica de la red se realiza mediante configuración electrónica en cuanto a los enlaces físicos de los transputers. Los enlaces lógicos se realizan mediante un fichero de configuración que no sólo permite su especificación sino también sirve para que el cargador ("loader") reconozca cada programa y lo cargue en su transputer de destino.

Claramente [Mat87] la topología óptima es dependiente de la aplicación.

B.1 Configuración estática y dinámica

Un sistema MIMD se caracteriza porque cada procesador tiene su propia memoria local, no compartida con la de los otros procesadores del sistema (Multiple Data), así como su propio conjunto de instrucciones a ejecutar (Multiple Instruction), posiblemente distinto de los cargados en los demás procesadores.

Para dividir adecuadamente una aplicación en sus partes independientes procesables simultáneamente, existen dos casos principales (ver [MS87]) en cuanto a la localización de las rutinas y su forma de intercomunicación:

- **Configuración estática** : Cada rutina o grupo de rutinas secuenciales independientes ejecutables simultáneamente se ubica en un transputer diferente. Ello implica poder distinguir unos transputers de otros y conocer de antemano el número de éstos en la red. Ello supone un fichero de configuración concreto para una red dada.
- **Configuración dinámica**, también conocida como *Processor Farming*. En este caso se ubica en el transputer raíz un programa *maestro*, que distribuye la información a una serie de programas idénticos —*esclavos* o *workers*—, cada uno de ellos situados en un transputer diferente. En este caso, no es necesario conocer de antemano el número de transputers existentes, ni cargar tareas distintas en distintos procesadores, ya que todos los esclavos son idénticos. El fichero es muy sencillo y sólo varía el cargador a utilizar.

En todos los casos cada programa mantiene su propio conjunto de datos, no compartido con los demás programas situados en otros transputers.

Un punto importante relacionado con los dos casos anteriormente mencionados es la comunicación y el trasvase de datos entre procesos y procesadores. Habitualmente se habilitan unos “threads”, concurrentes [Mat87] con el programa principal, cuya misión consiste en recibir los mensajes destinados al procesador en cuestión (configuración estática) y reenviar por la red los mensajes dirigidos a otros procesadores. Estos *routers* (ver Figura B.2) son dependientes de la topología de la red, aunque normalmente ocupan muy pocas líneas de código.

En el caso de configuración dinámica, los routers son siempre el mismo y su cometido suele ser tan elemental como interceptar el mensaje si el procesador está libre de trabajo, y redirigirlo “red arriba” en caso contrario. Esto plantea notables problemas cuando se traspasan grandes cantidades de información, si el tiempo de cálculo no es mucho mayor que el de transferencia de información.

Maximizar el rendimiento de un sistema de estas características depende de muchos factores (ver [Atk87]) entre los que se cuentan:

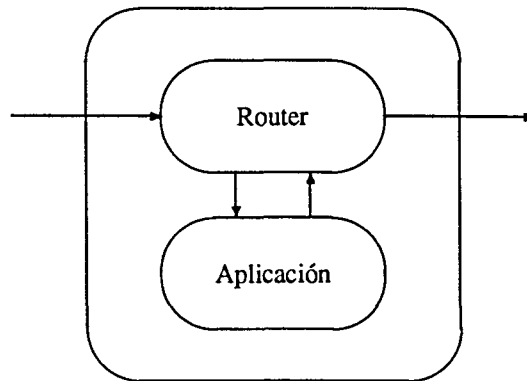


Figura B.2: Comunicación entre *routers* y aplicaciones

- La granularidad de la paralelización.
- La relación entre el volumen de cálculo a realizar por cada procesador y el volumen de información a recibir/enviar por el mismo.
- La posibilidad de incluir la parte más crítica o intensiva del cálculo en los 4Kbytes de RAM de cada procesador.
- Correr los procesos que utilizan los enlaces con alta prioridad, para evitar tiempos muertos esperando la recepción de mensajes.
- Desacoplar la comunicación del cálculo.

B.2 Programación y depuración

A pesar de que *Parallel C* posee un depurador simbólico a nivel de código fuente denominado *Tbug*, la programación de este tipo de máquinas presenta dificultades notables.

Si conceptualmente la realización de un algoritmo en *Parallel C* es relativamente sencilla y de fácil implementación, la depuración del mismo es dificultosa,

tanto más cuanto mayor es el tamaño de la red y cuanto mayor es la complicación de su topología. Programas que funcionan bien con cuatro transputers se cuelgan con 5 o más debido a saturación de las líneas de comunicación, o a la implementación deficiente de un router, por ejemplo.

Como problema añadido, si el error está en las rutinas de comunicación, no podremos emplear estas mismas para su depuración. Por otra parte, la versión 1.0 de *Tbug* simula en un solo transputer el comportamiento de una red y su trasiego de datos. Esto implica que si no hay suficiente memoria en el transputer, determinadas configuraciones sean de difícil depuración. La operatoria del mismo todavía ha de mejorarse, aunque supone un avance a tenerse que enfrentar con el problema sin más herramientas.

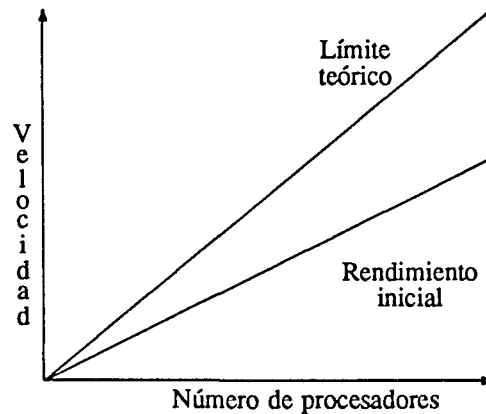


Figura B.3: Velocidad de proceso vs. numero de transputers

Como contrapartida, con una programación cuidadosa, se pueden obtener rendimientos muy altos de algoritmos intrínsecamente paralelizables, con un aumento de prestaciones cercano al lineal (ver Figura B.3) conforme aumenta el número de procesadores [Pac].

Apéndice C

Obtención de Vértices de Control

C.1 Introducción

La formulación mediante B-Splines proporciona un excelente sistema de representación de superficies, salvo por el hecho de que los puntos especificados por el usuario no están sobre la superficie. Incluso en los casos en que no se pretende interpolar una superficie conocida, sino diseñarla, es muy conveniente [BG80] poder encontrar los puntos de control que producen la superficie B-Spline que interpola un conjunto completamente especificado de puntos de la superficie, que llamaremos *de paso*.

En [WAG77] se desarrolla un algoritmo que permite hallar los puntos de control que interpolan un conjunto de puntos de paso dados, para B-Splines uniformes. Estos se caracterizan porque el espacio paramétrico está uniformemente subdividido. En este caso la resolución resulta particularmente eficiente. El artículo de [WAG77] resuelve el problema de hallar los nodos de control para curvas que son secciones de la superficie, e interpola la superficie según el otro parámetro mediante Splines Cardinales, que tienen la propiedad de interpolar los puntos de paso.

[BG80] resuelve también el problema, mediante la solución de un sistema lineal de ecuaciones.

En [BAN83] se describe un algoritmo análogo, en el que se encuentran los vértices de control a partir de los puntos de paso produciendo directamente los puntos de deBoor de una superficie B-Spline, pero partiendo de un supuesto

diferente y eliminando la necesidad de resolver un sistema lineal, usando en su lugar dos matrices previamente calculadas.

C.2 Un algoritmo

Seguiremos aquí la pauta establecida en el artículo de [BAN83], que es el que ha dado lugar a la implementación actual de nuestro sistema de conversión de puntos de paso a vértices de control.

Expresando la superficie en función de las bases de los Splines Cardinales, podemos interpolar toda la superficie con una formulación que contiene a todos los puntos de paso. No obstante perdemos el control local, ya que los Splines Cardinales poseen control global.

La definición de los Splines Cardinales nos permite escribir para n por m puntos

$$\mathbf{x}(u, v) = \sum_{i=1}^n \sum_{j=1}^m \mathbf{P}_{ij} C_i(u) C_j(v) \quad (\text{C.1})$$

en donde $\mathbf{P}_{ij} = (x_{ij}, y_{ij}, z_{ij})$ son los puntos de paso y $C_i(u)$ es el i -ésimo elemento de la base de Splines cardinales sobre el parámetro u , al igual que $C_j(v)$ es el j -ésimo elemento de la base unidimensional sobre el parámetro v . (Información más exhaustiva se encuentra en [dB78]).

Estas bases tienen las siguientes propiedades:

- $C_i(u)$ es una cúbica en todo el intervalo $u_k \leq u < u_{k+1}$ para $1 \leq k < n$
- $C_i \in C^2$ para $u_1 \leq u < u_n$
- $C_i(u_j) = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$
- $C_i(u)$ satisfacen condiciones específicas de contorno para $u = u_1$ y para $u = u_n$. En función de estas condiciones se obtienen distintos tipos de splines.

Si imponemos que $C'''(u_1) = L'''(u_1)$, siendo $L(u)$ el polinomio de Lagrange que interpola los 4 primeros puntos y $C'''(u_n) = L'''(u_n)$ con significado análogo para los 4 últimos obtenemos las condiciones denominadas de extremos curvados (*curved ends*). Imponiendo $C_i''(u_1) = C_i''(u_n) = 0$ obtenemos las condiciones denominadas extremos naturales (*natural ends*).

Utilizando parametrización uniforme ($u_k = k$) y la condición de contorno de extremos curvados, las bases se pueden calcular como [BAN83]:

$$C_i(u) = [t^3, t^2, t, 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_{k,i} \\ \delta_{k+1,i} \\ A_{k,i} \\ A_{k+1,i} \end{bmatrix} \quad (C.2)$$

con $k \leq u \leq k+1$ y $t = u - k$, siendo δ_{ij} la delta de Kronecker y A es la matriz que relaciona el vector de pendientes ($\dot{x}_i, \dot{y}_i, \dot{z}_i$) con el vector de puntos de paso sobre una dirección dada. Para la condición de extremos curvados A vale:

$$A = A_1^{-1} A_2 \quad (C.3)$$

donde

$$A_1 = \begin{bmatrix} 6 & 6 & 0 & 0 & \cdot & \cdot \\ 1 & 4 & 1 & 0 & \cdot & \cdot \\ 0 & 1 & 4 & 1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0 & 1 & 4 & 1 \\ \cdot & \cdot & 0 & 0 & 6 & 6 \end{bmatrix} \quad (C.4)$$

y

$$A_2 = \begin{bmatrix} -13 & 15 & -3 & 1 & \cdot & \cdot \\ -3 & 0 & 3 & 0 & \cdot & \cdot \\ 0 & -3 & 0 & 3 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & 0 & -3 & 0 & 3 \\ \cdot & \cdot & -1 & 3 & -15 & 13 \end{bmatrix} \quad (C.5)$$

Por otra parte, la misma superficie interpolada por los Splines Cardinales se puede construir mediante B-Splines, conociendo los $(n+2) \times (m+2)$ vértices de control apropiados $d_{i,j}$. Podemos escribir:

$$x(u, v) = \sum_{i=0}^{n+2} \sum_{j=0}^{m+2} d_{i,j} N_j^m(v) N_i^n(u) \quad (C.6)$$

Con estos dos tipos de interpolación tenemos la misma superficie expresada en términos de unas bases que proporcionan control global y en las que los puntos de paso intervienen directamente en su formulacion, y, por otra parte, en términos de las bases de B-Splines, con control local y en función de vértices de control. Esto sugiere igualar ambas formulaciones con el propósito de despejar los vértices de control en función de los puntos de paso.

Igualando C.1 a C.6 y teniendo en cuenta la relación entre bases (ver [BAN83]) se puede llegar a la siguiente relación matricial:

$$[D] = [\alpha_n][P][\alpha_m]^T \quad (C.7)$$

donde $[D]$ es la matriz de vértices de control de elementos $d_{i,j}$, $[P]$ es la matriz de puntos de paso de elementos $P_{i,j}$ y $[\alpha_k]$ es una matriz que, para condiciones de extremos curvados se calcula como:

$$[\alpha] = \begin{bmatrix} 1 & \dots & 0 \\ & M_1^{-1}M_2 & \\ 0 & \dots & 1 \end{bmatrix} \quad (C.8)$$

en donde

$$M_1 = \begin{bmatrix} 9 & -9 & 0 & \cdot & \cdot \\ 3 & 7 & 2 & \cdot & \cdot \\ 0 & 1 & 4 & 1 & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & 1 & 4 & 1 & 0 \\ \cdot & \cdot & 2 & 7 & 3 \\ \cdot & \cdot & 0 & -9 & 9 \end{bmatrix} \quad (C.9)$$

$$M_2 = \begin{bmatrix} 5 & -3 & -3 & 1 & \cdot \\ 0 & 12 & 0 & \cdot & \cdot \\ 0 & 0 & 6 & 0 & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & 0 & 6 & 0 & 0 \\ \cdot & \cdot & 0 & 12 & 0 \\ \cdot & 1 & -3 & -3 & 5 \end{bmatrix} \quad (C.10)$$

Este tipo de formulación tiene la ventaja de que se pueden precalcular las matrices α para distintos valores de k y guardarlas en disco. El cálculo de los vértices de control se reduce entonces a leer los valores de éstas y realizar la multiplicación de matrices pertinente, sin necesidad de resolver ningún sistema de ecuaciones.

Bibliografía

- [Atc84] D.A. Atchison. Third-order theory and aspheric spectacle lens design. *Ophthalmic and Physiological Optics*, 4(2):179–186, 1984.
- [Atk87] Phil Atkin. Performance maximisation. Inmos Bristol Technical Note 17, March 1987.
- [Ave07] O. Aves. Improvements in and relating to multifocal lenses and the like, and method of grinding the same. Patente Inglesa N° 15.735, 1907.
- [BAN83] P. Brunet, D. Ayala, and I. Navazo. An interactive algorithm for the generation of b-spline surfaces. *Procedures of the ICS*, 1983.
- [Ben61] A. G. Bennet. Variable and progressive power lenses, partes 1,2,3 y 4. *The Optician*, 160 y 161:Parte 1: 421–427, Parte 2: 533–538, Parte 3: 10–15 Parte 4: 14–22, 1970 (Vol 160) y 1971 (Vol 161).
- [BFK84] W Böhm, G Farin, and J Kahmann. A survey of curve and surface methods in cagd. *Computer Aided Geometric Design*, 0:1 – 60, 1984.
- [BG80] B. A. Barsky and D. P. Greenberg. Determining a set of b-spline control vertices to generate an interpolating surface. *Computer Graphics and Image Processing*, 14:203 – 226, 1980.
- [BS87] E.F. Barkan and D.H. Sklar. Method of making progressive lens surface and resulting article. Patente U.S.A. N° 4,676,610, 1987.
- [BW86] M. Born and E. Wolf. *Principles of Optics*. Pergamon Press, sixth (corrected) edition, 1986.
- [BZ46] C. Berens and J. Zuckerman. *Diagnostic Examination of the Eye*. Lippincott Company, 1946.

- [Car85] F. Carrasco. *Estudio Comparativo de algoritmos de generación de contornos de nivel*. Technical Report DMI02-85, ETSEIB UPC, Barcelona, 1985.
- [Cas85] Justiniano Casas. *Optica*. Editor J. Casas, 6ª edición, 1985.
- [Cre53] B. Cretin-Maitenaz. Systemes optiques a puissance localment variable. Patente francesa N° 1.095.375, 1953.
- [Dav78] J.K. Davis. Aspheric lenses: what's possible and what isn't. *Review of Optometry*, 115:68–74, 1978.
- [Dav81] Lawrence J. Davenport. Ophthalmic progressive power lens and method of making same. Patente U.S.A. N° 4.274.717, 1981.
- [dB78] Carl de Boor. *A Practical Guide to Splines*. Springer Verlag, 1978.
- [Duk61] S.t Duke-Elder. *System of Ophthalmology*. Volume II, Henry Kimpton, London, 1961.
- [Far90] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design*. Academic Press, second edition, 1990.
- [FL81] G Fürter and H. Lahres. Perfeccionamientos introducidos en una lente para gafas con efecto astigmático. Patente Española N° 501.281, 1981.
- [FL86] G Fürter and H. Lahres. Multi-focal spectacle lens with a dioptric power varying progressively between different zones of vision. Patente U.S.A. N° 4.606.622, 1986.
- [Fur85] G. Fürter. Optik konstruktion mit splines. *Optik*, 70(1):19–28, 1985.
- [FvD78] J. Foley and A. van Dam. *Fundamentals of interactive computer graphics*. Addison Wesley, 1978.
- [GOP*87] M. Gauthier, G. Obrecht, C. Pedrono, J-L. Vercher, and L. Stark. *Adaptive Optimization of Eye-Head Coordination with Degraded Peripheral Vision*, pages 201–210. Elsevier Science Publishers, 1987. en EYE MOVEMENTS : From Physiology to Cognition.
- [Gow14] H. O. Gowlland. Multifocal lens. Patente Canadiense N° 159.395, 1914.
- [Har88] E. Hartmann. Augen und kopfbewegungen. *Optometrie*, 2:13–18, 1988.

- [Inm88] Inmos. Ims t800 architecture. Inmos Bristol Technical Note 6, January 1988.
- [Jef57] J.H. Jeffree. Improvements in or relating to lenses. Patente Inglesa. N° 775,007, 1957.
- [KR81] G.A. Kitchen and W.J. Rupp. Lentille ophtalmique à puissance focale variable progressivement. Patente Francesa N° 2.476.333, 1981.
- [Leg45] Yves Legrand. *Optique Physiologique*. Volume 1, Editions de la Revue d'Optique, 1945.
- [Leg46] Yves Legrand. *Optique Physiologique*. Volume 2, Editions de la Revue d'Optique, 1946.
- [Leg61] Yves Legrand. Studies on the human cornea: application to the aphakic eye. Trans. Int. Ophtalmic Congress, 1961.
- [Lip85] Martin Lipschutz. *Geometría Diferencial. Compendios Schaum*, McGraw-Hill, 1985.
- [Mai62] B. Maitenaz. Récentes études relatives à l'utilisation de nouvelles surfaces progressives comme moyen de correction. Conferencia, 1962.
- [Mai66] B. Maitenaz. Four steps that led to varilux. *American Journal of Optometry*, 43:413-450, 1966.
- [Mai74] B. Maitenaz. Ophtalmic lenses with progressively varying focal length. Patente estadounidense N° 3.785.724, 1974.
- [Mat87] Philip Mattos. Program design for concurrent systems. Inmos Bristol Technical Note 5, February 1987.
- [Min63] G. Minkwitz. Über den Flächenastigmatismus bei gewissen symmetrischen Asphären. *Optica Acta*, 10(3):223-227, Julio 1963.
- [Mou88] Bernard Mouchet. *Les Verres Progressifs*. Optic 2000, Paris, 1988.
- [MS87] D. May and R. Shepherd. Communicating process computers. Inmos Bristol Technical Note 22, February 1987.
- [NM65] J.A. Nelder and R. Mead. *Computer Journal*, 7:308, 1965.
- [NR83] W. Newman and Sproull R. *Principles of Interactive Computer Graphics*. McGraw-Hill, 1983.
- [Obs78] Henri Obstfeld. *Optics in vision*. Butterworths, 1978.

- [Pac] J. Packer. Exploiting concurrency; a ray tracing example. Inmos Bristol Technical Note 7.
- [PC20] A. G. Poullain and D. H. J. Cornet. Perfeccionnement dans les verres de lunetterie et moyens de réaliser. Patente Francesa N° 418.583, 1920.
- [PFSV88] W.H. Press, B.P. Flannery, Teukolsky S.A., and W.T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1988.
- [Pie87] L. Piegl. Interactive data interpolation by rational bezier curves. *IEEE CG&A*, April:45 – 58, 1987.
- [Pog87] A. Pogorelov. *Geometry*. Mir Publishers, Moscow, 1987.
- [Pol71] E. Polak. *Computational Methods in Optimization*. Academic Press, NY, 1971.
- [POS87] C. Pedrono, G. Obrecht, and L. Stark. Eye-head coordination with laterally "modulated" gaze field. *American Journal of Optometry and Physiological Optics*, 64(11):853–860, 1987.
- [Pri90] D. J. Pritcher. Transputer applications on supernode. Proceedings of the first conference on Applications of Transputers, 1990.
- [PT87] L. Piegl and W Tiller. Curve and surface constructions using rational b-splines. *Computer Aided Design*, 19(9):485 – 498, 1987.
- [PZ90] W. Purgathofer and M. Zeiller. Configuring transputers for ray-tracing. Proceedings of the first conference on Applications of Transputers, 1990.
- [Qui88] Quintek. Fast9, user manual. 1988.
- [RA76] D. Rogers and J. Adams. *Mathematical elements for computer graphics*. McGraw-Hill, 1976.
- [Rod79] Optische Werke Rodenstock. Verre de lunettes. Patente Francesa N° 2.422.184, 1979.
- [Ros78] Bruno Rossi. *Fundamentos de Optica*. Editorial Reverté, 1978.
- [SA83] G Smith and D.A. Atchison. Effect of conicoid asphericity on the tscherring ellipses of spectacle lenses. *Journal of the Optical Society of America*, 73(4):441–445, Abril 1983.
- [Sab86] M. Sabin. A survey of contouring methods. *Computer Graphics Forum*, 5:325 – 340, 1986.

- [SF88] C. M. Sullivan and C. W. Fowler. Progressive addition and variable focus lenses. *Ophthalmic and Physiological Optics*, 8(3):402–414, Octubre 1988.
- [She68] D. Shepard. A two-dimensional interpolation function for irregularly-spaced data. Proceedings of 23rd National Conf. of ACM, 1968.
- [Slo87] S.W. Sloan. A fast algorithm for constructing delaunay triangulations in the plane. *Adv. Eng. Software*, 9(1):34–55, 1987.
- [SO85] T. Shinohara and S. Okazaki. Progressive multifocal ophtalmic lens. Patente Inglesa N° 2.146.791, 1985.
- [ST87] R. Shepherd and P. Thompson. Lies, damned lies and benchmarks. Inmos Bristol Technical Note 27, July 1987.
- [WAG77] Sheg-Chuan Wu, J. F. Abel, and D. P. Greenberg. An interactive computer graphics approach to surface representation. *Communications of the ACM*, 20(10):703 – 712, October 1977.
- [Wil81] P.R. Wilkinson. Progressive power ophtalmic lenses. Patente Europea. N° 0 027 339, 1981.
- [Win85] J.T. Winthrop. Progressive power ophtalmic lens. Patente U.S.A. N° 4.514.061, 1985.