

Presentación de Avance - Proyecto SWE

ILI384: Taller de Modelos y Métodos Cuantitativos

Rodrigo Naranjo

Martín Villanueva

5 de Enero 2016

1. Descripción del Problema

El problema consiste en modelar el sistema de *Shallow Water Equations* en 1D y 2D, de tal modo que se pueda determinar la evolución del sistema, dadas las ecuaciones diferenciales que modelan el problema, las condiciones iniciales y las condiciones de borde. Las SWE corresponden a un caso particular de las ecuaciones de Navier-Stokes, que se obtienen al hacer la suposición de que el fluido es incompresible y que no existe viscosidad, aunque esta última puede relajarse para obtener un sistema más realista. En el caso general (2D) las ecuaciones pueden escribirse como a continuación

$$\begin{aligned}\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial h}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial h}{\partial y} &= 0 \\ \frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(u(h-b)) + \frac{\partial}{\partial y}(v(h-b)) &= 0\end{aligned}\tag{1}$$

donde las variables de interés (a modelar) son $h(x, y, t)$ (nivel de agua), $u(x, y, t)$ (componente de velocidad de una columna de agua en x), $v(x, y, t)$ (componente de velocidad de una columna de agua en y). Adicionalmente deben ser conocidos $b(x, y)$, que corresponde a la *batimetría* que describe la forma del fondo marino y las condiciones de borde e iniciales del sistema.

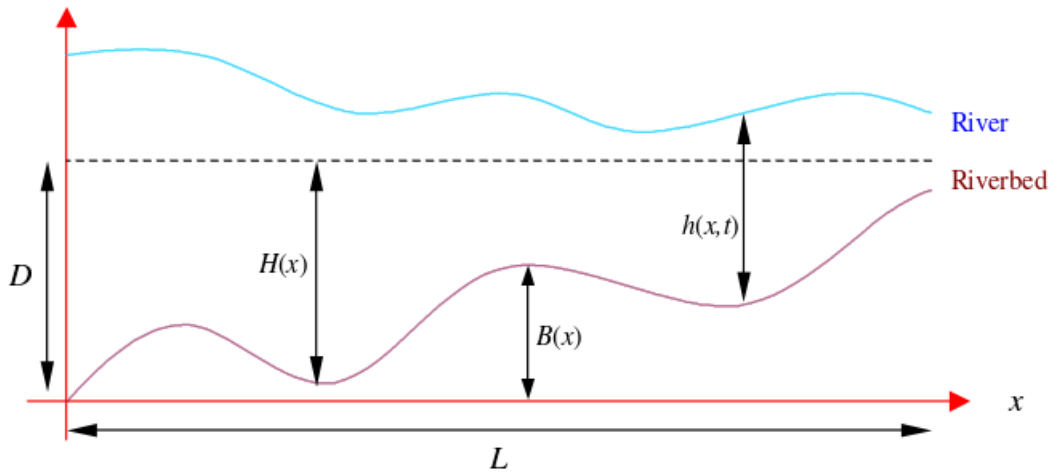


Figura 1: Esquema de SWE en una dimensión.

2. La Implementación

Para la resolución del problema se plantea un enfoque del tipo partículas, similar a *Shallow Water Particles* [1] donde la superficie del agua se modela por medio de la distribución que tome este conjunto de partículas. En el enfoque aquí propuesto, cada una de las variables de interés (h , u y v) se aproximan como una combinación lineal de funciones RBF

$$\sum_{i=1}^N \gamma_i(t) \Phi(\|\mathbf{x} - \xi_i(t)\|, \epsilon_i(t))\tag{2}$$

donde la función RBF a ocupar es de preferencia una función de soporte compacto, de modo que se acote el radio de influencia de cada partícula, y se generen de este modo sistemas *sparse*.

Lo que se quiere lograr sigue el enfoque *Evolutive RBF* como en [2], donde se obtienen ecuaciones de evolución (ODE system) para los parámetros dependientes del tiempo en cada RBF ($\gamma(t)$, $\xi_i(t)$ y $\epsilon_i(t)$). Las ventajas de esto, es que para obtener el estado del sistema en un tiempo $t + \Delta t$, requiere tan sólo computar las ecuaciones de evolución para cada RBF (de cada variable) y luego sumar los resultados respectivos en el espacio.

La implementación de la solución espera ser incremental, abordando la complejidad del problema paso por paso. En base a esto, se planea seguir el siguiente orden.

1. Inicialmente se propone resolver las *Burgers' Equations* en 1D sin viscosidad, definidas por:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0 \quad (3)$$

que es de interés pues es no lineal y se deriva de un modelo de convección-difusión, por lo tanto muy similar a las SWE.

2. El siguiente paso, consiste en resolver las SWE en 1D, a modo de obtener una idea de las propiedades y comportamiento de la solución propuesta, antes de avanzar a más dimensiones. Para este paso se pretende ocupar en primer lugar funciones RBF gaussianas, para luego dar con alguna implementación con funciones de soporte compacto.
3. Implementar la solución anterior al problema en 2D, considerando movimiento en el eje X e Y.
4. Implementar condiciones de borde, iniciales y batimetría utilizadas (por otros métodos de simulación) y probarlos con esta solución para establecer comparaciones, y verificar si se están obteniendo resultados correctos.
5. Posiblemente también será necesario agregar los términos de viscosidad al sistema, para analizar sus efectos y obtener resultados más realistas.
6. Buscar métodos para optimizar la ejecución de la solución encontrada: Paralelización/Implementación en GPU, Fast multipole method, aprovechar propiedades del sistema sparse, entre otras.

3. Dificultades

Se presentan aquí algunas de las dificultades que se conocen, y otras que eventualmente podrían aparecer en la implementación del proyecto.

- La batimetría $b(x, y)$ es, en general, una función discontinua (o que se conoce parcialmente), y por lo tanto debe hallarse una manera de aproximar sus derivadas.
- En simulaciones ocupando otros métodos (SPH *Smoothed Particle Hydrodynamics*), se ha notado que no tomar en cuenta la viscosidad, puede llevar a problemas de estabilidad numérica en las simulaciones.
- Probablemente en las simulaciones sea necesario ocupar una gran cantidad de partículas para modelar la superficie del agua, lo cual lleva a una gran cantidad de computación para determinar los estados siguientes del sistema. Por ello, quizás sea necesario paralelizar la ejecución, o buscar métodos más eficientes de computación (FMM Fast Multipole Method).

4. Burgers' Equation

4.1. Análisis Teórico

Para realizar una prueba preliminar antes de modelar las SWE, se procedió a resolver la Burgers' Equation (3), una ecuación que explica de manera simplificada el comportamiento de un fluido incompresible, por medio de las leyes de la conservación. Para realizar esto se define una solución $u(x, t)$ como una combinación lineal de RBFs, del siguiente modo

$$u(x, t) = \sum_i \phi_{h_i}(x, t) = \sum_i \gamma_i(t) \phi_{h_i}(x - \xi_i(t), \epsilon_i) \quad (4)$$

donde se utiliza $\phi(x, \epsilon) = e^{-(\epsilon x)^2}$. Siguiendo el enfoque *evolutivo* se propone a cada $\Phi(x, t)$ (se omite subíndice por simplicidad) como solución de (3), sin embargo para abordar el problema de la no linealidad, debe hacer una modificación al problema inicial. En vista de que se propone a $\Phi(x, t)$ como una solución a la PDE, es razonable aproximar $u(x, t)$ como una expansión de Taylor en torno a $(\xi(t), t)$ obteniendo

$$u(x, t) = u(\xi(t), t) + \frac{u_x(\xi(t), t)}{1!}(x - \xi(t)) + \dots \quad (5)$$

si tomamos sólo el primer término de esta expansión (de orden cero), y se reemplaza la solución propuesta $\Phi(x, t)$, entonces se obtiene

$$\frac{\partial}{\partial t} \Phi(x - \xi(t), \epsilon(t)) + u(\xi(t), t) \frac{\partial}{\partial x} \Phi(x - \xi(t), \epsilon(t)) = 0 \quad (6)$$

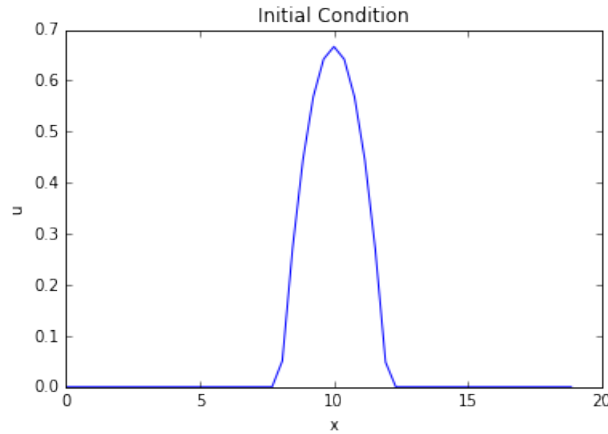
luego realizando el álgebra necesaria en (6), se obtienen las siguientes ecuaciones de evolución para los parámetros de cada RBF

$$\begin{aligned} \epsilon'(t) &= 0 \\ \gamma'(t) &= 0 \\ \xi'(t) &= u(\xi(t), t) \end{aligned}$$

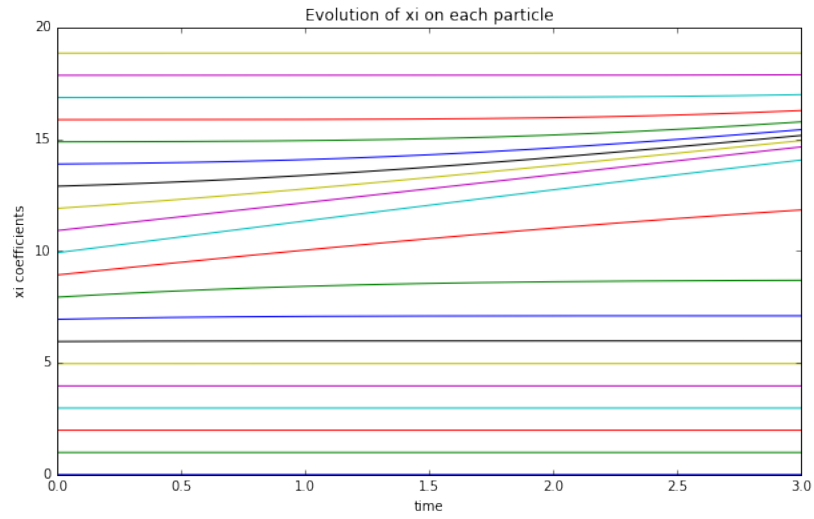
este sistema dice que cada una de las partículas mantiene su su forma y peso, pero que sin embargo pueden cambiar su posición (centro) en el tiempo, y por lo tanto la altura de la columna de agua en cada punto estará determinada sólo por la distribución de las partículas.

4.2. Análisis experimental

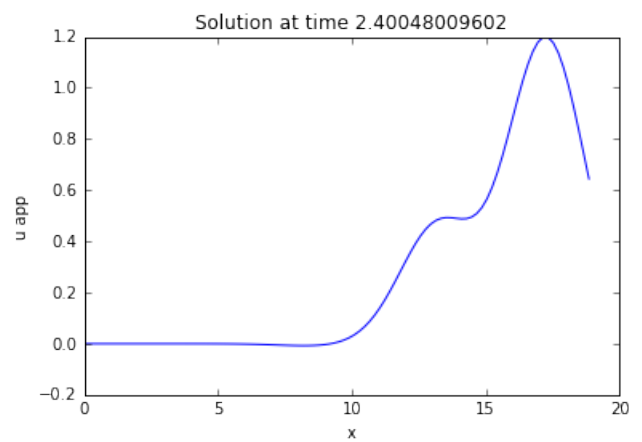
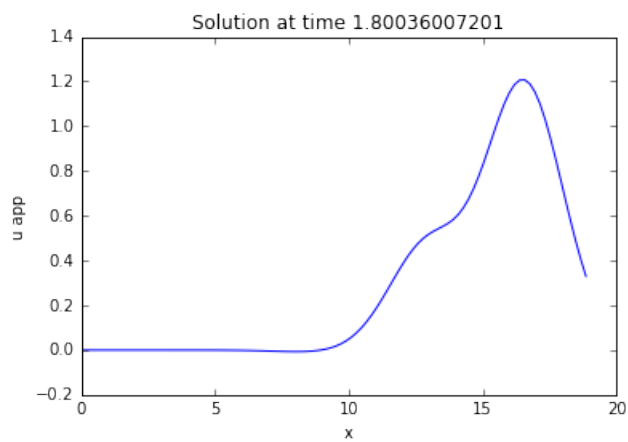
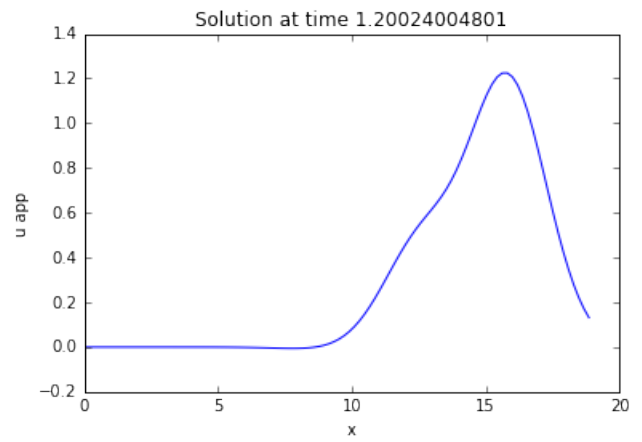
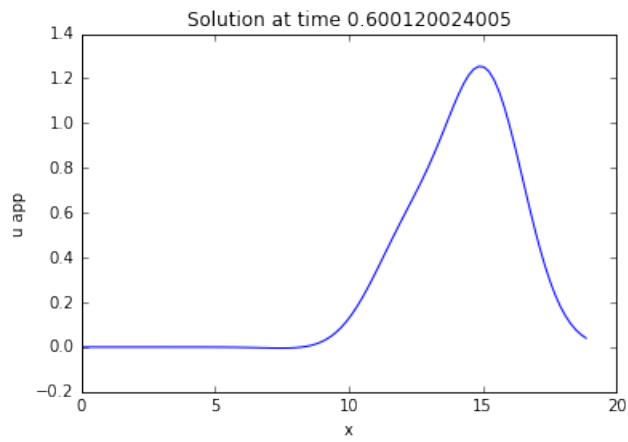
Se realizó un experimento modelando el comportamiento de la ecuación para el estado inicial $u(x, 0) = I_{x \in [8, 12]} \frac{(8-x)(x-12)}{12}$ como se muestra a continuación

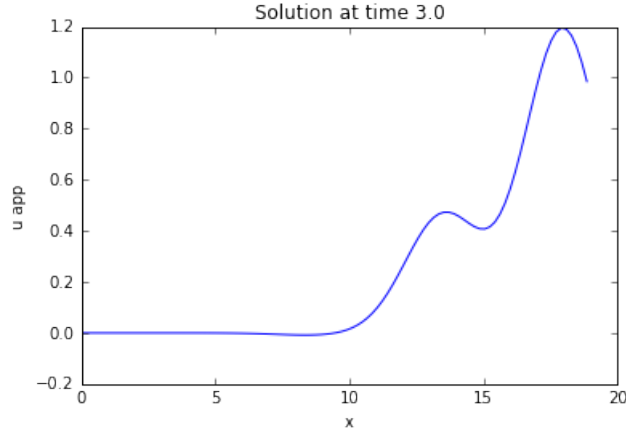


Computando simultáneamente las ecuaciones de evolución para cada una de las RBF, y luego sumando la combinación lineal en cada time step, se obtiene entonces la aproximación de la solución en el tiempo. Como se vio anteriormente el único parámetro que realmente varía es $\xi(t)$, que es la posición del centro de cada partícula. A continuación se muestra la evolución de cada uno de estos centros para un sistema de 20 partículas



Algunos resultados obtenidos para la evolución de la ecuación en el tiempo son los que se muestran a continuación:





Tal como se observa, el comportamiento modelado por la *Burgers' Equation* es bastante simple, consistiendo en un movimiento de la perturbación inicial $u(x, 0)$ hacia la derecha. Algunos de los detalles que pudieron observarse en la simulación, son la aparición de una perturbación más pequeña que parece moverse a menor velocidad que el resto del sistema. Tales problemas pueden ser abordados por medio de *remeshing*, esto es, pasado un intervalo de tiempo volver a realizar una interpolación RBF, para estabilizar las posiciones y velocidades del sistema.

5. Evolutive Shallow Water Equations

5.1. Derivación en 1D

El sistema a resolver corresponde a la versión 1D de las SWE, tomando por simplicidad la restricción de que la batimetría es constante y cero $b(x) = 0$, respecto al nivel de referencia. Así el sistema a trabajar es

$$\frac{\partial h}{\partial t} + h \frac{\partial u}{\partial x} + u \frac{\partial h}{\partial x} = 0 \quad (7)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + g \frac{\partial h}{\partial x} = 0 \quad (8)$$

Tal como se hizo anteriormente para la *Burgers' Equation*, se proponen como solución al sistema anterior, las siguientes dos funciones respectivamente:

$$h(x, t) = \sum_{i=1}^N \gamma_h(t) \phi(x - \xi_h(t), \epsilon_h(t)) + H_0 \quad (9)$$

$$u(x, t) = \sum_{i=1}^N \gamma_u(t) \phi(x - \xi_u(t), \epsilon_u(t)) + U_0 \quad (10)$$

donde H_0 y U_0 son constantes, que denotaremos *Sea level* y *Base velocity*. La idea es que la combinación lineal de RBFs en cada caso, capturen las variaciones por sobre los niveles base establecidos, dando así más estabilidad al enfoque evolutivo.

Para la derivación, consideraremos en primer lugar reemplazar la solución propuesta (9) en (7), tomando como conocidos u y u_x en un tiempo dado

$$\begin{aligned} \sum_{i=1}^N \frac{\partial}{\partial t} [\gamma_{h_i} \phi_{h_i}(x)] + \left(\sum_{i=1}^N \gamma_{h_i} \phi_{h_i}(x) + H_0 \right) u_x + u \sum_{i=1}^N \frac{\partial}{\partial x} [\gamma_{h_i} \phi_{h_i}(x)] &= 0 \\ \sum_{i=1}^N \frac{\partial}{\partial t} [\gamma_{h_i} \phi_{h_i}(x)] + \sum_{i=1}^N \gamma_{h_i} \phi_{h_i}(x) u_x + H_0 u_x + \sum_{i=1}^N u \frac{\partial}{\partial x} [\gamma_{h_i} \phi_{h_i}(x)] &= 0 \\ \sum_{i=1}^N \frac{\partial}{\partial t} [\gamma_{h_i} \phi_{h_i}(x)] + \sum_{i=1}^N \gamma_{h_i} \phi_{h_i}(x) u_x + \sum_{i=1}^N \frac{H_0 u_x}{N} + \sum_{i=1}^N u \frac{\partial}{\partial x} [\gamma_{h_i} \phi_{h_i}(x)] &= 0 \\ \sum_{i=1}^N \left(\frac{\partial}{\partial t} [\gamma_{h_i} \phi_{h_i}(x)] + \left(\gamma_{h_i} \phi_{h_i}(x) + \frac{H_0}{N} \right) u_x + u \frac{\partial}{\partial x} [\gamma_{h_i} \phi_{h_i}(x)] \right) &= 0 \end{aligned}$$

aquí se simplifica la notación $\gamma_{h_i}\phi_{h_i}(x) = \gamma_{h_i}(t)\phi(x - \xi_{h_i}(t), \epsilon_{h_i}(t))$. Adicionalmente se agrega una sumatoria sobre H_0 para poder formar una sola suma. De la última ecuación obtenida, se desprende que si cada partícula o RBF cumple con

$$\frac{\partial}{\partial t}[\gamma_{h_i}\phi_{h_i}(x)] + \left(\gamma_{h_i}\phi_{h_i}(x) + \frac{H_0}{N}\right)u_x + u \frac{\partial}{\partial x}[\gamma_{h_i}\phi_{h_i}(x)] = 0 \quad (11)$$

entonces su combinación lineal será una solución de (7). En (11) se aproximan u y u_x por los valores en los centros de las RBFs, esto es, $u \approx u(\xi(t), t)$ y $u_x \approx u_x(\xi(t), t)$ (Aproximación de Taylor de orden 0). Tal como lo interpreta L. Rossi en [2], esto quiere decir que para cada partícula, estamos aproximando su velocidad por la velocidad en su centro. Por lo tanto, mientras más amplio sea el alcance de cada partícula, mayor será el error que se está cometiendo en la aproximación de la velocidad.

Una derivación equivalente puede realizarse para $u(x, t)$, el reemplazar (10) en (8), esta vez tomando como conocidos u y h_x en un tiempo dado

$$\begin{aligned} \sum_{i=1}^N \frac{\partial}{\partial t}[\gamma_{u_i}\phi_{u_i}(x)] + u \left(\sum_{i=1}^N \frac{\partial}{\partial x}[\gamma_{u_i}\phi_{u_i}(x)] \right) + g h_x &= 0 \\ \sum_{i=1}^N \frac{\partial}{\partial t}[\gamma_{u_i}\phi_{u_i}(x)] + \left(\sum_{i=1}^N u \frac{\partial}{\partial x}[\gamma_{u_i}\phi_{u_i}(x)] \right) + \sum_{i=1}^N \frac{g h_x}{N} &= 0 \\ \sum_{i=1}^N \left(\frac{\partial}{\partial t}[\gamma_{u_i}\phi_{u_i}(x)] + u \frac{\partial}{\partial x}[\gamma_{u_i}\phi_{u_i}(x)] + \frac{g h_x}{N} \right) &= 0 \end{aligned}$$

donde se realiza la misma simplificación de notación, y el mismo truco para sumar gh_x . Del mismo modo que antes, si cada partícula cumple con

$$\frac{\partial}{\partial t}[\gamma_{u_i}\phi_{u_i}(x)] + u \frac{\partial}{\partial x}[\gamma_{u_i}\phi_{u_i}(x)] + \frac{g h_x}{N} = 0 \quad (12)$$

entonces su combinación lineal será una solución de (8). En (12) se aproximan u y h_x por los valores en los centros de las RBFs, generando el mismo tipo de error que antes.

Reemplazando las RBFs respectivas (Gaussianas) en (11) y (12), desarrollando el álgebra necesaria, y realizando las siguientes dos aproximaciones

$$\begin{aligned} e^{\epsilon_h(t)^2(x-\xi_h(t))^2} &= 1 + \epsilon_h(t)^2(x - \xi_h(t))^2 + O([x - \xi_h(t)]^3) \\ e^{\epsilon_u(t)^2(x-\xi_u(t))^2} &= 1 + \epsilon_u(t)^2(x - \xi_u(t))^2 + O([x - \xi_u(t)]^3) \end{aligned}$$

por series de Taylor centradas en $\xi_h(t)$ y $\xi_u(t)$ respectivamente, se obtiene el sistema dinámico (13) con las ecuaciones que definen la evolución de los parámetros, para las partículas de h y u

$$\begin{aligned} \gamma'_h(t) &= -\left(\frac{H}{N} + \gamma_h(t)\right)u_x \\ \xi'_h(t) &= u \\ \epsilon'_h(t) &= \frac{H u_x \epsilon_h(t)}{2N \gamma_h(t)} \\ \gamma'_u(t) &= -\frac{g h_x}{N} \\ \xi'_u(t) &= u \\ \epsilon'_u(t) &= \frac{g h_x \epsilon_u(t)}{2N \gamma_u(t)} \end{aligned} \quad (13)$$

Existen varios detalles interesantes en este sistema, y que se comentan a continuación:

1. Aunque pareciera que el sistema (13) está desacoplado (los parámetros de h no dependen de los de u , y viceversa) esto no es así, pues los parámetros $\gamma_h, \xi_h, \epsilon_h$ dependen de u y u_x , y los parámetros $\gamma_u, \xi_u, \epsilon_u$ dependen de h_x .
2. Las velocidades de las partículas/RBF's de h y u se mueven con la misma velocidad en su centro ($\xi'_h(t) = \xi'_u(t) = u$). Tomando en cuenta que además se ocupan la misma cantidad de partículas en cada caso (N), y con los mismos centros iniciales (detalle de implementación), esto implica que en todo tiempo exista una correspondencia una a uno, entre partículas de h y partículas de u .
3. A diferencia de las ecuaciones de evolución obtenidas para la *Burgers' Equation*, aquí los parámetros γ y ξ respectivos sí evolucionan. Esto difiere también del concepto de partícula planteado en *SWP* [1], donde las partículas de h no pueden cambiar su peso ni su forma, cumpliendo por definición la ecuación de continuidad.

4. Es importante notar que en la ecuación de evolución para ϵ , en ambos casos aparece en el denominador γ_h y γ_u respectivamente. Por lo tanto, estas dos ecuaciones deben tratarse con especial cuidado para no caer en indeterminaciones. Una posible forma de abordar este problema, es estableciendo los niveles base H_0 y U_0 por debajo del valor mínimo en cada caso.
5. Es interesante notar cómo se autorregulan los parámetros γ_h y ϵ_h para satisfacer la condición de continuidad. Si $\gamma_h \rightarrow 0$ entonces $\epsilon_h \rightarrow \infty$, lo cual quiere decir que si una partícula disminuye su peso/altura, entonces se distribuye en una extensión espacial más amplia (distribuye su masa de acuerdo a su altura).

5.2. Simulación en 1D

El proceso de simulación se compone básicamente de la ejecución de tres pasos: **Evolución**, **Reconstrucción** y **Verificación**.

1. **Evolución.** En cada time step, será necesario hacer evolucionar los parámetros de todas las partículas/RBFs del sistema, hacia los valores en un tiempo siguiente. Debido a que estas están desacopladas, esta operación es fácilmente vectorizable o paralelizable. Por lo demás son operaciones de muy bajo costo computacional, siendo esta la gran ventaja del enfoque evolutivo.
2. **Reconstrucción.** Con los valores de cada uno de los parámetros en un tiempo siguiente, ahora es posible reconstruir las funciones u , u_x , h y h_x como la combinación lineal de las correspondientes partículas. Esto es necesario, pues se debe conocer los valores de estas funciones en las posiciones de los centros de las partículas.
3. **Verificación.** Este paso corresponde a verificar la calidad de la soluciones obtenida en cada tiempo, estableciendo las condiciones bajo las cuales es necesario hacer *remeshing*. Algunas condiciones posibles son:
 - a) Si $\gamma = 0$ para alguna partícula de h o u , entonces el sistema (13) se indetermina. Para resolver este problema se puede intentar bajar el nivel base H_0 o U_0 , como alternativa al *remeshing*.
 - b) Como se analizó en la sección de derivación, la precisión espacial de nuestra solución está limitada por la extensión de cada una de las RBFs, vale decir, el método posee un error espacial $O(1/\epsilon_{\min}^2)$ ¹. Por lo tanto si en alguna de las partículas (tanto para h como u), ocurre que $\epsilon \rightarrow 0$, nos indica que nuestra solución va por mal camino.
 - c) El otro problema ocurre cuando las partículas empiezan a separarse considerablemente entre sí, formando *gaps* (algo como lo que sucedió en las simulaciones de *Burgers' Equation*). Para esto se puede establecer una distancia máxima entre partículas adyacentes $|\xi_i(t) - \xi_{i+1}(t)| \leq D_{\max}$, la cual una vez sobrepasada hace necesario hacer *remeshing*.

El proceso de *remeshing*, consiste en volver a la solución en el tiempo anterior, y realizar una interpolación *RBF* en esta tal como si fuese una condición inicial. Luego se sigue evolucionando a partir de los valores de los parámetros que se obtengan en esta interpolación.

¹El orden del término $1/\epsilon_{\min}$ se supone cuadrático, debido a que el procedimiento seguido es análogo al de L. Rossi [2], sin embargo esto debe ser demostrado.

Referencias

- [1] H. Lee and S. Han, “Solving the shallow water equations using 2D SPH particles for interactive applications,” *Vis Comput*, vol. 26, pp. 865–872, 2010.
- [2] L. F. Rossi, “Achieving high-order convergence rates with feforming basis functions,” *SIAM J. Sci. Comput.*, vol. 26, no. 3, pp. 885–906, 2005.