

TP #1

Etudiant : Roussel Desmond Nzoyem

UE : EDP 2 – Enseignant : Pr. Philippe Helluy

Date : 3 janvier 2021

1. Résolution numérique de l'équation de transport

Question 1.

Calculer la solution du problème suivant (c est une constante > 0).

$$\begin{cases} \partial_t u + c \partial_x u = 0, & x \in]0, L[, \quad t \in]0, T[\\ u(0, t) = e^{-t}, & t \in]0, T[\\ u(x, 0) = 0, & x \in]0, L[\end{cases}$$

Réponse

Nous procéderons par la méthode des courbes caractéristiques. Une courbe caractéristique est une courbe $t \mapsto \begin{pmatrix} x(t) \\ t \end{pmatrix}$ telle que la solution $u(x(t), t)$ y soit constante. Pour l'équation de transport à résoudre, les courbes caractéristiques sont des droites du plan de coefficient directeur $c > 0$. Calculons la solution du problème.

■ Dans un premier temps, pour tout $y \in \mathbb{R}$, on a :

$$\frac{d}{dt} u(y + ct, t) = \left(\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} \right) (y + ct, t) = 0, \quad 0 < y + ct < L, \quad 0 < t < T$$

C'est à dire

$$u(y + ct, t) = \text{Cste}, \quad 0 < y + ct < L, \quad 0 < t < T$$

On choisit la constante correspondant au cas limite $t = 0$

$$u(y + ct, t) = u(y, 0)$$

sous les conditions suivantes

$$\begin{aligned} & - \begin{cases} 0 < y + ct < L \\ 0 < y < L \end{cases} \implies ct < y + ct < L \\ & - 0 < t < T \end{aligned}$$

On effectue le changement de variable $y + ct = x$ pour obtenir la solution

$$u(x, t) = 0$$

définie sur l'ensemble :

$$\{(x, t) \in]0, L[\times]0, T[\mid \text{tel que } ct < x\}$$

■ D'autre part, pour tout $t' > 0$, on a :

$$\frac{d}{ds}u(cs, t' + s) = \left(\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} \right) (cs, t' + s) = 0, \quad 0 < cs < L, \quad 0 < t' + s < T$$

C'est à dire

$$u(cs, t' + s) = \text{Cste}, \quad 0 < cs < L, \quad 0 < t' + s < T$$

On choisi la constante correspondant à $s = 0$:

$$u(cs, t' + s) = u(0, t'), \quad 0 < cs < L, \quad s < t' + s < T$$

On effectue le changement de variable

$$\begin{cases} x = cs \\ t = t' + s \end{cases} \implies t' = t - \frac{x}{c}$$

Ceci permet d'écrire

$$u(x, t) = u(0, t - \frac{x}{c}) = e^{-t + \frac{x}{c}}$$

Sachant que $t' > 0$ (car $t' + s > s$), la solution donnée est définie sur :

$$\left\{ (x, t) \in]0, L[\times]0, T[\text{ tel que } \frac{x}{c} < t \right\}$$

On récapitule en écrivant la solution de l'équation de transport

$$\text{Pour } x \in]0, L[\text{ et } t \in]0, T[, \quad u(x, t) = \begin{cases} 0 & \text{si } ct < x \\ e^{-t + \frac{x}{c}} & \text{si } ct > x \end{cases}$$

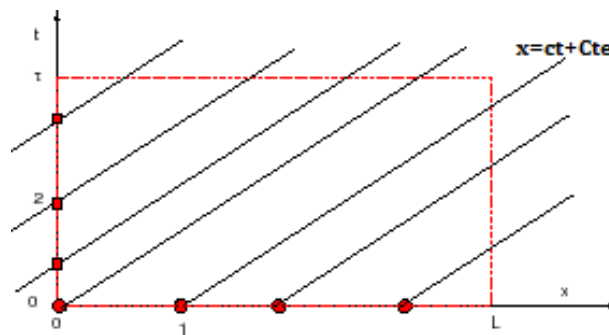


FIGURE 1 – Illustration de la solution de l'équation de Transport dans l'espace-temps à l'aide des courbes caractéristiques.

Question 2.

Montrer que le problème n'admet pas de solution lorsque $c < 0$.

Réponse

Procédons par l'absurde et supposons que le problème admette une solution $u(\cdot, \cdot)$.

Soit $x \in]0, L[$ et $t \in]0, T[$. Pour tout $s \in \mathbb{R}$, on a :

$$\frac{d}{ds}u(x + cs, t + s) = \left(\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} \right) (x + cs, t + s) = 0$$

Avec s t.q. $\begin{cases} 0 < x + cs < L \\ 0 < t + s < T \end{cases} \Rightarrow \begin{cases} -L < cs < L \\ -T < s < T \end{cases} \Rightarrow s \in K =] -\min(T, -\frac{L}{c}), \min(T, -\frac{L}{c})[$

On obtient donc :

$$\forall s \in K \quad u(x + cs, t + s) = Cste \quad (1)$$

En particulier, pour $s = 0 \in K$, on a $u(x + cs, t + s) = u(x, t)$

■ Supposons $-\frac{L}{c} < T$ i.e. $K =]\frac{L}{c}, -\frac{L}{c}[$; alors en prenant $s = -\frac{x}{c} \in K$, on a d'après (1) :

$$u(x + cs, t + s) = u(x, t) = u(0, t - \frac{x}{c})$$

On constate déjà que la condition sur la borne gauche du domaine permet de définir entièrement l'évolution de u (sans tenir compte de la condition initiale). On peut relier les deux conditions aux limites en posant $x = -ct'$ et $t = 0$:

— D'une part $u(x, t) = u(0, t - \frac{x}{c}) = u(0, t') = e^{-t'}$

— D'autre part $u(x, t) = u(-ct', 0) = 0$

Ceci pour tout $t' \in]0, -\frac{L}{c}[$. Ce qui est absurde car $e^{-t'} \neq 0$ pour tout $t' \in \mathbb{R}$.

■ Supposons au contraire $-\frac{L}{c} \geq T$ i.e. $K =]-T, T[$; on prends alors $s = -t \in K$, ce qui donne (d'après l'équation (1)) :

$$u(x + cs, t + s) = u(x, t) = u(x - ct, 0)$$

Posons $x = 0$ et $t = -\frac{x'}{c}$:

— D'une part $u(x, t) = u(x - ct, 0) = u(x', 0) = 0$

— D'autre part $u(x, t) = u(0, -\frac{x'}{c}) = e^{\frac{x'}{c}}$

Ceci pour tout $x' \in]0, -cT[$. Ce qui est absurde !

Dans les deux cas, on constate que le problème n'admet pas de solution. D'un point de vue physique, ce résultat était attendu. Ceci car pour $c < 0$, le transport de la quantité u se fait de la droite vers la gauche. Au lieu d'imposer la condition sur le bord gauche ($x = 0$), il aurait donc fallu fournir une condition sur le bord droit du domaine ($x = L$) pour que l'équation de transport admette une solution.

Question 3.

Écrire un programme qui résout l'équation de transport par la méthode de Godunov.

Réponse

Pour programmer la méthode de Godunov, il nous faut introduire certaines fonctions. Pour l'équation de transport, on les définit de la manière suivante :

— Le flux physique :

```
void fluxphy(double* w, double* flux) { flux[0] = _C * w[0]; }
```

— La vitesse maximale sur chaque cellule :

```
double lambda_max(double* u) { return _C; }
```

— La solution du problème de Riemann associé :

```
void riemann(double* a, double* b, double z, double* w) {
    if (z < _C) { // z = x/t
        w[0] = a[0];
    } else {
        w[0] = b[0];
    }
}
```

— Une solution exacte du problème : utilisé comme condition initiale ($t = 0$), et comme condition aux bords ($x = 0$ et $x = L$) :

```
void solexacte(double x, double t, double* w) {
    double uR = 0;
    double uL = exp(-(t - x / _C));

    if (x < _C * t) {
        w[0] = uL;
    } else {
        w[0] = uR;
    }
}
```

— La fonction de résolution du problème, utilisable tel quel pour d'autres problèmes

```
void godunov_solve(godunov* gd, double tmax) {
    double tnow = 0;
    int m = gd->m;
    while (tnow < tmax) {
        double vmax = 0;
        // Calcul de la vitesse max
        for (int i = 0; i < gd->N + 2; i++) {
            double vloc = lambda_max(gd->un + m * i);
            vmax = vmax > vloc ? vmax : vloc;
        }
        // Calcul du pas de temps
        gd->dt = gd->cfl * gd->dx / vmax;
        // Application du flux
        for (int i = 1; i < gd->N + 1; i++) {
            double flux[m];
            fluxnum(gd->un + i * m, gd->un + (i + 1) * m, flux);
            for (int iv = 0; iv < m; iv++) {
                gd->unp1[i * m + iv] =
                    gd->un[i * m + iv] - gd->dt / gd->dx * flux[iv];
            }
            fluxnum(gd->un + (i - 1) * m, gd->un + i * m, flux);
            for (int iv = 0; iv < m; iv++) {
                gd->unp1[i * m + iv] += gd->dt / gd->dx * flux[iv];
            }
        }
        // Mise à jour
        tnow += gd->dt;
        // Conditions aux limites
        int i = 0;
        solexacte(gd->xi[i], tnow, gd->unp1 + i * m);
        i = gd->N + 1;
        solexacte(gd->xi[i], tnow, gd->unp1 + i * m);
        // Preparation pour l'iteration suivante
        memcpy(gd->un, gd->unp1, (gd->N + 2) * m * sizeof(double));
    }
    gd->tfin = tnow;
}
```

On choisit de résoudre le problème avec $L = 2$, $T = 0.7$, $c = 1$ et cfl ¹. Pour différentes taille de maillage ($N = 50, 1000, 50000$), on obtient les résultats suivants :

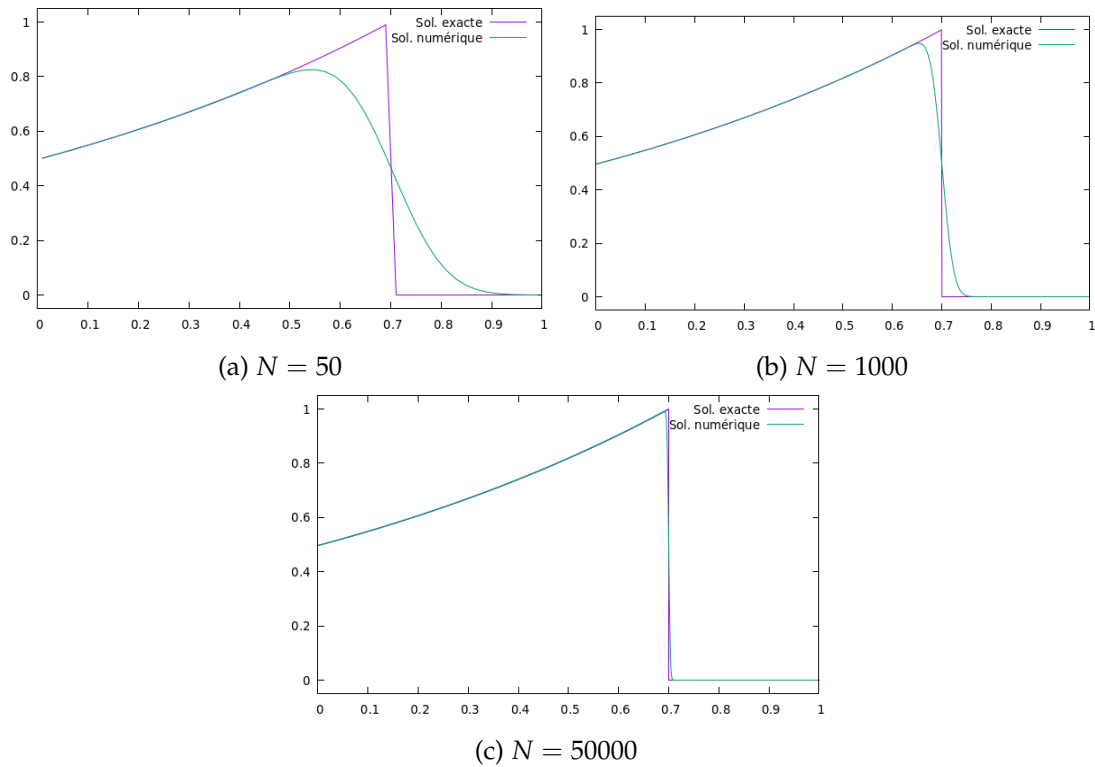


FIGURE 2 – Quelques résultats obtenus pour l'équation de transport au temps final $T = 0.7$.

Question 4.

Étude du taux de convergence. Vérifions la validité de notre programme sur l'équation de transport en traçant le logarithme de l'erreur en norme L^1 entre la solution exacte à l'instant T et la solution numérique en fonction de $\ln(\Delta x)$ où Δx est le pas de la subdivision.

Réponse

On obtient le tableau de convergence illustré ci-bas (tableau 1); son illustration est donnée à la figure 3.

1. Les détails sur le coefficient cfl sont donnés à la question 5.

N	Δx	Erreur L^1
10	0.100000	0.140029
40	0.025000	0.074383
160	0.006250	0.037323
640	0.001563	0.018697
2560	0.000391	0.009342
10240	0.000098	0.004669

TABLE 1 – Étude de convergence pour l'équation de transport

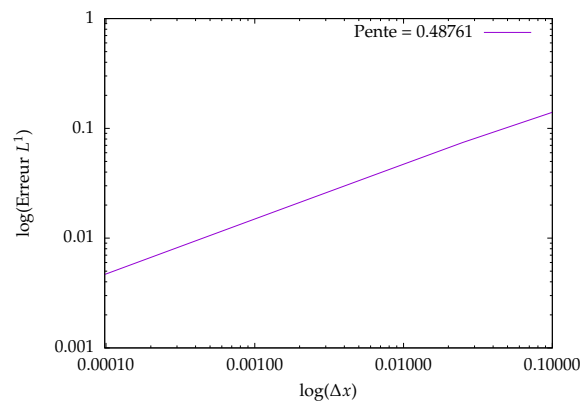


FIGURE 3 – Illustration de l'étude de convergence pour l'équation de transport

Le taux (ou l'ordre) de convergence estimé est donné soit par la pente de la courbe de la figure 3, soit par la formule

$$\text{ordre} \approx \frac{\log\left(\frac{\text{Err}(N)}{\text{Err}(2N)}\right)}{\log(2)}.$$

Un calcul avec $N = 5000$ nous donne un ordre de convergence proche de **0.50**. Ceci indique que le schéma de Godunov converge avec une vitesse de l'ordre de \sqrt{N} ².

Question 5.

Vérifions numériquement que le schéma devient instable lorsque la condition de CFL n'est pas satisfaite. Traçons un exemple lorsque $c\Delta t/\Delta x > 1$ et proche de 1.

Réponse

Comme précédemment, on prend $L = 2$, $T = 0.7$, $c = 1$ et $N = 1000$. Le coefficient cfl tel que $\Delta t = cfl \times \frac{\Delta x}{c}$ est pris égal 1.5, 1.01 et 0.999.

2. Cet ordre de convergence est indiqué pour les conditions aux limites et initiales imposées par le problème.

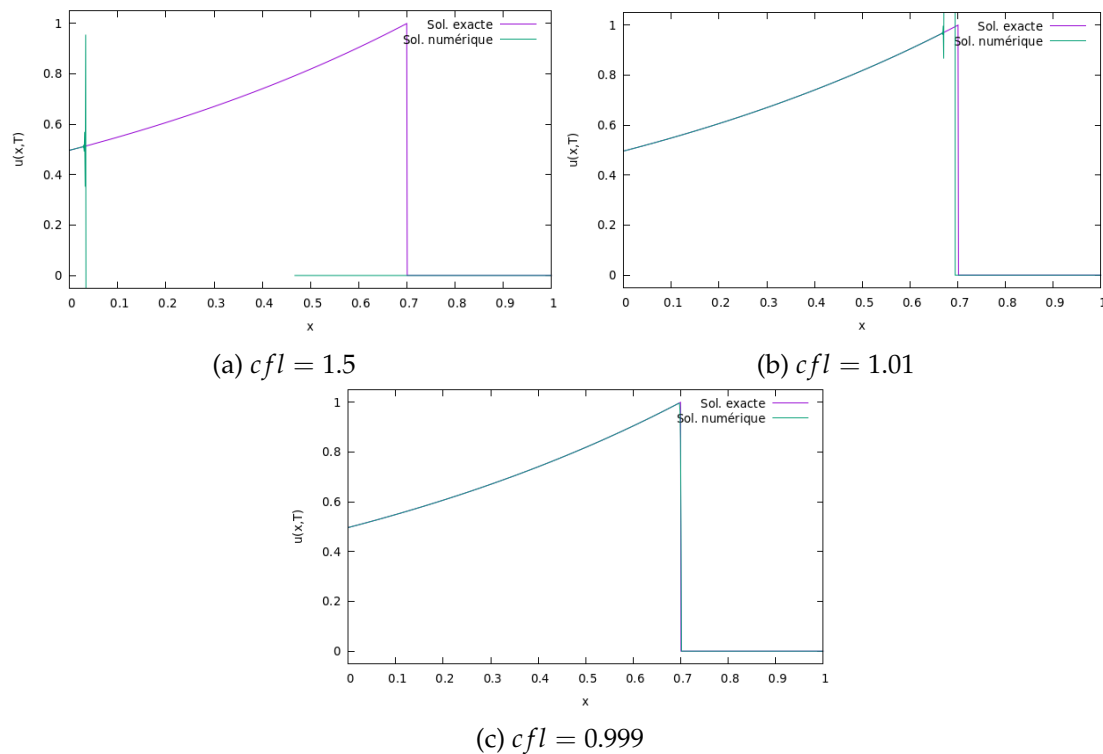


FIGURE 4 – Vérification numérique de l'instabilité de l'équation de transport

On vérifie ainsi que le schéma est instable lorsque la condition CFL n'est pas satisfaite (figures 4a et 4b). Par contre, lorsque la condition est vérifiée et le coefficient cfl est très proche de 1, la solution obtenue est considérablement meilleure qu'avec un coefficient cfl plus bas, pour la même taille de maillage ($N = 1000$). On peut observer ceci en comparant les figures 4c (où $cfl = 0.999$) et 2b (où $cfl = 0.5$).

2. Résolution numérique de l'équation de Burgers

Question 1.

Calculer la solution de Lax du problème suivant.

$$\begin{aligned} \partial_t u + \partial_x (u^2/2) &= 0, & x \in]-1, 2[, & t > 0 \\ u(0, t) &= 1 \\ u(x, 0) &= \begin{cases} 1 & \text{si } x < 0 \\ 1 - x & \text{si } 0 \leq x \leq 1 \\ 0 & \text{si } x > 1 \end{cases} \end{aligned}$$

Réponse

Ce problème se réécrit sous la forme $\partial_t u + \partial_x f(u) = 0$, avec $f : u \mapsto \frac{u^2}{2}$.

Considérons la courbe caractéristique $t \mapsto (x(t))$ sur laquelle la solution u est constante i.e. $u(x, t) = u(x(t), t) = \text{Cste}$.

Ceci donne

$$\frac{d}{dt}u(x(t), t) = \left(\frac{\partial u}{\partial t} + x'(t) \frac{\partial u}{\partial x} \right) (x(t), t) = 0$$

Par comparaison avec l'équation de Burgers qui s'écrit encore sous la forme

$$\left(\frac{\partial u}{\partial t} + u(x(t), t) \frac{\partial u}{\partial x} \right) (x(t), t) = 0$$

on déduit que suivant les courbes caractéristiques, on a $x'(t) = u(x(t), t) = \text{Cste}$. On peut donc écrire

$$x'(t) = u(x(0), 0)$$

Distinguons à présent les différents cas possibles d'abscisses à l'origine de la caractéristique :

— Si $x(0) < 0$, alors

$$x'(t) = u(x(0), 0) = 1 \implies x(t) = t + x(0) \implies x(0) = x(t) - t$$

Il s'en suit que pour $x - t < 0$, on a

$$\begin{aligned} u(x, t) &= u(x(t), t) = \text{Cste} \\ &= u(x(0), 0) \\ &= 1 \end{aligned}$$

— Si $0 \leq x(0) \leq 1$, alors

$$x'(t) = u(x(0), 0) = 1 - x(0) \implies x(t) = (1 - x(0))t + x(0) \implies x(0) = \frac{x(t) - t}{1 - t}$$

Il s'en suit que pour $0 \leq \frac{x - t}{1 - t} \leq 1$, on a

$$\begin{aligned} u(x, t) &= u(x(t), t) \\ &= u(x(0), 0) \\ &= 1 - \frac{x - t}{1 - t} \\ &= \frac{1 - x}{1 - t} \end{aligned}$$

— Si $x(0) > 1$, alors

$$x'(t) = u(x(0), 0) = 0 \implies x(t) = x(0)$$

Il s'en suit que pour $x > 1$, on a

$$\begin{aligned} u(x, t) &= u(x(t), t) \\ &= u(x(0), 0) \\ &= 0 \end{aligned}$$

On obtient donc la solution définie sur $] -1, 2[\times] 0, 1[$ par :

$$u(x, t) = \begin{cases} 1 & \text{si } x < t \\ \frac{1 - x}{1 - t} & \text{si } 0 \leq \frac{x - t}{1 - t} \leq 1 \\ 0 & \text{si } x > 1 \end{cases}$$

La fonction trouvée est bien définie et continue lorsque $t < 1$. Cependant, lorsque $t \geq 1$, les caractéristiques correspondants aux trois cas se croisent. Notre solution sera alors une solution faible si la pente σ de la courbe de discontinuité vérifie la relation de Rankine-Hugoniot (voir figure 5) :

$$f(u_L) - f(u_R) = \sigma(u_L - u_R) \implies \sigma = \frac{u_L + u_R}{2} = \frac{1+0}{2} = \frac{1}{2}$$

La vitesse de choc σ vérifie aussi le critère de Lax car

$$f'(u_L) = 1 > \sigma = \frac{1}{2} > 0 = f'(u_R).$$

En conclusion, la solution au sens de Lax est donnée par :

■ Lorsque $t < 1$, alors :

$$u(x, t) = \begin{cases} 1 & \text{si } x < t \\ \frac{1-x}{1-t} & \text{si } 0 \leq \frac{x-t}{1-t} \leq 1 \\ 0 & \text{si } x > 1 \end{cases}$$

■ Lorsque $t \geq 1$, alors :

$$u(x, t) = \begin{cases} 1 & \text{si } x < \frac{1}{2}t \\ 0 & \text{si } x > \frac{1}{2}t \end{cases}$$

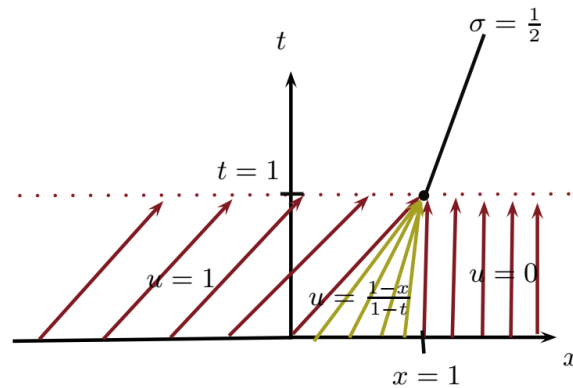


FIGURE 5 – Illustration de la solution de Lax du problème de Burgers dans l'espace-temps à l'aide des courbes caractéristiques.

Question 2.

Résoudre le problème de Riemann pour l'équation de Burgers.

$$\partial_t u + \partial_x (u^2/2) = 0 \quad (2)$$

$$u(x, 0) = \begin{cases} u_L & \text{si } x < 0 \\ u_R & \text{si } x > 0 \end{cases}$$

Réponse

Comme précédemment, les courbes caractéristiques sont les telle que $x'(t) = u(x(0), 0)$. On traite tout d'abord les deux cas suivant de façon indépendante :

— Si $x(0) < 0$ alors

$$x'(t) = u_L \Rightarrow x(t) = u_L t + x(0)$$

On obtient donc

$$u(x, t) = u_L \text{ pour } (x, t) \in D_L, \text{ avec } D_L = \left\{ (x, t) \in]-1, 2[\times \mathbb{R}_+^* \text{ t.q. } \frac{x}{t} < u_L \right\}$$

— Si $x(0) > 0$ alors

$$x'(t) = u_R \Rightarrow x(t) = u_R t + x(0)$$

On obtient donc

$$u(x, t) = u_R \text{ pour } (x, t) \in D_R, \text{ avec } D_R = \left\{ (x, t) \in]-1, 2[\times \mathbb{R}_+^* \text{ t.q. } \frac{x}{t} > u_R \right\}$$

Distinguons maintenant deux cas :

- Si $u_L > u_R$ alors les domaines D_L et D_R sont sécants dans l'espace temps; La solution présente donc une discontinuité dont la pente σ est donnée par la relation de Rankine-Hugoniot.

$$f(u_L) - f(u_R) = \sigma(u_L - u_R) \Rightarrow \sigma = \frac{u_L + u_R}{2}$$

On a donc obtenue une solution faible

$$u(x, t) = \begin{cases} u_L & \text{si } x < \sigma t \\ u_R & \text{si } x > \sigma t \end{cases}$$

Cette solution vérifie bien le critère de Lax car $f'(u_L) = u_L > \sigma = \frac{u_L + u_R}{2} > u_R = f'(u_R)$.

- Si au contraire $u_L < u_R$, les domaines D_L et D_R sont disjoints dans l'espace temps; la solution faible donnée dans le cas $u_L > u_R$ reste applicable, mais ne vérifie plus le critère de Lax.

On construit donc une solution acceptable en exprimant u sur

$$D_{\text{milieu}} = D_C = \left\{ (x, t) \in]-1, 2[\times \mathbb{R}_+^* \text{ t.q. } u_L < \frac{x}{t} < u_R \right\}.$$

On remarque d'entrée que si $u(x, t)$ est solution de l'équation (2), alors $u(\lambda x, \lambda t)$ est aussi solution (pour $\lambda \in \mathbb{R}$). On est donc amené à croire que la solution ne dépend que du rapport $\xi = x/t$ (i.e ξ est autosimilaire). Posons $u(x, t) = v(\xi)$. On remplace cela dans l'équation (2) pour trouver :

$$\left(\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} \right) (x, t) = \left(\frac{\partial v}{\partial t} + \frac{\partial f(v)}{\partial x} \right) (\xi) = 0$$

On développe le terme du milieu pour obtenir :

$$\begin{aligned} \left(\frac{\partial v}{\partial t} + \frac{\partial f(v)}{\partial x} \right) (\xi) &= \frac{\partial v}{\partial t}(\xi) + f'(v(\xi)) \frac{\partial v}{\partial x}(\xi) \\ &= \frac{\partial v}{\partial \xi}(\xi) \frac{\partial \xi}{\partial t}(t) + v(\xi) \frac{\partial v}{\partial \xi}(\xi) \frac{\partial \xi}{\partial x}(x) \\ &= \left(-\frac{x}{t^2} + \frac{v(\xi)}{t} \right) v'(\xi) \\ &= 0 \end{aligned}$$

On remarque que lorsque $v'(\xi) \neq 0$, l'expression de u sur D_C peut être obtenue par ³

$$-\frac{x}{t^2} + \frac{v(\xi)}{t} \implies v(\xi) = u(x, t) = \frac{x}{t}$$

On est donc amené à considérer la fonction suivante comme candidat pour le problème de Riemann (2).

$$u(x, t) = \begin{cases} u_L & \text{sur } D_L \\ \frac{x}{t} & \text{sur } D_C \\ u_R & \text{sur } D_R \end{cases}$$

Par construction, u est une solution forte de (2) sur $D_L \cup D_C \cup D_R$. En plus, elle est continue au niveau des zones de transition $\frac{x}{t} = u_L$ et $\frac{x}{t} = u_R$. Elle vérifie donc le critère de Lax.

En conclusion, la solution du problème de Riemann pour le problème de Burgers est donnée par (voir figure 6) :

■ Si $u_L < u_R$, alors la solution est l'onde de choc

$$u(x, t) = \begin{cases} u_L & \text{si } \frac{x}{t} < \frac{u_L + u_R}{2} \\ u_R & \text{si } \frac{x}{t} \geq \frac{u_L + u_R}{2} \end{cases}$$

■ Si $u_L \geq u_R$, alors la solution est l'onde dite de raréfaction

$$u(x, t) = \begin{cases} u_L & \text{si } \frac{x}{t} < u_L \\ u_R & \text{si } \frac{x}{t} > u_R \\ \frac{x}{t} & \text{sinon} \end{cases}$$

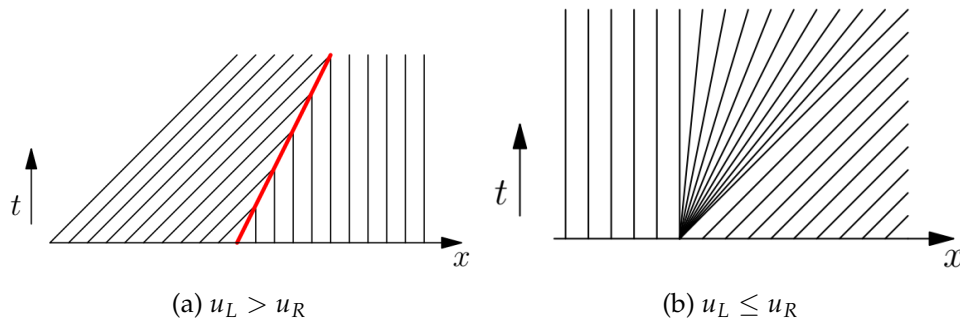


FIGURE 6 – Illustration de la solution du problème de Riemann à l'aide des courbes caractéristiques

Question 3.

- Écrire un programme qui résout le problème (2) par la méthode de Godunov.
- Comparer la solution exacte et la solution approchée aux instants $t = 1/2; t = 1; t = 2$ pour diverses finesses de maillage Δx .
- Effectuer une étude de taux de convergence.

3. En réalité, $v'(\xi)$ doit être non nul pour avoir des courbes caractéristiques non parallèles et ainsi accomplir la transition de u_L à u_R

Réponse

Le code de calcul pour la phase de résolution de l'équation de Burgers par la méthode de Godunov est rigoureusement identique à celui utilisé pour l'équation de transport. Cependant, quelques fonctions doivent être adaptées; elles sont toutes données ci-bas.

```
void fluxphy(double* w, double* flux) { flux[0] = w[0] * w[0] / 2; }

double lambda_max(double* u) { return fabs(u[0]); }

void riemann(double* a, double* b, double z, double* w) {
    if (a[0] > b[0]) {
        // choc
        double s = (a[0] + b[0]) / 2;
        if (z < s)
            w[0] = a[0];
        else
            w[0] = b[0];
    } else {
        // onde de détente
        if (z < a[0]) {
            w[0] = a[0];
        } else if (z > b[0]) {
            w[0] = b[0];
        } else
            w[0] = z;
    }
}

void solexacte(double x, double t, double* w) {
    if (x < t) {
        w[0] = 1;
    } else if (x > 1) {
        w[0] = 0;
    } else { // if ( 0 <= (x-t)/(1-t) && (x-t)/(1-t) <= 1.)
        w[0] = (1-x)/(1-t);
    }
}
```

Listing 1 – Flux physique, vitesse maximale, solution du problème de Riemann, et solution exacte pour l'équation de Burgers

Comparons à présent la solution exacte et la solution approchée aux temps indiqués, pour trois finesses de maillages. Les figures 7 à 9 montrent clairement que les solutions sont meilleures pour des maillages raffinés. Cependant, nous avons vu à la question 1) qu'une solution forte pour ce problème n'existait que pour $t \leq 1$. Ceci explique pourquoi le schéma numérique de Godunov ne retrouve pas la solution exacte (faible) que nous avons exprimé à la question précédente; ceci lorsque $t = 2$ (figures 7c, 8c et 9c).

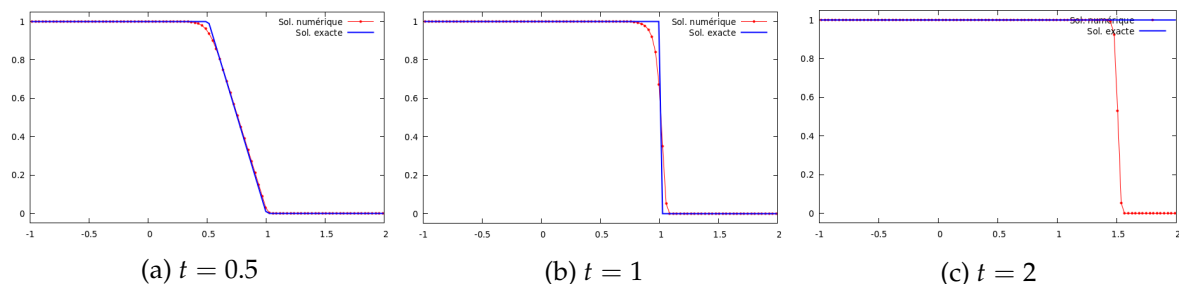


FIGURE 7 – Résolution de l'équation de Burgers par la méthode de Godunov, pour un maillage de taille $N = 100$ i.e $\Delta x = 3e - 2$.

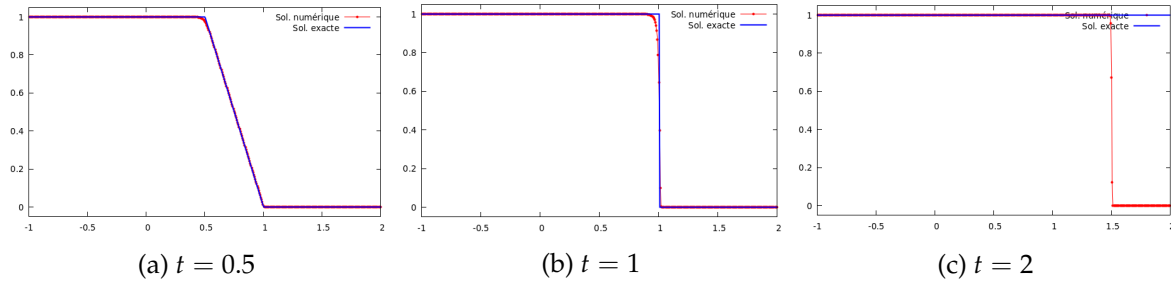


FIGURE 8 – Résolution de l'équation de Burgers par la méthode de Godunov, pour un maillage de taille $N = 500$ i.e $\Delta x = 6e - 3$.

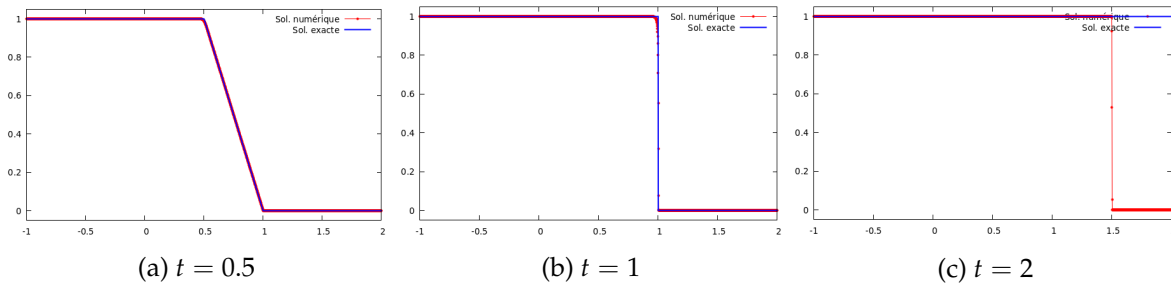


FIGURE 9 – Résolution de l'équation de Burgers par la méthode de Godunov, pour un maillage de taille $N = 2500$ i.e $\Delta x = 1.2e - 3$.

Effectuons à présent une étude du taux de convergence, pour les mêmes temps de simulation que précédemment. On obtient la figure 10 qui montre que la meilleure convergence est obtenue pour $t = 0.5$, avec un ordre de convergence d'environ 0.99. Ensuite vient le cas $t = 1$, avec un ordre de convergence proche de 0.76. Enfin, le cas $t = 2$ ne saurait converger dû à la non-existence d'une solution forte.

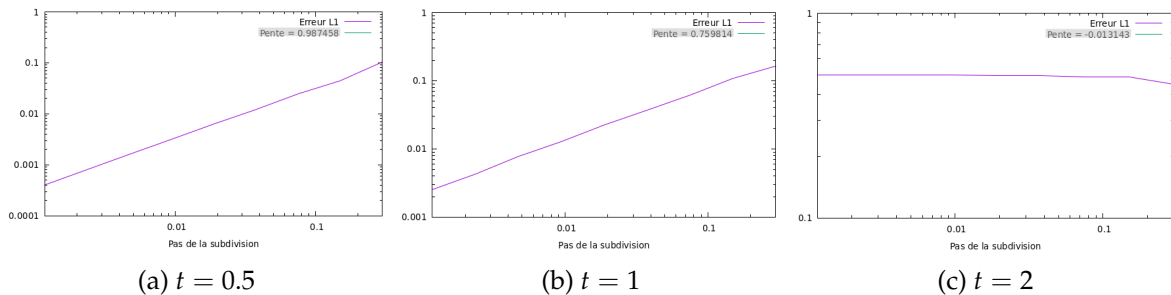


FIGURE 10 – Étude du taux de convergence pour l'équation de Burgers.

Question 4.

Décrire et programmer la correction MUSCL de van Leer. Vérifier que la précision du schéma est améliorée grâce à une étude de taux de convergence.

Réponse

Rappelons que le schéma de volumes finis en 1D s'écrit pour une maille i à l'itération n comme suit

$$\frac{w_i^{n+1} - w_i^n}{\Delta t} + \frac{F_{i+\frac{1}{2}}^n - F_{i-\frac{1}{2}}^n}{\Delta x} = 0 \quad (3)$$

où les flux numérique aux interfaces sont donnés par

$$\begin{aligned} F_{i+\frac{1}{2}}^n &= f(w_i^n, w_{i+1}^n) \\ F_{i-\frac{1}{2}}^n &= f(w_{i-1}^n, w_i^n) \end{aligned}$$

L'idée de la méthode MUSCL⁴ est de remplacer le flux numérique calculé à l'interface $i + \frac{1}{2}$ par un flux numérique calculé en des points plus proches de l'interface, et à un pas de temps correspondant. On notera ces points par $w_{i+\frac{1}{2},-}^{n+\frac{1}{2}}$ et $w_{i+\frac{1}{2},+}^{n+\frac{1}{2}}$ ⁵. On pose alors

$$\begin{aligned} w_{i+\frac{1}{2},-}^{n+\frac{1}{2}} &= w_i^n + s_i^n \frac{\Delta x}{2} + r_i^n \frac{\Delta t}{2} \\ w_{i+\frac{1}{2},+}^{n+\frac{1}{2}} &= w_{i+1}^n - s_{i+1}^n \frac{\Delta x}{2} + r_{i+1}^n \frac{\Delta t}{2} \end{aligned} \quad (4)$$

Pour le calcul des pentes en espace s_i^n pour une maille quelconque i , on pose :

$$\alpha = \frac{w_i^n - w_{i-1}^n}{\Delta x}, \quad \beta = \frac{w_{i+1}^n - w_i^n}{\Delta x}, \quad \gamma = \frac{w_{i+1}^n - w_{i-1}^n}{2\Delta x}$$

et on en déduit

$$s_i^n = \text{minmod}(\alpha, \beta, \gamma)$$

où la fonction minmod est donnée par :

$$\text{minmod}(\alpha, \beta, \gamma) = \begin{cases} \min(\alpha, \beta, \gamma) & \text{si } \alpha, \beta, \gamma > 0 \\ \max(\alpha, \beta, \gamma) & \text{si } \alpha, \beta, \gamma < 0 \\ 0 & \text{sinon} \end{cases}$$

Pour le calcul des pentes en temps, on se sert de la loi de conservation décrivant l'équation de Burgers

$$\partial_t w + \partial_x f(w) = 0$$

On prend donc la pente r_i^n telle que

$$r_i^n = -f'(w_i^n) s_i^n$$

Les pentes s_i^n et r_i^n ayant été calculées, on calcule le flux numérique aux points $w_{i+\frac{1}{2},-}^{n+\frac{1}{2}}$ et $w_{i+\frac{1}{2},+}^{n+\frac{1}{2}}$ grâce à l'équation (4), qu'on applique au schéma numérique. L'équation (3) devient alors

$$\frac{w_i^{n+1} - w_i^n}{\Delta t} + \frac{F_{i+\frac{1}{2}}^{n+\frac{1}{2}} - F_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x} = 0 \quad (5)$$

où

$$\begin{aligned} F_{i+\frac{1}{2}}^{n+\frac{1}{2}} &= f(w_{i+\frac{1}{2},-}^{n+\frac{1}{2}}, w_{i+\frac{1}{2},+}^{n+\frac{1}{2}}) \\ F_{i-\frac{1}{2}}^{n+\frac{1}{2}} &= f(w_{i-\frac{1}{2},-}^{n+\frac{1}{2}}, w_{i-\frac{1}{2},+}^{n+\frac{1}{2}}) \end{aligned}$$

En ce qui concerne l'implémentation, le code de calcul précédemment présenté⁶ doit être ajusté. En effet, nous devons rajouter les pentes lors de la résolution du système. On obtient le code de calcul pour la fonction `godunov_solve` ci-bas.

4. L'acronyme MUSCL est mis pour "Monotonic Upstream-centered Scheme for Conservation Laws". Nous décrivons ici la version de cette correction qui prend en compte à la fois les pentes en espace et en temps.

5. Notons qu'un raisonnement similaire est appliqué à l'interface $i - \frac{1}{2}$ pour calculer $w_{i-\frac{1}{2},-}^{n+\frac{1}{2}}$ et $w_{i-\frac{1}{2},+}^{n+\frac{1}{2}}$.

6. La fonction de résolution par la méthode de Godunov simple est la même pour l'équation de transport, et pour l'équation de Burgers.

```

void godunov_solve(godunov* gd, double tmax){
    double tnow = 0;
    int m = gd->m;
    while (tnow < tmax) {
        double vmax = 0;
        // calcul de la vitesse max
        for (int i = 0; i < gd->N + 2; i++) {
            double vloc = lambda_max(gd->un + m * i);
            vmax = vmax > vloc ? vmax : vloc;
        }
        // Calcul des pentes pour MUSCL
        double si[gd->N+1], ri[gd->N+1];
        for (int i = 1; i < gd->N + 1; i++) {
            double alpha = (gd->un[i] - gd->un[i-1])/gd->dx;
            double beta = (gd->un[i+1] - gd->un[i])/gd->dx;
            double gamma = (gd->un[i+1] - gd->un[i-1])/(2.0*gd->dx);
            si[i] = minmod(alpha, beta, gamma);
            ri[i] = - gd->un[i] * si[i];
        }
        gd->dt = gd->cfl * gd->dx / vmax;
        for (int i = 1; i < gd->N + 1; i++) {
            double flux[m];
            // Application de MUSCL avec la droite
            double uL[1] = {gd->un[i] + si[i]*gd->dx/2.0 + ri[i]*gd->dt/2.0};
            double uR[1] = {gd->un[i+1] - si[i+1]*gd->dx/2.0 + ri[i+1]*gd->dt/2.0};
            fluxnum(uL, uR, flux);
            for (int iv = 0; iv < m; iv++) {
                gd->unp1[i * m + iv] =
                    gd->un[i * m + iv] - gd->dt / gd->dx * flux[iv];
            }
            // Application de MUSCL avec la gauche
            uL[0] = gd->un[i-1] + si[i-1]*gd->dx/2.0 + ri[i-1]*gd->dt/2.0;
            uR[0] = gd->un[i] - si[i]*gd->dx/2.0 + ri[i]*gd->dt/2.0;
            fluxnum(uL, uR, flux);
            for (int iv = 0; iv < m; iv++) {
                gd->unp1[i * m + iv] += gd->dt / gd->dx * flux[iv];
            }
        }
        // mise à jour
        tnow += gd->dt;
        // conditions aux limites
        int i = 0;
        solexacte(gd->xi[i], tnow, gd->unp1 + i * m);
        i = gd->N + 1;
        solexacte(gd->xi[i], tnow, gd->unp1 + i * m);
        // Pour l'iteration suivante
        memcpy(gd->un, gd->unp1, (gd->N + 2) * m * sizeof(double));
    }
    gd->tfin = tnow;
}

```

Listing 2 – Fonction de résolution du problème de Burger par un schéma de Godunov, auquel on applique la correction MUSCL

Pour tester notre programme, nous reprenons l'étude faite à la question précédente. Les résultats sont présentés aux figures 11 à 13. En les comparant aux figures 7 à 9, on remarque immédiatement une meilleure approximation numérique, particulièrement perceptible pour un maillage grossier $N = 100$.

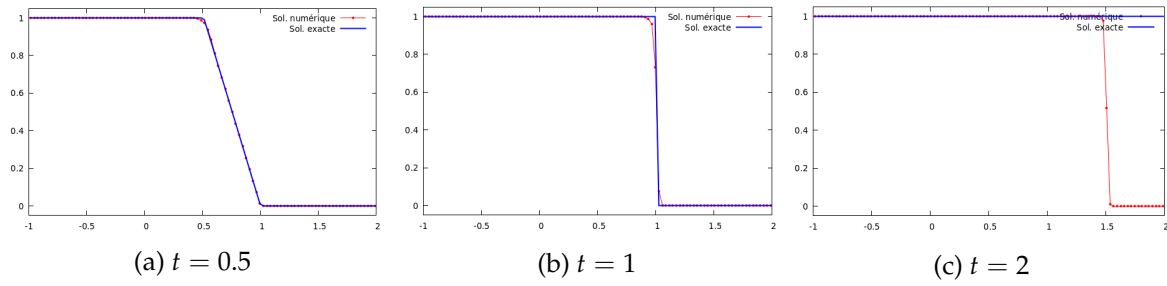


FIGURE 11 – Application de la correction de MUSCL au schéma de Godunov, pour un maillage de taille $N = 100$ i.e $\Delta x = 3e - 2$.

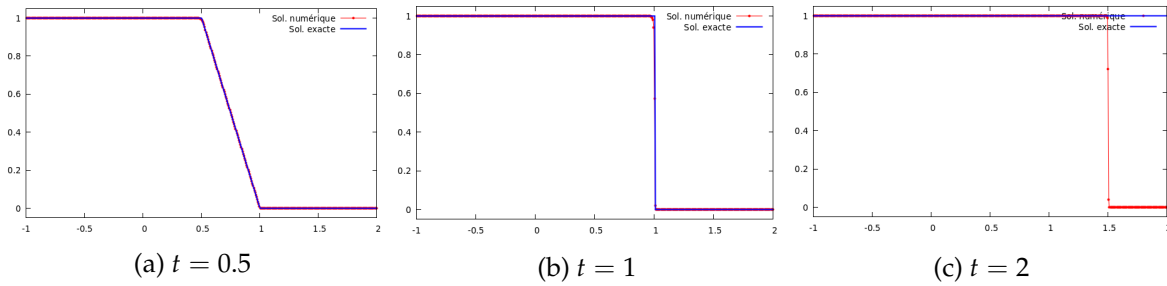


FIGURE 12 – Application de la correction de MUSCL au schéma de Godunov, pour un maillage de taille $N = 500$ i.e $\Delta x = 6e - 3$.

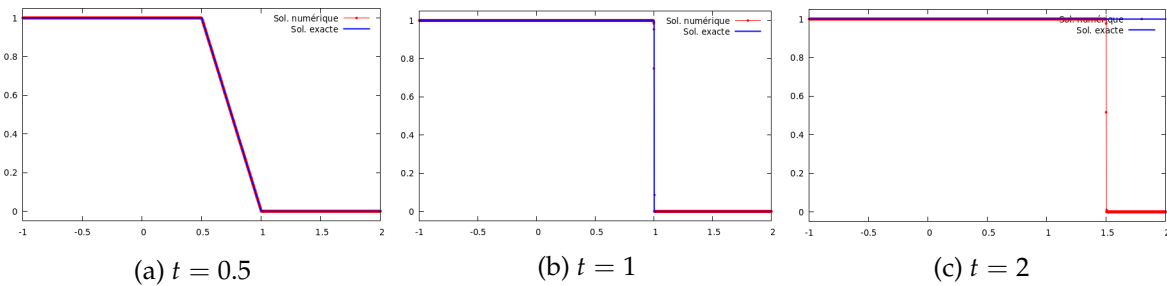


FIGURE 13 – Application de la correction de MUSCL au schéma de Godunov, pour un maillage de taille $N = 2500$ i.e $\Delta x = 1.2e - 3$.

L'étude du taux de convergence permet de constater la précision supérieure de la correction MUSCL de van Leer. Hormis le cas $t = 2$ qui est aberrant, on a, comparé à figure 10, des meilleures pentes. Ces pentes sont observables sur la figure figure 14 et un récapitulatif présenté au tableau 2. Il est intéressant de constater qu'en général, le problème de Burgers, bien qu'il soit non-linéaire, converge plus rapidement que le problème de transport (cf. tableau 1).

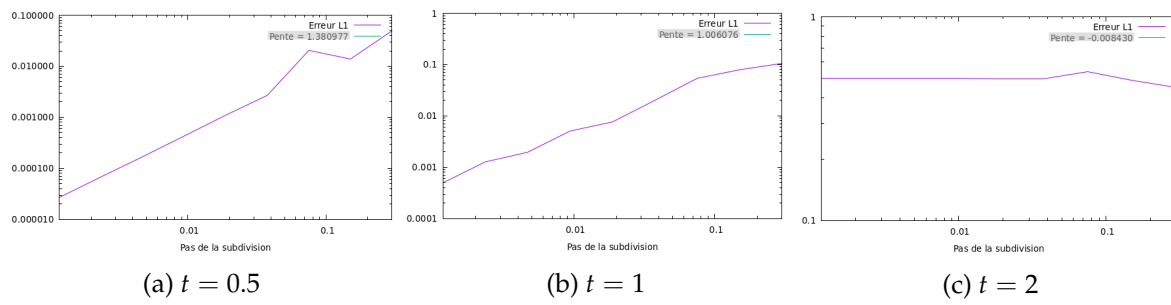


FIGURE 14 – Étude du taux de convergence de la méthode de Godunov avec la correction MUSCL.

Temps	Godunov Simple	Correction MUSCL
$t = 0.5$	0.9875	1.3810
$t = 1$	0.7598	1.0060
$t = 2$	-0.0131	-0.0084

TABLE 2 – Observation de l'amélioration de l'ordre de convergence avec la correction MUSCL. Le cas $t = 2$ est aberrant et n'intervient pas dans nos interprétations.