MF921 Topics in Dynamic Asset Pricing

Week 11

Yuanhui Zhao

Boston University

Chapter 14

An introduction to Finite Difference methods for PDEs in Finance

Agnès Tourin

Introduction

This note focuses on the practical design of finite difference methods for solving Hamilton–Jacobi–Bellman (HJB) equations that arise in quantitative finance. These equations typically model optimal decision-making over time, such as in portfolio selection or option pricing. Because these problems involve both time evolution and uncertainty (stochastic behavior), the resulting partial differential equations (PDEs) are usually of parabolic type.

A key framework used in the note is the Barles–Souganidis convergence theory. It shows that if a numerical scheme is consistent (approximates the correct PDE), monotone (preserves order in the solution), and stable (bounded behavior over time), then the numerical solution will converge to the correct viscosity solution of the HJB equation.

Finite difference methods are most useful in problems with low spatial dimension (1 to 3 variables). Their simplicity and ease of implementation make them attractive for many practical finance problems. In higher dimensions, they can be combined with Monte Carlo methods to retain computational tractability.

Introduction

This note focuses on the practical design of finite difference methods for solving Hamilton–Jacobi–Bellman (HJB) equations that arise in quantitative finance. These equations typically model optimal decision-making over time, such as in portfolio selection or option pricing. Because these problems involve both time evolution and uncertainty (stochastic behavior), the resulting partial differential equations (PDEs) are usually of parabolic type.

A key framework used in the note is the Barles–Souganidis convergence theory. It shows that if a numerical scheme is consistent (approximates the correct PDE), monotone (preserves order in the solution), and stable (bounded behavior over time), then the numerical solution will converge to the correct viscosity solution of the HJB equation.

Finite difference methods are most useful in problems with low spatial dimension (1 to 3 variables). Their simplicity and ease of implementation make them attractive for many practical finance problems. In higher dimensions, they can be combined with Monte Carlo methods to retain computational tractability.

Introduction

This note focuses on the practical design of finite difference methods for solving Hamilton–Jacobi–Bellman (HJB) equations that arise in quantitative finance. These equations typically model optimal decision-making over time, such as in portfolio selection or option pricing. Because these problems involve both time evolution and uncertainty (stochastic behavior), the resulting partial differential equations (PDEs) are usually of parabolic type.

A key framework used in the note is the Barles–Souganidis convergence theory. It shows that if a numerical scheme is consistent (approximates the correct PDE), monotone (preserves order in the solution), and stable (bounded behavior over time), then the numerical solution will converge to the correct viscosity solution of the HJB equation.

Finite difference methods are most useful in problems with low spatial dimension (1 to 3 variables). Their simplicity and ease of implementation make them attractive for many practical finance problems. In higher dimensions, they can be combined with Monte Carlo methods to retain computational tractability.

Barles-Souganidis framework

Consider the parabolic PDE

$$u_t + F(t, x, u, Du, D^2u) = 0 \text{ in } (0, T] \times \mathbb{R}^N \tag{1}$$

$$u(0, x) = u_0(x) \text{ in } \mathbb{R}^N \tag{2}$$

where F is Elliptic

$$F(t, x, u, p, A) \leq F(t, x, u, p, B), \text{ if } A \geq B.$$

For a sake of simplicity, we assume that u_0 is bounded in $\mathbb{R}^N$. Furthermore, we assume that (1), (2) satisfy a strong comparison principle.

Strong Comparison Principle

For (1), (2), If u is a viscosity subsolution (upper semi-continuous, bounded) and v is a viscosity supersolution (lower semi-continuous, bounded). Then:

$$u(t, x) \leq v(t, x) \quad \text{for all } (t, x) \in [0, T] \times \mathbb{R}^N$$

This guarantees uniqueness: if two solutions start from the same initial condition, they cannot cross.

Barles-Souganidis framework

Consider the parabolic PDE

$$u_t + F(t, x, u, Du, D^2u) = 0 \text{ in } (0, T] \times \mathbb{R}^N \tag{1}$$

$$u(0, x) = u_0(x) \text{ in } \mathbb{R}^N \tag{2}$$

where F is Elliptic

$$F(t, x, u, p, A) \leq F(t, x, u, p, B), \text{ if } A \geq B.$$

For a sake of simplicity, we assume that u_0 is bounded in $\mathbb{R}^N$. Furthermore, we assume that (1), (2) satisfy a strong comparison principle.

Strong Comparison Principle

For (1), (2), If u is a viscosity subsolution (upper semi-continuous, bounded) and v is a viscosity supersolution (lower semi-continuous, bounded). Then:

$$u(t, x) \leq v(t, x) \quad \text{for all } (t, x) \in [0, T] \times \mathbb{R}^N$$

This guarantees uniqueness: if two solutions start from the same initial condition, they cannot cross.

Barles-Souganidis framework

Consider the parabolic PDE

$$u_t + F(t, x, u, Du, D^2u) = 0 \text{ in } (0, T] \times \mathbb{R}^N \tag{1}$$

$$u(0, x) = u_0(x) \text{ in } \mathbb{R}^N \tag{2}$$

where F is Elliptic

$$F(t, x, u, p, A) \leq F(t, x, u, p, B), \text{ if } A \geq B.$$

For a sake of simplicity, we assume that u_0 is bounded in $\mathbb{R}^N$. Furthermore, we assume that (1), (2) satisfy a strong comparison principle.

Strong Comparison Principle

For (1), (2), If u is a viscosity subsolution (upper semi-continuous, bounded) and v is a viscosity supersolution (lower semi-continuous, bounded). Then:

$$u(t, x) \leq v(t, x) \quad \text{for all } (t, x) \in [0, T] \times \mathbb{R}^N$$

This guarantees uniqueness: if two solutions start from the same initial condition, they cannot cross.

Barles-Souganidis framework

The PDE operator F often comes from stochastic control problems, like optimal trading or investment.
Here, F is defined as:

$$F(t,x,r,p,X) = \sup_{\alpha \in A} \left\{ -\mathrm{tr}[a^{\alpha}(t,x)X] - b^{\alpha}(t,x)p - c^{\alpha}(t,x)r - f^{\alpha}(t,x) \right\},$$

where $a^{\alpha} = \frac{1}{2}\sigma^{\alpha}\sigma^{\alpha T}$.

Typically, the set of control $\mathcal{A}$ is compact or finite, all the coefficients in the equations are bounded and Lipschitz continuous in x, Hölder with coefficient $\frac{1}{2}$ in t and all the bounds are independent of α. Then the unique viscosity solution u of (1) is a bounded and Lipschitz continuous function and is the solution of the underlying stochastic control problem. The ideas, concepts and techniques actually apply to a broader range of optimal control problems. In particular, you can adapt the techniques to handle different situations, even possibly treat some delicate singular control problems.

Barles-Souganidis framework

The PDE operator F often comes from stochastic control problems, like optimal trading or investment.
Here, F is defined as:

$$F(t,x,r,p,X) = \sup_{\alpha \in A} \left\{ -\mathrm{tr}[a^{\alpha}(t,x)X] - b^{\alpha}(t,x)p - c^{\alpha}(t,x)r - f^{\alpha}(t,x) \right\},$$

where $a^{\alpha} = \frac{1}{2}\sigma^{\alpha}\sigma^{\alpha T}$.

Typically, the set of control $\mathcal{A}$ is compact or finite, all the coefficients in the equations are bounded and Lipschitz continuous in x, Hölder with coefficient $\frac{1}{2}$ in t and all the bounds are independent of α. Then the unique viscosity solution u of (1) is a bounded and Lipschitz continuous function and is the solution of the underlying stochastic control problem. The ideas, concepts and techniques actually apply to a broader range of optimal control problems. In particular, you can adapt the techniques to handle different situations, even possibly treat some delicate singular control problems.

Barles-Souganidis framework

The aim is to build an approximation scheme which preserves the Ellipticity. This discrete Ellipticity property is called monotonicity. The monotonicity, together with the consistency of the scheme and some regularity and boundedness conditions ensure its convergence to the unique viscosity solution of the PDE (1),(2). Without monotonicity, even a consistent scheme can converge to the wrong solution

The theory of viscosity solutions relies on comparison principles: the idea that a subsolution is always less than a supersolution. A non-monotone scheme can break this order at the discrete level. A computed subsolution becomes larger than the supersolution and the whole logic of convergence collapses.

A consistent scheme only tells you that you are approximating some PDE — but unless it is monotone, you might be approximating the wrong solution to the right PDE.

Barles-Souganidis framework

The aim is to build an approximation scheme which preserves the Ellipticity. This discrete Ellipticity property is called monotonicity. The monotonicity, together with the consistency of the scheme and some regularity and boundedness conditions ensure its convergence to the unique viscosity solution of the PDE (1),(2). Without monotonicity, even a consistent scheme can converge to the wrong solution

The theory of viscosity solutions relies on comparison principles: the idea that a subsolution is always less than a supersolution. A non-monotone scheme can break this order at the discrete level. A computed subsolution becomes larger than the supersolution and the whole logic of convergence collapses.

A consistent scheme only tells you that you are approximating some PDE — but unless it is monotone, you might be approximating the wrong solution to the right PDE.

Barles-Souganidis framework

The aim is to build an approximation scheme which preserves the Ellipticity. This discrete Ellipticity property is called monotonicity. The monotonicity, together with the consistency of the scheme and some regularity and boundedness conditions ensure its convergence to the unique viscosity solution of the PDE (1),(2). Without monotonicity, even a consistent scheme can converge to the wrong solution

The theory of viscosity solutions relies on comparison principles: the idea that a subsolution is always less than a supersolution. A non-monotone scheme can break this order at the discrete level. A computed subsolution becomes larger than the supersolution and the whole logic of convergence collapses.

A consistent scheme only tells you that you are approximating some PDE — but unless it is monotone, you might be approximating the wrong solution to the right PDE.

Barles-Souganidis framework

A numerical scheme is an equation of the following form

$$S(h, t, x, u_h(t, x), [u_h]_{t,x}) = 0 \quad \text{for } (t, x) \in G_h \setminus \{t = 0\} \tag{3}$$

$$u_h(0, x) = u_0(x) \quad \text{in } G_h \cap \{t = 0\} \tag{4}$$

- $h = (\Delta t, \Delta x)$: This is the discretization size. Time and space steps.
- $G_h = \Delta t \cdot \{0, 1, \ldots, n_T\} \times \Delta x \cdot \mathbb{Z}^N$: This defines a discrete space-time grid: time from 0 to T in steps of Δt, and space as an N-dimensional grid in steps of Δx.
- $u_h(t, x)$: This is the numerical approximation of the solution $u(t, x)$ on the grid.
- $[u_h]_{t,x}$: This stands for values of u_h at nearby grid points used to compute derivatives numerically (like neighbors in finite difference formulas).

Note: The numerical solution u_h is computed only at grid (mesh) points. But we can treat it like a continuous function by interpolating between those values. This idea connects the discrete world (numerical) with the continuous world (PDE).

Barles-Souganidis framework

A numerical scheme is an equation of the following form

$$S(h, t, x, u_h(t, x), [u_h]_{t,x}) = 0 \quad \text{for } (t, x) \in G_h \setminus \{t = 0\} \tag{3}$$

$$u_h(0, x) = u_0(x) \quad \text{in } G_h \cap \{t = 0\} \tag{4}$$

- $h = (\Delta t, \Delta x)$: This is the discretization size. Time and space steps.
- $G_h = \Delta t \cdot \{0, 1, \ldots, n_T\} \times \Delta x \cdot \mathbb{Z}^N$: This defines a discrete space-time grid: time from 0 to T in steps of Δt, and space as an N-dimensional grid in steps of Δx.
- $u_h(t, x)$: This is the numerical approximation of the solution $u(t, x)$ on the grid.
- $[u_h]_{t,x}$: This stands for values of u_h at nearby grid points used to compute derivatives numerically (like neighbors in finite difference formulas).

Note: The numerical solution u_h is computed only at grid (mesh) points. But we can treat it like a continuous function by interpolating between those values. This idea connects the discrete world (numerical) with the continuous world (PDE).

Barles-Souganidis framework

A numerical scheme is an equation of the following form

$$S(h, t, x, u_h(t,x), [u_h]_{t,x}) = 0 \quad \text{for } (t,x) \in G_h \setminus \{t=0\} \tag{3}$$

$$u_h(0,x) = u_0(x) \quad \text{in } G_h \cap \{t=0\} \tag{4}$$

- $h = (\Delta t, \Delta x)$: This is the discretization size. Time and space steps.
- $G_h = \Delta t \cdot \{0, 1, \dots, n_T\} \times \Delta x \cdot \mathbb{Z}^N$: This defines a discrete space-time grid: time from 0 to T in steps of Δt, and space as an N-dimensional grid in steps of Δx.
- $u_h(t,x)$: This is the numerical approximation of the solution $u(t,x)$ on the grid.
- $[u_h]_{t,x}$: This stands for values of u_h at nearby grid points used to compute derivatives numerically (like neighbors in finite difference formulas).

Note: The numerical solution u_h is computed only at grid (mesh) points. But we can treat it like a continuous function by interpolating between those values. This idea connects the discrete world (numerical) with the continuous world (PDE).

Barles-Souganidis framework

The theory requires the following assumptions

Monotonicity

$$S(h,t,x,r,u) \geq S(h,t,x,r,v) \text{ if } u \leq v.$$

The monotonicity assumption can be weakened somewhat. We only need it to hold approximately, with a margin of error that vanishes to 0 as h goes to 0.

Consistency

For every smooth function $\Phi(t,x)$ with bounded derivatives,

$$\lim_{h\to 0,(n\Delta t, i\Delta x)\to(t,x),c\to 0} S(h, n\Delta t, i\Delta x, \Phi(t,x)+c, [\Phi+c]_{t,x}) \\ = \Phi_t + F(t,x,\Phi(t,x), D\Phi(t,x), D^2\Phi(t,x)).$$

Stability

For every $h > 0$, the scheme has a solution u_h which is uniformly bounded independently of h.

Barles-Souganidis framework

The theory requires the following assumptions

Monotonicity

$$S(h,t,x,r,u) \geq S(h,t,x,r,v) \text{ if } u \leq v.$$

The monotonicity assumption can be weakened somewhat. We only need it to hold approximately, with a margin of error that vanishes to 0 as h goes to 0.

Consistency

For every smooth function $\Phi(t,x)$ with bounded derivatives,

$$\lim_{h\to 0,(n\Delta t,i\Delta x)\to(t,x),c\to 0} S(h,n\Delta t,i\Delta x,\Phi(t,x)+c,[\Phi+c]_{t,x}) \\ = \Phi_t + F(t,x,\Phi(t,x),D\Phi(t,x),D^2\Phi(t,x)).$$

Stability

For every $h>0$, the scheme has a solution u_h which is uniformly bounded independently of h.

Barles-Souganidis framework

The theory requires the following assumptions

Monotonicity

$$S(h,t,x,r,u) \geq S(h,t,x,r,v) \text{ if } u \leq v.$$

The monotonicity assumption can be weakened somewhat. We only need it to hold approximately, with a margin of error that vanishes to 0 as h goes to 0.

Consistency

For every smooth function $\Phi(t,x)$ with bounded derivatives,

$$\lim_{h\to 0,(n\Delta t, i\Delta x)\to(t,x),c\to 0} S(h, n\Delta t, i\Delta x, \Phi(t,x)+c, [\Phi+c]_{t,x}) \\ = \Phi_t + F(t,x,\Phi(t,x), D\Phi(t,x), D^2\Phi(t,x)).$$

Stability

For every $h>0$, the scheme has a solution u_h which is uniformly bounded independently of h.

Barles-Souganidis framework

Theorem (Barles-Souganidis[5])

Under the above assumptions, if the scheme (3),(4) satisfy the consistency, monotonicity and stability properties, its solution u_h converges locally uniformly to the unique viscosity solution of (1),(2).

Next, we will go through some example tests to illustrate this theory.

The heat equation

First, recall the classic explicit and implicit schemes for the linear heat equation in dimension 1 and verify that these schemes satisfy the required properties.

$$u_t - u_{xx} = 0 \text{ in } (0,T] \times \mathbb{R}. \tag{5}$$

$$u(0,x) = u_0(x) \tag{6}$$

We assume here that u_0 is continuous and bounded in $\mathbb{R}$. Of course, the linear heat equation does not require the machinery of viscosity solutions but falls into the scope of this theory and provides the opportunity to understand the connection between the theory for linear parabolic equations and the theory of viscosity solutions. More precisely, our goal here is to verify that the standard finite difference approximations for the heat equation are convergent in the Barles-Souganidis sense.

The heat equation

First, recall the classic explicit and implicit schemes for the linear heat equation in dimension 1 and verify that these schemes satisfy the required properties.

$$u_t - u_{xx} = 0 \text{ in } (0,T] \times \mathbb{R}. \tag{5}$$

$$u(0,x) = u_0(x) \tag{6}$$

We assume here that u_0 is continuous and bounded in $\mathbb{R}$. Of course, the linear heat equation does not require the machinery of viscosity solutions but falls into the scope of this theory and provides the opportunity to understand the connection between the theory for linear parabolic equations and the theory of viscosity solutions. More precisely, our goal here is to verify that the standard finite difference approximations for the heat equation are convergent in the Barles-Souganidis sense.

The standard explicit scheme

First of all, we define the mesh $\{(n\Delta t, i\Delta x) | n \in \{0, \ldots, N\}, i \in \mathbb{Z}\}$ where $(\Delta t, \Delta x)$ are the discretization steps and N is such that $N\Delta t = T$.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Since this scheme is explicit, it is very easy to compute at each time step $n+1$ the value of the approximation $\{u_i^{n+1} | i \in \mathbb{Z}\}$ from the value of the approximation at the time step n, namely $\{u_i^n | i \in \mathbb{Z}\}$

$$u_i^{n+1} = u_i^n + \Delta t \left\{ \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \right\}.$$

Furthermore, we initialize this algorithm by imposing the initial condition

$$u_i^0 = u_0(i\Delta x).$$

Here, we may define formally the scheme S by setting:

$$S(\Delta t, \Delta x, (n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^n, u_i^n, u_{i+1}^n]) = \frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Note that, in the last argument, only 3 grid points are used, instead of the whole grid.

The standard explicit scheme

First of all, we define the mesh $\{(n\Delta t, i\Delta x) | n \in \{0, \ldots, N\}, i \in \mathbb{Z}\}$ where $(\Delta t, \Delta x)$ are the discretization steps and N is such that $N\Delta t = T$.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Since this scheme is explicit, it is very easy to compute at each time step $n+1$ the value of the approximation $\{u_i^{n+1} | i \in \mathbb{Z}\}$ from the value of the approximation at the time step n, namely $\{u_i^n | i \in \mathbb{Z}\}$

$$u_i^{n+1} = u_i^n + \Delta t \left\{ \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \right\}.$$

Furthermore, we initialize this algorithm by imposing the initial condition

$$u_i^0 = u_0(i\Delta x).$$

Here, we may define formally the scheme S by setting:

$$S(\Delta t, \Delta x, (n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^n, u_i^n, u_{i+1}^n]) = \frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Note that, in the last argument, only 3 grid points are used, instead of the whole grid.

The standard explicit scheme

First of all, we define the mesh $\{(n\Delta t, i\Delta x) | n \in \{0, \dots, N\}, i \in \mathbb{Z}\}$ where $(\Delta t, \Delta x)$ are the discretization steps and N is such that $N\Delta t = T$.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Since this scheme is explicit, it is very easy to compute at each time step $n+1$ the value of the approximation $\{u_i^{n+1} | i \in \mathbb{Z}\}$ from the value of the approximation at the time step n, namely $\{u_i^n | i \in \mathbb{Z}\}$

$$u_i^{n+1} = u_i^n + \Delta t \left\{ \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \right\}.$$

Furthermore, we initialize this algorithm by imposing the initial condition

$$u_i^0 = u_0(i\Delta x).$$

Here, we may define formally the scheme S by setting:

$$S(\Delta t, \Delta x, (n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^n, u_i^n, u_{i+1}^n]) = \frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Note that, in the last argument, only 3 grid points are used, instead of the whole grid.

The standard explicit scheme

First of all, we define the mesh $\{(n\Delta t, i\Delta x) | n \in \{0, \dots, N\}, i \in \mathbb{Z}\}$ where $(\Delta t, \Delta x)$ are the discretization steps and N is such that $N\Delta t = T$.

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Since this scheme is explicit, it is very easy to compute at each time step $n+1$ the value of the approximation $\{u_i^{n+1} | i \in \mathbb{Z}\}$ from the value of the approximation at the time step n, namely $\{u_i^n | i \in \mathbb{Z}\}$

$$u_i^{n+1} = u_i^n + \Delta t \left\{ \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \right\}.$$

Furthermore, we initialize this algorithm by imposing the initial condition

$$u_i^0 = u_0(i\Delta x).$$

Here, we may define formally the scheme S by setting:

$$S(\Delta t, \Delta x, (n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^n, u_i^n, u_{i+1}^n]) = \frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Note that, in the last argument, only 3 grid points are used, instead of the whole grid.

The standard explicit scheme

This section is about verifying that the explicit scheme is consistent, meaning it approximates the true PDE correctly as the grid is refined. We're checking that:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} \approx u_t(t_n, x_i), \quad \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \approx u_{xx}(t_n, x_i)$$

If both approximations match the true derivatives up to small error, then the scheme is consistent with:

$$u_t = u_{xx}$$

Spatial Derivative Approximation (Center Difference)

We expand u_{i+1}^n and u_{i-1}^n around x_i using Taylor expansions:

$$u_{i+1}^n = u_i^n + u_x(n\Delta t, i\Delta x)\Delta x + \frac{1}{2}u_{xx}(n\Delta t, i\Delta x)\Delta x^2 + u_{xxx}(n\Delta t, i\Delta x)\frac{1}{6}\Delta x^3 +$$
$$\frac{1}{24}u_{xxxx}(n\Delta t, i\Delta x)\Delta x^4 + \Delta x^4 \epsilon(\Delta x),$$

and

$$u_{i-1}^n = u_i^n - u_x(n\Delta t, i\Delta x)\Delta x + \frac{1}{2}u_{xx}(n\Delta t, i\Delta x)\Delta x^2 - \frac{1}{6}u_{xxx}(n\Delta t, i\Delta x)\Delta x^3 +$$
$$\frac{1}{24}u_{xxxx}(n\Delta t, i\Delta x)\Delta x^4 + \Delta x^4 \epsilon(\Delta x).$$

The standard explicit scheme

This section is about verifying that the explicit scheme is consistent, meaning it approximates the true PDE correctly as the grid is refined. We're checking that:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} \approx u_t(t_n, x_i), \quad \frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} \approx u_{xx}(t_n, x_i)$$

If both approximations match the true derivatives up to small error, then the scheme is consistent with:

$$u_t = u_{xx}$$

Spatial Derivative Approximation (Center Difference)
We expand u_{i+1}^n and u_{i-1}^n around x_i using Taylor expansions:

$$u_{i+1}^n = u_i^n + u_x(n\Delta t, i\Delta x)\Delta x + \frac{1}{2}u_{xx}(n\Delta t, i\Delta x)\Delta x^2 + u_{xxx}(n\Delta t, i\Delta x)\frac{1}{6}\Delta x^3 +$$

$$\frac{1}{24}u_{xxxx}(n\Delta t, i\Delta x)\Delta x^4 + \Delta x^4 \epsilon(\Delta x),$$

and

$$u_{i-1}^n = u_i^n - u_x(n\Delta t, i\Delta x)\Delta x + \frac{1}{2}u_{xx}(n\Delta t, i\Delta x)\Delta x^2 - \frac{1}{6}u_{xxx}(n\Delta t, i\Delta x)\Delta x^3 +$$

$$\frac{1}{24}u_{xxxx}(n\Delta t, i\Delta x)\Delta x^4 + \Delta x^4 \epsilon(\Delta x).$$

The standard explicit scheme

Where $\lim_{\Delta x \to 0} \epsilon(\Delta x) = 0$. Then, adding up the two expansions, subtracting $2u_i^n$ from the left and right hand sides and dividing by Δx^2, one obtains

$$\frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} = u_{xx}(n\Delta t, i\Delta x) + \frac{1}{12} u_{xxxx}(n\Delta t, i\Delta x)\Delta x^2 + \Delta x^2 \epsilon(\Delta x),$$

and thus the truncation error for this approximation of the second spatial derivative is of order 2. Similarly the expansion

$$u_i^{n+1} = u_i^n + u_t(n\Delta t, i\Delta x)\Delta t + \frac{1}{2} u_{tt}(n\Delta t, i\Delta x)\Delta t^2 + \Delta t^2 \epsilon(\Delta t),$$

where $\lim_{\Delta t \to 0} \epsilon(\Delta t) = 0$ yields

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = u_t(n\Delta t, i\Delta x) + \frac{1}{2} u_{tt}(n\Delta t, i\Delta x)\Delta t + \Delta t \epsilon(\Delta t).$$

The truncation error for the approximation of the first derivative in time is of order 1 only. So the scheme is formally consistent with:

$$u_t = u_{xx},$$

as $\Delta t \to 0$ and $\Delta x \to 0$.

The standard explicit scheme

Where $\lim_{\Delta x \to 0} \epsilon(\Delta x) = 0$. Then, adding up the two expansions, subtracting $2u_i^n$ from the left and right hand sides and dividing by Δx^2, one obtains

$$\frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} = u_{xx}(n\Delta t, i\Delta x) + \frac{1}{12} u_{xxxx}(n\Delta t, i\Delta x)\Delta x^2 + \Delta x^2 \epsilon(\Delta x),$$

and thus the truncation error for this approximation of the second spatial derivative is of order 2. Similarly the expansion

$$u_i^{n+1} = u_i^n + u_t(n\Delta t, i\Delta x)\Delta t + \frac{1}{2} u_{tt}(n\Delta t, i\Delta x)\Delta t^2 + \Delta t^2 \epsilon(\Delta t),$$

where $\lim_{\Delta t \to 0} \epsilon(\Delta t) = 0$ yields

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = u_t(n\Delta t, i\Delta x) + \frac{1}{2} u_{tt}(n\Delta t, i\Delta x)\Delta t + \Delta t \epsilon(\Delta t).$$

The truncation error for the approximation of the first derivative in time is of order 1 only. So the scheme is formally consistent with:

$$u_t = u_{xx},$$

as $\Delta t \to 0$ and $\Delta x \to 0$.

The standard explicit scheme

Where $\lim_{\Delta x \to 0} \epsilon(\Delta x) = 0$. Then, adding up the two expansions, subtracting $2u_i^n$ from the left and right hand sides and dividing by Δx^2, one obtains

$$\frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2} = u_{xx}(n\Delta t, i\Delta x) + \frac{1}{12}u_{xxxx}(n\Delta t, i\Delta x)\Delta x^2 + \Delta x^2 \epsilon(\Delta x),$$

and thus the truncation error for this approximation of the second spatial derivative is of order 2. Similarly the expansion

$$u_i^{n+1} = u_i^n + u_t(n\Delta t, i\Delta x)\Delta t + \frac{1}{2}u_{tt}(n\Delta t, i\Delta x)\Delta t^2 + \Delta t^2 \epsilon(\Delta t),$$

where $\lim_{\Delta t \to 0} \epsilon(\Delta t) = 0$ yields

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = u_t(n\Delta t, i\Delta x) + \frac{1}{2}u_{tt}(n\Delta t, i\Delta x)\Delta t + \Delta t \epsilon(\Delta t).$$

The truncation error for the approximation of the first derivative in time is of order 1 only. So the scheme is formally consistent with:

$$u_t = u_{xx},$$

as $\Delta t \to 0$ and $\Delta x \to 0$.

The standard explicit scheme

For the heat equation explicit scheme:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n - 2u_i^n).$$

Rewriting:

$$u_i^{n+1} = u_i^n \left(1 - \frac{2\Delta t}{\Delta x^2}\right) + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n).$$

Furthermore, the approximation S is monotone if and only if S is decreasing in u_i^n, u_{i+1}^n and u_{i-1}^n. First of all, it is unconditionally decreasing with respect to both u_{i-1}^n and u_{i+1}^n. Secondly, it is only decreasing in u_i^n if the following CFL condition is satisfied:

$$(-1 + 2\frac{\Delta t}{\Delta x^2}) \leq 0$$

or equivalently

$$\Delta t \leq \frac{1}{2}\Delta x^2.$$

The standard explicit scheme

For the heat equation explicit scheme:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n - 2u_i^n).$$

Rewriting:

$$u_i^{n+1} = u_i^n \left(1 - \frac{2\Delta t}{\Delta x^2}\right) + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n).$$

Furthermore, the approximation S is monotone if and only if S is decreasing in u_i^n, u_{i+1}^n and u_{i-1}^n. First of all, it is unconditionally decreasing with respect to both u_{i-1}^n and u_{i+1}^n. Secondly, it is only decreasing in u_i^n if the following CFL condition is satisfied:

$$(-1 + 2\frac{\Delta t}{\Delta x^2}) \leq 0$$

or equivalently

$$\Delta t \leq \frac{1}{2}\Delta x^2.$$

The standard explicit scheme

Finally, we can prove the boundedness of the approximation independently of the discretization steps and hence the stability of the scheme recursively by showing the maximum principle

$$\max_i |u_i^n| \leq \max_i |u_i^0|$$

using the monotonicity property under the CFL condition.

Indeed, we have

$$\begin{aligned} u_i^{n+1} &= u_i^n(1 - 2\frac{\Delta t}{\Delta x^2}) + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n) \\ &\leq \max_i |u_i^n|(1 - 2\frac{\Delta t}{\Delta x^2}) + 2\frac{\Delta t}{\Delta x^2} \max_i |u_i^n| \\ &\leq \max_i |u_i^n|, \end{aligned}$$

to show reverse inequality we can reverse the time index in the logic and will end up bounding both directions.

This symmetry confirms:

$$\max_i |u_i^{n+1}| = \max_i |u_i^n| = \max_i |u_i^0|.$$

The maximum is non-increasing and bounded across all steps.

The standard explicit scheme

Finally, we can prove the boundedness of the approximation independently of the discretization steps and hence the stability of the scheme recursively by showing the maximum principle

$$\max_i |u_i^n| \leq \max_i |u_i^0|$$

using the monotonicity property under the CFL condition.
Indeed, we have

$$\begin{aligned} u_i^{n+1} &= u_i^n(1 - 2\frac{\Delta t}{\Delta x^2}) + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n) \\ &\leq \max_i |u_i^n|(1 - 2\frac{\Delta t}{\Delta x^2}) + 2\frac{\Delta t}{\Delta x^2} \max_i |u_i^n| \\ &\leq \max_i |u_i^n|, \end{aligned}$$

to show reverse inequality we can reverse the time index in the logic and will end up bounding both directions.
This symmetry confirms:

$$\max_i |u_i^{n+1}| = \max_i |u_i^n| = \max_i |u_i^0|.$$

The maximum is non-increasing and bounded across all steps.

The standard explicit scheme

Finally, we can prove the boundedness of the approximation independently of the discretization steps and hence the stability of the scheme recursively by showing the maximum principle

$$\max_i |u_i^n| \leq \max_i |u_i^0|$$

using the monotonicity property under the CFL condition.
Indeed, we have

$$\begin{aligned} u_i^{n+1} &= u_i^n(1 - 2\frac{\Delta t}{\Delta x^2}) + \frac{\Delta t}{\Delta x^2}(u_{i+1}^n + u_{i-1}^n) \\ &\leq \max_i |u_i^n|(1 - 2\frac{\Delta t}{\Delta x^2}) + 2\frac{\Delta t}{\Delta x^2} \max_i |u_i^n| \\ &\leq \max_i |u_i^n|, \end{aligned}$$

to show reverse inequality we can reverse the time index in the logic and will end up bounding both directions.
This symmetry confirms:

$$\max_i |u_i^{n+1}| = \max_i |u_i^n| = \max_i |u_i^0|.$$

The maximum is non-increasing and bounded across all steps.

The standard implicit scheme

For many financial applications, the explicit scheme turns out to be very inaccurate because the CFL condition forces the time step to be so small that the rounding error dominates the total computational error (computational error=rounding error+truncation error). Most of the time, an implicit scheme is preferred because it is unconditionally convergent, regardless of the size of the time step.

We now evaluate the second derivative at time $(n+1)\Delta t$ instead of time $n\Delta t$,

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2}.$$

We also need to initialize this scheme by setting

$$u_i^0 = u_0(i\Delta x).$$

Implementing an algorithm allowing to compute the approximation is less obvious here. This discrete equation may be converted into a linear system of equations and the algorithm will then consist in inverting a tridiagonal matrix. The truncation errors for smooth functions are the same as for the explicit scheme and the consistency follows from this analysis.

The standard implicit scheme

For many financial applications, the explicit scheme turns out to be very inaccurate because the CFL condition forces the time step to be so small that the rounding error dominates the total computational error (computational error=rounding error+truncation error). Most of the time, an implicit scheme is preferred because it is unconditionally convergent, regardless of the size of the time step.

We now evaluate the second derivative at time $(n+1)\Delta t$ instead of time $n\Delta t$,

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2}.$$

We also need to initialize this scheme by setting

$$u_i^0 = u_0(i\Delta x).$$

Implementing an algorithm allowing to compute the approximation is less obvious here. This discrete equation may be converted into a linear system of equations and the algorithm will then consist in inverting a tridiagonal matrix. The truncation errors for smooth functions are the same as for the explicit scheme and the consistency follows from this analysis.

The standard implicit scheme

For many financial applications, the explicit scheme turns out to be very inaccurate because the CFL condition forces the time step to be so small that the rounding error dominates the total computational error (computational error=rounding error+truncation error). Most of the time, an implicit scheme is preferred because it is unconditionally convergent, regardless of the size of the time step.

We now evaluate the second derivative at time $(n+1)\Delta t$ instead of time $n\Delta t$,

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2}.$$

We also need to initialize this scheme by setting

$$u_i^0 = u_0(i\Delta x).$$

Implementing an algorithm allowing to compute the approximation is less obvious here. This discrete equation may be converted into a linear system of equations and the algorithm will then consist in inverting a tridiagonal matrix. The truncation errors for smooth functions are the same as for the explicit scheme and the consistency follows from this analysis.

The standard implicit scheme

It turns out that, for any choice of the time step, the implicit scheme is monotone. In order to verify that claim, let us rewrite the implicit scheme using the notation S:

$$S(\Delta t, \Delta x, (n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^{n+1}, u_i^n, u_{i+1}^{n+1}]) =$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2}.$$

Since S is decreasing in u_i^n, u_{i+1}^{n+1} and u_{i-1}^{n+1} the implicit scheme is unconditionally monotone.

Besides, the implicit scheme is clearly consistent with the heat equation and is stable because, like the explicit scheme, it satisfies the maximum principle

$$\max_i |u_i^n| \leq \max_i |u_i^0|.$$

The standard implicit scheme

It turns out that, for any choice of the time step, the implicit scheme is monotone. In order to verify that claim, let us rewrite the implicit scheme using the notation S:

$$S(\Delta t, \Delta x, (n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^{n+1}, u_i^n, u_{i+1}^{n+1}]) =$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} - \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2}.$$

Since S is decreasing in u_i^n, u_{i+1}^{n+1} and u_{i-1}^{n+1} the implicit scheme is unconditionally monotone.

Besides, the implicit scheme is clearly consistent with the heat equation and is stable because, like the explicit scheme, it satisfies the maximum principle

$$\max_i |u_i^n| \leq \max_i |u_i^0|.$$

The standard implicit scheme

One may also consider alternately the so-called Crank-Nicolson scheme

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \theta \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2} + (1-\theta)\frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Observe that the choice $\theta = 0$ leads to the explicit scheme whereas the $\theta = 1$ yields the fully implicit scheme. For $\theta = 0.5$, the truncation error is of second order both in time and space. The condition for monotonicity and stability is here

$$\Delta t \leq \frac{\Delta x^2}{2(1-\theta)}$$

and in particular for $\theta = 0.5$ it becomes

$$\Delta t \leq \Delta x^2.$$

The standard implicit scheme

One may also consider alternately the so-called Crank-Nicolson scheme

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \theta \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2} + (1-\theta)\frac{u_{i+1}^n + u_{i-1}^n - 2u_i^n}{\Delta x^2}.$$

Observe that the choice $\theta = 0$ leads to the explicit scheme whereas the $\theta = 1$ yields the fully implicit scheme. For $\theta = 0.5$, the truncation error is of second order both in time and space. The condition for monotonicity and stability is here

$$\Delta t \leq \frac{\Delta x^2}{2(1-\theta)}$$

and in particular for $\theta = 0.5$ it becomes

$$\Delta t \leq \Delta x^2.$$

The Black-Scholes-Merton PDE

The price of a European call $u(t, x)$ for a time to expiration equal to t and for an asset value of x dollars at time $T - t$, satisfies the degenerate linear PDE

$$u_t + ru - \frac{1}{2}\sigma^2 x^2 u_{xx} - rxu_x = 0 \text{ in } (0, T] \times [0, +\infty)$$

$$u(0, x) = (x - K)^+$$

where $T > 0$ is the maturity, μ is the drift of the underlying asset and $\sigma > 0$ is its volatility.

Note that, in this example, the solution is no longer bounded but is nonnegative and grows at most linearly for large values of x. The Barles-Souganidis [5] framework can be slightly modified to accommodate the linear growth of the value function at infinity. One can also get rid of the degeneracy in the above Black-Scholes PDE by making the change of variable $y = \log s$.

The Black-Scholes-Merton PDE is linear and its Elliptic operator is degenerate. The first derivative u_x can be easily approximated in a monotone way using a forward Finite Difference

$$-rxu_x \approx -rx_i \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}.$$

The Black-Scholes-Merton PDE

The price of a European call $u(t, x)$ for a time to expiration equal to t and for an asset value of x dollars at time $T - t$, satisfies the degenerate linear PDE

$$u_t + ru - \frac{1}{2}\sigma^2 x^2 u_{xx} - rxu_x = 0 \text{ in } (0, T] \times [0, +\infty)$$

$$u(0, x) = (x - K)^+$$

where $T > 0$ is the maturity, μ is the drift of the underlying asset and $\sigma > 0$ is its volatility.

Note that, in this example, the solution is no longer bounded but is nonnegative and grows at most linearly for large values of x. The Barles-Souganidis [5] framework can be slightly modified to accommodate the linear growth of the value function at infinity.
One can also get rid of the degeneracy in the above Black-Scholes PDE by making the change of variable $y = \log s$.
The Black-Scholes-Merton PDE is linear and its Elliptic operator is degenerate. The first derivative u_x can be easily approximated in a monotone way using a forward Finite Difference

$$-rxu_x \approx -rx_i \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}.$$

The Black-Scholes-Merton PDE

The Black-Scholes-Merton PDE is linear and its Elliptic operator is degenerate. The first derivative u_x can be easily approximated in a monotone way using a forward Finite Difference

$$-rxu_x \approx -rx_i \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}.$$

One can, for instance, implement the implicit scheme

$$S(\Delta t, \Delta x,(n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^{n+1}, u_i^n, u_{i+1}^{n+1}]) \\ = \frac{u_i^{n+1} - u_i^n}{\Delta t} + ru_i^{n+1} - \frac{1}{2}\sigma^2(i\Delta x)^2 \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2} - ri\Delta x \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}.$$

The consistency is straightforward, each finite difference approximates its derivative to leading order. The monotonicity is unconditional and the standard stability requirement can be replaced by the following condition to accommodate the linear growth of the solution and its approximation at $+\infty$.

Stability For every h, the scheme has a solution u_h which is nonnegative and grows at most linearly.

The Black-Scholes-Merton PDE

The Black-Scholes-Merton PDE is linear and its Elliptic operator is degenerate. The first derivative u_x can be easily approximated in a monotone way using a forward Finite Difference

$$-rxu_x \approx -rx_i \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}.$$

One can, for instance, implement the implicit scheme

$$\begin{aligned} S(\Delta t, \Delta x, &(n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^{n+1}, u_i^n, u_{i+1}^{n+1}]) \\ =& \frac{u_i^{n+1} - u_i^n}{\Delta t} + ru_i^{n+1} - \frac{1}{2}\sigma^2 (i\Delta x)^2 \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2} - ri\Delta x \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}. \end{aligned}$$

The consistency is straightforward, each finite difference approximates its derivative to leading order. The monotonicity is unconditional and the standard stability requirement can be replaced by the following condition to accommodate the linear growth of the solution and its approximation at $+\infty$.

Stability For every h, the scheme has a solution u_h which is nonnegative and grows at most linearly.

The Black-Scholes-Merton PDE

The Black-Scholes-Merton PDE is linear and its Elliptic operator is degenerate. The first derivative u_x can be easily approximated in a monotone way using a forward Finite Difference

$$-rxu_x \approx -rx_i \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}.$$

One can, for instance, implement the implicit scheme

$$\begin{aligned} S(\Delta t, \Delta x, &(n+1)\Delta t, i\Delta x, u_i^{n+1}, [u_{i-1}^{n+1}, u_i^n, u_{i+1}^{n+1}]) \\ &= \frac{u_i^{n+1} - u_i^n}{\Delta t} + ru_i^{n+1} - \frac{1}{2}\sigma^2 (i\Delta x)^2 \frac{u_{i+1}^{n+1} + u_{i-1}^{n+1} - 2u_i^{n+1}}{\Delta x^2} - ri\Delta x \frac{u_{i+1}^{n+1} - u_i^{n+1}}{\Delta x}. \end{aligned}$$

The consistency is straightforward, each finite difference approximates its derivative to leading order. The monotonicity is unconditional and the standard stability requirement can be replaced by the following condition to accommodate the linear growth of the solution and its approximation at $+\infty$.

Stability For every h, the scheme has a solution u_h which is nonnegative and grows at most linearly.