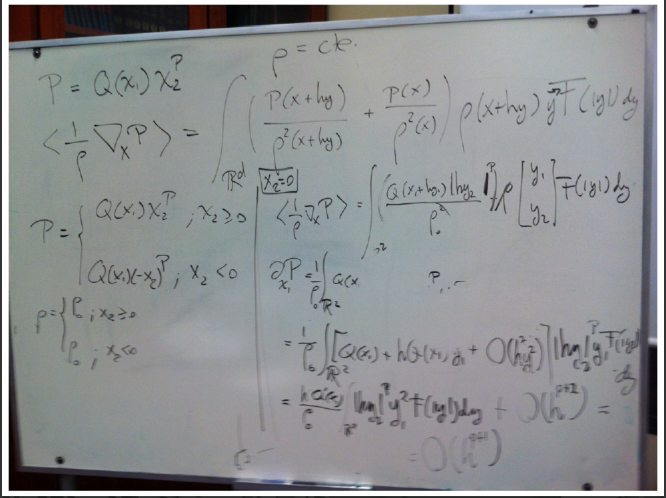
Consistency and Boundary Conditions in SPH

IBERIAN SPH 2015

Fabricio Macià
Universidad Politécnica de Madrid

Blackboard SPH



What is consistency?

When doing numerics, one replaces the *exact problem* we would like to solve:

$$Lu = f,$$

by a series of *approximate problems* (the numerical scheme):

$$L_h U_h = F_h,$$

What is consistency?

When doing numerics, one replaces the *exact problem* we would like to solve:

$$Lu = f,$$

by a series of *approximate problems* (the numerical scheme):

$$L_h U_h = F_h,$$

that can be solved numerically, hoping that

$$U_h \text{ is close to } u.$$

What is consistency?

The numerical scheme is said to be *consistent* provided:

the solution U_h of the approximate problems,

What is consistency?

The numerical scheme is said to be *consistent* provided:

the solution U_h of the approximate problems,

is an *approximate solution* of the *exact problem*:

$$LU_h = f + \mathcal{O}(h^r).$$

Why is consistency important?

Consistency is important because:

If the numerical scheme

$$L_h U_h = F_h$$

is both

consistent & stable

Why is consistency important?

Consistency is important because:

If the numerical scheme

$$L_h U_h = F_h$$

is both

consistent & stable

then it is convergent:

U_h converges to u as h goes to zero.

SPH

Start with a kernel function:

$$W(\mathbf{y}), \quad \mathbf{y} \in \mathbb{R}^d.$$

that is non-negative, smooth, radial, and satisfies:

$$\int_{\mathbb{R}^d} W(\mathbf{y}) \mathrm{d}\mathbf{y} = 1.$$

SPH

Start with a kernel function:

$$W(\mathbf{y}), \quad \mathbf{y} \in \mathbb{R}^d.$$

that is non-negative, smooth, radial, and satisfies:

$$\int_{\mathbb{R}^d} W(\mathbf{y}) \mathrm{d}\mathbf{y} = 1.$$

From this construct an approximation to the Dirac delta point mass:

$$W_h(\mathbf{y}) := \frac{1}{h^d} W\left(\frac{\mathbf{y}}{h}\right).$$

SPH

One starts by considering a (large) number of points in $\mathbb{R}^d$, the particles:

$$\mathbf{x}_1, \mathbf{x}_2, \quad \dots \quad \mathbf{x}_N,$$

with masses:

$$m_1, m_2, \quad \dots \quad m_N.$$

SPH

One starts by considering a (large) number of points in $\mathbb{R}^d$, the particles:

$$\mathbf{x}_1, \mathbf{x}_2, \quad \dots \quad \mathbf{x}_N,$$

with masses:

$$m_1, m_2, \quad \dots \quad m_N.$$

One then approximates a scalar field $u(\mathbf{x})$ by

$$U_h(\mathbf{x}_i) = \langle u \rangle_h (\mathbf{x}_i) := \sum_{j=1}^{N} \frac{m_j}{\rho_j} u(\mathbf{x}_j) W_h(\mathbf{x}_i - \mathbf{x}_j).$$

The parameter h is the effective *interacion range* between particles.

SPH discretization is a two-scale numerical method

The volume associated to each of the particles satisfies:

$$\frac{m_j}{\rho_j} \approx \varepsilon^d$$

where ε is the average nearest-neighbor distance. In practice:

$$\varepsilon \propto \frac{1}{N^{1/d}}.$$

SPH discretization is a two-scale numerical method

The volume associated to each of the particles satisfies:

$$\frac{m_j}{\rho_j} \approx \varepsilon^d$$

where ε is the average nearest-neighbor distance. In practice:

$$\varepsilon \propto \frac{1}{N^{1/d}}.$$

Therefore the SPH discretization is in fact an approximation to:

$$\langle u \rangle_h (\mathbf{x}_i) \approx \int_{\mathbb{R}^d} u(\mathbf{y}) W_h(\mathbf{x}_i - \mathbf{y}) \mathrm{d}\mathbf{y}.$$

We refer to this as the *continuous formulation* of SPH.

Differential operators and SPH

On then uses this idea to obtain discretizations of differential operators: gradients, Laplacian, divergence, etc.

For instance, the gradient is approximated by:

$$\langle \nabla u \rangle_h (\mathbf{x}_i) = \sum_{j=1}^{N} \frac{m_j}{\rho_j} u(\mathbf{x}_j) \nabla W_h(\mathbf{x}_i - \mathbf{x}_j).$$

Differential operators and SPH

On then uses this idea to obtain discretizations of differential operators: gradients, Laplacian, divergence, etc.

For instance, the gradient is approximated by:

$$\left\langle \nabla u \right\rangle_h (\mathbf{x}_i) = \sum_{j=1}^{N} \frac{m_j}{\rho_j} u(\mathbf{x}_j) \nabla W_h(\mathbf{x}_i - \mathbf{x}_j).$$

This arises from the continuous formulation:

$$\left\langle \nabla u \right\rangle_h (\mathbf{x}_i) \approx \int_{\mathbb{R}^d} u(\mathbf{y}) \nabla W_h(\mathbf{x}_i - \mathbf{y}) \mathrm{d}\mathbf{y}.$$

One always has that the SPH discretization of a differential operator has a continuous formulation.

Consistency of SPH

It turns out that SPH is a consistent numerical method

Consistency of SPH

It turns out that SPH is a consistent numerical method

provided work on $\mathbb{R}^d$, *i.e.* no boundaries are involved.

Consistency of SPH

It turns out that SPH is a consistent numerical method

provided work on $\mathbb{R}^d$, *i.e.* no boundaries are involved.

Possible inconsistencies can only appear at the continuous level. The discrete step does not cause any troubles.

Consistency of SPH

It turns out that SPH is a consistent numerical method

provided work on $\mathbb{R}^d$, *i.e.* no boundaries are involved.

Possible inconsistencies can only appear at the continuous level. The discrete step does not cause any troubles.

For instance, the continuous formulation of the gradient is exact:

$$\langle \nabla u \rangle_h (\mathbf{x}_i) = \nabla u((\mathbf{x}_i).$$

One has similar results for the discretization of divergences, Laplacians, etc.

SPH and Boundary Conditions

This is no longer the case if we replace infinite space $\mathbb{R}^d$ by

Ω a (bounded) region (usually the fluid domain).

In most interesting cases, the field $u(\mathbf{y})$ is only defined for $\mathbf{y} \in \Omega$ and one imposes on the boundary $\partial\Omega$ a boundary condition:

$$u(\mathbf{y}) = U_B, \quad \text{for } \mathbf{y} \in \partial\Omega.$$

For fluid fields one usually has: no-slip, free slip, Robin B.C.

SPH and Boundary Conditions: truncation

The naive way: replace integrals over $\mathbb{R}^d$ by integrals over Ω:

$$\langle \nabla u \rangle_h (\mathbf{x}_i) \approx \int_\Omega u(\mathbf{y}) \nabla W_h(\mathbf{x}_i - \mathbf{y}) \mathrm{d}\mathbf{y}.$$

SPH and Boundary Conditions: truncation

The naive way: replace integrals over $\mathbb{R}^d$ by integrals over Ω:

$$\langle \nabla u \rangle_h (\mathbf{x}_i) \approx \int_\Omega u(\mathbf{y}) \nabla W_h(\mathbf{x}_i - \mathbf{y}) \mathrm{d}\mathbf{y}.$$

This is completely inconsistent!!!

SPH and Boundary Conditions: truncation

The naive way: replace integrals over $\mathbb{R}^d$ by integrals over Ω:

$$\langle \nabla u \rangle_h (\mathbf{x}_i) \approx \int_\Omega u(\mathbf{y}) \nabla W_h(\mathbf{x}_i - \mathbf{y}) \mathrm{d}\mathbf{y}.$$

This is completely inconsistent!!!

One is missing a (usually big) term coming from the integration by parts, and most importantly, the fraction of volume of the kernel range tends to zero as we approach the boundary.

SPH and Boundary Conditions: boundary integrals

First solution: include the term coming from integration by parts and renormalize the SPH kernel.

SPH and Boundary Conditions: boundary integrals

First solution: include the term coming from integration by parts and renormalize the SPH kernel.

An example. Pressure gradient on an interval $\Omega = (a, b)$:

$$\left\langle \frac{dp}{dx} \right\rangle_h (x) = \frac{1}{\gamma_h(x)} \int_a^b p(x') \frac{dW_h}{dx}(x - x') \, \mathrm{d}x' + \frac{1}{\gamma_h(x)} \left[p(b) W_h(x - b) - p(a) W_h(x - a) \right]$$

where the normalization factor is defined as:

$$\gamma_h(x) := \int_a^b W_h(x - y) \, \mathrm{d}y.$$

SPH and Boundary Conditions: boundary integrals

Similar ideas go back to Shepard, Belytshcko, etc....

F. Macià, L.M. González, J.L. Cercós-Pita, and A. Souto-Iglesias. A boundary integral SPH formulation: consistency and applications to ISPH and WCSP. *Progress in Theoretical Physics*, **128**(3) (2012), 439–462.

In the same line of ideas: Ferrand *et al.*, Amicarelli *et al.*, and many others.

SPH and Boundary Conditions: boundary integrals

Similar ideas go back to Shepard, Belytshcko, etc....

F. Macià, L.M. González, J.L. Cercós-Pita, and A. Souto-Iglesias. A boundary integral SPH formulation: consistency and applications to ISPH and WCSP. *Progress in Theoretical Physics*, **128**(3) (2012), 439–462.

In the same line of ideas: Ferrand *et al.*, Amicarelli *et al.*, and many others.

Main drawback. It is not so easy and efficient to implement the computation of boundary integrals. Can get complicated in 3-d.

SPH and Boundary Conditions: ghost particles

One introduces a (thin) layer of non-physical particles outside Ω close to the boundary $\partial\Omega$. The so-called *ghost particles*.

SPH and Boundary Conditions: ghost particles

One introduces a (thin) layer of non-physical particles outside Ω close to the boundary $\partial\Omega$. The so-called *ghost particles*.

In the continuous formulation of SPH this amounts to extending the field $u(\mathbf{x})$ for $\mathbf{x}$ outside Ω in order to obtain an extended field:

$$\overline{u}(\mathbf{x}), \text{ defined for } \mathbf{x} \in \mathbb{R}^d.$$

SPH and Boundary Conditions: ghost particles

One introduces a (thin) layer of non-physical particles outside Ω close to the boundary $\partial\Omega$. The so-called *ghost particles*.

In the continuous formulation of SPH this amounts to extending the field $u(\mathbf{x})$ for $\mathbf{x}$ outside Ω in order to obtain an extended field:

$$\bar{u}(\mathbf{x}), \text{ defined for } \mathbf{x} \in \mathbb{R}^d.$$

And then, one applies usual SPH.

SPH and Boundary Conditions: ghost particles

There are many ways to do that:

- Constant extension.

SPH and Boundary Conditions: ghost particles

There are many ways to do that:

- Constant extension.
- Symmetric extension.

SPH and Boundary Conditions: ghost particles

There are many ways to do that:

- Constant extension.
- Symmetric extension.
- Antisymmetric extension.

SPH and Boundary Conditions: ghost particles

There are many ways to do that:

- Constant extension.
- Symmetric extension.
- Antisymmetric extension.
- Takeda’s method

among others.

SPH and Boundary Conditions: ghost particles

There are many ways to do that:

- Constant extension.
- Symmetric extension.
- Antisymmetric extension.
- Takeda’s method

among others.

Advantages: very easy to implement!

SPH and Boundary Conditions: ghost particles

In

F. Macià, M. Antuono, A.Colagrossi, and L.M. González. Theoretical analysis of the no-slip boundary condition enforcement in SPH methods. *Progress in Theoretical Physics*, **125**(6) (2011), 1091–1121.

we analyze the consistency of enforcing B.C. using this approach in a simple setting (unidirectional fields, flat boundaries).

SPH and Boundary Conditions: ghost particles

In

F. Macià, M. Antuono, A.Colagrossi, and L.M. González. Theoretical analysis of the no-slip boundary condition enforcement in SPH methods. *Progress in Theoretical Physics*, **125**(6) (2011), 1091–1121.

we analyze the consistency of enforcing B.C. using this approach in a simple setting (unidirectional fields, flat boundaries).

It turns that none of these extension methods gives *simultaneously* a consistent discretization for all the differential operators one needs to discretize the Navier-Stokes system.

SPH and Boundary Conditions: ghost particles

Consistency of the ghost particle method is tightly related to the differentiability properties of the extended field $\overline{u}(\mathbf{x})$ at points $\mathbf{x}$ of the boundary $\partial\Omega$.

In general, big derivatives (or more precisely, big *modulus of continuity*) of the extended fields at points close to the boundary gives rise to inconsistencies.

SPH and Boundary Conditions: ghost particles

Consistency of the ghost particle method is tightly related to the differentiability properties of the extended field $\overline{u}(\mathbf{x})$ at points $\mathbf{x}$ of the boundary $\partial\Omega$.

In general, big derivatives (or more precisely, big *modulus of continuity*) of the extended fields at points close to the boundary gives rise to inconsistencies.

Making this precise is a bit technical, though.

SPH and MPS: consistency goes both ways

Our continuous formulation/two-scale approach to the analysis of SPH has allowed us to prove that the Moving-Particle Semi-Implicit Method is essentially equivalent to SPH.

SPH and MPS: consistency goes both ways

Our continuous formulation/two-scale approach to the analysis of SPH has allowed us to prove that the Moving-Particle Semi-Implicit Method is essentially equivalent to SPH.

The main difference is that MPS uses different kernels W_h to compute the discretizations of the gradient and the Laplacian.

SPH and MPS: consistency goes both ways

Our continuous formulation/two-scale approach to the analysis of SPH has allowed us to prove that the Moving-Particle Semi-Implicit Method is essentially equivalent to SPH.

The main difference is that MPS uses different kernels W_h to compute the discretizations of the gradient and the Laplacian.

There is a precise “dictionary”, that allows to translate any consistency result on SPH to a result on MPS, and the other way round.

SPH and MPS: consistency goes both ways

Results in this direction can be found in:

1. A. Souto-Iglesias, F. Macià, L.M. González, and J.L. Cercós-Pita. On the consistency of MPS. *Computer Physics Communications*, **184**(3) (2013), 732–745.
2. A. Souto-Iglesias, F. Macià, L.M. González, and J.L. Cercós-Pita. Addendum to: "On the consistency of MPS". *Computer Physics Communications*, **185**(2) (2014), 595–598.