
PROYECTO INVESTIGADOR

Simulaciones Hidrodinámicas
Alternativas Basadas en Partículas

CONCURSO DE ACCESO A
CUERPOS DOCENTES UNIVERSITARIOS

DEPTO. DE CIENCIAS APLICADAS A LA INGENIERÍA NAVAL
ESCUELA TÉCNICA SUPERIOR DE INGENIEROS NAVALES

UNIVERSIDAD POLITÉCNICA DE MADRID

Documentación presentada por el candidato
Daniel Duque Campayo

3 de noviembre de 2011

DATOS DE LA PLAZA

Fecha de la resolución:	Resolución de 5 de julio de 2011
Publicación en B.O.E.:	19 de julio de 2011, 12467
Numero de la plaza:	04.
Centro:	E.T.S.I. Navales.
Cuerpo:	Profesores Titulares de Universidad.
Departamento:	0806 - Ciencias Aplicadas a la Ingeniería Naval
Área de conocimiento:	385 - Física Aplicada
Perfil docente:	“Física I”, “Física II”, Plan 2009, “Física General II” y “Física General III” Plan 2002
Perfil investigador:	2204 - Física de Fluidos. 2211 - Física del Estado Sólido
Dedicación:	Completa.

AGRADECIMIENTOS

El presente Proyecto recoge los elementos fundamentales de mi línea de investigación actual. Al cambiar de centro de trabajo, de la Universidad Autónoma de Madrid a la Politécnica, decidí también cambiar a una línea de investigación más adecuada al centro: de simulación molecular a simulación hidrodinámica. Por fortuna, los trabajos de Pep Español y sus colaboradores supusieron un puente ideal entre las dos materias. Aproximadamente en ese momento comencé a colaborar con Pep en estos temas. A él quiero agradecer en especial la posibilidad que me ha brindado.

Además, mi avance en la comprensión de la hidrodinámica y de la simulación se ha visto enormemente facilitado gracias a la colaboración con los miembros de investigación del Canal de Ensayos Hidrodinámicos de la ETSI Navales, en particular y sin ningún orden (ya que establecerlo sería difícil), Juan Miguel Sánchez, Leo González, Fabricio Macía, José Luis Cercós y Jesús Gómez Goñi.

Para terminar, en un ámbito personal he de agradecer el enorme apoyo de mi esposa, Ana Granados, en todo lo relativo a mi dedicación docente e investigadora; en particular, en la redacción de los presentes Proyectos.

ÍNDICE GENERAL

Índice general	VII
1. Presentación	1
1.1. La simulación en hidrodinámica	1
1.2. La simulación de partículas	2
1.3. Requisitos de la formulación	2
1.4. Dificultades	3
1.5. Posibles soluciones	3
1.5.1. Trabajo previo	4
1.5.2. Trabajo futuro	4
2. Resumen de hidrodinámica	5
2.1. Puntos de vista de Euler y de Lagrange	5
2.2. Cinemática de una partícula	6
2.3. Conservación de la masa	8
2.3.1. Incompresibilidad	9
2.4. Cambio del momento	10
2.4.1. Fuerzas de superficie	10
2.4.2. Presión	12
2.4.3. Ecuación de Euler	12
2.4.4. Fuerzas viscosas	12
2.4.5. Ecuación de Navier-Stokes	13
2.4.6. Conservación de la energía	14
2.4.7. Ecuaciones constitutivas	16
2.5. Otros aspectos	16
2.5.1. Escalado y similitud	17
2.5.2. Condiciones de contorno	18
3. Geometría computacional	21
3.1. Algunos conceptos	21
3.1.1. Envolverte convexa	21
3.1.2. Triangulaciones	21
3.1.3. Celdas de Voronoi	24
3.1.4. Otros conceptos	25
3.2. Las bibliotecas CGAL	25
3.2.1. Programación orientada a objetos	25
3.2.2. Plantillas (<i>templates</i>)	26

3.2.3.	Contenedores (<i>containers</i>)	28
3.2.4.	Ejemplo: diagrama de Voronoi.	28
4.	Simulación hidrodinámica	31
4.1.	Simulaciones de Euler	31
4.1.1.	Diferencias finitas	32
4.1.2.	Elementos finitos	32
4.1.3.	Volúmenes finitos	32
4.2.	Elementos finitos	32
4.2.1.	Método de Galerkin	33
4.2.2.	Elementos finitos en 1D	35
4.2.3.	Elementos finitos en 2D	37
4.3.	Simulaciones de Lagrange	38
4.3.1.	Formulación genérica	39
4.3.2.	SPH	42
5.	Métodos basados en diagramas de Voronoi	47
5.1.	Operadores diferenciales	47
5.2.	Dificultades	48
5.3.	Convergencia del laplaciano	49
5.3.1.	Una dimensión	50
5.3.2.	Dos dimensiones	53
5.3.3.	Resumen del laplaciano	55
6.	Métodos de máxima entropía	59
6.1.	Introducción	59
6.2.	LME	61
6.2.1.	Varias dimensiones	62
6.2.2.	Propiedades de las funciones base	62
6.3.	SME	64
6.3.1.	Consistencia redefinida	65
6.3.2.	Varias dimensiones	67
6.3.3.	Propiedades de las funciones base	68
7.	Máxima entropía para partículas	69
7.1.	Adaptación a partículas	69
7.2.	Derivadas de las funciones base	70
7.2.1.	Primeras derivadas	70
7.2.2.	Segundas derivadas	71
7.2.3.	Detalles técnicos	72
7.3.	Resultados	74

8. Trabajo futuro	77
8.1. Aplicar lo desarrollado	77
8.2. Base física para los métodos de máxima entropía	78
8.3. Ajuste por gaussianas	79
8.4. Otros métodos de máxima entropía	80
8.5. Una formulación genérica	80
Referencias	81
Índice de figuras	85
Índice de cuadros	87
A. Derivadas de las funciones base	89
A.1. Funciones de forma	89
A.2. Primeras derivadas	90
A.3. Segundas derivadas	93

CAPÍTULO 1:

PRESENTACIÓN

The particle method is not only an approximation of the continuum fluid equations, but also gives the rigorous equations for a particle system which approximates the molecular system underlying, and more fundamental than the continuum equations.

Von Neumann, J. 1944, “Proposal and analysis of a new numerical method for the treatment of hydrodynamical shock problems”. En: A. Taub (ed.): Von Neumann Collected Works. Pergamon, (1963).

El presente Proyecto Investigador tiene por título *Simulaciones Hidrodinámicas Alternativas Basadas en Partículas*. Se presenta aquí una descripción de la estructura general del Proyecto.

1.1. La simulación en hidrodinámica

Las ecuaciones de la hidrodinámica se conocen desde el trabajo durante el siglo XVIII de grandes científicos como Daniel Bernoulli, Euler, d’Alembert, Lagrange, y Laplace. A mediados del siglo XIX se añadieron los términos de viscosidad por parte, entre otros, de Navier, Cauchy, Poisson y Stokes. Quedaban de este modo perfectamente formuladas desde el punto de vista matemático lo que hoy llamamos ecuaciones de Navier-Stokes. Se puede encontrar un resumen de la derivación de algunas leyes en el capítulo 2.

A pesar de ciertos avances a principios del siglo XX gracias al trabajo de Prandtl, Reynolds, Lord Rayleigh, lo cierto es que encontrar soluciones de las ecuaciones, o tan sólo propiedades generales de estas soluciones, demostró ser una tarea muy difícil. La situación sería paralela a la que se daba en física estadística, otro campo perfectamente formulado desde finales del siglo XIX pero muy difícil de abordar desde el punto de vista matemático. Igual que en este caso el desarrollo de la termodinámica aplicada se desgajó del estudio teórico, la ciencia aplicada de la hidráulica se alejaría de la hidrodinámica, más teórica y difícil de aplicar.

En ambos casos, el desarrollo de los ordenadores y la informática a partir de los años 60 del s. XX impulsaron de forma decisiva el progreso teórico y la convergencia con el mundo aplicado. La simulación numérica hacía posible los “experimentos teóricos”, mediante los cuales se podía simular el comportamiento de sistemas que

obedecen con precisión las leyes físicas y los diversos supuestos (por ejemplo, las condiciones de contorno), a diferencia de la situación en un laboratorio. Las condiciones del sistema se pueden conocer con una precisión muy elevada. Por ello, la mecánica de fluidos computacional (*Computational fluid dynamics, CFD*) es en nuestros días una técnica fundamental en la investigación fundamental de los fluidos, y también en sus aplicaciones ingenieriles e industriales. Se discutirá esta técnica en el capítulo 4.

Es necesario precisar que, a pesar de su importancia, la simulación numérica no es la única herramienta que ha contribuido a los recientes avances en el campo. Por ejemplo, los ordenadores también han revolucionado las técnicas experimentales de toma de datos, tratamiento estadístico y visualización. Los propios aparatos de medida han experimentado un fuerte avance, con la invención de velocímetros (*hot-wire, hot-film, Doppler*), sensores de presión y temperatura miniaturizados, trazadoras de diversos tipos (burbujas, humo, colorantes)... Todo ello hace que las experiencias con modelos en túneles de viento y canales hidrodinámicos sean todavía muy relevantes.

1.2. La simulación de partículas

La simulación de la dinámica de los fluidos mediante técnicas numéricas puede dividirse en dos partes. Una de ellas, que denominamos “euleriana” es la más tradicional y asentada; en esta se emplean mallas fijas sobre las cuales se calculan las propiedades del fluido. La otra, que llamaremos aquí “de partículas” o “lagrangiana” mueve las distintas partículas del fluido.

En este Proyecto, examinaremos el concepto de partícula bien pronto, comenzando ya con el resumen teórico del capítulo 2. Las dos formulaciones numéricas se discutirán en el capítulo 4

Ambas formulaciones, si bien equivalentes desde el punto de vista teórico, son bastantes diferentes en su implementación numérica. La formulación lagrangiana, en particular, ofrece en principio una mayor versatilidad, por ejemplo cuando aparecen superficies libres. Sin embargo, su desarrollo ha ido muy por detrás de la formulación euleriana, lo cual se refleja en las numerosas dificultades que todavía se encuentran a un nivel bastante básico.

1.3. Requisitos de la formulación

Se pueden proponer diferentes definiciones de partícula de fluido que pueden ser interesantes desde un punto de vista computacional. La dificultad estriba, en todo caso, en formular de manera adecuada la hidrodinámica para estas partículas. En concreto, la calidad de la formulación de las ecuaciones de la hidrodinámica para partículas (su discretización o “particularización”) depende fundamentalmente de cómo se definan los operadores derivadas espaciales; en particular, el gradiente, la

divergencia y el laplaciano.

Una propiedad fundamental que se puede exigir de estos operadores es la *consistencia*: que proporcionen la respuesta correcta para campos sencillos. En particular:

- Las derivadas de un campo constante deben ser nulas; llamaremos esta propiedad consistencia de orden cero.
- Las primeras derivadas de un campo lineal deben ser constantes, su laplaciano debe ser nulo; llamaremos esta propiedad consistencia de primer orden.
- Las primeras derivadas de un campo cuadrático (parabólico) deben ser lineales, su laplaciano debe ser constante; llamaremos esta propiedad consistencia de segundo orden.

En general, se puede exigir consistencia de orden más alto; sin embargo, la exigencia de coexistencia de segundo orden es un requisito suficiente, ya que las ecuaciones de Navier-Stokes llegan hasta segundo orden de derivadas espaciales (laplaciano). Además, la solución exacta de estas ecuaciones es parabólica en ciertos casos, como el conocido flujo de Poiseuille.

Otro requisito importante, que también se tendrá en cuenta, es el de *convergencia*: en los casos en los que el método no presenta consistencia, se debe comprobar que la solución numérica aproximada converge en cierto modo hacia el valor exacto según se va incrementando la resolución (es decir, el número de partículas). Típicamente, la solución corresponde a una consistencia superior (por ejemplo, se puede explorar la convergencia a una parábola en un método que sólo satisface consistencia de orden cero o de primer orden), o bien a una forma funcional distinta a las anteriores, típicamente una función trigonométrica.

1.4. Dificultades

La principal dificultad de las formulaciones de partículas se resume en el hecho de que la formulación más conocida hoy en día, la hidrodinámica de partículas suavizadas (*Smoothed Particle Hydrodynamics*, SPH) no satisface *ninguna* de las consistencias anteriores. Por supuesto, se han introducido modificaciones de ella para satisfacer al menos la consistencias más bajas, nula y de primer orden. Sin embargo, estas modificaciones introducen nuevos problemas en la formulación; por lo general, se dejan de satisfacer la conservación del momento lineal y angular.

Un problema quizá secundario, pero también importante, es la dificultad práctica de introducir condiciones en la frontera, debido al amplio rango de interacción de las partículas en SPH.

1.5. Posibles soluciones

Una opción es examinar otras alternativas a la definición de partícula del método SPH. Una posibilidad es asociar a cada partícula el volumen de su celda de Voronoi

asociada, como se discute en la sección 5. El concepto de celda de Voronoi se discute en el capítulo 3, así como otros conceptos de geometría computacional. Históricamente, el método fue propuesto por Mar Serrano y Pep Español en 2001 [1], aunque pronto se vio que el método es consistente tan sólo hasta primer orden.

El autor ha estado involucrado en algunos aspectos de este enfoque. Por claridad, separamos aquí las contribuciones pasadas a este campo y las que se proponen como trabajo futuro.

1.5.1. Trabajo previo

Como se verá en el capítulo 5.3, se ha explorado un enfoque tomado del método de elementos finitos, muy bien conocido. Por desgracia, este enfoque también plantea problemas: el laplaciano sigue sin satisfacer convergencia.

Otra posibilidad es el reciente método de máxima entropía. A este método se le dedica el capítulo 7. Los resultados son más prometedores, a pesar de lo cual hay que ser cautos. Primeramente, el método es bastante complejo y alejado de los habituales. En segundo lugar, y por estas mismas razones, es difícil conectarlo con la formulación teórica; de hecho, la implementación que se discute no conserva el momento lineal, a no satisfacer el principio de acción y reacción.

1.5.2. Trabajo futuro

En este estado de cosas, se dedicará el último capítulo, el 8, a discutir posibles vías de avanzar en este campo. Una posibilidad clara es intentar reconducir los métodos de máxima entropía hacia los enfoques más tradicionales. En el camino, es posible que ciertas propiedades, como el principio de acción y reacción, se recuperen.

Por último, se expone brevemente en la sección 8.5 un enfoque más genérico en el que las funciones peso que se utilizan para describir las partículas podrían ser de dos espacios funcionales distintos. Un juego sería para interpolar campos continuos a partir de partículas discretas; otro, para lo contrario: discretizar campos continuos en un conjunto de partículas.

CAPÍTULO 2:

RESUMEN DE HIDRODINÁMICA

Ποταμοῖσι τοῖσιν αὐτοῖσιν ἐμβαίνουσιν, ἕτερα καὶ ἕτερα ὕδατα ἐπιρρεῖ

Aguas siempre nuevas fluyen sobre aquellos que entran en los mismos ríos.

Heráclito [2].

Se presenta aquí una muy breve introducción a las ecuaciones de la hidrodinámica. Tan sólo nos limitamos a introducir los conceptos imprescindibles en el ámbito de este trabajo: la simulación hidrodinámica mediante partículas. Por ello, se enfatiza un punto de vista “lagrangiano”, basado en partículas de fluido, que difiere del que suele ser habitual, el “euleriano”.

2.1. Puntos de vista de Euler y de Lagrange

A la hora de expresar las distintas ecuaciones, se ha venido distinguiendo dos puntos de vista complementarios. El primero, llamado “lagrangiano”¹ es el más similar a un enfoque mecánico habitual: el énfasis se pone en la trayectoria de las *partículas* del fluido.

Estas partículas se entienden hoy como se entendían hace 200 años: desde el punto de vista matemático son puntos, pero se entiende que desde el punto de vista físico son porciones materiales dotadas de magnitudes macroscópicas. Hoy sabemos que la materia es discreta, pero el tamaño atómico y molecular es tan pequeño que esta definición sigue siendo válida en casi todos los casos.

La trayectoria de estas partículas estará descrita por un vector de posición r que cambia con el tiempo. Si la partícula estaba en el punto r_0 en el instante inicial, la colección de todas las trayectorias del fluido con el tiempo es

$$r = r(r_0, t),$$

lo que matemáticamente sería un mapeo del espacio $r_0 \rightarrow r$ con un parámetro t .

Por definición, las partículas siguen una velocidad, que es un campo que satisfará

$$v = \frac{\partial r}{\partial t}. \quad (2.1)$$

¹La nomenclatura es históricamente incorrecta, ya que ambos puntos de vista se encuentran en el trabajo pionero de Euler.

En la Ref. [3], p. xxx incluso se escribe $(\partial r/\partial t)_{r_0}$ para enfatizar que se trata de una derivada sólo temporal.

En el punto de vista “euleriano”, en cambio, se supone un marco fijo de referencia sobre el cual cambian los campos hidrodinámicos. La trayectoria de las partículas pasa a jugar un papel menor y el campo principal es más bien el de velocidades. Esto es también natural para un fluido, ya que las partículas en muchos casos siguen trayectorias que vienen de lejos, y van a sitios también lejanos, mientras que su velocidad en cada punto puede ser más constante (como ilustra la cita de Heráclito al comienzo de este capítulo).

En este sentido, la variación temporal de cualquier magnitud física referente al fluido, A , no vendrá dada simplemente por su derivada con respecto a su dependencia explícita del tiempo. Como también la posición de la partícula puede variar con el tiempo, la variación vendrá dada por la regla de la cadena:

$$\frac{dA}{dt} = \frac{\partial A}{\partial t} + \frac{\partial A}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial A}{\partial y} \frac{\partial y}{\partial t} + \frac{\partial A}{\partial z} \frac{\partial z}{\partial t}.$$

Como se ha discutido, la derivada de la posición de la partícula con respecto al tiempo es la velocidad, así que podemos escribir

$$\frac{dA}{dt} = \frac{\partial A}{\partial t} + (v \cdot \nabla)A,$$

donde hay que entender el operador $v \cdot \nabla$ como

$$v \cdot \nabla = \frac{\partial x}{\partial t} \frac{\partial}{\partial x} + \frac{\partial y}{\partial t} \frac{\partial}{\partial y} + \frac{\partial z}{\partial t} \frac{\partial}{\partial z}.$$

Esta derivada se denomina justamente derivada particular (“de las partículas”), o material².

2.2. Cinemática de una partícula

No nos referimos aquí a la cinemática de una partícula puntual, tan conocida. Más bien, a la acción de un campo de velocidades sobre una de nuestras partículas fluidas. Nos limitamos aquí a proporcionar una lista de expresiones relevantes, refiriendo al lector a la excelente descripción de la Ref. [4], p. 19.

Una partícula puede distorsionarse de cuatro maneras distintas bajo un campo de velocidades. Cada una de ellas tiene una tasa, o ritmo, de deformación (*strain rate*) asociada.

Traslación

²También hay que recordar los numerosos otros nombres que se usan para esta expresión: derivada convectiva, advectiva, sustantiva, sustancial, lagrangiana, hidrodinámica, de Stokes, siguiendo el movimiento. . . También distintos símbolos, como “ D/Dt ”.

Es fácil ver que la tasa de traslación es, casi por definición, la propia velocidad v .

Rotación

La tasa de rotación viene dada por la velocidad angular $d\Omega/dt$. Debido a un molesto factor $1/2$, se suele definir un vector que es el doble de la velocidad angular

$$\omega = 2 \frac{d\Omega}{dt}.$$

Esta magnitud, de gran importancia, se denomina *vorticidad*. Su expresión es particularmente sencilla:

$$\omega = \nabla \times v;$$

es decir, es el rotacional del campo de velocidades.

Por ejemplo,

$$\omega_z = \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y}$$

Deformación de cizalladura

La deformación de cizalladura, o de corte (*shear strain*), tiene tasas asociadas

$$\epsilon_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right),$$

donde $i \neq j$. Vemos que estas magnitudes son simétricas, $\epsilon_{ij} = \epsilon_{ji}$.

Dilatación

La tasa asociada a la dilatación, definida como el cambio relativo de la longitud de la partícula en la dirección i resulta ser

$$\epsilon_{ii} = \frac{\partial v_i}{\partial x_i},$$

es decir, la componente i de la divergencia del campo de velocidades

$$\epsilon_{ii} = (\nabla \cdot v)_i.$$

Tensor de deformación

Por tanto, se puede definir un tensor conjunto para la tasa de cizalladura-dilatación. Este tensor cumple las propiedades que pueden esperarse de un tensor de segundo orden simétrico. En particular, existen tres invariantes: la traza, el determinante, y una cantidad adicional. En términos de los autovalores del vector (llamados “tasas de deformación principales”) las expresiones para estas cantidades son muy sencillas:

- $\epsilon_1 + \epsilon_2 + \epsilon_3 = \nabla \cdot v$ (traza)
- $\epsilon_1 \epsilon_2 \epsilon_3$ (determinante)
- $\epsilon_1 \epsilon_2 + \epsilon_2 \epsilon_3 + \epsilon_3 \epsilon_1$.

También puede definirse un tensor ∇v que sea el gradiente de la velocidad (también llamada la jacobiana³) con componentes:

$$\nabla v_{ij} = v_{i,j} = \frac{\partial v_i}{\partial x_j};$$

en notación tensorial sería un producto directo de dos vectores:

$$\nabla v = \nabla \otimes v.$$

Es interesante que siempre podemos dividir el tensor en una parte simétrica y otra antisimétrica

$$v_{i,j} = \frac{1}{2}(v_{i,j} + v_{j,i}) + \frac{1}{2}(v_{i,j} - v_{j,i}).$$

Es fácil comprobar que esto equivale a

$$\frac{\partial v_i}{\partial x_j} = \epsilon_{ij} + \frac{1}{2}\omega_k.$$

Es decir, las derivadas espaciales de la velocidad se deben a deformaciones de cizalladura y dilatación, más un término de rotación, que rota la partícula pero no la distorsiona. (Este término sólo está presente si $i \neq j$, en cuyo caso ω_k corresponde a la otra componente cartesiana, con el signo adecuado a la permutación (i, j, k)).

Más adelante utilizaremos la parte simetrizada del tensor gradiente de velocidades, $\overline{\nabla v}$:

$$\overline{\nabla v}_{ij} = \frac{1}{2}(v_{i,j} + v_{j,i}) \quad \overline{\nabla v} = \frac{1}{2}(\nabla v + (\nabla v)^t).$$

2.3. Conservación de la masa

La manera más elegante de obtener las ecuaciones de movimiento es a partir de ecuaciones de conservación. Para empezar, supongamos que una partícula tiene masa m . Desde el punto de vista lagrangiano, la conservación de la masa es simplemente

$$m = m_0, \tag{2.2}$$

donde m_0 es la masa inicial. Si el volumen de la partícula es \mathcal{V} y su densidad, por tanto $\rho = m/\mathcal{V}$, la expresión anterior significa

$$\rho\mathcal{V} = \rho_0\mathcal{V}_0,$$

donde ρ_0 es la distribución inicial de densidad. (ver [3], p. y [4] p. 61). Matemáticamente, esta expresión puede entenderse como un cambio de variables que sólo sería válido dentro de una integración; el jacobiano del cambio de \mathcal{V}_0 a \mathcal{V} sería ρ/ρ_0 .

³En castellano podemos distinguir con el género la jacobiana, la matriz con las derivadas, y el jacobiano, su determinante.

Desde el punto de vista euleriano, en cambio, la conservación se escribiría

$$\frac{dm}{dt} = 0.$$

Poniendo $m = \rho\mathcal{V}$,

$$\mathcal{V}\frac{d\rho}{dt} + \rho\frac{d\mathcal{V}}{dt} = 0.$$

Como vimos en la sección anterior, el cambio relativo de \mathcal{V} es la parte de dilatación (la diagonal) del tensor de deformaciones, que a su vez es la divergencia de la velocidad:

$$\frac{1}{\mathcal{V}}\frac{d\mathcal{V}}{dt} = \nabla \cdot v. \quad (2.3)$$

Es decir,

$$\frac{d\rho}{dt} = -\rho\nabla v, \quad (2.4)$$

una expresión que Monaghan denomina, jocosamente, “de la convergencia”, ya que $-\nabla v$ sería “la convergencia” de v [5].

Esta expresión, recordando que la derivada temporal es la derivada material, puede escribirse también

$$\frac{\partial\rho}{\partial t} + \nabla\rho v = 0, \quad (2.5)$$

ecuación que se conoce como “de continuidad”.

Una manera más habitual de llegar a esta ley es la siguiente: si la masa en una región del espacio es $\int_V \rho$, su cambio vendrá dado por el flujo de materia a través de la superficie que rodea esta masa, $-\int_S(\rho v) \cdot n$ (el signo menos se introduce porque el vector normal a la superficie, n , apunta por convención hacia afuera, así que un integrando positivo representa una pérdida). Introduciendo la derivada temporal en la integral y aplicando el teorema de la divergencia de Gauss se llega a la misma conclusión.

2.3.1. Incompresibilidad

Si un fluido es incompresible, su masa no puede cambiar (ni temporal ni espacialmente). Esto implica que el campo de velocidades es solenoidal:

$$\nabla \cdot v = 0,$$

como se concluye de las diversas expresiones de la conservación de la masa que acabamos de describir. Por ejemplo, desde el punto de vista lagrangiano que ρ sea constante significa que el volumen de una partícula, \mathcal{V} , también lo será (dado que la masa m siempre se conserva). Según los argumentos cinemáticos de la sección anterior, la dilatación de una partícula es proporcional a la divergencia de la velocidad, que debe ser nula por tanto.

Desde el punto de vista euleriano, la conclusión es la misma, a partir de la ecuación de la convergencia (2.4) o de la de continuidad (2.5).

2.4. Cambio del momento

La segunda ley de Newton expresa que el momento lineal de una partícula cambia cuando existen fuerzas netas sobre ésta. El momento de una partícula es $p = mv$; desde el punto de vista lagrangiano,

$$\frac{\partial mv}{\partial t} = F.$$

Como hemos visto que la masa se conserva, podemos dividir por el volumen de la partícula \mathcal{V} y escribir

$$\rho \frac{\partial v}{\partial t} = f,$$

donde $f = F/\mathcal{V}$ es la fuerza por unidad de volumen.

Esta expresión es, desde el punto de vista euleriano,

$$\rho \frac{dv}{dt} = \rho \left(\frac{\partial v}{\partial t} + v \cdot \nabla v \right) = f.$$

Es en esta ecuación donde queda patente la principal dificultad de las ecuaciones de la hidrodinámica: resultan ser no lineales. Por ello su desarrollo ha sido tan lento, a pesar de haber sido propuestas históricamente a la vez que las de la electricidad y magnetismo. Estas últimas, a pesar de su carácter más abstracto (en términos de campos que no son directamente visibles), sí son lineales, y verifican una serie de propiedades que fallan para la hidrodinámica, tales como: existencia y unicidad de soluciones, movimiento oscilatorio con espectro continuo de frecuencias (una característica principal del flujo turbulento).

Podría objetarse que la expresión equivalente lagrangiana sí parece lineal. De hecho, la no linealidad está “escondida bajo la alfombra”, ya que las posiciones de las partículas deben actualizarse según $dr/dt = v$. Las dos propuestas son, de hecho, idénticas, al menos a este nivel, exacto.

La fuerza f se puede descomponer, por lo general, en dos tipos:

- Fuerzas de volumen. La más común es la gravedad, que tiene la conocida forma $F = mg$ (al menos, cerca de la superficie terrestre). Así pues, $f = \rho g$. En astrofísica y cosmología estas fuerzas son, por supuesto, más complicadas; en magnetohidrodinámica aparecen también campos eléctricos y magnéticos.
- Fuerzas de superficie: de presión y viscosas.

Pasamos a discutir las últimas en más detalle.

2.4.1. Fuerzas de superficie

Estas fuerzas operan sobre una partícula de fluido por contacto; salvo en la frontera del fluido, este contacto es con partículas circundantes. Si tomamos una

partícula cúbica de volumen $\mathcal{V} = dx dy dz$, la fuerza neta en la dirección x será en general debida a esfuerzos, o tensiones (*stresses*) sobre cada una de las seis caras del cubo. Estos esfuerzos se consideran positivos si se ejercen hacia fuera de la partícula (en el sentido del vector normal a la superficie, según el convenio habitual).

Si tomamos tres caras del cubo perpendiculares y con áreas $dydz$, $dx dy$ y $dx dz$ la fuerza de contacto sobre la partícula a través de estas caras será:

$$dF_{x,\text{tres caras}} = \tau_{xx} dydz + \tau_{xy} dx dz + \tau_{xz} dx dy, \quad (2.6)$$

donde τ_{ij} son los esfuerzos correspondientes.

Como existen otras tres caras paralelas, la fuerza *neta* sobre el cubo en la dirección x resulta estar relacionada con el cambio de los esfuerzos en sus direcciones correspondientes. Por ejemplo

$$d\tau_{xx} \approx \frac{\partial \tau_{xx}}{\partial x} dx.$$

Así pues,

$$dF_{x,\text{neta}} = \left(\frac{\partial \tau_{xx}}{\partial x} dx \right) dydz + \left(\frac{\partial \tau_{xy}}{\partial y} dy \right) dx dz + \left(\frac{\partial \tau_{xz}}{\partial z} dz \right) dx dy$$

Así pues, la fuerza por unidad de volumen será

$$f_x = \frac{dF_{x,\text{neta}}}{\mathcal{V}} = \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z}.$$

Para tener en cuenta las otras componentes de f , se puede definir un tensor de esfuerzos τ que, como el tensor de deformaciones ϵ es también de orden dos y simétrico. La conexión con las fuerzas de superficie es, en general

$$f = \nabla \cdot \tau;$$

en coordenadas cartesianas,

$$f_i = \frac{\partial \tau_{ij}}{\partial x_j}.$$

El caso anterior corresponde a $i = 1$.

La ecuación del momento es, por tanto:

$$\rho \frac{dv}{dt} = \rho g + \nabla \cdot \tau. \quad (2.7)$$

Para cerrar esta ecuación hace falta encontrar una expresión de los esfuerzos en función de la velocidad, lo cual por lo general supone conectar de algún modo el tensor de esfuerzos, τ , con el de deformaciones, ϵ .

2.4.2. Presión

Por la definición de presión habitual, como una fuerza por unidad de superficie que siempre es normal a ésta, está claro que las componentes diagonales de τ son la presión misma; salvo un cambio de signo, ya que una tensión positiva se ejerce hacia dentro de la partícula.

Por ejemplo, la fuerza neta en la componente x debida a la presión será:

$$dF_{x,\text{neto}} = - \left(\frac{\partial p}{\partial x} dx \right) dydz + 0dx dz + 0dx dy,$$

Es fácil ver que si p crece con x la fuerza neta en esta dirección será negativa, como debe ser.

En este caso,

$$f = -\nabla p,$$

simplemente.

En general, el tensor de esfuerzos contendrá un término $-p$ en cada uno de las componentes diagonales, pero las otras fuerzas de superficie (que, como queda claro, son de cizalladura) contribuirán tanto a la diagonal como a los elementos fuera de ella. Estas otras fuerzas son debidas a la viscosidad en un fluido.

2.4.3. Ecuación de Euler

Si la viscosidad se supone despreciable, nuestra ecuación del momento queda

$$\rho \frac{dv}{dt} = \rho g - \nabla p,$$

la llamada *ecuación de Euler*. Esta es la ecuación que describe los fluidos inviscibles, también llamados ideales.

Recordamos una vez más que la aceleración no es lineal:

$$\frac{dv}{dt} = \frac{\partial v}{\partial t} + v \cdot \nabla v$$

2.4.4. Fuerzas viscosas

Si, por el contrario, existen fuerzas viscosas, estas contribuirán al tensor de esfuerzos. El caso más sencillo es el de los fluidos *newtonianos*, para los cuales el tensor de esfuerzos es proporcional al de deformaciones.

Expresando esta linealidad, y teniendo en cuenta la isotropía de los fluidos, se puede escribir en la base que diagonaliza los dos tensores (la diagonalización es simultánea al ser ambos proporcionales):

$$\tau_1 = -p + K\epsilon_1 + \lambda(\epsilon_1 + \epsilon_2 + \epsilon_3),$$

siendo ésta la expresión más genérica permitida. Al rotar los ejes para pasar a otro sistema de referencia, la conclusión es:

$$\tau_{ii} = -p + K\epsilon_{ii} + \lambda\nabla \cdot v,$$

y para los elementos de fuera de la diagonal, con $j \neq i$:

$$\tau_{ik} = K\epsilon_{ij}.$$

Una aplicación de estas ecuaciones a un fluido newtoniano entre dos placas planas paralelas revela que el esfuerzo en el fluido es

$$\tau_{xy} = \frac{KV}{2h},$$

donde V es la velocidad relativa de las placas y h su separación. Históricamente, esta relación se utiliza para definir (y medir) el coeficiente de viscosidad (dinámica) μ :

$$\tau_{xy} = \mu \frac{V}{h},$$

así pues $K = 2\mu$.

En cuanto a λ , esta cantidad sería un segundo coeficiente de viscosidad, denominado coeficiente de Lamé en elasticidad. Fijémonos que la traza del tensor de esfuerzos satisfará

$$-\frac{\text{Tr}}{3} = p - (\lambda + \frac{2}{3}\mu)\nabla \cdot v.$$

El papel de la “presión extra” $-(\lambda + 2\mu/3)\nabla \cdot v$ suele por lo general ignorarse, alegando una de estas dos condiciones:

- La combinación $\lambda + 2\mu/3$ es nula. Esto define un fluido “stokesiano”.
- El flujo es prácticamente incompresible, así que $\nabla \cdot v$ es despreciable.

La primera condición, totalmente *ad hoc* parece no sostenerse en la práctica; de hecho, λ parece ser positivo en las (escasas) medidas que se han realizado, así que a duras penas puede satisfacerse. La segunda, en cambio, sí puede sostenerse en muchas ocasiones. Al final, sólo hay algunos problemas de interés en los cuales el término no puede (o debe) ignorarse, como ondas de choque o atenuación de ondas de sonido.

2.4.5. Ecuación de Navier-Stokes

Recogiendo todas las contribuciones al tensor de esfuerzos resulta la expresión vectorial

$$\rho \frac{dv}{dt} = \rho g - \nabla p + \nabla \cdot (\mu \overline{\nabla v} + \lambda(\nabla \cdot v)\bar{1}). \quad (2.8)$$

En ella, $\bar{1}$ es el tensor identidad de orden dos, y $\overline{\nabla v}$ es la simetrización del tensor gradiente de velocidades introducida anteriormente.

Alternativamente, con subíndices:

$$\rho \frac{dv_i}{dt} = \rho g_i - \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\mu \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right) + \delta_{ij} \lambda \nabla \cdot v, \right]$$

donde se aplica el criterio de Einstein (suma implícita sobre subíndices repetidos).

Para un fluido incompresible el término $\nabla \cdot v$ desaparece. Si además la viscosidad puede considerarse constante (en el caso de los líquidos, esto implica generalmente que la temperatura es constante),

$$\rho \frac{dv}{dt} = \rho g - \nabla p + \mu \nabla^2 v. \quad (2.9)$$

Esta ecuación suele ser el punto de partida de muchas teorías o simulaciones de fluidos viscosos, a pesar de las numerosas aproximaciones que se presuponen. De nuevo, repetimos que la derivada temporal es la sustancial.

2.4.6. Conservación de la energía

La energía de una partícula es la suma de energía interna, energía cinética y energía potencial:

$$E = U + \frac{1}{2}mv^2 - mg \cdot r, \quad (2.10)$$

donde U es la energía interna, y los otros dos términos son los habituales para la energía cinética y la potencial gravitatoria (la única que consideramos en este trabajo). Recordando que m es constante, la variación temporal de la energía será

$$\frac{dE}{dt} = \frac{dU}{dt} + mv \cdot \frac{dv}{dt} - mg \cdot v. \quad (2.11)$$

Por otro lado, la conservación de la energía viene dada por la primera ley de la termodinámica, que reza:

$$dE = dQ + dW,$$

donde el cambio de energía de una partícula dE , puede deberse al calor aportado a la misma dQ , o al trabajo realizado sobre la misma.

En cuanto al calor, si se define un *flujo de calor* por unidad de área q , de nuevo podemos plantear un problema similar al que relacionaba fuerza y esfuerzos. Por ejemplo, en la cara x izquierda tendremos un aporte de calor igual a:

$$\frac{dQ_{x,\text{izda}}}{dt} = q_x dydz.$$

En la dirección x el incremento neto de calor es:

$$\frac{dQ_{x,\text{neto}}}{dt} = - \left(\frac{\partial q_x}{\partial x} dx \right) dydz.$$

El resultado acaba siendo

$$\frac{dQ}{dt} = -\nu \nabla \cdot q.$$

El flujo de calor suele considerarse bien aproximado por la ley de Fourier, que establece que estos flujos son proporcionales al gradiente de temperaturas (y de signo contrario, obviamente):

$$q = -\kappa \nabla T,$$

donde κ es la conductividad térmica. En este caso,

$$\frac{dQ}{dt} = \nu \nabla \kappa \nabla T.$$

Para el trabajo, el argumento es similar: hay que evaluar la tasa de trabajo en cada una de las caras. En la cara x izquierda esta tasa viene dada por la fórmula de la potencia mecánica:

$$\frac{dW_{x,izda}}{dt} = -(v_x \tau_{xx} + v_y \tau_{xy} + v_z \tau_{xz}) dy dz;$$

el signo menos se debe a que consideramos el trabajo *sobre* la partícula de las fuerzas de contacto que aparecían en la ec. (2.6) y siguientes; recordemos que los esfuerzos son positivos en la dirección hacia fuera de la partícula. Así pues, en la dirección x el incremento neto de trabajo es:

$$\frac{dW_{x,neto}}{dt} = \left(\frac{\partial \sum_j v_x \tau_{xj}}{\partial x} dx \right) dy dz.$$

Teniendo en cuenta todas las caras,

$$\frac{dW}{dt} = \nu \nabla (v \cdot \tau).$$

Aplicemos la regla de la cadena:

$$\nabla (v \cdot \tau) = v \cdot (\nabla \cdot \tau) + \tau : \nabla v,$$

donde “:” significa reducción completa de dos tensores: $a : b = \sum_{i,j} a_{ij} b_{ij}$. Del primer término, sabemos de la ecuación del momento, (2.7):

$$\nabla \cdot \tau = \rho \left(\frac{dv}{dt} - g \right).$$

Así pues,

$$\nu v \cdot (\nabla \cdot \tau) = m v \cdot \frac{dv}{dt} - m g \cdot v.$$

Llega ahora el momento de igualar las dos expresiones para el cambio de energía, y vemos que parte de la ecuación del trabajo cancela la parte de la expresión (2.11) correspondiente al cambio en energía cinética y gravitatoria. El resultado final es:

$$\frac{1}{\nu} \frac{dU}{dt} = \nabla \kappa \nabla T + \tau : \nabla v.$$

Definiendo la energía interna por unidad de masa $u = U/m$,

$$\rho \frac{du}{dt} = \nabla \kappa \nabla T + \tau : \nabla v.$$

Otra expresión alternativa se puede obtener dividiendo el tensor de esfuerzos en una parte viscosa y otra de presión, diagonal

$$\tau = \tau' - p\bar{1}.$$

Entonces,

$$\tau : \nabla v = \tau' : \nabla v - p \nabla \cdot v.$$

Pero de la ecuación de continuidad, podemos escribir:

$$-p \nabla \cdot v = \frac{p d\rho}{\rho dt}.$$

Desarrollando la expresión,

$$\frac{p d\rho}{\rho dt} = -\rho \left(-\frac{p d\rho}{\rho^2 dt} + \frac{1 dp}{\rho dt} - \frac{1 dp}{\rho dt} + \right) = -\rho \left(-\frac{d(p/\rho)}{dt} - \frac{1 dp}{\rho dt} \right).$$

La ecuación de la energía puede reescribirse de este modo:

$$\rho \frac{dh}{dt} = \frac{dp}{dt} + \nabla \kappa \nabla T + \tau' : \nabla v,$$

donde la entalpía por unidad de masa es $h = u + p/\rho$.

El término $\tau' : \nabla v$ se denomina “función de disipación”, y puede comprobarse que es siempre positivo. Esta es una expresión de la segunda ley: la viscosidad representa siempre una pérdida de energía.

2.4.7. Ecuaciones constitutivas

En las tres ecuaciones principales de la hidrodinámica se ve que existen siete campos. Se puede suponer que tres de ellos son los primarios y los otros cuatro función de ellos. Una elección común es escoger p , v y T y expresar los otros cuatro en función de ellos. Los campos son ρ , μ , h y κ . La relaciones $\rho = \rho(p, T)$ y $h = h(p, T)$ vienen dada por la ecuación de estado termodinámica; es habitual suponer μ y κ constantes.

2.5. Otros aspectos

Por no extendernos demasiado, evitaremos totalmente algunos aspectos de gran interés pero que quedan fuera de este proyecto. Sin embargo, hay otros que merecen ser discutidos, aunque sea brevemente.

2.5.1. Escalado y similitud

Como las ecuaciones de la hidrodinámica son bastante complicadas, es conveniente, como en otros muchos ámbitos, definir variables sin dimensiones físicas. Esto conlleva una serie de ventajas:

- Simplicidad en las ecuaciones, al tener que acarrear menos parámetros.
- Generalidad en los resultados, que se aplican ahora a multitud de situaciones físicas.
- Identificación de los parámetros relevantes.

El método más habitual es identificar en un problema estos parámetros relevantes:

Longitud L La longitud característica del problema. Por ejemplo: el radio de un cilindro inmerso en un fluido, el radio de una tubería, la eslora de un barco.

Propiedades del flujo libre Estas son las magnitudes físicas que corresponderían a un problema simplificado. En particular, nos interesan la densidad del flujo libre, ρ_0 , su presión p_0 y ...

Velocidad v_0 Una velocidad de referencia, por lo general correspondiente al flujo libre, puntos lejanos o fronteras móviles: velocidad de un fluido lejos de un obstáculo, velocidad de crucero de un barco. . . En un problema de convección, puede tomarse la velocidad típica de convección, $v_0 = \mu_0/(\rho_0 L)$.

En términos de estas variables, se pueden definir magnitudes sin dimensiones:

$$\begin{aligned}
 x^*_i &= \frac{x_i}{L} & \nabla^* &= L \nabla \\
 t^* &= \frac{v_0}{L} t & \frac{\partial}{\partial t^*} &= \frac{L}{v_0} \frac{\partial}{\partial t} \\
 v^* &= \frac{v}{v_0} & \rho^* &= \frac{\rho}{\rho_0} \\
 p^* &= \frac{p - p_0}{\rho_0 v_0^2}
 \end{aligned}
 \tag{2.12}$$

Es interesante ver qué sucede con las ecuaciones de la hidrodinámica al usar estas magnitudes sin dimensiones. La ecuación de continuidad, (2.5), no cambia en absoluto al pasar a estas magnitudes. Es decir, la continuidad no se ve afectada por los parámetros del fluido.

La ecuación de Navier-Stokes en su forma más sencilla, (2.9), se transforma en otra casi idéntica en variables sin dimensiones, pero esta vez el término viscoso, en vez de un prefactor μ pasa a tener otro.

$$\mu \rightarrow \frac{\mu}{\rho_0 v_0 L}.$$

Esta expresión no tiene dimensiones; es tradicional definir más bien su inversa

$$\text{Re} = \frac{\rho_0 v_0 L}{\mu},$$

el famoso número de Reynolds⁴. Por tanto, hemos identificado la combinación de parámetros que determina cómo de viscoso es un flujo. De hecho también se puede escribir

$$\text{Re} = \frac{\rho_0 v_0^2 L^2}{\mu v_0 L},$$

lo cual puede verse como un cociente entre las fuerzas inerciales (de arrastre) y las fuerzas viscosas.

Se pueden considerar estos límites:

Re **pequeña** En este caso, μ es grande comparada con $\rho_0 v_0 L$, y el flujo es viscoso. En este caso, dominan los efectos viscosos sobre los inerciales.

Re **grande** En este caso, μ es pequeña comparada con $\rho_0 v_0 L$, y el flujo es poco viscoso. En este caso, dominan los efectos inerciales. Sin embargo, se suele argumentar que los efectos viscosos son siempre importantes, aunque estén confinados a una capa límite. Es en este límite cuando puede producirse el fenómeno de la turbulencia.

Otro parámetro que se desprende de la ecuación sin dimensiones es el siguiente:

$$\text{Fr} = \frac{v_0^2}{gL},$$

el número de Froude. Este número es importante siempre que la gravedad lo es. En particular, para fenómenos de superficie libre tales como olas y todo tipo de problemas navales, al menos para longitudes típicas superiores al centímetro. A escalas inferiores, la tensión superficial empieza a ser importante (y, finalmente, dominante); el número relevante es entonces el de Weber, We .

Una descripción exhaustiva de todas las variables hidrodinámicas (algunas de ellas asociadas a condiciones de contorno) y los diversos números adimensionales que aparecen queda fuera de este trabajo, referimos de nuevo a textos canónicos como la Ref. [4].

2.5.2. Condiciones de contorno

Como es bien sabido, la solución de una ecuación diferencial depende de las condiciones iniciales y de las condiciones de contorno. Las primeras especifican por lo general la forma de sus campos en el instante inicial $t = 0$. Las segundas se refieren a los valores que toman los campos en las fronteras de los dominios. Es

⁴En rigor, sería μ_0 , pero hemos supuesto en esta ecuación simplificada que μ es constante.

decir, en mecánica de fluidos, en las paredes de los recipientes o “lejos” del sistema en cuestión.

Este tema un enorme interés y sutileza; ver por ejemplo la Ref. [4] para una discusión de las condiciones en la frontera. Sin embargo, los método de partículas en los que nos centraremos sustituyen estas sutilezas por otras, algunas de las cuales son problemas abiertos hoy día. Así que expondremos simplemente los dos tipos de condiciones de contorno más habituales:

de Dirichlet Se especifica en valor de la función en todos los puntos del contorno. Si este valor es nulo, la condición es *de Dirichlet homogénea*. Por ejemplo, el valor de cada una de las componentes de la velocidad es nulo en las condiciones *sin deslizamiento (no slip)*.

de Neumann Se especifica la derivada de la función en todos los puntos del contorno. De nuevo, si esta es nula la condición es homogénea. Por ejemplo, en la superficie libre de un líquido la componente normal de la velocidad tiene una derivada tangencial que se suele considerar despreciable.

Las condiciones pueden ser, por supuesto, una mezcla de estas dos. Además, el asunto de dónde deben definirse estas condiciones, y dónde su valor es un resultado, es asimismo delicado.

CAPÍTULO 3:

GEOMETRÍA COMPUTACIONAL

En esta sección proporcionamos una introducción al área de geometría computacional. No pretendemos avanzar mucho en esta área, tan sólo introducir los conceptos básicos que serán necesarios más adelante. Referimos al lector que quiera ampliar sus conocimientos al enciclopédico trabajo de A. Okabe *et al*, Ref. [6]. La idea que subyace a este campo de la matemática aplicada es el desarrollo de algoritmos que resuelvan problemas geométricos.

3.1. Algunos conceptos de geometría computacional

En muchos casos, en los que consideramos en particular, la geometría es simplemente la euclídea, incluso plana. Sin embargo, hay un gran número de problemas cuya resolución numérica eficaz está lejos de ser trivial. Mencionamos a continuación algunos de especial interés en nuestro estudio.

3.1.1. Envolverte convexa

Dada una serie de puntos en el espacio con coordenadas x_a (que llamaremos nodos), existe un polígono (en tres dimensiones, polihedro) que contiene a todos los puntos en su interior y cuya área es mínima. Éste se denomina envolvente, o casco, convexo (*convex hull*). En el caso de la simulación, la importancia es evidente, dado que es vital especificar adecuadamente las condiciones en la frontera.

Existen algoritmos específicos para hallar la envolvente convexa; nosotros hemos utilizado los que se basan en triangulaciones, como se explica en la siguiente sección. Esta elección se debe a que estos métodos son tan eficaces como otros, y a que en muchas ocasiones es deseable construir una triangulación en todo caso.

3.1.2. Triangulaciones

Sin entrar en muchos detalles matemáticos, la idea intuitiva de una triangulación es que indica la figura 3.1: una colección de triángulos que no se intersecan, cuyos

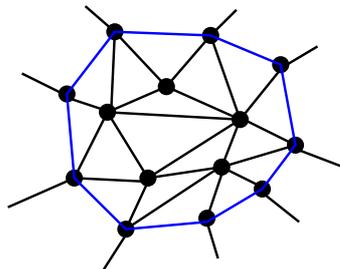


Figura 3.1: Ejemplo de una triangulación. La envoltura convexa es el polígono azul. Los nodos pertenecientes a ella se consideran veces del vértice infinito en ciertas implementaciones.

vértices son los nodos¹. La triangulación llena el espacio dentro de la envoltura conexa del conjunto de nodos. Todos los nodos están rodeados por triángulos, salvo los que quedan más “expuestos”, que forman parte justamente de la envoltura conexa. Para tratar estos nodos igual que los otros se puede introducir un vértice infinito, que está conectado a éstos y situado en el infinito (una idea similar a la de Riemann, que asigna el polo norte de una esfera al punto del infinito del plano proyectivo). Este es el método implementado en CGAL. La triangulación define además una relación de vecindad entre nodos que estén conectados por un lado de un triángulo.

De todas las posibles triangulaciones (cuyo número puede ser enorme, ya que crece como 2^N), existe una única (salvo casos degenerados, como se discutirá) llamada de Delaunay, introducida por Delone en 1924². Una de sus posibles definiciones es la siguiente propiedad:

Propiedad. Círculos vacíos: La triangulación de Delaunay verifica que las circunferencias circunscritas de uno de los triángulos no contienen ningún nodo en su interior.

La figura 3.2 muestra dos posibilidades de triangulación de cuatro nodos. De las dos posibles, la dibujada en azul es la de Delaunay, al cumplir la condición.

Esta condición indica claramente qué casos pueden ser problemáticos: en configuraciones llamadas “degeneradas” puede que haya dos posibilidades de triangulación; en este caso, habrá un nodo sobre circunferencia circunscrita. Este caso se muestra en la figura 3.3.

Otra propiedad muy importante de estas triangulaciones queda patente en la figura 3.2:

¹Una triangulación es un ejemplo de una teselación: un recubrimiento del espacio mediante teselas “iguales”. En este caso, son iguales al ser todos triángulos (tetrahedros en tres dimensiones, símplexes en general), aunque por supuesto su forma varía.

²La historia de estos conceptos es bastante embrollada. Boris N. Delone, escribía su nombre a la manera francesa, Delaunay, en sus primeros artículos (aunque en realidad su nombre familiar era francés, “de Launay”). El concepto se remonta en realidad a Voronoï en 1908, quien lo introduzco como dual de su diagrama, aunque fue Delone quien expuso la condición de círculos vacíos.

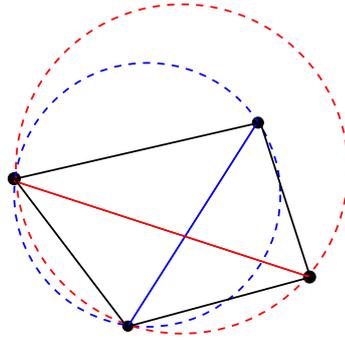


Figura 3.2: Dos posibles triangulaciones de cuatro nodos. La azul es la de Delaunay (el círculo circunscrito no engloba al cuarto nodo), la roja no lo es.

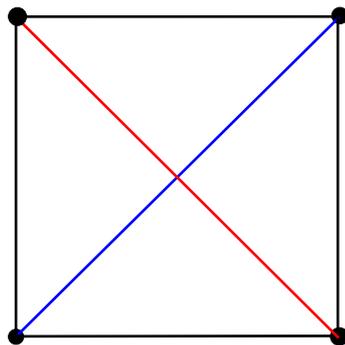


Figura 3.3: Caso degenerado: tanto la conexión roja como la azul corresponden a triangulaciones de Delaunay.

Propiedad. *Triángulos más equiláteros:* la triangulación de Delaunay es la que produce los ángulos internos más abiertos.

Es decir, los triángulos resultantes son siempre “más equiláteros” que los de otras triangulaciones. Esto puede demostrarse rigurosamente y tiene importantes consecuencias en el método de elementos finitos, como se discutirá más adelante, en el cual los triángulos “poco equiláteros” no son deseables.

Existen otras propiedades importantes, por ejemplo:

Propiedad. *Distancia mínima:* los vértices más próximos entre sí están necesariamente unidos por la triangulación de Delaunay.

3.1.3. Celdas de Voronoi

Para una triangulación dada, se puede hallar el diagrama dual trazando las mediatrices de cada uno de los lados de los triángulos. En el caso de la triangulación de Delaunay, la malla dual resulta ser el diagrama de Voronoi, de enorme importancia³. Por la propia definición, los nodos de esta malla son los circuncentros de los triángulos de Delaunay. La propiedad más famosa de esta teselación es la siguiente:

La celda de Voronoi correspondiente a un nodo es lugar geométrico de puntos que están más cerca del nodo que de otros nodos.

La simplicidad de este concepto explica lo antiguo que en realidad. Aunque se adjudica su definición precisa a G. Voronoi⁴, quien introdujo el concepto formalmente (aunque en una red periódica) en 1907, en realidad existía una definición precisa en un trabajo de Dirichlet de 1850. La idea aparece, más informal, ya en la teoría de los vórtices de Descartes (1644).

Su generalidad confiere a este diagrama su importancia en ámbitos muy diversos:

- Asignación de recursos por zonas geográficas (centrales de bomberos, delegaciones de correos, hospitales . . .)
- Demarcaciones de animales territoriales.
- En física del estado cristalino son de gran importancia, aunque en el espacio recíproco (de Fourier); las celdas se denominan de Wigner-Seitz.

Es interesante que en muchos de estos ámbitos el concepto haya sido “redescubierto” de cero.

A pesar de la mayor simplicidad de la definición del diagrama de Voronoi, en muchos casos es más sencillo implementar una triangulación de Delaunay y calcular las propiedades del diagrama de Voronoi a partir de ella.

³La expresión correcta es diagrama y no teselación, ya que el recubrimiento del espacio es mediante objetos de distinta naturaleza.

⁴De nuevo, el apellido se escribe “Voronoi”, “Voronoï”, o “Voronoy”.

3.1.4. Otros conceptos

Además de los casos discutidos, existen otros muchos de gran interés.

Distancias mínimas: Destaca el problema de encontrar el nodo más cercano a otro dado en un conjunto; o, más generalmente, los k nodos más cercanos a él. Este problema es ubicuo en simulación, y suele tratarse mediante “linked lists” o listas de vecinos de Verlet. Es interesante tener en cuenta que existen algoritmos muy eficaces para resolver este problema, basados en triangulaciones de Delaunay. De hecho, se puede demostrar que el nodo más cercano a otro debe ser vecino de Delaunay (o Voronoi).

Formas α : a la hora de buscar una definición de “frontera” de un dominio está claro que la envolvente convexa es, en general, una opción muy pobre. Por su convexidad, cualquier convavidad, con todos sus detalles, quedará borrada por una línea recta que conecte sus extremos. Para retener estos detalles se han propuesto alternativas. Una de ellas son las formas α : dado un radio r , relacionado por α ($r = \alpha^2$ en CGAL, pero la relación depende del autor), ...

3.2. Las bibliotecas Cgal

La implementación de rutinas de geometría computacional es, por lo general, bastante difícil. Por lo general, es muy sencillo implementar un método que funcione pero tenga un mal escalado con el número de puntos. Por ejemplo, es trivial encontrar el nodo más cercano a uno dado probando todas las $N - 1$ posibilidades; es mucho más difícil mejorar esto y conseguirlo en tiempo que escale como, por ejemplo, $\log N$.

Igual que se hace en otras áreas que han sido muy estudiadas, como por ejemplo el álgebra lineal computacional, lo más indicado es utilizar bibliotecas desarrolladas por expertos. En nuestro caso, hemos venido empleando las las bibliotecas CGAL, desarrolladas por un consorcio de universidades y centros de investigación en su mayor parte europeos [7]. Se trata de una iniciativa de código abierto de gran complejidad (más de xxx líneas de código a día de hoy) y, lo que es muy importante en la práctica, dotada de una documentación exhaustiva y muy útil.

En este campo, la gran complejidad de las estructuras hace que muchos lenguajes de bajo nivel, como `fortran` o `c` no sean realmente adecuados. Por ello, las bibliotecas CGAL se han decantado desde su fundación por una implementación en `c++`. Más aún, han utilizado algunos de los conceptos más avanzados del language. Destacamos brevemente algunos.

3.2.1. Programación orientada a objetos

En este paradigma de programación se definen entidades llamadas objetos que engloban datos (como las “estructuras” o “tipos” de lenguajes más sencillos) pero también código ejecutable. Este código accede principalmente a “sus” datos. De

hecho, la idea es “proteger” los datos de un objeto, de tal modo que todas las consultas y modificaciones de datos se hagan a través de código.

Por ejemplo, en programación tradicional se podría definir un vector bidimensional v con dos componentes cartesianas, $v.x$ y $v.y$. Para calcular su longitud tendríamos una rutina separada que se invocaría así: `long(v)`.

En un enfoque orientado a objetos tendríamos, en cambio, un objeto v , y la longitud sería una función propia (“miembro” del objeto), `v.long()`. Esta función tiene acceso a los datos internos del objeto. Remitimos al libro clásico del inventor de C++, B. Stroustrup, Ref. [8], y a la excelente obra de Yang con énfasis en aplicaciones científicas, Ref. [9].

3.2.2. Plantillas (*templates*)

Las librerías consisten básicamente en código fuente con plantillas que se “instancian” en tiempo de compilación según las necesidades del programador. En resumen, las plantillas suponen un esquema de programación muy flexible, de tal modo que se pueden escribir rutinas enteras que valen para cualquier tipo de variable.

<pre> int idiv(int i, int j) { return i/j; } double ddiv(int i, double j) { return i/j; } // ... int a=idiv(2,3); // a=0 double b=ddiv(2,3); // b=0.66... </pre>	<pre> int div(int i, int j) { return i/j; } double div(int i, double j) { return i/j; } // ... int a=div(2,3); // a=0 double b=div(2,3.0); // b=0.66... </pre>
<pre> template<typename tt> tt div(int i, tt j) { return i/j; } // ... int a=div(2,3); // a=0 double b=div(2,3.0); // b=0.66... double c= div<double>(2,3); // c=0.66... </pre>	

Cuadro 3.1: Ejemplos de estilos de programación: “tradicional”, con sobrecarga y con plantillas.

Imaginemos que queremos programar una rutina que divida dos números, distinguiendo entre división entera y entre coma flotante (un asunto, por lo demás, que suele ser problemático en aplicaciones reales); se puede plantear este problema de tres maneras, como se muestra en la tabla 3.1. El enfoque tradicional consistiría en definir dos funciones casi idénticas, pero con nombres distintos. En `c++` se permite “sobrecargar” las funciones, de modo que dos funciones distintas pueden tener el mismo nombre y el compilador selecciona una u otra dependiendo de (generalmente) los argumentos de la función. Si embargo, se ve que en este caso los cuerpos de las dos funciones son idénticos. Por medio de las plantillas, se puede escribir un código genérico que el compilador adapta (“instancia”) según haga falta. En este ejemplo, el operador de división “/” está definido para enteros y números de coma flotante, pero el código funcionaría para cualquier objeto que tuviera “/” definido (vectores en el espacio, puntos, ...).

3.2.3. Contenedores (*containers*)

Como los de la biblioteca estándar de plantillas (STL). Se trata de una generalización de los vectores (*arrays*) utilizando toda la maquinaria de la programación orientada a objetos. Se pueden definir, por ejemplo, objetos de tipo `vector` que almacenen cualquier clase de datos. Estos vectores pueden crecer o menguar a voluntad, y todas las tareas de mantenimiento están automatizadas.

Una particularidad es que los índices no están particularmente recomendados. En lugar de ellos, pueden utilizarse iteradores, que comienzan en cierto elemento y van recorriendo el contenedor.

3.2.4. Ejemplo: diagrama de Voronoi.

Como ejemplo, el código del cuadro 3.2, tomado de la documentación de CGAL, recorre los bordes⁵ de una triangulación de Delaunay y examina si corresponden a segmentos finitos del diagrama dual de Voronoi o a segmentos infinitos, devolviendo el número total de los dos tipos.

Comienza con una cabecera que incluye el código necesario mediante comandos `#include` y sigue con una serie de `typedef` que son básicamente “alias” y se introducen por aligerar el código (por ejemplo, en la primera línea de este tipo “`K`” sustituye a la expresión tan larga que tiene a su izquierda).

En la función principal, `main()`, se crea una triangulación de Delaunay. La manera en que se hace puede resultar algo oscura, pero consiste tan sólo en “cebar” el archivo con las coordenadas cartesianas, de nombre `data/voronoi.cin`, a la función `insert`, que es miembro de los objetos de tipo `Triangulation`.

Una vez la triangulación está creada (no tenemos por qué saber los detalles de esta construcción), podemos acceder a sus propiedades. En este ejemplo, se recorren

⁵Los bordes (*edges*) de una triangulación son simplemente los segmentos que conectan los vértices.

todos los bordes de la triangulación mediante el iterador `eit`. Fijémonos que, en el bucle `for` se crea este iterador, haciéndolo inicialmente igual que el “primer” borde (no sabemos cuál es, realmente) y se hace que vaya recorriendo todos los bordes (mediante el operador `++` hasta que se pasa del “último”).

Lo contenido dentro del bucle es quizá más difícil de entender, ya que la función `dual` devuelve un objeto genérico, debido justamente a que el segmento dual de un borde puede ser o bien un segmento finito o una semirecta (o “rayo”). La función `object_cast` es la que discrimina cuál caso es el apropiado; dependiendo del resultado se incrementa el contador `ns` o `nr`.

Convengamos que, aparte de algunas dificultades técnicas como las descritas en el párrafo anterior, la utilización de este tipo de códigos es mucho más sencilla que iniciar un desarrollo de cero de estas rutinas geométricas. De hecho, los cálculos de las expresiones que aparecen en el capítulo 5 no son, conceptualmente, mucho más complicados que el ejemplo que se ha descrito.

```

#include <CGAL/Exact_predicates_inexact_constructions_kernel↵
    .h>
#include <CGAL/Delaunay_triangulation_2.h>

#include <fstream>

typedef CGAL::Exact_predicates_inexact_constructions_kernel ↵
    K;

typedef CGAL::Delaunay_triangulation_2<K>   Triangulation;
typedef Triangulation::Edge_iterator        Edge_iterator;
typedef Triangulation::Point                Point;

int main( )
{
    std::ifstream in("data/voronoi.cin");
    std::istream_iterator<Point> begin(in);
    std::istream_iterator<Point> end;
    Triangulation T;
    T.insert(begin, end);

    int ns = 0;
    int nr = 0;
    for (
        Edge_iterator eit =T.edges_begin();
        eit !=T.edges_end(); ++eit) {
        CGAL::Object o = T.dual(eit);
        if (CGAL::object_cast<K::Segment_2>(&o)) {++ns;}
        else if (CGAL::object_cast<K::Ray_2>(&o)) {++nr;}
    }
    std::cout << "The Voronoi diagram has " << ns << " finite ↵
        edges "
                << " and " << nr << " rays" << std::endl;
    return 0;
}

```

Cuadro 3.2: Cálculo de propiedades de un diagrama de Voronoi mediante las rutinas CGAL.

CAPÍTULO 4:

SIMULACIÓN HIDRODINÁMICA

The numerical calculations were so lengthy, that we came close to the ultimate limit of what can be done in this way. I myself had no part in this. I did try once or twice to set up and work out such a calculation, but then it would turn out that I had made a mistake, so that it had to be done all over again by others.

Hendrik Lorentz (1926), comentando las dificultades de integrar las ecuaciones de la hidrodinámica en una dimensión para los cálculos de la Afsluitdijk, el Gran Dique de Cierre en Holanda. De Carlo Beenakker, Instituto Lorentz de Física Teórica.

Entramos ahora en el dominio de la simulación numérica. La mecánica de fluidos computacional (*Computational fluid dynamics, CFD*) es un campo que ha venido creciendo exponencialmente desde sus comienzos en los años 60. Los avances en el campo siguen, por supuesto, los de los recursos informáticos. Sin embargo, también los algoritmos y las ideas subyacentes han evolucionado notablemente durante estos años. Comentamos brevemente los métodos más tradicionales de resolución de las ecuaciones para centrarnos en los basados en partículas.

4.1. Simulaciones de Euler

La simulación numérica implica por lo general la definición de un dominio finito. Por esto, y por el uso tradicional del punto de vista euleriano, se suele definir una malla fija en el espacio sobre la cual se discretizan las ecuaciones de la hidrodinámica. Estas ecuaciones se integran temporalmente, calculando la evolución temporal en intervalos temporales también discretizados (otra dimensión adicional de la malla, en el fondo).

Dependiendo del tipo de malla, de la discretización y de la integración temporal se han venido proponiendo una serie de métodos más o menos sofisticados y con más o menos méritos. No es nuestro propósito proporcionar una cronología detallada de estos desarrollos, sino tan sólo resaltar los aspectos de algunos de ellos que nos serán de utilidad en el futuro.

4.1.1. Diferencias finitas

Estos esquemas parten de una discretización del espacio para formar una malla. Los distintos operadores se identifican con diferencias entre los valores de una malla.

Un ejemplo muy conocido sería la expresión para aproximar la primera derivada de una función en una dimensión mediante diferencias centradas:

$$\left. \frac{df}{dx} \right|_a \doteq \frac{f_{a+1} - f_{a-1}}{x_{a+1} - x_{a-1}}.$$

4.1.2. Elementos finitos

Estos métodos aparecieron como una extensión de las diferencias finitas a mallas más genéricas. Son para nosotros muy importantes, así que presentaremos en la sección (4.2) una exposición más detallada de ellos.

4.1.3. Volúmenes finitos

Un punto de alternativo de de las diferencias finitas consiste en visualizar la malla dual y considerar el transporte a través de ésta. Las distintas ecuaciones de conservación establecen los flujos adecuados, obteniéndose así un método computacional alternativo.

Nos interesan sobre todo las aplicaciones de este método a mallados basados en el diagrama de Voronoi. En la sección 5.3 discutiremos algunos aspectos de estas aplicaciones.

4.2. Elementos finitos

El método de los elementos finitos es extremadamente conocido en matemática aplicada. La definición precisa depende del autor. Para nosotros, un elemento finito es una función base. Además, nos restringiremos al caso en que los elementos finitos son funciones lineales a trozos (en varias dimensiones, afines a trozos). Estos elementos suelen ser utilizados dentro de una formulación de Galerkin, como se discutirá. En este trabajo “método de elementos finitos” significará por tanto “elementos finitos afines aplicados en un método de Galerkin”.

En una dimensión, el elemento finito sobre el nodo a es un triángulo de altura 1 sobre el nodo a , que va a 0 linealmente en los nodos vecinos $a - 1$ y $a + 1$. En los dos nodos de los extremos la definición es la misma, salvo que no existe nodo izquierdo o derecho, según qué lado. Es fácil ver que estos elementos constituyen una base para cualquier curva poligónica; es decir, una curva lineal a trozos que cambia de pendiente en los nodos. Otra característica importante de estos elementos es que son “casi” ortogonales: la integral sobre la recta real de dos elementos a y b es nula, salvo que b sea vecino de a (o el propio a).

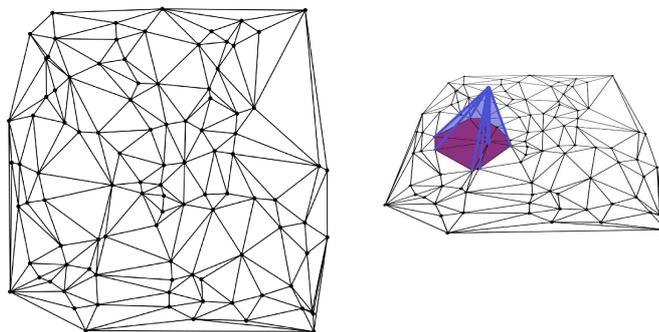


Figura 4.1: Triangulación de Delaunay de un conjunto aleatorio de nodos en un cuadrado. Elemento finito definido sobre uno de los nodos.

En dimensiones superiores, la idea es la misma: definir una base para las funciones afines a trozos. Si las superficies son trianguladas (una opción, por cierto, ubicada en muchos campos), los elementos deben definirse sobre triangulaciones. De todas las posibles, se suele argumentar que la de Delaunay es la más adecuada, al proporcionar los triángulos más equiláteros (ver sección 3.1.2). El procedimiento es, por tanto:

- Para un nodo dado, hallar sus triángulos de Delaunay incidentes
- Con los triángulos, definir un polígono (que no tiene por qué ser convexo).
- Levantar en el nodo un “mástil” de altura 1
- Unir la parte superior del mástil con los vértices vecinos del nodo (que comparten triángulos con él).

4.2.1. Método de Galerkin

Los elementos finitos, definidos arriba, son funciones afines. Sus derivadas son, por tanto, algo patológicas: las primeras serán constantes a trozos (es decir, discontinuas) y las segundas similares a deltas de Dirac (nulas o infinitas, siendo poco rigurosos).

Una manera de regularizar estas derivadas es mediante un método de Galerkin; si bien este método es completamente general y puede emplearse con funciones de cualquier tipo. Nos centramos en las segundas derivadas, sobre todo el laplaciano por ser éste el término que suele ser más delicado en las ecuaciones.

Por ejemplo, tomemos la ecuación de Poisson:

$$\nabla^2 f = g.$$

En el ámbito de las ecuaciones en derivadas parciales, se suele conocer el término de fuente g y el problema es determinar f . En nuestro caso nos preocupamos más

bien por una reconstrucción: conocemos f y la tarea es calcular un laplaciano g que cumpla una serie de requisitos. En todo caso, la idea de Galerkin es la misma: introduciendo funciones base $\{s_a\}$, transferimos el problema de Poisson a lo que denomina una “forma débil”

$$\int s_a \nabla^2 f = \int g s_a.$$

Lo interesante de la idea de Galerkin de reformular el problema en forma débil queda claro si se escribe:

$$\int s_a (\nabla^2 f - g) = 0.$$

El integrando es el error: $e = \nabla^2 f - g$, que sería nulo para la solución exacta del problema. Lo que se pretende es, si el error no puede hacerse cero en todo punto, al menos puede “echarse” fuera del espacio funcional que se considera, haciendo que sea ortogonal a todas las funciones base. Se encuentran ideas similares en otras áreas del cálculo numérico, por ejemplo en la aproximación mediante polinomios de Chebychev [10].

Si ahora las funciones anteriores se aproximan por

$$f(x) \doteq \sum_a s_a(x) f_a \quad g(x) \doteq \sum_a s_a(x) g_a$$

el problema acaba reduciéndose a

$$\sum_b f_b \int s_a \nabla^2 s_b = g_b \int s_a s_b.$$

Si ahora aplicamos el teorema de Gauss, despreciando el término de borde (lo cual debe ser debidamente justificado y quizá modificado cerca de las fronteras del dominio):

$$\sum_b \Lambda_{ab} f_b = \sum_b m_{ab} g_b,$$

donde la matrix de rigidez es

$$\Lambda_{ab} = - \int (\nabla s_a) \cdot (\nabla s_b)$$

y la matrix de masas es

$$m_{ab} = \int s_a s_b.$$

Esta es la formulación habitual del método de Galerkin donde, como se ha dicho, se debe resolver el problema de álgebra lineal

$$\Lambda \vec{f} = m \vec{g},$$

para obtener f_a a partir de g_a .

En nuestra implementación (sección 5.3) el problema es el inverso, más sencillo. Además, planteamos el problema de una manera algo distinta: si definimos el valor de cualquier función en un nodo como

$$g_a = \frac{1}{\int s_a} \int g s_a$$

(la división es necesaria para normalizar las funciones peso, si es que no lo están, de tal modo que se cumpla consistencia de orden cero). En este caso, la forma débil pasa a ser

$$\int s_a \nabla^2 f = \int g s_a = g_a \int s_a.$$

El argumento sigue el mismo cauce, pero la ecuación final es

$$\sum_b \Lambda_{ab} f_b = m'_a g_a,$$

donde la “matriz” de masas sería simplemente diagonal, con elementos $m'_a = \int s_a$. Como casi siempre se satisface consistencia de orden cero, $\sum_b s_b = 1$ así que esta reducción de una matriz m_{ab} a otra diagonal $m'_{ab} = m'_a \delta_{ab}$ se llama en este campo “aproximación de masa acumulada” (*lumped mass approximation*). Correspondería a sumar todos los elementos de cada línea de la matriz m y colocar la suma en el elemento diagonal de una matriz diagonal m' . En muchos casos esta aproximación es plausible; en nuestro caso particular es no es tan siquiera una aproximación, sino una consecuencia del planteamiento del problema (ver en todo caso la sección de trabajo futuro 8, donde tendremos algo que decir respecto a esta aproximación.)

4.2.2. Elementos finitos en 1D

En una dimensión es muy sencillo evaluar las integrales que aparecen en la matriz de rigidez y de masas. El resultado final para el laplaciano (en la aproximación de masa acumulada) es:

$$(\nabla^2 f)_a \doteq \frac{2}{h_{a-} + h_{a+}} \left[\frac{f_{a+1}}{h_{a+}} + \frac{f_{a-1}}{h_{a-}} - \left(\frac{1}{h_{a-}} + \frac{1}{h_{a+}} \right) f_a \right]$$

donde h_{a-} es el intervalo entre la posición del nodo a y el de su izquierda, $h_{a-} = x_a - x_{a-1}$ y $h_{a+} = x_{a+1} - x_a$ el de la derecha.

Esta expresión es, de hecho, la habitual en diferencias finitas (ver sección 4.1.1). Si los intervalos son todos iguales a h , la expresión es muy conocida:

$$(\nabla^2 f)_a \doteq \frac{1}{h^2} [f_{a+1} + f_{a-1} - 2f_a.]$$

Las propiedades de esta aproximación son excelentes: verifica consistencia de orden cero, $\nabla^2 c \doteq 0$, consistencia de primer orden, $\nabla^2 x \doteq 0$ y consistencia de segundo

orden, $\nabla^2 x^2 \doteq 2$, para cualquier conjunto de nodos. Lo último es sorprendente: las funciones base tienen segundas derivadas patológicas; sin embargo, las primeras derivadas, dentro de un método de Galerkin, proporcionan la segunda derivada exacta.

Para completar la discusión, los elementos de la matriz de masas son

$$m_{a,a} = \frac{1}{3}(h_{a-} + h_{a+})$$

$$m_{a,a-1} = \frac{1}{6}h_{a-}$$

$$m_{a,a+1} = \frac{1}{6}h_{a+}$$

Puede comprobarse que se recupera la expresión para la masa acumulada, $m_a = m_{a,a} + m_{a,a-1} + m_{a,a+1}$.

A la hora de implementar estas expresiones (que son lo bastante sencillas para requerir un pequeño programa en un programa de cálculo estilo matlab, u octave) es interesante que la matriz de rigidez es, en un enfoque ingenuo, singular. Por ejemplo, para 4 nodos equiespaciados (ponemos $h = 1$ por claridad):

$$\Lambda = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix};$$

una matriz claramente singular (basta sumar filas sucesivamente). El álgebra lineal nos está indicando, a su manera, que el problema está mal condicionado porque falta una parte importante: las condiciones en la frontera. Por ejemplo, si estas son de Dirichlet homogéneas en los dos extremos, estas condiciones se reflejan en la matriz de rigidez, que pasa a ser:

$$\Lambda = \begin{pmatrix} -2 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -2 \end{pmatrix}, \quad (4.1)$$

la cual ya no es singular. En general, la idea es introducir “a mano” las condiciones en la frontera en las matrices de rigidez y de masas. Para esto último puede ser útil introducir nodos de frontera que luego no aparecen en las matrices. En la matriz anterior, habría entonces un “nodo 0”, a la izquierda del primero, y otro “ $N + 1$ ” a la derecha del último.

Otra opción es añadir explícitamente estos dos nodos de frontera a la matriz, escribiendo dos filas adicionales bastante triviales:

$$\Lambda = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

La matriz de masas deberá tener dos filas análogas; por último, el vector g debe contener dos elementos adicionales, correspondientes al valor de la función f en los dos extremos del dominio (para el caso homogéneo sería ambos cero, pero pueden usarse otros valores). De este modo forzamos a que la función f satisfaga los valores deseados en los extremos, mientras la conexión con los nodos internos está garantizada por la parte original de las matrices. Un inconveniente es que las matrices ya no son simétricas.

La última opción es quizá más elegante: consiste en alterar la parte derecha de la ecuación de Poisson para incluir las condiciones de contorno adecuadas, partiendo de las de Dirichlet homogéneas (4.1), ver [11], p. 377.

4.2.3. Elementos finitos en 2D

Las integrales son más complicadas en dos dimensiones, pero son bastante asumibles, dado que los elementos son lineales a trozos. Es evidente que un nodo sólo tiene solapamiento con sus vecinos de Voronoi (o Delaunay). Así pues, las integrales son sobre triángulos. El resultado final es:

$$m_a(\nabla^2 f)_a \doteq \frac{1}{2} \sum_b f_b (\cot \theta_{ab} + \cot \theta'_{ab}), \quad (4.2)$$

donde $m_a = A/3$, un tercio del área de la base del elemento (la suma de las áreas de los triángulos incidentes) y los dos ángulos son los dos ángulos externos compartidos por el nodo a y su vecino b (ver figura 4.2). Esta es la conocida “fórmula de la cotangente”. Como esta función trigonométrica tiende a 0 cuando los ángulos tienen al recto, esto significa que el papel de un nodo b es menor cuando sus triángulos son abiertos en la dirección que une a y b . En cambio, cuando son cerrados o abiertos su papel es muy grande, al tender la cotangente a infinito cuando su argumento tiende a cero o a dos ángulos rectos. Vemos que esta expresión del laplaciano incorpora por tanto una noción de la distorsión de una triangulación.

Cuando $a = b$, la fórmula es la que satisface consistencia de orden cero:

$$\sum_b (\cot \theta_{ab} + \cot \theta'_{ab}), \quad \rightarrow \quad (\nabla^2 c)_a \doteq 0.$$

En la práctica, es más útil utilizar este hecho que evaluar la expresión completa.

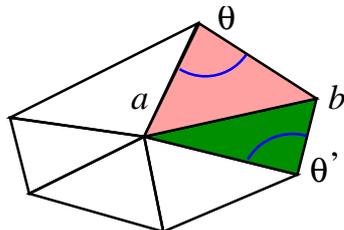


Figura 4.2: Los dos ángulos de la fórmula de la cotangente.

Como este método es ampliamente utilizado, han aparecido numerosos trabajos sobre las propiedades de esta aproximación [12, 13].

En cuanto a la matriz de masas, los elementos son

$$m_{a,a} = \frac{A}{6},$$

con la misma área A que aparecía en m_a , y

$$m_{a,b} = \frac{A_{ab}}{12},$$

donde A_{ab} es el área de los dos triángulos compartidos por a y b . Sumando para cada triángulo, dado que existen dos nodos distintos de a , se tendrá

$$\sum_b m_{a,b} = \frac{A_{ab}}{6} + 2 \frac{A_{ab}}{12} = \frac{A_{ab}}{3}.$$

Sumando a todos los triángulos incidentes a a se recupera la expresión para la masa acumulada, $m_a = A/3$.

4.3. Simulaciones de Lagrange

Las simulaciones de partículas, o lagrangianas, siguen el otro punto de vista histórico, el lagrangiano. Se definen por tanto partículas de fluido que deben seguir el campo de velocidades. Esto conlleva la dificultad de que no puede definirse ya una malla fija para el sistema. De hecho, el propio concepto de malla debe ser redefinido: o bien se abandona totalmente o bien se debe remallar a cada paso temporal (o, al menos, cada cierto tiempo).

Estas simulaciones pueden ser superiores a las eulerianas, más tradicionales, por distintos motivos [5]:

- El tratamiento de la advección es exacto (dentro de los límites de la integración temporal). En cambio, los métodos eulerianos presentan generalmente una dependencia del mallado, que debe escogerse con cuidado, y falta de invarianza galileana que se pone de manifiesto si se sobreimpone un campo constante de velocidades.

- Los sistemas con más de una componente son fáciles de simular: una partícula es de un tipo o de otro. En particular, el tratamiento de la superficie libre es muy sencillo: es la superficie que separa dos componentes. Se pueden obtener resultados razonables sin imponer condiciones en esta superficie.
- El cálculo numérico se realiza en las zonas donde hay partículas. Esto es fundamental a la hora de simular sistemas donde la materia está muy concentrada en ciertas regiones, y hay grandes vacíos entre ellas. Por ello uno de los orígenes del método SPH, y una de sus principales aplicaciones, es en cosmología y astrofísica, donde la materia acaba en muchas ocasiones concentrada en regiones muy reducidas. También es importante en problemas con fragmentación y gotas.
- Un aspecto relacionado con el anterior: no es necesario intuir dónde va a ser necesaria más precisión. En una simulación euleriana es por lo general necesario refinar la malla en las zonas donde se prevé más dificultad; en una lagrangiana, este refinamiento puede irse modificando en el tiempo y el espacio.
- Para problemas con fragmentación, el enfoque euleriano es mucho más sencillo, ya que no hay que preocuparse demasiado por las condiciones de contorno. Por ello, el método SPH es casi estándar en problemas de explosivos.
- La simulación numérica de partículas es muy similar a la dinámica molecular. Al estar este método muy desarrollado, se pueden utilizar las múltiples técnicas desarrolladas para él, como por ejemplo las listas de vecinos de Verlet o las celdas enlazadas, para acelerar la ejecución. También se pueden adaptar las técnicas que no pueden aplicarse directamente, como los integradores temporales: en dinámica molecular las fuerzas dependen por lo general de las posiciones solamente; si éstas dependen de la velocidad también los algoritmos deben ser adaptados. En este último punto, hay una fuerte analogía con la dinámica de partículas disipativa (DPD), una técnica de simulación molecular donde aparecen fuerzas de fricción dependientes de la velocidad.

Históricamente, estas simulaciones comienzan con la propuesta de la hidrodinámica de partículas suavizadas (*Smoothed Particle Hydrodynamics*, SPH). Hoy en día es claramente el método más popular de simulación lagrangiana.

Sin embargo, antes de presentar el método, y otras teorías más alternativas, es conveniente formular los métodos de partículas dentro de un formalismo genérico. Seguiremos la Ref. [14], uno de los trabajos más claros sobre este tema.

4.3.1. Formulación genérica

En la sección 2.1 se hizo hincapié en el punto de vista de Lagrange. Este punto de vista es el natural para formular un método de partículas lagrangianas, igual que la imagen de Euler es la natural para los métodos de malla fija.

Una partícula en nuestra simulación será, por tanto, un pequeño sistema físico con sus magnitudes correspondientes. Para una partícula a , de una total de N , tendremos: la masa M_a , volumen \mathcal{V}_a , velocidad v_a , energía interna $e_a \dots$

Una aplicación directa de las ecuaciones (2.1) y (2.2) nos proporciona estas ecuaciones de movimiento de una partícula lagrangiana:

$$\begin{aligned} \frac{dM_a}{dt} &= 0 \\ \frac{dr_a}{dt} &= 0 \end{aligned} \tag{4.3}$$

El cambio en velocidad vendrá dado por la ecuación de Navier-Stokes (2.8), “particularizada”:

$$\rho_a \frac{dv_a}{dt} = \rho_a g - \nabla_a p + \nabla_a \cdot (\mu_a \overline{\nabla_a v} + \lambda_a (\nabla_a \cdot v) \bar{\mathbf{1}}). \tag{4.4}$$

En esta ecuación todos los términos son bastante fáciles de evaluar salvo los operadores diferenciales espaciales. En efecto, con ∇_a nos referimos en ella a “una aproximación para el operador ∇ en la partícula a ”. La dificultad radica justamente en cómo definir un operador que tenga propiedades numéricas adecuadas.

Divergencia

Para avanzar en este problema de una manera sistemática, fijémonos en la ecuación (2.3) para el cambio de volumen de una partícula, y apliquémosla a una partícula:

$$\frac{d\mathcal{V}_a}{dt} = \mathcal{V}_a \nabla_a \cdot v. \tag{4.5}$$

En este momento, formulemos esta importante condición:

Propiedad. *Volumen de la partícula:* El volumen de una partícula dada viene determinado por la posición del conjunto de las partículas.

Esta condición es bastante genérica y no especifica de momento cuál es procedimiento para obtener el volumen de una partícula. Como veremos, distintos procedimientos conducen a distintas aproximaciones.

Si admitimos esto, la regla de la cadena nos dice que el cambio en el volumen será

$$\frac{d\mathcal{V}_a}{dt} = \sum_b \frac{d\mathcal{V}_a}{dR_b} \cdot \frac{dR_b}{dt} = \sum_b \frac{d\mathcal{V}_a}{dR_b} \cdot v_b. \tag{4.6}$$

Aquí se ha escrito $d\mathcal{V}_a/dR_b$ para el vector que expresa el cambio del volumen \mathcal{V}_a cuando cambia la posición de la partícula b . Es decir, su componente cartesiana i vendrá dada por $d\mathcal{V}_a/dx_{b,i}$.

Comparando la expresión con (4.5), vemos que una aproximación a la divergencia en la partícula a sería

$$\nabla_a \cdot v \doteq \frac{1}{\mathcal{V}_a} \sum_b \frac{d\mathcal{V}_a}{dR_b} \cdot v_b. \quad (4.7)$$

En rigor, esta sería una aproximación a la divergencia de la velocidad. Sin embargo, como estamos buscando una expresión general para la divergencia, esta fórmula debe tomarse como válida para cualquier campo vectorial, no sólo el de velocidades.

El gradiente

Para obtener una expresión para el gradiente, consideremos la ecuación de Euler para una partícula,

$$\rho_a \frac{dv_a}{dt} = -\nabla_a p, \quad (4.8)$$

donde omitimos el término gravitacional, que no conlleva ninguna dificultad.

En la sección anterior conectamos una ecuación tomada del continuo con otra procedente de la regla de la cadena y de la condición del volumen de una partícula. Ahora seguiremos el mismo camino, definiendo la energía de una partícula a como en (2.10):

$$E_a = U_a + \frac{1}{2} m_a v_a^2. \quad (4.9)$$

La energía total del sistema es

$$E = \sum_a E_a = \sum_a \left(U_a + \frac{1}{2} m_a v_a^2 \right).$$

Igual que en el caso continuo, (2.11), el cambio de la energía será

$$\frac{dE}{dt} = \sum_a \left(m_a v_a \cdot \frac{dv_a}{dt} + \frac{dU_a}{d\mathcal{V}_a} \frac{d\mathcal{V}_a}{dt} \right). \quad (4.10)$$

Por otro lado, conocemos la relación termodinámica

$$-p = \frac{\partial U}{\partial V},$$

válida para un proceso adiabático. Así pues,

$$\frac{dE}{dt} = \sum_a \left(m_a v_a \cdot \frac{dv_a}{dt} - p_a \frac{d\mathcal{V}_a}{dt} \right). \quad (4.11)$$

Introduciendo la ecuación (4.6),

$$\frac{dE}{dt} = \sum_a \left(m_a v_a \cdot \frac{dv_a}{dt} - p_a \sum_b \frac{d\mathcal{V}_a}{dR_b} \cdot v_b \right). \quad (4.12)$$

Para recuperar la ecuación de Euler, este cambio de energía debe ser nulo:

$$\frac{dE}{dt} = 0 \quad \rightarrow \quad \sum_a m_a v_a \cdot \frac{dv_a}{dt} = \sum_a p_a \sum_b \frac{d\mathcal{V}_a}{dR_b} \cdot v_b. \quad (4.13)$$

Vemos que una posibilidad de satisfacer esta ecuación es definiendo

$$m_a \frac{dv_a}{dt} = \sum_b \frac{d\mathcal{V}_b}{dR_a} p_b. \quad (4.14)$$

Una comparación con la ecuación de Euler (4.8) revela que acabamos de introducir una aproximación para el gradiente de la presión:

$$\nabla_a p \doteq -\frac{1}{\mathcal{V}_a} \sum_b \frac{d\mathcal{V}_b}{dR_a} p_b. \quad (4.15)$$

Esta es una expresión similar a la de la divergencia (4.7), pero con sutiles diferencias: un cambio de signo y una inversión de a y b en la derivada. Además, estos dos operadores son mutuamente adjuntos en cierto sentido, como se discute en [14].

4.3.2. SPH

Este método fue propuesto en 1977 de manera casi simultánea por Lucy [15] y Gingold y Monaghan [16]. Fue la segunda referencia la que lo bautizó con el término con que se conoce hoy en día, hidrodinámica de partículas suavizadas (*Smoothed Particle Hydrodynamics*, SPH).

La idea era utilizar las técnicas de dinámica molecular, bastante bien asentadas por aquel entonces, para atacar problemas de hidrodinámica. La mecánica molecular, en resumen, consiste en resolver las ecuaciones de movimiento de Newton para un conjunto de moléculas. Sus “partículas” son, por tanto, centros de interacción móviles descritos por magnitudes físicas moleculares, microscópicas. Por ejemplo: masa, carga, momento dipolar; pero nunca presión ni viscosidad. La ventaja de el programa SPH es la relativa facilidad de implementar un método lagrangiano de este modo. Por ejemplo, no requiere ningún mallado (si acaso, y por motivos de eficacia numérica, una simple lista de vecinos que se utiliza a la hora de calcular las interacciones).

El método introduce una función núcleo (*kernel*) W para describir las interacciones entre partículas. Esta función tiene por lo general forma de campana, es isotrópica, y se anula para puntos lejanos (tiene “soporte compacto”). El alcance de esta función es muy importante.

La manera más natural de obtener una formulación SPH dentro del formalismo genérico de la sección 4.3.1 es a partir de una definición de volumen de una partícula

$$\frac{1}{\mathcal{V}_a} = \sum_b W(r_{ab}),$$

donde $r_{ab} = |r_a - r_b|$. Las funciones núcleo tienen por tanto dimensiones de inversa de volumen.

Si una partícula a tiene numerosas vecinas b dentro del soporte de su núcleo, entonces el sumatorio tendrá un valor elevado y su volumen será pequeño. Si, por el contrario, no hay ninguna, el sumatorio sólo tendrá el término $a = b$ y el volumen será muy grande (este límite identifica además $w(0)$ como la inversa del volumen mínimo de una partícula).

Utilizando esta definición de volumen en la ecuación general de la divergencia (4.7) se llega directamente a esta expresión:

$$\nabla_a \cdot v \doteq \mathcal{V}_a \sum_{b \neq a} \nabla W_{ab} \cdot v_{ab}, \quad (4.16)$$

donde se quiere decir

$$\nabla W_{ab} = \nabla W(r_{ab})$$

y v_{ab} es la velocidad relativa de las partículas a y b :

$$v_{ab} = v_a - v_b.$$

De hecho, se han propuesto diversas definiciones de la divergencia en SPH, y esta correspondería a una de ellas, muy similar a los discutidos en el artículo de referencia de Monaghan [5] (es idéntica a una de ellas si las masas de las partículas son iguales, $m_a = m_b$; también puede deducirse exactamente si en la ecuación (2.13) de esta referencia se usa $\Phi = 1/\mathcal{V}$).

En cuanto al gradiente, la ecuación genérica (4.15) se puede escribir así:

$$\nabla_a p \doteq \frac{1}{\mathcal{V}_a} \sum_{b \neq a} (\mathcal{V}_a^2 p_a + \mathcal{V}_b^2 p_b) \nabla W_{ab}. \quad (4.17)$$

Esta expresión es casi idéntica a la recomendada por Monaghan [17, 5] (para ser precisos, las masas deben ser iguales para que se recupere exactamente la misma expresión).

Estas expresiones pueden servir de partida para desarrollar un método de simulación completo, que requiere atender numerosos detalles, entre otros:

La ecuación de estado. Para cerrar el sistema de ecuaciones, se debe relacionar presión y densidad. Como las aplicaciones suelen ser hidrodinámicas, el sistema debe ser incompresible. Así pues, la ecuación de estado debe presentar una subida drástica de la presión con la densidad (se suele utilizar la de ecuación de Cole, que es muy sencilla). Alternativamente, se puede intentar implementar una incompresibilidad rigurosa, lo cual conlleva bastantes dificultades técnicas.

Derivadas de orden superior. Por lo general, el laplaciano no se implementa como la divergencia del gradiente, ni tampoco se utilizan derivadas segundas del núcleo; por el contrario, se emplean otras expresiones equivalentes en teoría pero muchos más fáciles de implementar en la práctica (tan complicadas como las expresiones anteriores para la divergencia y el gradiente).

Las condiciones de contorno. Como los núcleos de las partículas se extienden bastante, y son isotrópicos por lo general, las condiciones de contorno son complicadas de implementar.

Integración temporal. Aunque no presenta ni mucho menos las dificultades de la integración espacial (al ser el tiempo unidimensional, y no estar discretizado de ninguna manera concreta), lo cierto es que es necesario hacer que el sistema evolucione en el tiempo con un algoritmo que tenga características adecuadas. Por ejemplo, integradores simplécticos, como el de Verlet en alguna de sus variantes.

No es este el lugar para una discusión de todos estos aspectos del método SPH, ni de sus múltiples aplicaciones. Como se ha dicho, el método es, con gran diferencia, el más utilizado para simulaciones lagrangianas. Existe una fuerte comunidad académica que emplea este método. Por ejemplo, la ERIC (*European Research Interest Group*, Grupo de Interés de Investigación Europeo) SPHERIC, dentro de la comunidad ERCOFTAC (*European Research Community On Flow, Turbulence, and Combustion*) aglutina 12 grupos directivos (*Steering Comitee*), 52 grupos europeos adicionales y 23 de fuera de Europa. Cada año organiza un taller sobre el método, en cuyas dos últimas convocatorias el candidato ha presentado comunicaciones (en sesiones de “teorías alternativas”) [18, 19]. Las aplicaciones industriales también cabida en este grupo, con diversos grupos del campo de la hidráulica, energía, geología, obras públicas y explosivos.

La aplicación más existosa del método es, sin duda, en el campo de los efectos especiales. El programa informático RealFlow, desarrollado por la compañía Next Limit, es hoy en día el más destacado a la hora de desarrollar efectos realistas de líquidos. La compañía recibió un Premio de la Academia por Logros Técnicos (de la Academia de las Artes y las Ciencias Cinematográficas de EEUU) por su trabajo en *El Señor de los Anillos*, donde se simulaba agua y otros fluidos, como lava. Sin embargo, la compañía al parecer no utiliza SPH, sino otro método (mallado de Boltzmann) para su programa de simulación ingenieril XFlow.

Quizá esto último represente un punto de vista extendido: a pesar de la potencia y simplicidad de SPH, parece difícil ir “más allá” con él, y llegar hasta un desarrollo y aplicación comparables a los métodos de dinámica de fluidos computacional basados en mallas eulerianas. El método debe ser, por tanto, refinado considerablemente de manera apropiada para cada aplicación. Los recursos informáticos pueden ser, además, bastante intensivos si se requiere precisión: una característica de SPH es que se puede aumentar la precisión aumentando el alcance del núcleo. Esto es por un lado positivo, ya que al menos existe esta posibilidad; pero un núcleo muy extendido amplía el número de vecinas de cada partícula, aumentando considerablemente el coste de la simulación en términos de tiempo de ejecución y también de memoria.

Una alternativa es considerar otros métodos basados en partículas, aprovechando las lecciones aprendidas de SPH (que son numerosas, en particular en lo relativo a los detalles técnicos mencionados arriba). Un enfoque muy interesante, pero que no

desarrollaremos en detalle aquí, es aplicar ideas del método de los volúmenes finitos. Referimos al interesante trabajo de V. Springel [20].

CAPÍTULO 5:

MÉTODOS BASADOS EN DIAGRAMAS DE VORONOI

La idea de las formulaciones de partículas basadas en diagramas de Voronoi es sencilla: usar la celda de Voronoi de una partícula a para definir sus propiedades. Se trata de una idea original de M. Serrano y P. Español [1], que luego ha sido discutida por el mismo grupo investigador [21, 14, 18] y también por otros [22].

En realidad, basta con definir el volumen de una partícula, \mathcal{V}_a , como el volumen de su celda de Voronoi asociada. El formalismo genérico de la sección 4.3.1 puede emplearse entonces, y sólo basta determinar la variación de \mathcal{V}_a cuando cambia R_b , la posición de la partícula b (sea b igual a a o no).

5.1. Operadores diferenciales

Los cálculos resultan ser bastante farragosos (ver apéndice de [1]), pero lo interesante es que este cambio viene dado por magnitudes geométricas bastante intuitivas:

$$\frac{\partial \mathcal{V}_a}{\partial r_b} = \mathcal{A}_{ab} \left(\frac{u_{ab}}{2} - \frac{c_{ab}}{r_{ab}} \right).$$

Las expresiones que aparecen en ella son (ver figura 5.1):

- El área (o longitud en 2d) de la cara de Voronoi común a a y b , \mathcal{A}_{ab} .
- La distancia entre a y b , r_{ab} .
- El vector unitario de a hacia b , u_{ab} .
- El vector c_{ab} tiene una definición más complicada: va del punto medio entre a y b hasta el centro de masas de la cara de Voronoi común a a y b . Si el vector u_{ab} es “radial”, éste sería “transversal” o “de cizalla”.

Además, si $b = a$,

$$\frac{\partial \mathcal{V}_a}{\partial r_b} = - \sum_{b \neq a} \frac{\partial \mathcal{V}_b}{\partial r_a}.$$

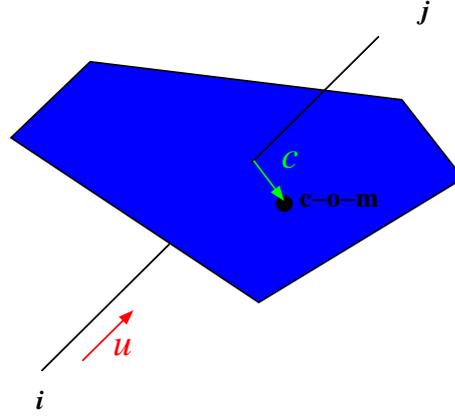


Figura 5.1: Magnitudes geométricas que aparecen en el cambio del volumen de una celda de Voronoi.

Esta propiedad implica que la divergencia de un campo constante será nula; utilizándola, una manera concisa de expresar el operador de la divergencia (4.7) es:

$$\nabla_a \cdot v \doteq \frac{1}{\mathcal{V}_a} \sum_{b \neq a} \mathcal{A}_{ab} \left[\frac{\mathbf{u}_{ab}}{2} - \frac{\mathbf{c}_{ab}}{R_{ij}} \right] \cdot (v_b - v_a). \quad (5.1)$$

Como se ve, la contribución de la partícula b a a es en todo caso proporcional al área (longitud en 2d) de la cara en contacto común. Luego, hay una parte central de la divergencia, proporcional a $u_{ab} \cdot (v_b - v_a)$, y una parte transversal proporcional a $u_{ab} \cdot (v_b - v_a)$.

También puede escribirse la ecuación para el gradiente (4.15) de este modo:

$$\nabla_a p \doteq \frac{1}{\mathcal{V}_a} \sum_{b \neq a} \mathcal{A}_{ab} \left[\frac{\mathbf{u}_{ab}}{2} (p_a + p_b) - \frac{\mathbf{c}_{ab}}{R_{ij}} (p_a - p_b) \right], \quad (5.2)$$

donde se ha simetrizado la expresión utilizando el hecho

$$\sum_b \mathcal{A}_{ab} \mathbf{u}_{ab} = 0,$$

lo cual es fácil de demostrar como una consecuencia del teorema de la divergencia de Gauss aplicado a campos constantes.

5.2. Dificultades

En la referencia[23], Mar Serrano informaba de que esta implementación fallaba parcialmente en una prueba muy sencilla: un campo externo de cizalla con forma sinusoidal. Más adelante, en la referencia [14] Pep Español y la misma autora analizaban las expresiones para los operadores diferenciales, dejando de lado por

claridad las simulaciones hidrodinámicas completas. Se consideraban partículas dispuestas sobre el cuadrado unidad (tanto aleatoria como regularmente), en las cuales se imponía el campo

$$P(r) = P_0 \text{sen}(k \cdot r)$$

(no se dice en el artículo, pero creemos que $P_0 = 1$ y $k = 2\pi$). Se evalúa el gradiente según la fórmula 5.2, y se compara con el exacto,

$$g(r) = P_0 k \cos(k \cdot r) \vec{u}_x$$

mediante un error L_1 :

$$L_1 = \frac{1}{N} \sum_a |(\nabla P)_a - g(r_a)|.$$

En la figura 5.2 se representa, a la izquierda, los resultados para distribuciones regulares y aleatorias. Como es de esperar, la convergencia es más lenta en el segundo caso, pero el operador diferencial converge adecuadamente.

La sorpresa aparece al hacer lo análogo para la divergencia. Primero se impone un campo vectorial

$$v(x, y) = v_0 \text{sen}(k \cdot r)$$

cuya divergencia es

$$d(x, y) = v_0 \cdot k \cos(k \cdot r).$$

El error converge igual de bien en distribuciones regulares, pero la convergencia se destruye totalmente en distribuciones al azar. Este resultado es un auténtico obstáculo para una simulación de partículas. Incluso si se consigue una formulación que no involucre la divergencia, lo cual es bastante viable, el laplaciano que aparece en el término viscoso se verá forzosamente “contaminado” por la divergencia. Esto es obvio si éste se evalúa como la divergencia del gradiente. Por supuesto, todavía puede simularse un fluido inviscible, como se hace en la Ref. [21], ya que en la ecuación de Euler no aparece la viscosidad

5.3. Convergencia del laplaciano

En artículos de Pep Español, Ignacio Zúñiga y Jesús G. Anero [24, 25] se utilizaba de nuevo el diagrama de Voronoi en el contexto de la teoría de transporte. Se identificó que las funciones de correlación temporales presentaban una patológica singularidad en el origen. Una posible solución era utilizar elementos finitos como funciones base, lo cual llevaba directamente a la expresión de la cotangente para el laplaciano, Ec. (4.2).

Parece natural, por tanto, utilizar esta misma idea dentro del contexto de las simulaciones de partículas. La esperanza es circunvalar las dificultades expuestas en la sección 5.2 y evaluar directamente el operador laplaciano, evitando utilizar la divergencia. En esta sección se presenta material original que explora esta posibilidad.

Consideraremos el problema en una y dos dimensiones, que discutimos por separado.

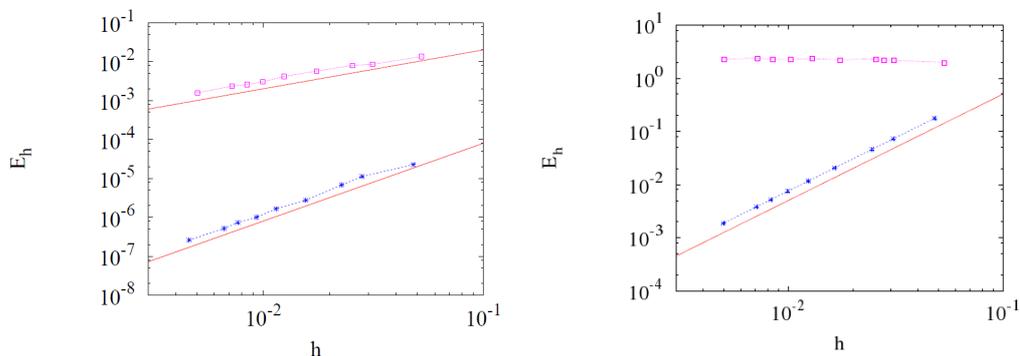


Figura 5.2: Convergencia de los operadores gradiente (izquierda) y divergencia (derecha) para partículas al azar en dos dimensiones. Se dibuja el error L_1 en función de la distancia promedio entre partículas $h = 1/\sqrt{N}$, en escalas logarítmicas. De [14].

5.3.1. Una dimensión

De manera similar a lo explicado en la sección 5.2, en una dimensión colocaremos N partículas sobre el segmento $[0, 1]$ en estas posibles disposiciones:

- Sobre una malla regular, con las partículas en posiciones $x_a = a/(N-1) - 1/2$, $a \in [1, N]$.
- Colocadas al azar, con posiciones tomadas de números aleatorios entre 0 y 1 (éstos no incluidos).

Nuestra función prueba es

$$f(x) = \frac{1}{(2\pi)^2} \text{sen}(2\pi x),$$

que cumple las condiciones de Dirichlet homogéneas $f(-1/2) = f(1/2) = 0$. Su laplaciano exacto es

$$g(x) = f''(x) = -\text{sen}(2\pi x).$$

Discutimos tanto los resultados de aplicar el método de elementos finitos completo,

$$\vec{g} \doteq m^{-1} \Lambda \vec{f},$$

como el de masa acumulada,

$$\vec{g} \doteq m_0^{-1} \Lambda \vec{f}.$$

Este último es muy rápido computacionalmente al no requerir inversión matricial.

En la figura 5.3 vemos los resultados de aplicar los dos métodos en una malla regular para $N = 10$. El acuerdo es claramente excelente, incluso para este número tan bajo. En 5.4, con el mismo N pero con posiciones al azar, se aprecia que el resultado no es tan bueno. Las partículas que están muy alejadas causan grandes

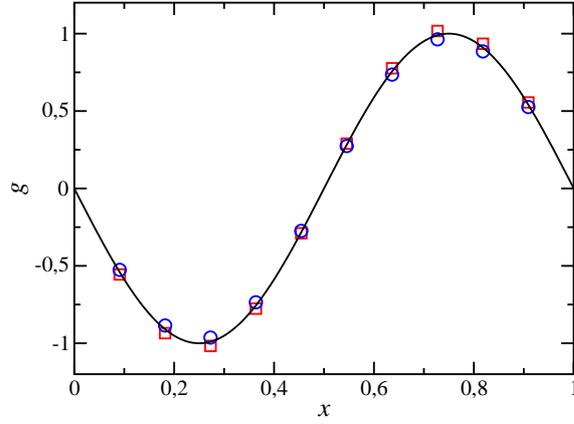


Figura 5.3: Laplaciano aproximado por el método FEM en la versión completa (círculos azules) y de masa acumulada (cuadrados rojos) para partículas equiespaciadas en una dimensión. En línea continua negra, la solución exacta.

desviaciones respecto del valor exacto, en ambas versiones. En esta distribución en particular, existe un gran vacío de partículas en la parte izquierda del segmento, y ambas versiones fallan, como es natural. Sin embargo, la versión de masa acumulada parece fallar también en zonas con alta concentración de partículas, mientras que la completa no. Esto se debe a la multiplicación por la matriz m , que suaviza estas fluctuaciones.

Para cuantificar la convergencia, utilizaremos el error L_2 relativo:

$$L_2 = \sqrt{\frac{\sum (g(x_a) - g_a)^2}{\sum (g_a)^2}}. \quad (5.3)$$

En la figura 5.5 se muestra esta magnitud para el caso regular y aleatorio. En el primer caso se observa una clara convergencia de las dos versiones del método, con resultados casi indistinguibles en la escala empleada. Un ajuste a una ley de potencia revela una convergencia como $L_2 \sim N^{-2}$. En el caso aleatorio, en cambio, las dos versiones se distinguen claramente; la de masa completa sigue convergiendo con el mismo exponente -2 , pero la de masa acumulada converge más lentamente, como $L_2 \sim N^{-1}$.

Para completar el estudio, dibujamos también la convergencia del problema inverso (de Poisson):

$$\vec{g} \doteq \Lambda^{-1} m \vec{f},$$

o, en versión de masa acumulada,

$$\vec{g} \doteq \Lambda^{-1} m_0 \vec{f}.$$

En la figura 5.6 se ve que las aproximaciones convergen; esta vez hay poca diferencia entre las dos, y obtenemos un ajuste aproximado a la ley $L_2 \sim N^{-2,4}$. Es decir, la convergencia es mejor que en problema directo.

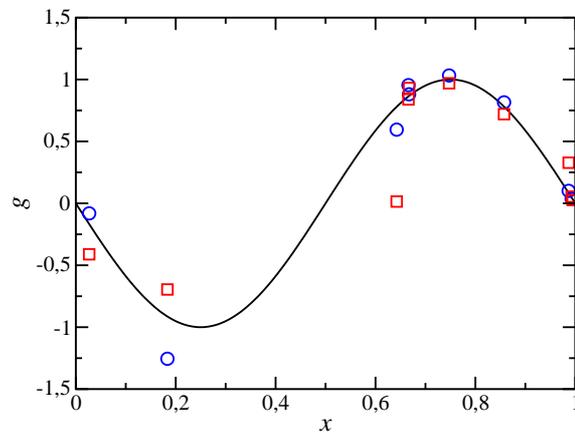


Figura 5.4: Igual que en la figura 5.3, pero para distribución al azar de partículas.

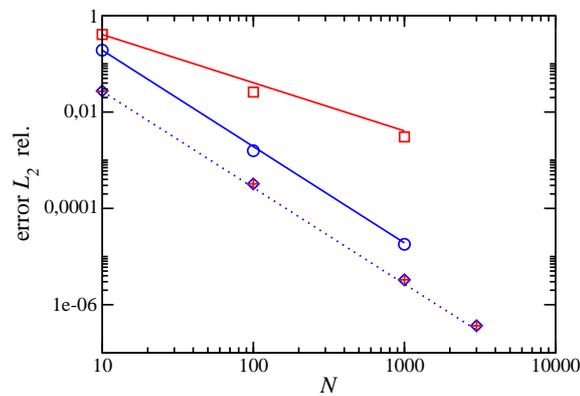


Figura 5.5: Convergencia del laplaciano aproximado por el método FEM. Círculos azules: versión completa en disposición aleatoria; diamantes azules: versión completa en disposición regular; cuadrados rojos: versión de masa acumulada en disposición aleatoria; cruces rojas: versión de masa acumulada en disposición regular. Las líneas azules son ajustes a los puntos mediante leyes de potencia de exponente -2 ; la línea roja, ajuste mediante ley de potencia de exponente -1 .

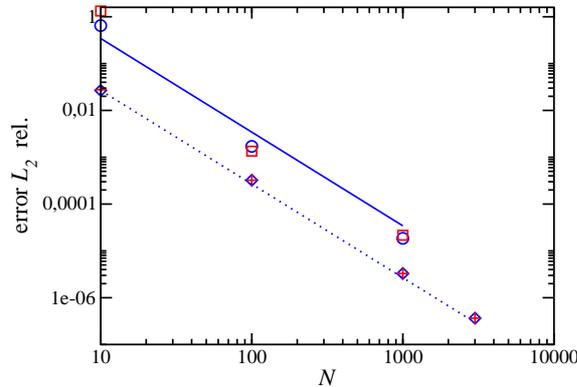


Figura 5.6: Convergencia del problema de Poisson aproximado por el método FEM. Símbolos como en 5.5.

5.3.2. Dos dimensiones

En dos dimensiones colocaremos N partículas sobre el cuadrado unidad $[0, 1] \times [0, 1]$ en estas posibles disposiciones:

- Sobre una malla regular cuadrada, con las partículas en posiciones $x_a = a/(L - 1) - 1/2$, $a \in [1, L]$, con $L = \sqrt{N}$, y similarmente para y_a .
- Colocadas al azar, con componentes cartesianas tomadas de números aleatorios entre 0 y 1 (éstos no incluidos).
- Como estado intermedio, también contemplamos una perturbación de la malla regular cuadrada, en la que cada partícula puede desplazarse al azar dentro de un cuadrado cuyo lado es un 10% de la distancia entre partículas.
- Por último, utilizamos el generador de mallas (“mallador”) de la aplicación GiD [26], diseñado justamente para calcular mallas que tengan buenas propiedades.

En este caso la función que utilizamos es

$$f(x) = \frac{1}{(2\pi)^2} \text{sen}(2\pi x) \text{sen}(2\pi y),$$

que cumple condiciones de Dirichlet homogéneas en el cuadrado unidad y cuyo laplaciano exacto es

$$g(x) = \nabla^2 f(x) = -2 \text{sen}(2\pi x) \text{sen}(2\pi y).$$

En la figura 5.7 se muestran 900 partículas colocadas al azar sobre el cuadrado unidad. Se dibuja también la triangulación de Delaunay correspondiente. Además se incluyen las “partículas de frontera”, necesarias a la hora de implementar las

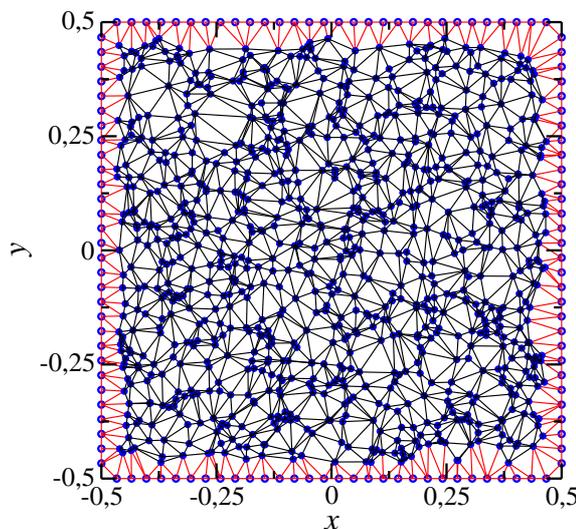


Figura 5.7: Partículas al azar sobre el cuadrado unidad. Círculos azules sólidos: partículas originales; círculos azules vacíos: partículas del borde añadidas. Segmentos negros: bordes de Delaunay internos; segmentos rojos: bordes externos.

condiciones de contorno (en una dimensión bastaba con dos, que situábamos en 0 y 1). Estas partículas reciben el valor 0 y se tienen en cuenta como vecinas de las partículas interiores (los bordes de la triangulación de color rojo), aunque no se calcula el laplaciano sobre ellas.

Igual que antes, calculamos el laplaciano mediante dos versiones: la completa y la de masa acumulada. En la figura 5.8 se muestra el resultado de la segunda versión para las posiciones aleatorias de la figura 5.7. Evidentemente, el resultado no es muy satisfactorio, con bastante dispersión cerca de los extremos de la función.

Para cuantificar esta dispersión y su dependencia del número de puntos, de nuevo utilizamos el error relativo L_2 , ecuación (5.3). Los resultados se muestran en la figura 5.9. Se puede concluir lo siguiente:

- El método de elementos finitos no es convergente en disposiciones desordenadas (aleatorias o perturbadas), en ninguna de las dos versiones.
- La aproximación de masa acumulada es, al contrario que en una dimensión, mejor que la completa.
- En mallas ordenadas el método sí converge. El ajuste revela que la convergencia es de orden N^{-1} , más lenta que en una dimensión. Esta convergencia es muy sensible al orden de la disposición de partículas: sólo se dibujan los resultados para N que son cuadrados perfectos; en otros casos, los errores son superiores debido a la imposibilidad de crear una malla cuadrada perfecta.
- El método también converge en mallas de buena calidad generadas por el programa especializado GiD. En este segundo caso, una visualización de la malla

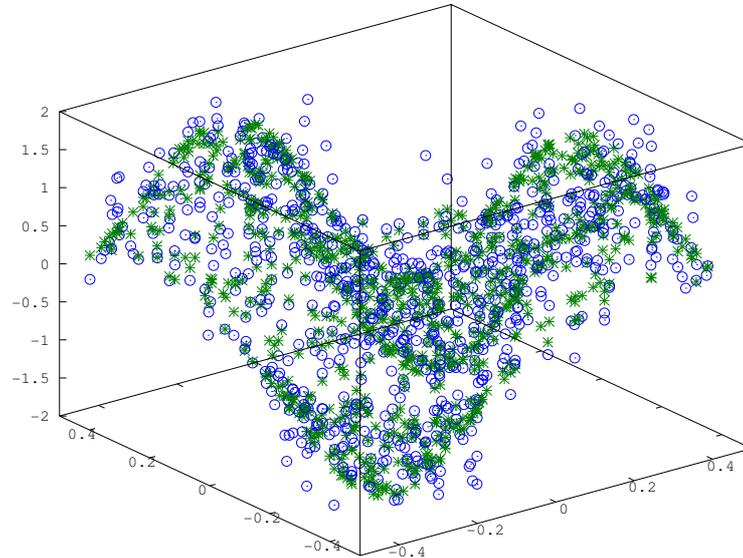


Figura 5.8: Aproximación de elementos finitos al laplaciano en 2D (círculos azules), comparada con el laplaciano exacto $g(x)$ (asteriscos verdes) .

(figura 5.10) permite comprobar que se trata de una malla regular triangular, salvo en ciertas “costuras” diagonales. La presencia de estas costuras empeora la convergencia, pero su importancia disminuye al crecer N .

5.3.3. Alternativas para el laplaciano: un resumen

Este resultado negativo es, por así decirlo, el último de una serie de fracasos relativos a la evaluación del laplaciano. Llevamos a cabo en esta sección una panorámica de esta serie, en orden no necesariamente cronológico.

Elementos finitos

Acabamos de discutir, en la sección anterior, el fracaso de este enfoque. Este hecho, que hemos constatado de manera independiente, parece ser conocido en la comunidad de optimización geométrica, donde se define el operador de Laplace-Beltrami sobre superficies[27].

Para completar la discusión, señalamos que es también posible formular expresiones para la divergencia y el gradiente mediante el método de elementos finitos. Es fácil ver que la matriz de masas es la misma, pero que en vez de la matriz de

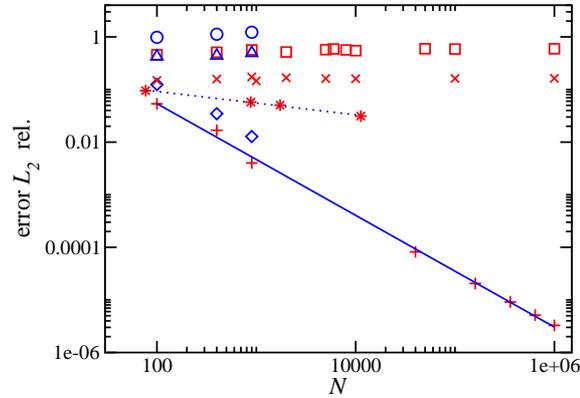


Figura 5.9: Convergencia del laplaciano aproximado por el método FEM en 2D. Círculos azules: versión completa en disposición aleatoria; diamantes azules: versión completa en disposición regular; triángulos azules: versión completa en disposición cuadrada perturbada; cuadrados rojos: versión de masa acumulada en disposición aleatoria; cruces rojas: versión de masa acumulada en disposición regular; espas rojas: versión de masa acumulada en disposición cuadrada perturbada. Línea azul continua: ajuste a una ley de potencia de exponente -1 ; línea azul punteada: ajuste a una ley de potencia genérica (el exponente resulta ser de $-0,22$).

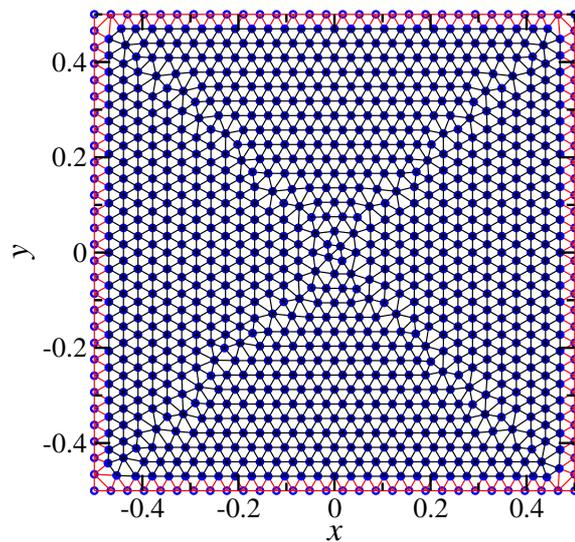


Figura 5.10: Partículas sobre el cuadrado unidad generadas mediante el programa GiD. Símbolos como en figura 5.7.

rigidez hay que considerar la matriz de vectores con componentes

$$\mathbf{G}_{ab} = \int s_a \nabla s_b.$$

De este modo se pueden obtener aproximaciones para la divergencia y el gradiente que, curiosamente, convergen para disposiciones sobre mallas regulares (su error L_2 , como N^{-1}) y también para posiciones aleatorias (como $N^{-1/2}$). Sin embargo, la combinación de ambas para formar el laplaciano sigue sin converger para disposiciones aleatorias (para regulares sí lo hace, como $N^{-1/4}$).

Elementos de Voronoi

Como se ha discutido en la sección 5.2, el operador divergencia no converge para disposiciones desordenadas, así que no puede combinarse con el gradiente (que sí converge) para formar un laplaciano convergente.

Laplaciano SPH

Como se discute claramente en [5], a partir de la fórmula para la interpolación de una función en SPH,

$$f(r) \doteq \sum_b \mathcal{V}_b W(r - r_b) f_b,$$

con el volumen SPH

$$\frac{1}{\mathcal{V}_b} = \sum_c W(r_{bc}),$$

uno escribiría

$$\nabla^2 f(r) \doteq \sum_b \mathcal{V}_b f_b \nabla^2 W(r - r_b). \quad (5.4)$$

Sin embargo, esta expresión directa para el laplaciano presenta una serie de desventajas:

- extrema sensibilidad a la regularidad de las posiciones de las partículas, un fenómeno claramente similar a lo que hemos visto arriba para elementos finitos.
- el hecho de que $\nabla^2 W$ es una función que tiene valores negativos lleva a fenómenos sin sentido físico; por ejemplo en la ley de Fourier de la difusión del calor uno podría tener flujo de calor de partículas frías a otras calientes.

Por ello, se suele emplear esta expresión,

$$\nabla^2 f(r) \doteq \sum_b \mathcal{V}_b (f_a - f_b) F(r - r_b), \quad (5.5)$$

donde la función F está definida por

$$\mathbf{r}F = \nabla W.$$

Por ejemplo, para un kernel gaussiano $W = \exp(-\alpha r^2)$, la función sería simplemente $F = -2\alpha W$.

Esta expresión presenta un comportamiento típico en SPH: no converge si la resolución se mantiene constante; es decir, si el número de vecinos (en promedio) de cada partícula no cambia. Esto se consigue escalando la anchura de la función núcleo W igual que la distancia media entre partículas. Si, por el contrario, la anchura se fija pero se incrementa el número de partículas, la expresión sí converge (a un coste computacional cada vez más elevado, claro está).

Otros

En la Referencia [27] se propone una fórmula que, si se inspecciona con cierta atención, resulta ser la expresión SPH (5.4) para un núcleo gaussiano (salvo un cambio menor en la definición de \mathcal{V}_b). La falta de comunicación entre distintas comunidades científicas explica que estos investigadores, del área de gráficos informáticos y optimización geométrica, desconozcan al parecer este método.

Los conceptos geométricos del método basado en el diagrama de Voronoi han aparecido en formulaciones similares. Por ejemplo, en la Referencia [28] se presentan varias expresiones bastante intuitivas basadas en triangulaciones y sus propiedades geométricas. El candidato ha comprobado que éstas también fracasan para el laplaciano. En la Referencia [29] se propone una expresión para el laplaciano a partir del método de los covolumenes que es extremadamente sencilla:

$$(\nabla^2 f)_a \doteq \frac{1}{\mathcal{V}_a} \sum_b \frac{\mathcal{A}_{ab}}{r_{ab}} (f_a - f_b);$$

la cual, por desgracia, tampoco converge.

Por último, el propio candidato exploró un método basado en mínimos cuadrados móviles (MLS)[18]. En este enfoque se impone consistencia hasta segundo orden por construcción, mediante un ajuste por mínimos cuadrados. Por desgracia, este método presenta numerosos problemas de estabilidad y de conservación.

CAPÍTULO 6:

MÉTODOS DE MÁXIMA ENTROPÍA

6.1. Introducción

Ciertos métodos basados en el concepto de *entropía máxima*, extremadamente conocido en el ámbito de la inferencia estadística, se han propuesto recientemente en el ámbito de la mecánica computacional.

La idea principal es que, para lograr la máxima flexibilidad en las funciones base, hay que abandonar la idea que éstas puedan tener una forma determinada prescrita. Por el contrario, deben ser evaluadas punto por punto según ciertos criterios. En general, estas funciones deben ser positivas:

$$s_a(x) > 0 \quad \forall x.$$

Además, un requisito básico es la consistencia de orden cero, que hace que una función constante se reproduzca exactamente:

$$\sum_a s_a(x) > 0 \quad \forall x.$$

Si ahora aislamos un punto x cualquiera, los valores de las funciones base son simplemente un conjunto de números $\{s_a\}$. Los anteriores requisitos son idénticos a los que se requieren de una serie de probabilidades asociadas a los posibles estados de un sistema. En este sentido, parece natural tomar ideas de la teoría de la probabilidad. En particular, la teoría de la información de Shannon introduce el concepto de *entropía* de un sistema, inspirado por la entropía física debida a L. Boltzmann:

$$S = - \sum_a s_a \log s_a$$

(en lo sucesivo, omitimos la referencia al punto x donde se evalúan estas cantidades). Se supone que la función $x \log x$ se extiende analíticamente de modo que $0 \log 0 = 0$.

Esta entropía cumple algunas propiedades atractivas:

- La entropía es *máxima*, e igual a $N \log N$ cuando no conocemos nada sobre el estado del sistema. En este caso, $\{s_a\} = \{1/N, 1/N, \dots\}$. Además, si se añaden más estados y N aumenta, la entropía aumenta.
- La entropía es *mínima*, y nula, si el sistema está en un estado b : $\{s_a\} = \{0, 0, \dots, 1, \dots\}$. En este caso, si se añaden más estados, pero estos son “imposibles” (tienen probabilidad nula), la entropía sigue siendo mínima.

- Se puede demostrar que la entropía combinada de dos sistemas independientes estadísticamente es la suma de las entropías $S = S_1 + S_2$. Esta propiedad depende de la forma funcional concreta de la entropía, $x \log x$ mientras que las anteriores se pueden cumplir con cualquier función con derivada segunda positiva que se haga cero en 0 y 1 (p.e. $x(1-x)$).

Todas estas propiedades son muy conocidas en el área de física estadística, sobre todo cuando los estados son discretos, como en mecánica estadística cuántica. En física se define en realidad

$$S' = -k_B \sum_a s_a \log s_a,$$

donde la constante de Boltzmann k_B establece la correspondencia fundamental entre entropía y energía (además de proporcionar las dimensiones físicas adecuadas de [energía]/[temperatura]). Usamos en esta sección primas para distinguir las magnitudes físicas de las adimensionales

En general, se suele buscar el extremo de una función objetivo de este tipo:

$$F = -S + L_0,$$

donde L_0 es una ligadura que contiene un multiplicador de Langrange para satisfacer la condición de consistencia de orden cero:

$$L_0 = \alpha \left(\sum_a s_a - 1 \right).$$

El extremo se alcanza para $s_a = 1/N$, que es el resultado trivial cuando no se aporta más información.

Para hacer que el método sea local e incorpore información sobre los nodos, en los métodos de máxima entropía se introduce, como es habitual en física estadística, una energía que debe minimizarse. La forma más típica de esta energía es

$$E = \beta \sum_a s_a (x - x_a)^2,$$

donde $\beta \geq 0$ es un parámetro de control: los nodos que estén más lejos de x están penalizados con respecto a los que estén más cerca, por contribuir más a la energía. La función que se minimiza ahora es

$$F = -S + E + L_0.$$

De nuevo, la conexión con la mecánica estadística es clara; en ésta se minimiza la energía libre de Helmholtz

$$F' = E' - TS',$$

donde T es la temperatura. O también

$$\beta F' = \beta E' - S'/k_B,$$

donde $\beta = 1/(k_B T)$ (históricamente, la denominación β procede además de su condición de multiplicador de Lagrange [30].) Así que la función de F de los métodos de máxima entropía es la energía libre de Helmholtz adimensionalizada, y lo mismo se aplica a E y S .

Se puede por tanto considerar el límite $\beta \rightarrow 0$ como el límite “frío”, que es el dominado por la energía, y $\beta \rightarrow \infty$ como el “caliente”, dominado por la entropía.

El extremo de la función se obtiene facilmente:

$$s_a = \frac{1}{Z} e^{-\beta(x-x_a)^2},$$

donde Z es la denominada función de partición:

$$Z = \sum_b e^{-\beta(x-x_b)^2}.$$

Queda patente que el resultado son funciones base gaussianas, de anchura $\approx 1/\sqrt{\beta}$: más finas en el límite frío y más anchas en el caliente.

Este resultado es, de hecho, idéntico al método SPH con funciones núcleo (“kernel”) gaussianas, corregidas a la manera de Shepard (a través del término Z) para satisfacer consistencia de orden cero. Es trivial recuperar cualquier función núcleo SPH que se desee cambiando la definición de la energía.

6.2. LME

La primera propuesta en aparecer históricamente consiste en una mejora del caso sencillo arriba expuesto para conseguir consistencia a primer orden:

$$\sum_a s_a x_a = x.$$

El término de ligadura con el multiplicador de Lagrange necesario es:

$$L_1 = \lambda \left(\sum_a s_a (x_a - x) \right).$$

La función objetivo que hay que extremizar es ahora:

$$F = -S + E + L_0 + L_1.$$

Ya no es posible llevar a cabo esta tarea de manera analítica, pero sí se puede hallar el extremo con respecto a los valores s_a , con el siguiente resultado:

$$s_a = \frac{1}{Z} \exp(f_a), \tag{6.1}$$

donde

$$f_a = \lambda(x - x_a) - \beta(x - x_a)^2 \tag{6.2}$$

y la función de partición es, como antes,

$$Z = \sum_{a=1}^N f_a. \quad (6.3)$$

Al insertar estas expresiones en la función original, el problema queda reducido a encontrar el mínimo de la función $\log Z$ con respecto a λ :

$$\min_{\lambda} \log Z. \quad (6.4)$$

Una vez este mínimo se ha encontrado, los valores en el mínimo de λ , que llamaremos λ^* , y la función en el mínimo $\log Z^*$ pueden evaluarse para obtener las funciones base según (6.1).

6.2.1. Varias dimensiones

En la discusión anterior, se han desarrollado expresiones que valen sólo para una dimensión espacial. En varias dimensiones, las expresiones son muy similares. La energía es:

$$E = \beta_{ij} \left[\sum_a s_{ai}(x_i - x_{ai})(x_j - x_{aj}) \right],$$

es decir, β es ahora un tensor de dimensión 2 y tamaño $d \times d$. Los subíndices a y b se reservan para enumerar los nodos, y los i, j, \dots para las coordenadas cartesianas. Se supone la regla de suma de Einstein para estas últimas: cuando hay índices repetidos se debe sumar para todas las dimensiones espaciales. Como es obvio, el tensor debe ser simétrico, ya que $x_i x_j = x_j x_i$.

El término de ligadura para la consistencia de primer orden pasa a ser

$$L_1 = \lambda_i \left(\sum_a s_{ai}(x_{ai} - x_i) \right),$$

así que λ es ahora un vector de longitud d .

Al extremizar la función, la solución tiene la misma forma, ecuación (6.1), pero el exponente es:

$$f_a = \lambda_i(x_i - x_{ai}) - \beta_{ij}(x_i - x_{ai})(x_j - x_{aj}). \quad (6.5)$$

Por supuesto, la minimización no es ahora con respecto a un sólo parámetro, sino con respecto a cada una de las componentes de λ : d componentes independientes en d dimensiones.

6.2.2. Propiedades de las funciones base

Igual que en el caso sencillo descrito anteriormente, el parámetro libre β regula la anchura de las funciones. A la hora de fijar este parámetro, es importante darse

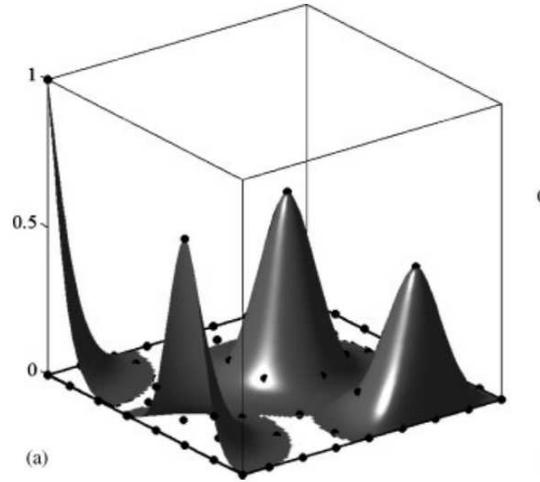


Figura 6.1: Funciones base LME en dos dimensiones. De [31].

cuenta de que lo relevante no es su valor numérico, sino su relación con la distancia entre nodos h . Escribiendo, en una dimensión

$$f_a = \lambda x - \beta x^2 = \lambda h \frac{x}{h} - \beta h^2 \frac{x^2}{h^2},$$

queda claro que lo relevante es βh^2 en realidad. En cambio, aunque es más elegante tomar λh como valor, este cambio es menos importante porque el valor de λ se determina al minimizar la función objetivo. Este hecho no cambia en dimensiones superiores, salvo si se define una h anisotrópica, una posibilidad interesante que hemos explorado, aunque no en profundidad.

En los casos sencillos considerados en [31], h es un escalar constante. A la hora de aplicar este método, en cambio, resulta necesario adecuar h al entorno local de cada nodo, porque si no un valor “frío” para puntos espaciados puede resultar “caliente” para otros más cercanos.

Una propiedad muy interesante de las funciones base resultantes de LME es su comportamiento en la frontera: a pesar de tener una anchura mucho mayor que la de los elementos finitos, satisfacen la misma condición de frontera que ellos. Esta condición, denominada de “casi- δ ” en [31] consiste en que todas las funciones son idénticamente nulas en la frontera del dominio (para ser exactos, en su envolvente convexa); excepto las correspondientes a los nodos que están en la frontera: para estos las funciones alcanzan su máximo. Como se ve en la figura 6.1, las funciones base nunca salen de la envolvente convexa, a pesar de que esta envolvente no se calcula nunca explícitamente.

Otra conexión sorprendente con la geometría computacional es el límite muy frío: cuando $\beta \rightarrow \infty$ el término de entropía es despreciable, y puede considerarse la función objetivo

$$F = E + L_0 + L_1.$$

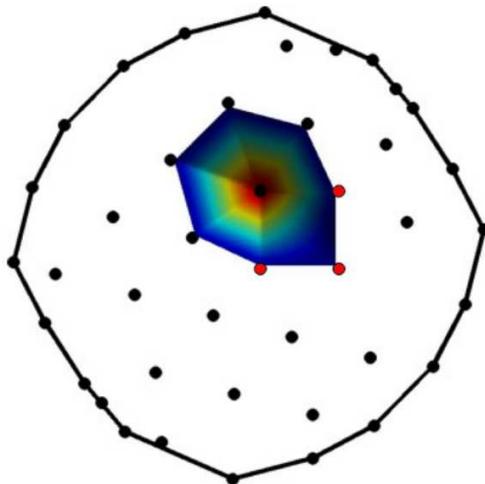


Figura 6.2: Método LME en el límite muy frío para una configuración de nodos cuya triangulación de Delaunay sería degenerada. De [31].

Las funciones base correspondientes al extremo de esta función son, como demostró Rajan [32], los elementos finitos definidos sobre la triangulación de Delaunay. El cálculo de estos elementos suele seguir métodos de geometría computacional, como se ha explicado, pero este método llega a ellos de una manera totalmente distinto. De hecho, el método es algo superior a la triangulación de Delaunay: si existe una configuración degenerada, como la de la figura 6.2, en la que hay cuatro nodos situados en los vértices de un cuadrado, hay dos triangulaciones de Delaunay posibles. En este caso, este método no proporciona ninguna de las dos, sino una función base suavizada, de un orden superior (cuadrático y no lineal).

6.3. SME

Este método se propuso como una extensión de LME para conseguir consistencia a segundo orden. Esta consistencia puede imponerse así

$$\sum_a s_a (x - x_a)^2 = 0,$$

una condición que, junto con las consistencias a orden cero y primer orden implica

$$\sum_a s_a x_a^2 = x^2.$$

La ligadura de Lagrange correspondiente sería pues

$$L_2 = \eta \left(\sum_a s_a x_a^2 - x^2 \right).$$

Es evidente que esta “ligadura” es exactamente el término de la energía de LME. Es decir: la idea sería considerar β , que es un parámetro fijo en LME, una variable con respecto a la cual minimizar, igual que λ . La función objetivo sería entonces

$$F = -S + E + L_0 + L_1,$$

igual que antes; pero ahora el extremo corresponde a

$$\min_{\lambda, \beta} \log Z. \tag{6.6}$$

6.3.1. Consistencia redefinida

Sin embargo, esta función objetivo no lleva a nada nuevo. Para ser más precisos, la solución de la función objetivo siempre conducirá al límite $\beta \rightarrow \infty$, es decir, la solución de elementos finitos sobre la triangulación de Delaunay hallada por Rajan. Existen dos maneras de explicar este hecho (que, desde luego, un cálculo numérico confirma inmediatamente).

La primera, que se puede encontrar en [31] y [33] (más detallada en el segundo), es de origen geométrico. En particular, se desarrolla en el “espacio de la solución”: un espacio $d + 1$ formado por los nodos y los valores de cierta función definida en ellos. Igual que en el espacio d , puede definirse una envolvente conexas de estos nodos-solución. Una función parabólica definida en los nodos estará necesariamente por fuera de esta envolvente conexas, y sólo la tocará en los propios nodos. Sin embargo, una condición necesaria para que la función objetivo tenga solución en un punto x es que este punto esté dentro de la envolvente conexas. Como la función está fuera, la conclusión es que una función parabólica nunca puede recuperarse de esta manera; salvo en los propios nodos. Sin embargo, incluso en estos es de esperar que no se recupere una función parabólica, sino la propia envolvente.

Una manera alternativa de enfocar el problema es a partir de la propia forma de $\log Z$. Para garantizar que la función tenga un mínimo, podemos examinar su límite cuando λ y β son muy grandes en valor absoluto.

Es fácil demostrar lo siguiente:

Límite $\lambda \rightarrow \infty$ En este límite, $\log Z \rightarrow \lambda(x - x_b)$, donde x_b es el punto más ... a x por la derecha...

Límite $\lambda \rightarrow \infty$ En este límite, $\log Z \rightarrow \lambda(x - x_c)$, donde x_c es el punto más ... a x por la derecha...

Límite $\beta \rightarrow -\infty$ En este límite, $\log Z \rightarrow -\beta(x - x_d)^2$, donde x_d es el punto más ... alejado...

Límite $\beta \rightarrow -\infty$ Este es el límite conflictivo, ya que $\log Z$ puede hacerse arbitrariamente pequeño.

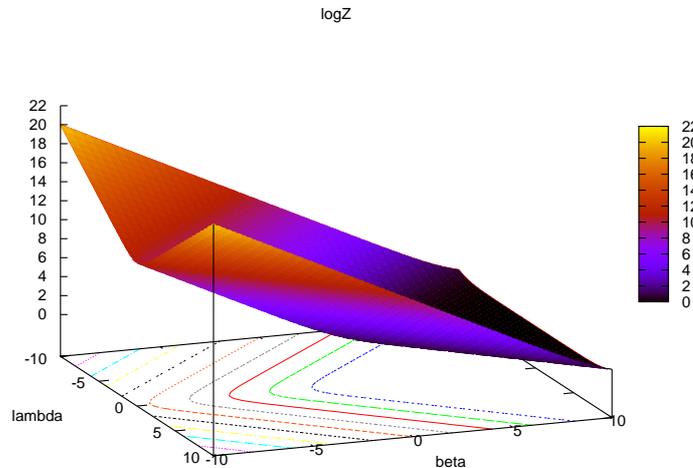


Figura 6.3: Dibujo de la función $\log(1 + \exp(-x - y) + \exp(x - y))$.

En la figura 6.3 se muestra un ejemplo sencillo, para sólo dos puntos. La situación cualitativa no cambia si se añaden más, como se ha explicado, ni tampoco en dimensiones superiores.

Una manera de evitar esto es, sencillamente, añadir una pendiente a la función objetivo, que “atraiga” al mínimo del infinito. Esto se muestra en la figura 6.3.

Al añadir esta pendiente, realmente estamos contemplando la siguiente minimización:

$$\min_{\lambda, \beta} \left(\beta g + \log \sum_a e^{\lambda(x-x_a) - \beta(x-x_a)^2} \right),$$

Otra manera de escribir esto mismo es:

$$\beta g + \log \sum_a e^{\lambda(x-x_a) - \beta(x-x_a)^2} = \log \sum_a e^{\lambda(x-x_a) - \beta[(x-x_a)^2 - g]},$$

lo que significa que nuestra ligadura original L_2 ha sido cambiada por esta otra:

$$L'_2 = \beta \sum_a s_a [(x - x_a)^2 - g].$$

Esta ligadura corresponde a imponer:

$$\sum_a s_a x_a^2 = x^2 + g.$$

Es decir: una función cuadrática no puede reproducirse exactamente, pero una función cuadrática desplazada por una función $g(x) > 0$, sí podría. Por ello, esta función se denomina “de salto” (*gap function*). Este salto, corresponde, además, al

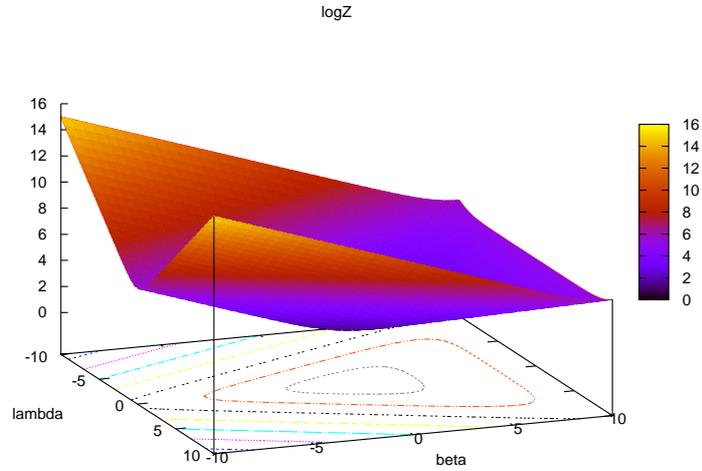


Figura 6.4: Dibujo de la función $\log(1 + \exp(-x - y) + \exp(x - y)) + 0,5y$.

desplazamiento vertical que habría que aplicarle a una parábola para meterla dentro de la envolvente conexa.

Una vez que este hecho se ha asimilado, se siguen una serie de conclusiones acerca de la forma general de la función $g(x)$:

Las funciones base son

$$s_a = \frac{1}{Z} \exp(f_a), \quad (6.7)$$

donde

$$f_a = \lambda(x - x_a) - \beta((x - x_a)^2 - g) \quad (6.8)$$

6.3.2. Varias dimensiones

De nuevo, la discusión anterior se ha limitado básicamente a una dimensión. En varias, las expresiones son similares.

El término de ligadura para la consistencia de segundo orden sería

$$L_2 = \beta_{ij} \left(\sum_a s_{ai} (x_{ai} - x_i) (x_{aj} - x_j) \right),$$

y el modificado,

$$L'_2 = \beta_{ij} \left(\sum_a s_{ai} [(x_{ai} - x_i) (x_{aj} - x_j) - g_{ij}] \right).$$

Así pues, la función $g(x)$ pasa a ser un tensor de orden 2, igual que β (y también simétrico).

Las funciones base tienen la misma forma, ecuación (6.7), pero el exponente es:

$$f_a = \lambda_i(x_i - x_{ai}) - \beta_{ij}((x_i - x_{ai})(x_j - x_{aj}) - g_{ij}). \quad (6.9)$$

Por supuesto, la minimización no es ahora con respecto a dos parámetros, sino con respecto a cada una de las componentes de λ y β : $2 + 3 = 5$ componentes independientes en dos dimensiones (β es simétrico, así que tiene 3 y no 4 componentes independientes), y $3 + 6 = 9$ en tres dimensiones.

6.3.3. Propiedades de las funciones base

Igual que en LME, es ahora la función g la que regula la anchura de las funciones. A la hora de fijarla, de nuevo lo relevante no es su valor numérico en cada punto, sino su relación con la distancia entre nodos h . Escribiendo, en una dimensión

$$f_a = \lambda x - \beta(x^2 - g) = \lambda h \frac{x - x_a}{h} - \beta h^2 \left[\frac{(x - x_a)^2}{h^2} - \frac{g}{h^2} \right], \quad (6.10)$$

así que lo relevante es gh^{-2} .

En los casos sencillos considerados en [33], h es un escalar constante.

De nuevo, las funciones base resultantes de SME, a pesar de tener una anchura mucho mayor que la de los elementos finitos, satisfacen la misma condición de frontera que ellos. De hecho, la apariencia de las funciones base es similar, pero estas nuevas funciones satisfacen la propiedad crucial de cumplir la consistencia de segundo orden (al menos en cuanto a sus derivadas).

CAPÍTULO 7:

MÁXIMA ENTROPÍA PARA PARTÍCULAS

Este capítulo recoge material propio muy reciente acerca de la posibilidad de utilizar los métodos de máxima entropía para simulaciones de partículas.

7.1. Adaptación a partículas

Los métodos de máxima entropía que se han propuesto emplean técnicas de Galerkin. Para ello es necesario evaluar las funciones base y sus derivadas en una serie de puntos espaciales. Como estos métodos requieren integraciones del producto de las funciones (para la matriz de masas) y de sus gradientes (para la de rigidez), es necesario evaluar las funciones en un conjunto de puntos mayor que el de los nodos.

La cuestión de escoger bien estos puntos y asignarles pesos se conoce con el nombre de “cuadratura”. Esta técnica es compleja ya que las funciones base no tienen soporte compacto; además, se extienden más allá de los primeros vecinos. De hecho, parece que en una dimensión son deseables unos 4 vecinos por partícula (dos a cada lado), y en dos dimensiones, unos 30. Los métodos de cuadratura suelen realizarse sobre una triangulación de Delaunay.

Por otro lado, un enfoque conceptual de cuadratura nos sacaría fuera del marco general de simulaciones de partículas. Según este, la interacción sobre una depende tan sólo de propiedades de las demás partículas. En este sentido, la introducción de “nodos” de cuadratura resulta extraña.

Así pues, para evitar la complejidad asociada a los cálculos de cuadratura desde un punto de vista práctico, y para mantenernos dentro del enfoque de partículas, hemos considerado la opción de aproximar directamente las sucesivas derivadas de los campos reconstruidos:

$$\begin{aligned} f(x) &\approx \sum_{a=1}^N s_a(x) f_a \\ \nabla f(x) &\approx \sum_{a=1}^N \nabla s_a(x) f_a \\ \nabla^2 f(x) &\approx \sum_{a=1}^N \nabla^2 s_a(x) f_a. \end{aligned} \tag{7.1}$$

Además, no estamos por lo general interesados en cualquier punto espacial x ,

sino en el que corresponda a una partícula b . Por ejemplo,

$$\nabla_b^2 f \approx \sum_{a=1}^N \nabla^2 s_a(x_b) f_a.$$

7.2. Derivadas de las funciones base

Para las primeras derivadas de (7.1), y para aproximar las segundas en un método de Galerkin, es necesario calcular las derivadas espaciales de las funciones base, ∇s_a . Discutimos esto en primer lugar, para luego pasar a las segundas derivadas.

7.2.1. Primeras derivadas

La idea es sencilla; por claridad la exponemos para LEM en una dimensión. Igual que para calcular s_a necesitamos conocer el multiplicador λ , para calcular $\nabla s_a = s'_a$ necesitamos conocer su derivada λ' . Esto se debe a la regla de la cadena:

$$\frac{ds_a}{dx} = \frac{\partial s_a}{\partial x} + \frac{\partial s_a}{\partial \lambda} \frac{\partial \lambda}{\partial x}. \quad (7.2)$$

Las derivadas parciales de s_a con respecto a λ y a x son muy sencillas. Por otro lado, la variación de λ está ligada al hecho de que la función objetivo $\log Z$ debe permanecer siempre en un mínimo:

$$\frac{\partial \log Z}{\partial d\lambda} = 0 \quad \text{siempre.}$$

Esto implica

$$\frac{d}{dx} \frac{\partial \log Z}{\partial d\lambda} = 0.$$

Una nueva aplicación de la regla de la cadena lleva a:

$$\frac{\partial^2 \log Z}{\partial \lambda^2} \lambda' + \frac{\partial^2 \log Z}{\partial \lambda \partial x} = 0,$$

de donde puede despejarse λ' , ya que las segundas derivadas de $\log Z$ pueden calcularse de manera bastante sencilla.

La situación en SME es similar, pero como ahora aparece un nuevo multiplicador de Lagrange β , obtenemos un sistema de 2 ecuaciones con 2 incógnitas: λ' y β' . Los términos que aparecen son casi todas las derivadas segundas posibles de $\log Z$ respecto a λ , β y x .

En dimensiones superiores, aparecen subíndices correspondientes a las componentes cartesianas, pero las ecuaciones siguen siendo similares. Este caso, el más general, es el que se muestra en el Apéndice A.1. Es necesario resolver un problema de álgebra lineal, ecuaciones (A.16) y (A.17), o más esquemática, ecuación (A.18). La matriz involucra de nuevo casi todas las posibles derivadas segundas de $\log Z$,

que se indican en las ecuaciones (A.20 – A.26) (sólo falta segunda derivada espacial (A.27)). La solución del sistema proporciona todas las derivadas $d\lambda_{i,j}$ and $d\mu_{st,j}$. En 2D, estas son 12 incógnitas: 4 para las derivadas de λ y 8 para las de β (aunque, en realidad, sólo 9 son independientes gracias a las simetrías). Estas derivadas pueden finalmente combinarse para calcular ∇s_a , ver ec. (A.12), que no es más que la generalización de (7.2).

Una complejidad añadida a la hora de implementar el método es que los métodos de álgebra lineal se aplican a vectores; nuestras incógnitas son más bien tensores. Por lo tanto, en algún momento (al principio del programa, o justo antes de utilizar las rutinas de álgebra lineal) hace falta “traducir” los tensores a vectores, mediante convenios similares al de Voigt. En la práctica, nosotros hemos numerado los ejes cartesianos 0 o 1, de tal modo que un elemento de un tensor (sea del orden que sea) vendrá dado por un número en base binaria; por ejemplo, $(11)b = 3$. El primer elemento lleva asignado un índice “0”, pero eso es en todo caso estándar en C y C++.

Este procedimiento aparece en las referencias [31, 33], con pequeños cambios de notación (sobre todo con la primera referencia, más antigua).

En resumen, el procedimiento es:

- Minimizar $\log Z$ para calcular λ y β .
- Calcular las segundas derivadas de $\log Z$, ecuaciones (A.20 – A.26).
- Utilizarlas para montar la matriz y el vector independiente en el problema de álgebra lineal (A.18).
- Resolver el problema, por ejemplo mediante descomposición LU. Algunas ecuaciones son redundantes, por ejemplo la correspondiente a μ_{xy} es igual que la de μ_{yx} . Estas ecuaciones deben ser eliminadas con cuidado, ya que si no muchos algoritmos devuelven un error, al ser la matriz singular. Hemos utilizado el método LU que está implementado en el proyecto de bibliotecas científicas GNU, GSL [34].
- Las derivadas se pueden calcular de (A.14).

7.2.2. Segundas derivadas

Se pueden obtener las segundas derivadas volviendo a aplicar el razonamiento de las primeras. De nuevo, comenzamos con LME en una dimensión, y en el Anexo A.1 se puede encontrar el caso más general: SME en varias dimensiones.

La regla de la cadena indica que ahora necesitaremos la segunda derivada de λ , λ'' :

$$\frac{d^2 s_a}{dx^2} = \frac{\partial^2 s_a}{\partial x^2} + 2 \frac{\partial^2 s_a}{\partial \lambda \partial x} \lambda' + \frac{\partial^2 s_a}{\partial \lambda^2} (\lambda')^2 + \frac{\partial s_a}{\partial \lambda} \lambda'' = 0. \quad (7.3)$$

De nuevo, la función objetivo $\log Z$ debe permanecer en un mínimo, y esto implica que, también para la segunda derivada,

$$\frac{d^2}{dx^2} \frac{\partial \log Z}{\partial d\lambda} = 0.$$

La aplicación de la regla de la cadena lleva a:

$$\frac{\partial^2 \log Z}{\partial \lambda^2} \lambda'' + \frac{\partial^3 \log Z}{\partial \lambda^3} (\lambda')^2 + 2 \frac{\partial^3 \log Z}{\partial \lambda^2 \partial x} \lambda' + \frac{\partial^3 \log Z}{\partial \lambda \partial^2 x} = 0. \quad (7.4)$$

Las terceras derivadas de $\log Z$ pueden calcularse así que, con λ' conocida de la sección anterior, puede despejarse λ'' .

Este sencillo esquema se convierte en farragoso en SME, debido a la aparición de β (cuya segunda derivada es también necesaria). Por último, en dimensiones superiores el sistema final de ecuaciones es bastante impresionante; además, está plagado de sutilezas y su implementación es bastante ardua. Sin embargo, su ejecución numérica es bastante trivial. Resumimos los pasos necesarios:

- Resolver el problema de la sección anterior (primeras derivadas).
- Calcular las terceras derivadas de $\log Z$, ecuaciones (A.34) a (A.43).
- El resultado es un problema de álgebra lineal cuya matriz es la misma que en la sección anterior. Esto puede comprobarse en el caso sencillo de (7.4): el coeficiente de λ'' ya estaba calculado de antes. Desde el punto de vista numérico esto es crucial, ya que se puede emplear la misma descomposición (LU, por ejemplo) de la sección anterior. Lo que cambia es el vector independiente.
- Hallar la solución del problema de álgebra lineal por columnas¹, para obtener las segundas derivadas espaciales de λ y β .
- Finalmente, calcular el tensor de segundas derivadas de la Ec. (A.29).

7.2.3. Detalles técnicos

A la hora de aplicar estos métodos a una simulación de partículas hay que afrontar una serie de problemas técnicos.

El principal es que, a diferencia de las distribuciones de nodos consideradas en las Refs. [31, 33], aplicadas a problemas de elasticidad, las configuraciones de las partículas en una simulación pueden ser bastante desordenadas. Por ello es preciso determinar adecuadamente el valor de g . Como se ve en la figura 7.1, en un sistema con dos zonas de distinto valor de h un cálculo en el que no se ha tenido en cuenta este hecho produce unos resultados poco aceptables: la partícula señalada en rojo a la izquierda tiene muchos vecinos, pero la de la derecha tiene pocos.

¹Las segundas derivadas son tensores de orden tres, que pueden considerarse matrices cuyas columnas son tensores de orden dos “traducidos” como hemos explicado anteriormente.

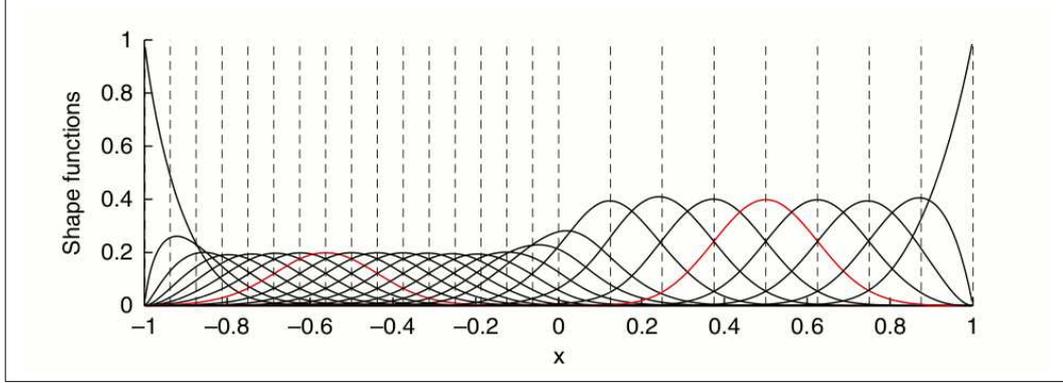


Figura 7.1: Funciones base SME para un sistema con dos valores de h muy distintos. De [33].

Para ser más precisos, es h lo relevante; en una implementación numérica los exponentes f_a de la Ec. 6.10 se evaluarían así:

$$f_a = (\lambda h_a) \frac{x - x_a}{h_a} - (\beta h_a)^2 \left[\frac{(x - x_a)^2}{h_a^2} - \frac{g}{h_a^2} \right], \quad (7.5)$$

de modo que todas las distancia se dividen por el valor de h_a local, y el valor de $g^* = g/h_a^2$, que es adimensional, está fijo para todo el sistema. Es decir, el valor de g en cada punto es $g_a = g^* h_a^2$: más pequeño si las partículas están juntas y más grande si están separadas. Esta elección solucionaría la situación de la figura 7.1.

En una dimensión g^* puede tomarse alrededor de 0,25, como se sugiere en [33]; en 2D, puede ser sencillamente $0,25I$, proporcional al tensor identidad. Una posibilidad, mencionada en [33] y examinada por nosotros (con poco éxito, de momento), es considerar tensores más complejos, que se adapten a la anisotropía local.

Así pues, una etapa previa en cada paso temporal de la simulación es la determinación de un valor de h local. Una opción ingenua sería tomar el valor del vecino más próximo. Esta idea hay que descartarla, ya que en general los otros vecinos pueden ser mucho más lejanos, y h debe ser lo bastante grande para englobar un número “suficiente” de vecinos.

De hecho, en una dimensión un criterio que da mejores resultados es tomar la distancia del primer vecino (a la derecha o izquierda) que esté más alejado. Esto se generalizaría en 2D de manera elegante: tomar h como la distancia al vecino de Delaunay más lejano; de esta manera, los más cercanos también quedan englobados. El parentesco entre las funciones base y los elementos finitos sobre la triangulación de Delaunay (aunque haya quedado más difuminado que para el método LME) hace que esta elección sea natural. Hemos comprobado que este criterio da buenos resultados; por otro lado, esto exige el cálculo de la triangulación de Delaunay, que justamente se pretendía evitar.

Así pues, en 2D hemos utilizado un método más sencillo, que produce resultados análogos: tomar h como la distancia de la sexta partícula más cercana. Que sea la

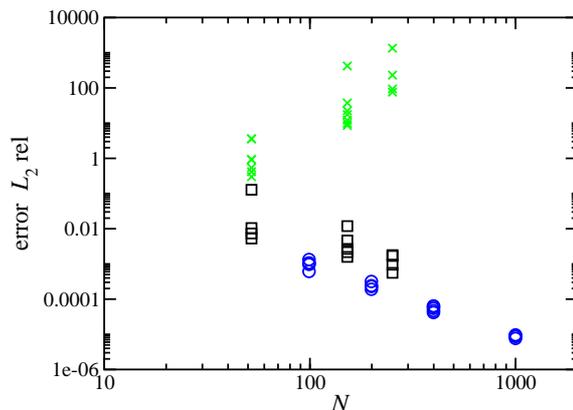


Figura 7.2: Convergencia del laplaciano aproximado por los métodos FME, LME y SME para N partículas al azar en una dimensión. Círculos azules: método FME; cuadrados negros: método SME; cruces verdes: LME.

sexta justamente está motivado porque, en promedio, una partícula en una configuración aleatoria tiene seis vecinos; en todo caso, con 5 o 7 vecinos no cambian los resultados apenas. Esta distancia se determina, en la implementación actual, de una manera poco refinada: se evalúan, para una partícula dada, las distancias a todas las demás, se ordenan y se toma el sexto valor. Al menos, la lista de distancias puede aprovecharse para almacenar una lista de vecinos, descartando los que estén más allá de, aproximadamente $3h$.

En cuanto al

7.3. Resultados

De manera similar a la sección 5.3, consideramos una función de prueba

$$f(x) = \frac{1}{(2\pi)^2} \cos(2\pi x),$$

tanto en 1D como en 2D.

Igual que allí, calculamos el error relativo L_2 como medida de la convergencia de los métodos, Ec. (5.3). En la figura 7.2 se representa, en escala logarítmica, este error en función del número de nodos, distribuidos al azar en una dimensión. Como se ve, el método FME converge perfectamente. No se puede decir lo mismo del LME, que fracasa de manera bastante evidente. En cambio, el SME sí converge, a pesar de ser sensible, como el LME, al desorden de cada configuración (como queda plasmado en la dispersión de los datos).

No parece, por tanto, que en una dimensión compense el esfuerzo que supone la implementación de estos métodos. Por supuesto, la gran esperanza es que en dos dimensiones cambien los resultados, ya que hemos visto que el método FEM no converge. Afortunadamente, parece que esto es así: en la figura 7.3 se muestra

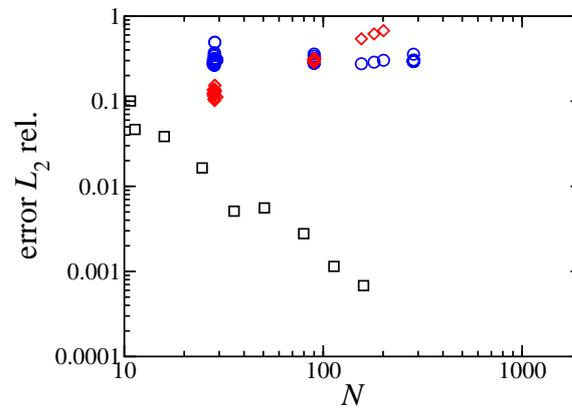


Figura 7.3: Convergencia del laplaciano aproximado por los métodos FME, LME y SME para N partículas al azar en dos dimensiones. Círculos azules: método FME; cuadrados negros: método SME; rombos rojos: SPH.

cómo, mientras el método FEM no converge en absoluto, el método SME que hemos descrito sí lo hace. Se trata, de hecho, del primer resultado positivo en nuestra búsqueda de métodos de partículas. En esta gráfica se han excluido los resultados del método LME (que no converge) y, para completar la discusión, se han introducido resultados SPH. Estos no convergen, ya que el alcance del kernel se varía como $h = 1/\sqrt{N}$, de modo que el número de vecinos sea, en promedio, siempre el mismo.

CAPÍTULO 8:

TRABAJO FUTURO

En este capítulo exponemos algunas líneas de investigación que prolongarían los resultados presentados en los capítulos anteriores.

8.1. Aplicar lo desarrollado

Se puede observar en este Proyecto una búsqueda continua de métodos que vayan más allá de las técnicas habituales de partículas, en particular SPH. Sin embargo, por el camino se han ido hallando métodos que, si bien imperfectos, ya contienen mejoras considerables.

Por ejemplo, una simulación de Voronoi con el término del laplaciano tomado del método FEM, como se discute en la sección 5.3, puede ser ya comparable, o superior, a otras técnicas. En realidad, no se ha explorado suficientemente este tipo de simulaciones.

A modo de ejemplo, en la Figura 8.1 se exponen instantáneas de una simulación de este tipo, en un estado todavía muy rudimentario. Se trata de un problema típico de rotura de presa (“dam break”) en el cual un sistema en 2D de partículas equilibradas experimenta la súbita retirada de una de las paredes laterales. A pesar del poco desarrollo que ha tenido este trabajo, los resultados presentan algunas características razonables, como que se satisfaga adecuadamente la ley barométrica de presiones en el estado inicial.

Quizá este tipo de trabajo sea más rutinario y menos excitante que desarrollar nuevos métodos, pero también es más completo al tener que considerar toda una serie de problemas técnicos, como los que se mencionan al final de la sección 4.3.2 sobre SPH. Puede por ello ser idóneo para proyectos de máster y de doctorado.

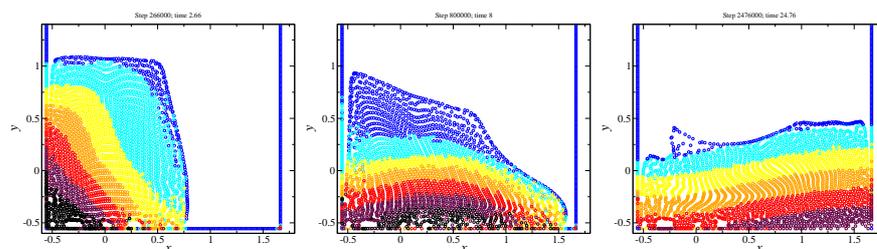


Figura 8.1: Tres etapas de un problema de rotura de presa. Los colores indican presión.

8.2. Base física para los métodos de máxima entropía

No hemos hecho énfasis apenas en una serie de requisitos físicos que una teoría de partículas debería verificar. Por un lado está la conservación de diversas magnitudes físicas en la simulación de la parte sin viscosidad (parte denominada a veces “conservada”) de las ecuaciones de movimiento para las partículas:

Momento lineal. Las ecuaciones deben ser invariantes bajo una traslación espacial arbitraria.

Momento angular. También deben serlo para una rotación arbitraria.

Energía. Lo mismo para una traslación temporal. La conservación de la energía está presente en la derivación de la expresión para el gradiente de la Sección 4.3.1 sobre las simulaciones de partículas en general. Así pues, las fuerzas de presión obtenidas de este modo cumplen automáticamente esta conservación (siempre que el método numérico de integración temporal sea adecuado en este sentido, como sucede para los llamados simplécticos).

El hecho de que las invarianzas conlleven la conservación de magnitudes físicas se debe al conocido teorema de Noether. Referimos al lector a la Ref. [14] para más información sobre propiedades de conservación. Nos limitamos a resumir que la formulación de Voronoi considerada por este grupo (descrita en la sección 5), así como la habitual SPH (sección 4.3.2), en menor grado.

Por otro lado, la parte viscosa (o “no conservada”) debe satisfacer, además:

Segunda ley. El laplaciano debe ser siempre semi-definido negativo.

Esta propiedad garantiza que el término viscoso siempre conlleve un frenado del sistema. Su denominación se debe a la Segunda Ley de la Termodinámica; no hemos considerado las ecuaciones hidrodinámicas de magnitudes termodinámicas, pero en su forma más sencilla la ley de Fourier para la conducción térmica (la “ecuación del calor”) es:

$$\nabla^2 T = \kappa \frac{dT}{dt},$$

donde T es la temperatura y κ el coeficiente de conducción térmica. Puede verse que si el laplaciano no es siempre semi-definido negativo, la temperatura puede aumentar en los sitios calientes y disminuir en los fríos, violando así la Segunda Ley. De igual manera, la ecuación de difusión de Fick también requiere que la concentración de un soluto vaya de los sitios donde hay más a donde hay menos.

Por desgracia, el método SME en la implementación “directa” 7.1 no satisface esta propiedad, como hemos verificado numéricamente. Tampoco satisface, como es obvio, la reciprocidad de acción y reacción, ya que, en general:

$$\nabla^2 s_a(x_b) \neq \nabla^2 s_b(x_a).$$

En cambio, una formulación como en los elementos finitos sí la satisface:

$$\int s_a \nabla^2 s_b = \int s_b \nabla^2 s_a = - \int \nabla s_a \nabla s_b;$$

las dos primeras expresiones son iguales porque al integrar por partes se llega a una expresión manifiestamente simétrica (siempre que se puede integrar con partes; puede que las condiciones de contorno compliquen este argumento). También es sencillo demostrar que la expresión es semi-definida negativa. Esto significa que el laplaciano podría introducirse a la manera de elementos finitos con las funciones base SME, que es lo que se hace en el artículo original [33]. Sin embargo, nos gustaría evitar esta evaluación por los motivos enumerados en 7.1.

También hay que mencionar la importancia del programa GENERIC en el desarrollo de estas ideas, comenzando con el artículo seminal Ref. [1]. Este programa establece matemáticamente un marco de teorías que cumplen las propiedades mencionadas, y además permite enlazar correctamente con el nivel mesoscópico, en el cual están presentes fluctuaciones brownianas. Es importante tratar que la teoría resultante se adapte a esta estructura en lo posible.

8.3. Ajuste por gaussianas

Aunque es, en principio, imposible determinar la forma que van a tener las funciones base en los métodos LME y SME, es cierto que, por lo general siguen siendo “campanas”. Como no tienen soporte compacto es por lo tanto tentador aproximarlas por gaussianas. Por supuesto, se perderían *todas* las propiedades de consistencia de las funciones SME (incluyendo la de orden cero), pero si en la práctica se consigue a cambio convergencia a orden cero, primero y segundo, esto justificaría el procedimiento.

El ajuste sería sencillo, ya que conocemos el valor de la función de forma s_a en el nodo a , así como sus primeras y segundas derivadas. La Ecuación (A.14) para $r = r_a$ (es decir, $x = 0$) se reduce a:

$$\left. \frac{ds_a}{dx_i} \right|_a = -s_a d\mu_{st,i} g_{st}$$

(es decir, las funciones de forma SME *no* tienen un máximo en su nodo correspondiente, aunque las LME sí); la Ecuación (A.29) pasa a ser

$$\begin{aligned} \left. \frac{d^2 s_a}{dx_i dx_k} \right|_a &= \frac{ds_{a,i} ds_{a,k}}{s_a} + s_a \{ \\ &\quad - d^2 \mu_{st,ik} g_{st} - (2\mu_{il} + d\lambda_{l,i})(2\mu_{kj} + d\lambda_{j,k}) g_{jl} + \\ &\quad g_{mn} g_{st} d\mu_{mn,i} d\mu_{st,k} \cdot \} \end{aligned}$$

Estas derivadas nos darían los parámetros que necesitamos para nuestras gaussianas que, como puede verse, serían anisotrópicas:

$$G(r) = A \exp[-\lambda \cdot r - r^t \cdot \sigma \cdot r],$$

donde λ es un vector en d dimensiones y σ un tensor simétrico en $d \times d$ dimensiones.

Una vez determinadas las gaussianas aproximadas, todas las integrales que aparecen en un tratamiento de Galerkin (con masa acumulada o no) se evalúan muy fácilmente. Por lo que sabemos, no se han planteado nunca métodos de Galerkin basados en funciones gaussianas. Este método es bastante sencillo de analizar; evitamos dar aquí los detalles, pero el resultado final resulta ser casi idéntico a la expresión SPH directa, ec. (5.4), que justamente se evita por sus malas propiedades.

8.4. Otros métodos de máxima entropía

El método SME que se ha aplicado a partículas está adaptado del original. La teoría general tras los métodos de máxima entropía es bastante genérica, así que pueden desarrollarse métodos alternativos basados en ella. Un obstáculo básico en este sentido es a la hora de definir ligaduras. Como se ha discutido, las ligaduras son de tipo local, para forzar consistencia de orden cero, primer orden y (en la medida de lo posible) segundo orden. Si se intentan definir propiedades no locales se producía lo que desde el punto de vista de la mecánica estadística sería un “acoplamiento” entre los valores de las funciones peso en distintos puntos. Este problema sería mucho más difícil de abordar, y exigiría métodos de muestreo por importancia, como el algoritmo de Monte Carlo de Metropolis.

8.5. Una formulación genérica

Una idea reciente de Pep Español es referente a que, por lo general, se suele introducir un juego de funciones peso que se emplea tanto para reconstruir funciones a partir de números aislados (el uso general de ellas en el presente Proyecto) como para asignar valores a partir de una función continua. La más conocida opción para el segundo caso es la función delta de Dirac, que equivale a, simplemente, tomar los valores de f_a en un punto con coordenadas x_a como $f(x_a)$:

$$f_a \int f(x) \delta(x - x_a) = f(x_a).$$

Sin embargo, se puede considerar un juego distinto a las delta de Dirac, y distinto también al juego de funciones que se utiliza para la reconstrucción. La exigencia de que estos dos juegos cumplan condiciones básicas se traduce en una serie de requisitos.

Es muy interesante comprobar que, con ciertos criterios sencillos, se llega de este modo a la fórmula de elementos finitos para el laplaciano *en su modalidad de matriz de masa completa* (es decir, sin “lumped mass”, aproximación de masa acumulada). Si esta versión de la teoría convergiera en 2D este hecho sería de enorme relevancia; por desgracia, como ya discutimos en la Sección 5.3.2 esto no es así. Sin embargo, este punto de vista merece ser explorado en más detalle.

REFERENCIAS

- [1] Mar Serrano and Pep Español. Thermodynamically consistent mesoscopic fluid particle model. *Physical Review E*, 64(4):046115, Sep 2001. 4, 47, 79
- [2] H A Diels. *Die Fragmente der Vorsokratiker*. Weidmann, Berlín, Alemania, 2 edition, 1903. 5
- [3] L. D. Landau and E. M. Lifshitz. *Fluid Mechanics, Second Edition: Volume 6 (Course of Theoretical Physics)*. Course of theoretical physics / by L. D. Landau and E. M. Lifshitz, Vol. 6. Butterworth-Heinemann, 2 edition, 1987. 6, 8
- [4] Frank M. White. *Viscous fluid flow*. McGraw-Hill Higher Education, 2006. 6, 8, 18, 19
- [5] J J Monaghan. Smoothed particle hydrodynamics. *Reports on Progress in Physics*, 68(8):1703, 2005. 9, 38, 43, 57
- [6] Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, and Sung Nok Chiu. *Spatial tessellations: Concepts and applications of Voronoi diagrams*. Probability and Statistics. Wiley, Nueva York, NY, EEUU, 2 edition, 2000. 21
- [7] CGAL. CGAL web site, 2011. <http://www.cgal.org>. 25
- [8] B. Stroustrup. *The C++ programming language*. Addison-Wesley, 2000. 26
- [9] D. Yang. *C++ and object-oriented numeric computing for scientists and engineers*. Springer, 2001. 26
- [10] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical recipes in C (2nd ed.): the art of scientific computing*. Cambridge University Press, Nueva York, NY, EEUU, 1992. 34
- [11] K Eriksson, D Estep, P Hansbo, and C Johnson. *Computational Differential Equations*. Studentlitteratur, Lund, Suecia, 1996. 37
- [12] Max Wardetzky, Saurabh Mathur, Felix Kälberer, and Eitan Grinspun. Discrete laplace operators: no free lunch. In *Proceedings of the fifth Eurographics symposium on Geometry processing*, pages 33–37, Aire-la-Ville, Suiza, 2007. Eurographics Association. 38
- [13] Max Wardetzky. Convergence of the cotangent formula: An overview. In A I Bobenko, John M Sullivan, Peter Schröder, and Günter Ziegler, editors, *Discrete Differential Geometry*. Birkhäuser, Basilea, Alemania, 2008. 38

- [14] Pep Español and Mar Serrano. Voronoi fluid particles & tessellation fluid dynamics. In Rien van de Weijgaert, Gert Vegter, Jelle Ritzerveld, and Vincent Icke, editors, *Tessellations in the Sciences Virtues, Techniques and Applications of Geometric Tilings*. Kluwer/Springer, 2009. 39, 42, 47, 48, 50, 78
- [15] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal*, 82:1013–1024, 1977. 42
- [16] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. *Mon. Not. Roy. Astron. Soc.*, 181:375–389, 1977. 42
- [17] J. J. Monaghan. Smoothed particle hydrodynamics. *Annual Review of Astronomy and Astrophysics*, 30:543–574, 1992. 43
- [18] D. Duque and P. Español. Discrete differential operators for Voronoi particle dynamics. *Proceeding of the 5th SPHERIC Workshop*, 2010. 44, 47, 58
- [19] D. Duque and P. Español. A maximum entropy method for particle simulations. In Thomas Rung and Christian Ulrich, editors, *6th International SPHERIC Workshop*. Schriftenreihe Schiffbau, Hamburgo, Alemania, 2011. 44
- [20] Volker Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. *Monthly Notices of the Royal Astronomical Society*, 401(2):791–851, 2010. 45
- [21] M Serrano, P Español, and I Zúñiga. Voronoi fluid particle model for Euler equations. *JOURNAL OF STATISTICAL PHYSICS*, 121(1-2):133–147, OCT 2005. 1st International Conference for Mesoscopic Methods in Engineering and Science (ICMMES), Braunschweig, GERMANY, JUL 25, 2004-JUL 30, 2005. 47, 49
- [22] S. Hess and V. Springel. Particle hydrodynamics with tessellation techniques. *arXiv:0912.0629 [astro-ph]*, April 2010. 47
- [23] M. Serrano. Comparison between smoothed dissipative particle dynamics and Voronoi fluid particle model in a shear stationary flow. *Physica A: Statistical Mechanics and its Applications*, 362(1):204–209, March 2006. 48
- [24] Pep Español and Ignacio Zúñiga. On the definition of discrete hydrodynamic variables. *The Journal of Chemical Physics*, 131(16):164106+, 2009. 49
- [25] Pep Español, Jesus G. Anero, and Ignacio Zúñiga. Microscopic derivation of discrete hydrodynamics. *JOURNAL OF CHEMICAL PHYSICS*, 131(24), DEC 28 2009. 49
- [26] GiD. GiD 10 Reference Manual, 2011. 53

-
- [27] Mikhail Belkin, Jian Sun, and Yusu Wang. Discrete laplace operator on meshed surfaces. In *Proceedings of the twenty-fourth annual symposium on Computational geometry*, SCG '08, pages 278–287, New York, NY, USA, 2008. ACM. 55, 58
- [28] Qiang Du, Vance Faber, and Max Gunzburger. Centroidal voronoi tessellations: Applications and algorithms. *SIAM Review*, 41(4):637+, 1999. 58
- [29] R. A. Nicolaides. Direct discretization of planar Div-Curl problems. *SIAM Journal on Numerical Analysis*, 29(1):32+, 1992. 58
- [30] Kerson Huang. *Statistical mechanics / Kerson Huang*. Wiley, 2nd ed. edition, 1987. 61
- [31] M. Arroyo and M. Ortiz. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering*, 65(13):2167–2202, 2006. 63, 64, 65, 71, 72
- [32] V T Rajan. Optimality of the delaunay triangulation in R^d . *Discrete and Computational Geometry*, 12(2):189–202, 1994. 64
- [33] C J Cyron, M Arroyo, and M Ortiz. Smooth, second order, non-negative mesh-free approximants selected by maximum entropy. *International Journal for Numerical Methods in Engineering*, 79(13):1605–1632, 2009. 65, 68, 71, 72, 73, 79, 89, 93
- [34] Mark Galassi, Jim Davies, James Theiler, Brian Gough, Gerard Jungman, Michael Booth, and Fabrice Rossi. *Gnu Scientific Library: Reference Manual*. Network Theory Ltd., 3 edition, 2003. 71, 90

ÍNDICE DE FIGURAS

3.1. Ejemplo de una triangulación	22
3.2. Dos posibles triangulaciones de cuatro nodos	23
3.3. Caso degenerado	23
4.1. Triangulación de Delaunay aleatoria y elemento finito	33
4.2. Los dos ángulos de la fórmula de la cotangente.	38
5.1. Magnitudes geométricas en el cambio del volumen de Voronoi	48
5.2. Convergencia de grad. y div. con Voronoi	50
5.3. Laplaciano aproximado para partículas equiespaciadas, 1D	51
5.4. Laplaciano aproximado para partículas al azar, 1D	52
5.5. Convergencia del laplaciano aproximado por el método FEM.	52
5.6. Convergencia del problema de Poisson por FEM	53
5.7. Partículas al azar sobre el cuadrado unidad.	54
5.8. Aproximación de elementos finitos al laplaciano en 2D	55
5.9. Convergencia del laplaciano por FEM en 2D	56
5.10. Partículas sobre el cuadrado unidad generadas por por GiD	56
6.1. Funciones base LME en dos dimensiones	63
6.2. Método LME para una configuración degenerada	64
6.3. Dibujo de la función	66
6.4. Dibujo de la función	67
7.1. Funciones base SME para un sistema con dos valores de h	73
7.2. Convergencia del laplaciano por FEM, LME y SME en 1D.	74
7.3. Convergencia del laplaciano por FEM, LME y SME en 2D.	75
8.1. Etapas de un problema de rotura de presa	77

ÍNDICE DE CUADROS

3.1. Ejemplos de estilos de programación	27
3.2. Cálculo de Voronoi mediante las rutinas CGAL	30

APÉNDICE A:

DERIVADAS DE LAS FUNCIONES BASE

A.1. Funciones de forma

Nos centraremos en el valor de la función de forma s_a debida al punto (“nodo” o “partícula”) a , situado en r_a en el espacio R^d , en otro punto r . El vector relativo de posición es $x_a = r_a - r$; por simplicidad, omitiremos en muchas ocasiones el subíndice, salvo cuando sea necesario especificarlo, y escribiremos tan solo “ x ”.

Definimos

$$f_a = \lambda_i x_i + \mu_{ij}(x_i x_j - g_{ij}). \quad (\text{A.1})$$

En esta expresión, λ es un vector de dimensión d , μ y g son tensores de grado 2 de dimensión $d \times d$, y se sobreentiende el convenio de suma de Einstein. Mantenemos el nombre de μ para hacer nuestra notación similar a la de [33], pero en el resto de esta obra se utiliza $\beta = -\mu$.

En 1D esto significa simplemente

$$f_a = \lambda x + \beta(x^2 - g), \quad (\text{A.2})$$

y en 2D,

$$f_a = \lambda_x x_x + \lambda_y x_y + \beta_{xx}(x_x^2 - g_{xx}) + \beta_{yy}(x_y^2 - g_{yy}) + 2\beta_{xy}(x_x x_y - g_{xy}). \quad (\text{A.3})$$

En el último término anticipamos el hecho de que β debe ser un tensor simétrico.

La función de forma s_a tiene el valor

$$s_a = \frac{1}{Z} \exp(f_a), \quad (\text{A.4})$$

donde Z es denominada “función de partición” y es:

$$Z = \sum_{a=1}^N f_a. \quad (\text{A.5})$$

Nuestra tarea es resolver el problema

$$\min_{\lambda, \mu} \log Z. \quad (\text{A.6})$$

Observamos que la derivada de $\log Z$ con respecto a cualquier variable ξ puede escribirse, gracias a la regla de la cadena:

$$\frac{\partial \log Z}{\partial \xi} = \sum_a s_a \frac{\partial f_a}{\partial \xi}. \quad (\text{A.7})$$

Si probamos $\xi = \lambda_i$ obtenemos

$$\frac{\partial \log Z}{\partial \lambda_i} = \sum_a s_a x_i \stackrel{*}{=} 0, \quad (\text{A.8})$$

donde el asterisco sobre el signo $=$ significa a partir de ahora que la igualdad sólo rige en los valores para los cuales $\log Z$ es mínimo; es decir, se satisface la Ec. (A.6). En este caso, la igualdad es justamente la condición de mínimo, que implica consistencia a primer orden; este es justamente el motivo por el cuál se introdujeron los multiplicadores de Lagrange λ .

Similarmente, si $\xi = \mu_{st}$ obtenemos

$$\frac{\partial \log Z}{\partial \mu_{st}} = \sum_a s_a (x_s x_t - g_{st}) \stackrel{*}{=} 0, \quad (\text{A.9})$$

Para completar todas las posibilidades, si $\xi = x_i$ obtenemos

$$\frac{\partial \log Z}{\partial x_i} = \sum_a s_a (\lambda_i + \mu_{ik} x_k) \stackrel{*}{=} \lambda_i, \quad (\text{A.10})$$

que no tiene por qué ser cero en el mínimo en general.

La minimización se lleva a cabo numéricamente mediante, por ejemplo, gradientes conjugados (ya que las fórmulas anteriores proporcionan el gradiente de la función que hay que minimizar); no es necesario ningún cálculo adicional para el valor de las funciones de forma s_a . Hemos empleado para estos métodos numéricos estándar las bibliotecas GSL (Gnu Scientific Libraries) en lenguaje C [34].

A.2. Primeras derivadas

Ahora deseamos calcular

$$\frac{ds_a}{dx_i} = \nabla_i s_a = ds_{a,i}, \quad (\text{A.11})$$

donde la notación “ \cdot_i ” significa el componente i -ésimo de una derivada espacial (aun así, mantenemos la letra “ d ” for claridad).

Por la regla de la cadena,

$$s_{a,i} = \frac{\partial s_a}{\partial x_i} + \frac{\partial s_a}{\partial \lambda_j} d\lambda_{j,i} + \frac{\partial s_a}{\partial \mu_{st}} d\mu_{st,i}. \quad (\text{A.12})$$

La primera derivada es fácil de obtener:

$$ds_{a,i} \stackrel{*}{=} 2\mu_{ij}x_j, \quad (\text{A.13})$$

y podemos reducir la Ec. (A.12) a:

$$ds_{a,i} = s_a[2\mu_{ij}x_j + d\lambda_{j,i}x_j + d\mu_{st,i}(x_sx_t - g_{st}).] \quad (\text{A.14})$$

Para los otros dos términos, dado que la Ec. (A.8) debe satisfacerse en todo punto, tenemos

$$\frac{d}{dx_i} \frac{\partial \log Z}{\partial \lambda_k} = 0. \quad (\text{A.15})$$

La regla de la cadena entonces lleva a

$$\begin{aligned} \frac{d}{dx_i} \frac{\partial \log Z}{\partial \lambda_k} &= \frac{\partial^2 \log Z}{\partial \lambda_k \partial x_i} \\ &+ \frac{\partial^2 \log Z}{\partial \lambda_k \partial \lambda_j} \frac{\partial \lambda_j}{\partial x_i} + \frac{\partial^2 \log Z}{\partial \lambda_k \partial \mu_{st}} \frac{\partial \mu_{st}}{\partial x_i} = 0, \end{aligned} \quad (\text{A.16})$$

y de la Ec. (A.9),

$$\begin{aligned} \frac{d}{dx_i} \frac{\partial \log Z}{\partial \mu_{st}} &= \frac{\partial^2 \log Z}{\partial \mu_{st} \partial x_i} \\ &+ \frac{\partial^2 \log Z}{\partial \mu_{st} \partial \lambda_j} \frac{\partial \lambda_j}{\partial x_i} + \frac{\partial^2 \log Z}{\partial \mu_{st} \partial \mu_{uv}} \frac{\partial \mu_{uv}}{\partial x_i} = 0. \end{aligned} \quad (\text{A.17})$$

Estas dos ecuaciones tienen la forma de un sistema de ecuaciones lineales que hay que resolver para $d\lambda_{i,j}$ and $d\mu_{st,j}$. En forma matricial esquemática,

$$\begin{pmatrix} D^2 \log Z_{\lambda\lambda} & D^2 \log Z_{\lambda\mu} \\ D^2 \log Z_{\lambda\mu} & D^2 \log Z_{\mu\mu} \end{pmatrix} \begin{pmatrix} d\lambda \\ d\mu \end{pmatrix} + \begin{pmatrix} D^2 \log Z_{\lambda x} \\ D^2 \log Z_{\mu x} \end{pmatrix} = \vec{0}. \quad (\text{A.18})$$

Ahora necesitamos evaluar todas las derivadas de segundo orden de $\log Z$ que aparecen en estas expresiones. Para obtenerlas sistemáticamente observamos que, derivando de nuevo la Ec. (A.7):

$$\frac{\partial^2 \log Z}{\partial \xi \partial \phi} = \sum_a s_a \left[\left(\frac{\partial f_a}{\partial \phi} - \frac{\partial \log Z}{\partial \phi} \right) \frac{\partial f_a}{\partial \xi} + \frac{\partial^2 f_a}{\partial \xi \partial \phi} \right] \quad (\text{A.19})$$

Así pues obtenemos las siguientes expresiones.

$$\frac{\partial^2 \log Z}{\partial \lambda_i \partial x_j} \stackrel{*}{=} 2g_{ik}\mu_{jk} + \delta_{ij} \quad (\text{A.20})$$

$$\frac{\partial^2 \log Z}{\partial \lambda_i \partial \lambda_j} \stackrel{*}{=} g_{ij} \quad (\text{A.21})$$

(obsérvese la necesidad de que g sea simétrico).

$$\frac{\partial^2 \log Z}{\partial \lambda_i \partial \mu_{st}} \stackrel{*}{=} (j3)_{ist}, \quad (\text{A.22})$$

donde $j3$ es un tensor de rango 3 de componentes

$$(j3)_{ist} = \sum_a s_a x_i x_s x_t. \quad (\text{A.23})$$

En general, utilizaremos tensores de rango n

$$(jn)_{i\dots} = \sum_a s_a x_i \cdots, \quad (\text{A.24})$$

hasta $n = 6$. Nótese que $j2 \stackrel{*}{=} g$.

$$\frac{\partial^2 \log Z}{\partial x_i \partial \mu_{st}} \stackrel{*}{=} 2\mu_{ik} (j3)_{kst} \quad (\text{A.25})$$

$$\frac{\partial^2 \log Z}{\partial \mu_{st} \partial \mu_{uv}} \stackrel{*}{=} (j4)_{stuv} - g_{st} - g_{uv} \quad (\text{A.26})$$

$$\frac{\partial^2 \log Z}{\partial x_i \partial x_j} \stackrel{*}{=} 4\mu_{il} \mu_{jm} g_{lm} + 2g_{ij} \quad (\text{A.27})$$

El procedimiento es, por tanto:

- Resolver el problema de la sección anterior.
- Calcular segundas derivadas, Ecs. de (A.20) a (A.27). (Un cálculo previo de $j3$ y $j4$ es recomendable).
- Usar estas para construir la matriz y el vector independiente del problema de álgebra lineal de la Ec. (A.18).
- Resolver el problema lineal. El sistema debe reducirse antes, ya que algunas ecuaciones son redundantes. Por ejemplo, la ecuación para μ_{xy} es la misma que para μ_{yx} ; esto puede llevar a errores con algunos códigos numéricos. Hemos empleado la decompocisión LU estándar, según está implementada en las librerías GSL.
- Calcular las derivadas espaciales deseadas mediante la Ec. (A.14).

A.3. Segundas derivadas

Los resultados anteriores pueden encontrarse, con menos detalles, en la Ref. [33]. Lo que sigue representa material original.

Ahora deseamos calcular

$$\frac{d^2 s_a}{dx_i dx_k} = d^2 s_{a,ik}. \quad (\text{A.28})$$

Por la regla de la cadena, y tras algunos pasos intermedios,

$$\begin{aligned} d^2 s_{a,ik} = & \frac{ds_{a,i} ds_{a,k}}{s_a} + \\ & s_a \{ \\ & (2d\mu_{ij,k} + 2d\mu_{kj,i} d^2 \lambda_{j,ik}) x_j + d^2 \mu_{st,ik} (x_s x_t - g_{st}) - \\ & (2\mu_{il} + d\lambda_{l,i})(2\mu_{kj} + d\lambda_{j,k}) g_{jl} - \\ & (2\mu_{il} + d\lambda_{l,i}) d\mu_{st,k} x_s x_t x_l - (2\mu_{kj} + d\lambda_{j,k}) d\mu_{st,i} x_s x_t x_j - \\ & (x_m x_n x_s x_t - g_{mn} g_{st}) d\mu_{mn,i} d\mu_{st,k}. \\ & \} \end{aligned} \quad (\text{A.29})$$

Todas las expresiones que aparecen se han discutido en las secciones anteriores, salvo las segundas derivadas espaciales $d^2 \lambda$ and $d^2 \mu$. Ahora, debemos imponer que la segunda derivada espacial de la Ec. (A.8) siga siendo nula:

$$\frac{d^2}{dx_l dx_i} \frac{\partial \log Z}{\partial \lambda_k} = 0. \quad (\text{A.30})$$

De este modo, llegamos a

$$\begin{aligned} & (\partial^2 \log Z / \partial \lambda^2)_{kj} (d^2 \lambda)_{j,il} + (\partial^2 \log Z / \partial \lambda \partial \mu)_{kst} (d^2 \mu)_{st,il} + \\ & (\partial^3 \log Z / \partial \lambda^3)_{kjs} (d\lambda)_{j,i} (d\lambda)_{s,l} + (\partial^3 \log Z / \partial \lambda^2 \partial \mu)_{kfst} (d\lambda)_{j,i} (d\mu)_{st,l} + \\ & (\partial^3 \log Z / \partial \lambda^2 \partial \mu)_{kfst} (d\lambda)_{j,l} (d\mu)_{st,i} + (\partial^3 \log Z / \partial \lambda \partial^2 \mu)_{kstuv} (d\mu)_{st,l} (d\mu)_{uv,i} + \\ & (\partial^3 \log Z / \partial \lambda^2 \partial x)_{kj,i} (d\lambda)_{j,l} + (\partial^3 \log Z / \partial \lambda^2 \partial x)_{kj,l} (d\lambda)_{j,i} + \\ & (\partial^3 \log Z / \partial \lambda \mu \partial x)_{kst,i} (d\mu)_{st,l} + (\partial^3 \log Z / \partial \lambda \mu \partial x)_{kst,l} (d\mu)_{st,i} + \\ & (\partial^3 \log Z / \partial \lambda \mu \partial^2 x)_{k,il} = 0. \end{aligned} \quad (\text{A.31})$$

Un resultado similar se sigue de

$$\frac{d^2}{dx_l dx_i} \frac{\partial \log Z}{\partial \lambda_{st}} = 0. \quad (\text{A.32})$$

De nuevo, llegamos a un problema de álgebra lineal que debe ser resuelto. El “vector” de soluciones es ahora una matriz: cada columna j de ella corresponde a $d^2 \lambda_{ij}$; el índice de fila i corresponde a la dirección cartesiana i .

Para calcular las nuevas derivadas de $\log Z$ que hacen falta, tenemos el resultado general

$$\begin{aligned} \frac{\partial^3 \log Z}{\partial \xi \partial \phi \partial \psi} = & \sum_a s_a \{ \\ & \left(\frac{\partial f_a}{\partial \psi} - \frac{\partial \log Z}{\partial \psi} \right) \left(\left[\frac{\partial f_a}{\partial \phi} - \frac{\partial \log Z}{\partial \phi} \right] \frac{\partial f_a}{\partial \xi} + \frac{\partial^2 f_a}{\partial \xi \partial \phi} \right) + \\ & \frac{\partial^2 f_a}{\partial \xi \partial \psi} \left[\frac{\partial f_a}{\partial \phi} - \frac{\partial \log Z}{\partial \phi} \right] + \\ & \frac{\partial f_a}{\partial \psi} \left(\frac{\partial^2 f_a}{\partial \phi \partial \psi} - \frac{\partial^2 \log Z}{\partial \phi \partial \psi} \right) + \\ & \left. \frac{\partial^3 f_a}{\partial \xi \partial \phi \partial \psi} \right\} \end{aligned} \quad (\text{A.33})$$

De este modo, obtenemos las siguientes derivadas.

$$(\partial^3 \log Z / \partial \lambda^3)_{ijk} \stackrel{*}{=} (j3)_{ijk} \quad (\text{A.34})$$

$$(\partial^3 \log Z / \partial \lambda^2 \partial \mu)_{ijst} \stackrel{*}{=} (j4)_{ijst} - g_{ij} g_{st} \quad (\text{A.35})$$

$$(\partial^3 \log Z / \partial \lambda^2 \partial x)_{ijk} \stackrel{*}{=} 2(j3)_{ijl} \mu_{kl} \quad (\text{A.36})$$

$$(\partial^3 \log Z / \partial \lambda \partial^2 \mu)_{istuv} \stackrel{*}{=} (j5)_{istuv} - (j3)_{sti} g_{uv} - (j3)_{uvi} g_{st} \quad (\text{A.37})$$

$$\begin{aligned} (\partial^3 \log Z / \partial^3 \mu)_{stuvpq} \stackrel{*}{=} & (j6)_{stuvpq} - \\ & (j4)_{stuv} g_{pq} - (j4)_{stp} g_{uv} - (j4)_{uvs} g_{st} + \\ & 2g_{st} g_{uv} g_{pq} \end{aligned} \quad (\text{A.38})$$

$$(\partial^3 \log Z / \partial \lambda^2 \partial x)_{ij,k} \stackrel{*}{=} 2(j3)_{lij} \mu_{kl} \quad (\text{A.39})$$

$$\begin{aligned} (\partial^3 \log Z / \partial \lambda \partial \mu \partial x)_{ist,k} \stackrel{*}{=} & g_{is} \delta_{tk} + g_{it} \delta_{sk} + \\ & 2\mu_{kl} [(j4)_{list} - g_{li} g_{st}] \end{aligned} \quad (\text{A.40})$$

$$\begin{aligned} (\partial^3 \log Z / \partial \mu^2 \partial x)_{stuv,k} \stackrel{*}{=} & 2\mu_{kl} [(j5)_{stuvl} - (j3)_{uvs} g_{st} - (j3)_{stl} g_{uv}] + \\ & (j3)_{uvs} \delta_{tk} + (j3)_{uvt} \delta_{sk} + (j3)_{stu} \delta_{vk} + (j3)_{stv} \delta_{uk} \end{aligned} \quad (\text{A.41})$$

$$(\partial^3 \log Z / \partial \lambda \partial^2 x)_{i,jk} \stackrel{*}{=} 4\mu_{kl} \mu_{jm} (j3)_{lmi} \quad (\text{A.42})$$

$$(\partial^3 \log Z / \partial \mu \partial^2 x)_{st,jk} \stackrel{*}{=} 4\mu_{kl}\mu_{jm}[(j4)_{lmst} - g_{lm}g_{st}] + 2\mu_{jl}[g_{sl}\delta_{tk} + g_{tl}\delta_{sk}] + 2\mu_{kl}[g_{sl}\delta_{tj} + g_{tl}\delta_{sj}] \quad (\text{A.43})$$

Ahora el procedimiento sería:

- Resolver el problema de las secciones anteriores.
- Calcular las derivadas segundas, Ecs. de (A.34) a (A.43). (Se recomienda un cálculo previo de $j4$, $j5$ y $j6$.)
- En el problema de álgebra lineal que se obtiene la matriz resulta ser la misma que en la sección anterior. Si se emplea la decomposición LU no hace falta por tanto volver a descomponer la matriz y pueden usarse las mismas matrices L y U . El “vector” independiente (una matriz, en realidad), sí cambia.
- Encontrar la solución al problema de álgebra lineal columna a columna para encontrar las segundas derivadas espaciales de λ y μ .
- Calcular las derivadas deseadas a partir de la Ec. (A.29).

Queremos recalcar que en estas expresiones suponemos que el tensor g es constante en un entorno local del punto en cuestión, así que sus derivadas espaciales no aparecen. Sin embargo, no hacemos ninguna otra suposición adicional. Por ejemplo, si g es simplemente el tensor identidad multiplicado con un escalar (que es justamente nuestra elección), muchas de las expresiones se simplifican considerablemente.