Stabilization of a string network: optimal decay rates and numerical implementations

M. Mehrenberger

University of Strasbourg and TONUS project (INRIA), France

Stability of nonconservative systems
Valenciennes, July 04, 2016

Joint work with M. Jellouli, Faculté des Sciences de Monastir, Tunisie

Outline

- I. Introduction
- II. Theoretical results (*)
- III. Numerical results (*)
- IV. Semi-Lagrangian implementation (new)
- V. Conclusion

(*) M. Jellouli, M. Mehrenberger, *Optimal decay rates for the stabilization of a string network*, Comptes Rendus Mathematiques, 2014.

The case of one string

- Vibrating elastic string fixed at end point

$$\partial_t^2 u(t,x) - \partial_x^2 u(t,x) = 0, \text{ on } (0,\ell), \; u(t,\ell) = 0$$

- Dissipation condition $\partial_x u(t,0) = \alpha \partial_t u(t,0)$ at origin
- Then the energy of the system satisfies

$$E(t) \le C \exp(-\gamma_\alpha t), \; \gamma = \frac{1}{\ell} \log \left| \frac{1+\alpha}{1-\alpha} \right|$$

- If $\alpha = 1$, then $E(t) = 0$, for $t \ge 2\ell$.

The case of one string

- See V. Komornik, *The Multiplier Method*
- Explicit solution via d'Alembert formula
- Exponential stabilization for $\alpha \neq 1$
- Controllability for $\alpha = 1$ (stabilization in finite time)

Question: what happens for a network of strings?

Some related results

- Star-shaped with stabilization at the common node or tree-shaped with stabilization at the root (Ammari, Jellouli, Khenissi, 2004/5)

  no exponential stability in the non degenerate case
- Pointwise stabilisation of a string viewed as 2 star shaped strings
  - (Ammari, Henrot, Tucsnak, 2001) Dirichlet/Neumann
    - exponential stabilization is obtained if and only if the lengths satisfy $\frac{\ell_1}{\ell_1+\ell_2} = \frac{p}{q}$, with p and q odd numbers
    - best decay rate, fixing total length $\ell_1 + \ell_2$ obtained for equal lengths $\ell_1 = \ell_2 = \ell$
    - best decay rate is $\gamma = \frac{\ln(3)}{\ell}$
  - (Nicaise, Valein, 2010) Dirichlet/Dirichlet, degenerate case
    - Energy limit E_∞ is identified
    - exponential decrease of energy to E_∞
- New results of L. Rosier and coauthors concerning *finite time stabilization*.
- New result of M. Jellouli, 2015: characterization of best decay rates with spectral analysis

A tree-shaped configuration

- $N \geq 3$ vibrating elastic strings

$$\partial_t^2 u_j(t,x) - \partial_x^2 u_j(t,x) = 0,\ t > 0,\ x \in (0,\ell_j),\ j = 1,\dots,N$$

- Continuity: $u_1(t,\ell_1) = u_j(t,0),\ j = 2,\dots,N$
- Dissipation at the root of the tree

$$\partial_x u_1(t,0) = \alpha \partial_t u_1(t,0),\ t \geq 0\ (\alpha > 0)$$

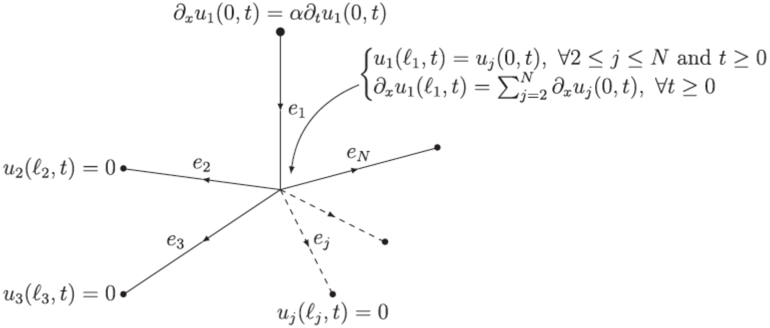
- Dirichlet at other exterior nodes $u_j(t,\ell_j) = 0,\ t \geq 0,\ j = 2,\dots,N$
- Transfert condition

$$\partial_x u_1(t,\ell_1) = \sum_{j=2}^{N} \partial_x u_j(t,0),\ t \geq 0,$$

- Initial condition

$$u_j(0,x) = a_j(x),\ \ \partial_t u_j(0,x) = b_j(x),\ x \in [0,\ell_j],\ j = 1,\dots,N.$$

Picture [1]



[1] borrowed from `Jellouli, 2015`

Questions

- What is the energy limit E_∞?
- Do we have exponential decrease to E_∞?
- What is the best decay rate?
- Can we have stabilization in finite time?
- How do numerical schemes behave?

In the sequel, we suppose that all the lengths are equal

$$\ell_j = \ell,\ j = 1, \dots, N.$$

The case $N = 2$ can be recasted to $N = 1$ with length $\ell_1 + \ell_2$.

Energy and energy limit

The energy is defined as

$$E(t) = \frac{1}{2}\sum_{j=1}^{N} \left\|\partial_x u_j(t)\right\|^2 + \frac{1}{2}\sum_{j=1}^{N} \left\|\partial_t u_j(t)\right\|^2,$$

with $\| \ \|$ for the norm of $L^2(0,\ell)$

Theorem

The energy limit is given by

$$E_\infty = \frac{1}{2(N-1)}\sum_{j=2}^{N}\sum_{k=j+1}^{N}\left(\left\|a'_k - a'_j\right\|^2 + \left\|b_k - b_j\right\|^2\right).$$

$E_\infty = 0$, for $a'_j = a'_2,\ b_j = b_2,\ j = 2, \ldots, N$.

Exponential decrease to E_∞ and best decay rate

Theorem

If $\alpha \neq \alpha_0$,

$$0 \leq E(t) - E_\infty \leq C_{\alpha,N}\left(\|a_1'\|^2 + \|b_1\|^2 + \left\| \sum_{j=2}^{N} a_j' \right\|^2 + \left\| \sum_{j=2}^{N} b_j \right\|^2 \right) e^{-\gamma t}$$

If $\alpha = \alpha_0$,

$$0 \leq E(t) - E_\infty \leq C_{\alpha,N}\left(\|a_1'\|^2 + \|b_1\|^2 + \left\| \sum_{j=2}^{N} a_j' \right\|^2 + \left\| \sum_{j=2}^{N} b_j \right\|^2 \right) t^2 e^{-\gamma_0 t},$$

where $\gamma_0 = \frac{1}{\ell} \log(1 + 2\frac{\sqrt{N-1}+1}{N-2})$ *and* $\gamma < \gamma_0$ *for* $\alpha \neq \alpha_0 = \frac{2\sqrt{N-1}}{N}$.

Remarks

- No stabilization in finite time, except for special initial data
- Proof based on operators of type τ (Ammari, Jellouli, 2007/10)
- Explicit computations based on d'Alembert formula
- Difficult to make the computations when lengths are different
- Star-shaped configuration should be feasible with same approach
- Explicit values permit to check the code
- Fixing total length L, we have $\ell = L/N$ and
  - $L\gamma_0$ is the worst for $N = 4$ (value is $4\log(2+\sqrt{3}) \simeq 5.2678$)
  - $L\gamma_{\alpha=1}$ is the worst for $N = \infty$ (value is 2)

Semi-discretization

- Classical finite difference space semi-discretization
- Discrete energy is decreasing; what is the limit?

Proposition

For $\phi \in \mathbb{R}^{2M}$*, and numerical solution* $V(t) = \exp(tA)\phi$*, we can write* $\phi = \phi_1 + \phi_2$ *where* ϕ_1 *belongs to the space* Λ_1 *of eigenvectors relative to the eigenvalues* λ *of A, with* $\Re(\lambda) = 0$ *and* ϕ_2 *belongs to the space* Λ_2 *of generalized eigenvectors relative to the eigenvalues* λ *of A, with* $\Re(\lambda) < 0$*. We then have*

$$E_{h,\phi,\infty} := \lim_{t\to\infty} E_{h,\phi}(t) = E_{h,\phi_1}(0),$$

and this value does not depend on $\alpha > 0$*.*

Numerical results

- We consider $d = 3$ strings of length $\ell = 1$
- N is now the number of points in each string
- initial condition non zero on string 2
  - $u_2^0(x) = \sin^2(\pi x)$
  - $u_2^0(x) = \sin^2(8\pi x)$

exact/semi-discrete time evolution of $E(t) - E_\infty$

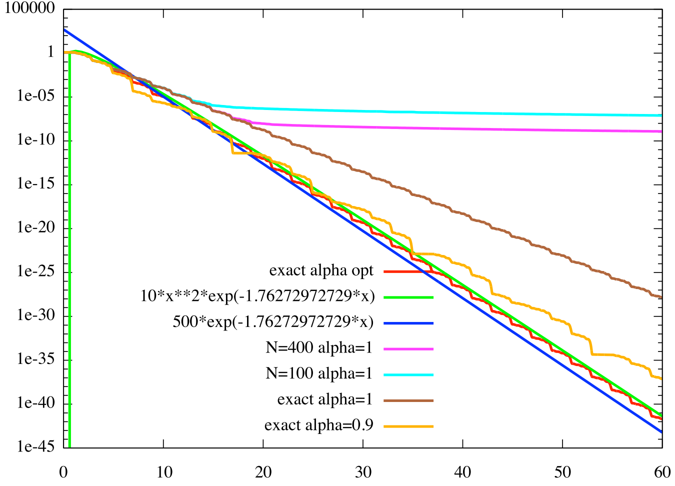


Figure: $\alpha \in \{0.9, 1, \alpha_{\rm opt} \simeq 0.9428\}$

exact/semi-discrete time evolution of $E(t) - E_\infty$

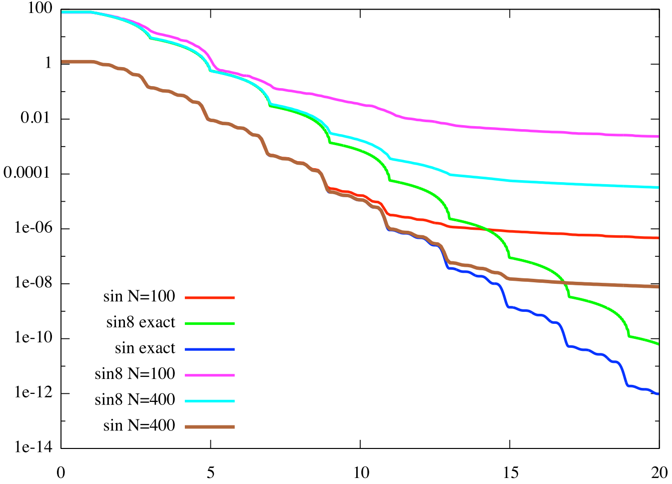
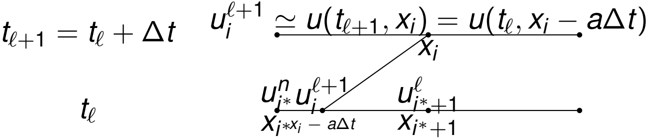


Figure: different initial conditions, $\alpha = \alpha_{\mathrm{opt}}$ }

Moving to a semi-Lagrangian scheme

The case of 1d constant advection

$$\partial_t u + a\partial_x u = 0, \; u = u(t,x)$$



- Characteristics are exact
- Lagrange interpolation :
  - Degree 1 (linear): x_{i^*}, x_{i^*+1}
  - Degree 3 (cubic): $x_{i^*-1}, x_{i^*}, x_{i^*+1}, x_{i^*+2}$
- Cubic splines interpolation
- Hermite interpolation :
  - f'_{i^*} : $x_{i^*-2}, x_{i^*-1}, x_{i^*}, x_{i^*+1}, x_{i^*+2}$

The case of Lagrange interpolation

Some known results

- L^2 stability `Strang, 1962`
- $L^q,\ q \geq 1$ stability for odd degree `Després, 2009`
- The scheme is equivalent to a Lagrange Galerkin scheme (`Pironneau, 1982`) for odd degree ≤ 13 `Ferretti, 2010`
  $\Rightarrow$ other proof of L^2stability

About SL-LG equivalence

Ferretti, 2010

- SL and LG are equivalent for the $1d$ constant advection, if we can find a function ϕ such that

$$\int_{\mathbb{R}} \phi(\eta + y)\phi(y)dy = \psi(y) \qquad \textbf{auto-correlation integral}$$

- ψ describes the Semi Lagrangian (SL) scheme
- ϕ describes the Lagrange Galerkin (LG) scheme
- In Fourier

$$\hat{\psi}(\omega) = \left|\hat{\phi}(\omega)\right|^2$$

- Example: for degree 3, we have

$$\hat{\psi}(\omega) = \frac{8(6+\omega^2)\sin(\omega/2)^4}{3\omega^4} \in \mathbb{R}^+$$

A direct proof of SL-LG equivalence

Algebraic form of the Fourier transform valid for *arbitrary* odd degree [2] (conjectured in Ferretti, 2010)

- Aim: prove that $S(\omega) = \int_0^1 \sum_{\ell=-d}^{d+1} L_\ell(x) \exp\left(i\,(\ell - x)\,\omega\right) dx \in \mathbb{R}^+$
- Compact formula for the derivative Boyer/Després's lecture notes
  $S'(\omega) = (-1)^d \frac{2^{2d+1}}{(2d+1)!} \sin^{2d+1}\left(\frac{\omega}{2}\right) \sigma(\omega)$
- **Integration by parts** for the factor $\sigma(\omega) = \int_0^1 \cos\left(\left(x - \frac{1}{2}\right)\omega\right) w\,(x)\, dx,\;\; w(x) = \prod_{j=-d}^{d+1}(x - j)$
- Recognise the primitive thanks to relation

$$w^{(2k+1)}(0) = -\frac{d+1}{2k+2} w^{(2k+2)}(0),\;\; k = 0, \ldots, d.$$

- Final explicit form

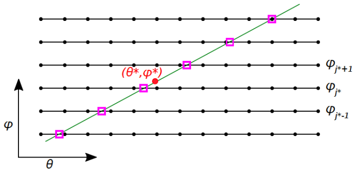
$$S(\omega) = (-1)^d \frac{2^{2d+1}}{(2d+1)!} \sin^{2d+2}\left(\frac{\omega}{2}\right) \sum_{k=0}^{d} \frac{w^{(2k+2)}(0)}{k+1} \frac{(-1)^k}{\omega^{2k+2}}$$

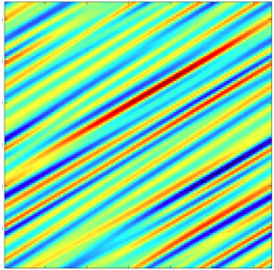
$\Rightarrow$ New proof of L^2 stability of SL scheme for constant advection

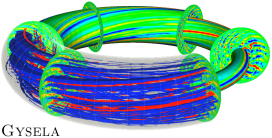
[2] Ferretti-M, WIP

Other proof from aligned interpolation[3]

Interpolation along a fixed oblic direction







GYSELA

$\Rightarrow$ Reconstruction of the values necessary by interpolation in θ

$\Rightarrow$ Reconstruction in the aligned direction

[3] Latu-Güclü-M-Ottaviani-Sonnendrücker, submitted

Other proof from aligned interpolation

- We first treat the case of $\lambda = \frac{b_\theta N_\theta}{b_\varphi N_\varphi}$ rational
  - *2d* symbol writes as a convex combination of 1*d* symbols in aligned direction
    - coefficients are discrete Fourier transform
    - Discrete Fourier transform is real
    - Discrete Fourier transform is **nonnegative**
- Case of λ real by density

Design of a semi-Lagrangian scheme for a string network

- d'Alembert formula is exact but solution can become complex
- finite difference scheme subject to CFL condition
- semi-Lagrangian method is the discrete analog of d'Alembert
- treatment of boundary
- link with Inverse Lax-Wendroff method [4]

[4] Tan-Shu, 2010; Chang-Filbet, 2013

Rewriting of the wave equation

Considering the wave equation

$$\partial_t^2 u - c^2 \partial_x^2 u = 0,$$

we can rewrite it as [5]

$$\begin{cases} \partial_t u + c\partial_x p = 0 \\ \partial_t p + c\partial_x u = 0. \end{cases}$$

We set

$$\begin{aligned} w^+ &= (u+p)/2 \\ w^- &= (u-p)/2 \end{aligned}$$

so that

$$\begin{cases} \partial_t w^+ + c\partial_x w^+ = 0 \\ \partial_t w^- - c\partial_x w^- = 0 \end{cases}$$

[5] see e.g. `Del Pino-Jourdren, 2006`

Initial condition

From

$$\begin{cases} u(t=0) = a_j \\ \partial_t u(t=0) = b_j \end{cases}$$

we get

$$\begin{cases} w_j^+(0,x) = \frac{1}{2}a_j(x) - \frac{1}{2c}\int_0^x b_j(y)dy - k_j \\ w_j^-(0,x) = \frac{1}{2}a_j(x) + \frac{1}{2c}\int_0^x b_j(y)dy + k_j \end{cases}$$

where k_j is an arbitrary constant.
The energy is then given by

$$E(t) = \frac{1}{2}\sum_{j=1}^{N}\int_0^{\ell_j} \left|\partial_x w^+ + \partial_x w^-\right|^2 + c^2\left|\partial_x w^+ - \partial_x w^-\right|^2$$

Boundary condition at root

At root, we have

$$\partial_x u = \alpha \partial_t u.$$

Note that

$$c\partial_x u = c\partial_x w^- + c\partial_x w^+ = \partial_t w^- - \partial_t w^+$$

and

$$\alpha c \partial_t u = \alpha c \partial_t w^- + \alpha c \partial_t w^+,$$

so that

$$\partial_t w^+ = \frac{1-\alpha c}{1+\alpha c}\partial_t w^-.$$

Integrating, we get

$$w^+(t+\Delta t, 0) = w^+(t,0) + \frac{1-\alpha c}{1+\alpha c}\left(w^-(t, c\Delta t) - w^-(t,0)\right).$$

Boundary condition at root

This leads to

$$w_1^+(t+\Delta t, x_k) = w_1^+(t+\Delta t - x_k/c_1 + x_k/c_1, x_k) = w_1^+(t+\Delta t - x_k/c_1, 0)$$
$$= w_1^+(t,0) + \frac{1-\alpha c_1}{1+\alpha c_1}\left(w_1^-(t, c_1\Delta t - x_k) - w_1^-(t,0)\right),$$

when $x_k < c_1\Delta t$, while

$$w_1^+(t+\Delta t, x_k) = w_1^+(t, x_k - c_1\Delta t),$$

otherwise.
Note that we have a *weak* restriction on time step

$$c_1\Delta t \le \ell_1.$$

Boundary condition at other exterior nodes

At other exterior nodes, we have $\partial_t u = 0$
This leads to $\partial_t w^+ + \partial_t w^- = 0$
By integration, we get

$$w^-(t+\Delta t,\ell) = w^-(t,\ell) + w^+(t,\ell) - w^+(t,\ell - c\Delta t)$$

For $j = 2,\ldots,N$, this leads to

$$\begin{aligned} & w_j^-(t+\Delta t, x_k) = w_j^-(t+\Delta t - (\ell_j - x_k)/c_j + (\ell_j - x_k)/c_j, x_k) \\ & = w_j^-(t+\Delta t - (\ell_j - x_k)/c_j, \ell_j) \\ & = w_j^-(t,\ell_j) + w_j^+(t,\ell_j) - w_j^+(t,\ell_j - c_j\Delta t + \ell_j - x_k) \end{aligned}$$

when $\ell_j - x_k < c_j \Delta t$, while

$$w_j^-(t+\Delta t, x_k) = w_j^-(t, x_k + c_j\Delta t)$$

otherwise. We have again the constraint $c_j \Delta t \leq \ell_j$.

Boundary condition at interior node

Suppose for simplicity that $N = 3$. We have

$$\partial_x u_1(\ell_1) = \partial_x u_2(0) + \partial_x u_3(0).$$

It rewrites

$$\partial_x w_1^+(\ell_1) + \partial_x w_1^-(\ell_1) = \partial_x w_2^+(0) + \partial_x w_2^-(0) + \partial_x w_3^+(0) + \partial_x w_3^-(0)$$

That is

$$\begin{array}{l} -\frac{1}{c_1}\partial_t w_1^+(\ell_1) + \frac{1}{c_1}\partial_t w_1^-(\ell_1) \\ = -\frac{1}{c_2}\partial_t w_2^+(0) + \frac{1}{c_2}\partial_t w_2^-(0) \\ -\frac{1}{c_3}\partial_t w_3^+(0) + \frac{1}{c_3}\partial_t w_3^-(0) \end{array}$$

$$\begin{array}{l} -\frac{1}{c_1} w_1^+(t+\Delta t, \ell_1) + \frac{1}{c_1} w_1^+(t, \ell_1) + \frac{1}{c_1} w_1^-(t+\Delta t, \ell_1) - \frac{1}{c_1} w_1^-(t, \ell_1) \\ = -\frac{1}{c_2} w_2^+(t+\Delta t, 0) + \frac{1}{c_2} w_2^+(t, 0) + \frac{1}{c_2} w_2^-(t+\Delta t, 0) - \frac{1}{c_2} w_2^-(t, 0) \\ -\frac{1}{c_3} w_3^+(t+\Delta t, 0) + \frac{1}{c_3} w_3^+(t, 0) + \frac{1}{c_3} w_3^-(t+\Delta t, 0) - \frac{1}{c_3} w_3^-(t, 0) \end{array}$$

Boundary condition at interior node

$$
\begin{aligned}
&-\tfrac{1}{c_1}w_1^+(t+\Delta t,\ell_1)+\tfrac{1}{c_1}w_1^+(t,\ell_1)+\tfrac{1}{c_1}w_1^-(t+\Delta t,\ell_1)-\tfrac{1}{c_1}w_1^-(t,\ell_1)\\
&=-\tfrac{1}{c_2}w_2^+(t+\Delta t,0)+\tfrac{1}{c_2}w_2^+(t,0)+\tfrac{1}{c_2}w_2^-(t+\Delta t,0)-\tfrac{1}{c_2}w_2^-(t,0)\\
&-\tfrac{1}{c_3}w_3^+(t+\Delta t,0)+\tfrac{1}{c_3}w_3^+(t,0)+\tfrac{1}{c_3}w_3^-(t+\Delta t,0)-\tfrac{1}{c_3}w_3^-(t,0)
\end{aligned}
$$

$$
\begin{aligned}
&-\tfrac{1}{c_1}w_1^+(t,\ell_1-c_1\Delta t)+\tfrac{1}{c_1}w_1^+(t,\ell_1)+\tfrac{1}{c_1}w_1^-(t+\Delta t,\ell_1)-\tfrac{1}{c_1}w_1^-(t,\ell_1)\\
&=-\tfrac{1}{c_2}w_2^+(t+\Delta t,0)+\tfrac{1}{c_2}w_2^+(t,0)+\tfrac{1}{c_2}w_2^-(t,c_2\Delta t)-\tfrac{1}{c_2}w_2^-(t,0)\\
&-\tfrac{1}{c_3}w_3^+(t+\Delta t,0)+\tfrac{1}{c_3}w_3^+(t,0)+\tfrac{1}{c_3}w_3^-(t,c_3\Delta t)-\tfrac{1}{c_3}w_3^-(t,0)
\end{aligned}
$$

This leads to the first relation

$$
\begin{aligned}
&\tfrac{1}{c_1}w_1^-(t+\Delta t,\ell_1)+\tfrac{1}{c_2}w_2^+(t+\Delta t,0)+\tfrac{1}{c_3}w_3^+(t+\Delta t,0)\\
&=\tfrac{1}{c_1}w_1^-(t,\ell_1)+\tfrac{1}{c_2}w_2^+(t,0)+\tfrac{1}{c_3}w_3^+(t,0)\\
&+\tfrac{1}{c_1}w_1^+(t,\ell_1-c_1\Delta t)-\tfrac{1}{c_1}w_1^+(t,\ell_1)\\
&+\tfrac{1}{c_2}w_2^-(t,c_2\Delta t)-\tfrac{1}{c_2}w_2^-(t,0)\\
&+\tfrac{1}{c_3}w_3^-(t,c_3\Delta t)-\tfrac{1}{c_3}w_3^-(t,0)
\end{aligned}
$$

Boundary condition at interior node

We have also the conditions for $j = 2, \ldots, N$

$$\partial_t u_1(\ell_1) = \partial_t u_j(0).$$

This leads to

$$w_1^+(t+\Delta t, \ell_1) + w_1^-(t+\Delta t, \ell_1) = w_1^+(t, \ell_1) + w_1^-(t, \ell_1)$$
$$+ w_j^+(t+\Delta t, 0) + w_j^-(t+\Delta t, 0) - w_j^+(t, 0) - w_j^-(t, 0)$$

that is

$$w_1^+(t, \ell_1 - c_1\Delta t) + w_1^-(t+\Delta t, \ell_1) = w_1^+(t, \ell_1) + w_1^-(t, \ell_1)$$
$$+ w_j^+(t+\Delta t, 0) + w_j^-(t, c_j\Delta t) - w_j^+(t, 0) - w_j^-(t, 0)$$

$$w_1^-(t+\Delta t, \ell_1) - w_j^+(t+\Delta t, 0) = w_1^-(t, \ell_1) - w_j^+(t, 0)$$
$$+ w_1^+(t, \ell_1) - w_1^+(t, \ell_1 - c_1\Delta t) + w_j^-(t, c_j\Delta t) - w_j^-(t, 0)$$

Boundary condition at interior node

Matrix to solve, $A = L^{-1}U$, for example for $N = 4$

$$A = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 1/c_1 & 1/c_2 & 1/c_3 & 1/c_4 \end{pmatrix}$$

with

$$L = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 1/c_2 & 1/c_3 & -\sum_{i=1}^{N-1} 1/c_i & 1 \end{pmatrix}, \quad U = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & \sum_{i=1}^{N} 1/c_i \end{pmatrix}$$

Boundary condition at interior node

Now, we use as before

$$w_1^-(t+\Delta t, x_k) = \begin{cases} w_1^-(t + (x_k + c_1\Delta t - \ell_1)/c_1, \ell_1), \ x_k + c_1\Delta t > \ell_1 \\ w_1^-(t, x_k + c_1\Delta t), \ x_k + c_1\Delta t \leq \ell_1 \end{cases}$$
$$w_j^+(t+\Delta t, x_k) = \begin{cases} w_j^+(t - (x_k - c_j\Delta t)/c_j, 0), \ x_k - c_j\Delta t < 0 \\ w_j^+(t, x_k - c_j\Delta t), \ x_k - c_j\Delta t \geq 0 \end{cases}$$

So, if $\ell_1 - x_k < c_1\Delta t$, we need to compute for $\delta t = \Delta t - (\ell_1 - x_k)/c_1$

$$\begin{array}{l} w_1^+(t, \ell_1 - c_1\delta t) \\ w_j^-(t, c_j\delta t), \ j = 2, \ldots, N \end{array}$$

If $x_k < c_2\Delta t$, we need to compute the same quantities for $\delta t = \Delta t - x_k/c_2$, and so on until N instead of 2.

Interpolation

We use **Hermite** interpolation [6], instead of Lagrange interpolation.
Fix $N_1 \in \mathbb{N}^*$ and $\varepsilon > 0$
We divide $(0, \ell_1)$ into N_1 cells $dx = \ell_1 / N_1$

- First cell

$$0, \varepsilon, (1-\varepsilon)dx, (1+\varepsilon)dx$$

- Cell $\ell \in \{2, \ldots, N_1 - 1\}$

$$(\ell - 1 - \varepsilon)dx, (\ell - 1 + \varepsilon)dx, (\ell - \varepsilon)dx, (\ell + \varepsilon)dx$$

- Cell number N_1

$$(N_1 - 1 - \varepsilon)dx, (N_1 - 1 + \varepsilon)dx, (N_1 - \varepsilon)dx, N_1 dx = \ell_1$$

[6] JET schemes `Seibold-Rozales-Navé, 2012`

Interpolation

On a cell (a, b), we have for cell $\ell \in \{2, \ldots, N_1 - 1\}$

$$x_1 = a - \varepsilon dx,\ x_2 = a + \varepsilon dx,$$
$$x_3 = b - \varepsilon dx,\ x_4 = b + \varepsilon dx$$

and use Hermite interpolation using points

$$f_a = (f(x_1) + f(x_2))/2,\ f'_a = (f(x_2) - f(x_1))/(2\varepsilon),$$
$$f_b = (f(x_3) + f(x_4))/2,\ f'_b = (f(x_4) - f(x_3))/(2\varepsilon)$$

For $\alpha \in (0, 1)$

$$f_{(1-\alpha)a+\alpha b} = (1-\alpha)^2(2\alpha+1)f_a + \alpha^2(3-2\alpha)f_b + \alpha(1-\alpha)^2 f'_a - \alpha^2(1-\alpha)f'_b$$

Adaption for first and last cell

Interpolation

Letting $\varepsilon \to 0$, we return to classical Hermite interpolation, using

$$f'_{(1-\alpha)a+\alpha b} = 6\alpha(1-\alpha)(f_b - f_a) + (3\alpha-1)(\alpha-1)f'_a + \alpha(3\alpha-2)f'_b$$

We use a uniform mesh and have to locate the cells and displacement inside the cell

We consider first the first string with $x_k = (k-1)\frac{\ell_1}{N_1}, k = 1, \dots, N_1 + 1$.

- $-c_1 N_1 \Delta t/\ell_1 = -k_1 + \alpha_1,\ k_1 \in \mathbb{Z},\ 0 \le \alpha_1 < 1$.
- If $\Delta t = 0$, we take $k_1 = 1,\ \alpha_1 = 1$. Then, we can suppose $k_1 \in \{1, \dots, N_1\}$.
- $x_{k_1+1} - c_1 \Delta t$: displacement α_1 in cell 1 ...
- $x_{N_1+1} - c_1 \Delta t$: displacement α_1 in cell $N_1 + 1 - k_1$
- $c_1 N_1 \Delta t/\ell_1 = k_1 - 1 + 1 - \alpha_1$
- $c_1 \Delta t - x_1$: displacement $1 - \alpha_1$ in cell k_1 ...
- $c_1 \Delta t - x_{k_1}$: displacement $1 - \alpha_1$ in cell 1

Interpolation

- $x_1 + c_1 \Delta t$: displacement $1 - \alpha_1$ in cell k_1 ...
- $x_{N_1 - k_1 + 1} + c_1 \Delta t$: displacement $1 - \alpha_1$ in cell N_1
- $2\ell_1 - x_k - c_1 \Delta t = \frac{\ell_1}{N_1}(2N_1 - k - k_1 + 1 + \alpha_1)$
- $2\ell_1 - x_{N_1 - k_1 + 2} - c_1 \Delta t$: displacement α_1 in cell N_1 ...
- $2\ell_1 - x_{N_1 + 1} - c_1 \Delta t$: displacement α_1 in cell $N_1 + 1 - k_1$

Finally, we need to compute

- w_1^+ at each cell with shift α_1
- w_1^- at each cell with shift $1 - \alpha_1$

This is not enough

Interpolation

We write

$$\Delta t = \frac{\ell_j}{c_j N_j}\left(k_j - 1 + 1 - \alpha_j\right),\ j = 1, \ldots, N.$$

We define $k_{j,j'}(s),\ \alpha_{j,j'}(s)$ by

$$\frac{\ell_j}{c_j N_j}\left(k_{j,j'}(s) - 1 + 1 - \alpha_{j,j'}(s)\right) = \frac{\ell_{j'}}{c_{j'} N_{j'}}\left(s - 1 + 1 - \alpha_{j'}\right)$$

We need here

$$k_{j,1}(1), \ldots, k_{j,1}(k_1) = k_j,\ \alpha_{j,1}(1), \ldots, \alpha_{j,1}(k_1) = \alpha_j$$

- $c_j(\Delta t - (\ell_1 - x_{N_1-k_1+2})/c_1)$: displacement $\alpha_{j,1}(1)$ in cell $k_{j,1}(1)$ ...
- $c_j(\Delta t - (\ell_1 - x_{N_1+1})/c_1)$: displacement $\alpha_{j,1}(k_1)$ in cell $k_{j,1}(k_1)$

Interpolation

Now, we consider the other strings $j = 2, \ldots, N$
Now, $x_k = (k-1)\frac{\ell_j}{N_j}, k = 1, \ldots, N_j + 1$.

- $x_{k_j+1} - c_j\Delta t$: displacement α_j in cell 1 $\ldots$
- $x_{N_j+1} - c_j\Delta t$: displacement α_j in cell $N_j + 1 - k_j$
- $c_j\Delta t - x_1$: displacement $1 - \alpha_j$ in cell k_j $\ldots$
- $c_j\Delta t - x_{k_j}$: displacement $1 - \alpha_j$ in cell 1
- $x_1 + c_j\Delta t$: displacement $1 - \alpha_j$ in cell k_j $\ldots$
- $x_{N_j-k_j+1} + c_j\Delta t$: displacement $1 - \alpha_j$ in cell N_j
- $2\ell_j - x_{N_j-k_j+2} - c_j\Delta t$: displacement α_j in cell N_j $\ldots$
- $2\ell_j - x_{N_j+1} - c_j\Delta t$: displacement α_j in cell $N_j + 1 - k_j$

This is again not enough

Interpolation

We have for $j' \neq j$

$$c_{j'}(\Delta t - \frac{x_k}{c_j}) = \frac{c_j' \ell_j}{c_j N_j}(k_j - k + 1 - 1 + 1 - \alpha_j)$$
$$= \frac{\ell_{j'}}{N_{j'}}(k_{j'j}(k_j - k + 1) - 1 + 1 - \alpha_{j'j}(k_j - k + 1))$$

- $c_{j'}(\Delta t - x_1/c_j)$: displacement $\alpha_{j',j}(k_j)$ in cell $k_{j',j}(k_j)$ ...
- $c_{j'}(\Delta t - x_{k_j}/c_j)$: displacement $\alpha_{j',j}(1)$ in cell $k_{j',j}(1)$

It remains to treat the case

$$\ell_1 - c_1(\Delta t - \frac{x_k}{c_j}) = \ell_1 - \frac{\ell_1}{N_1}(k_{1j}(k_j - k + 1) - 1 + 1 - \alpha_{1j}(k_j - k + 1))$$
$$= \frac{\ell_1}{N_1}(N_1 - k_{1j}(k_j - k + 1) + \alpha_{1j}(k_j - k + 1))$$

- $\ell_1 - c_1(\Delta t - \frac{x_1}{c_j})$: displacement $\alpha_{1,j}(k_j)$ in cell $N_1 + 1 - k_{1,j}(k_j)$ ...
- $\ell_1 - c_1(\Delta t - \frac{x_{k_j}}{c_j})$: displacement $\alpha_{1,j}(1)$ in cell $N_1 + 1 - k_{1,j}(1)$

Special discretization

We suppose that all the lengths and speeds are rational.
Taking $M \in \mathbb{N}^*$ large enough such that $N_j = M\ell_j/c_j \in \mathbb{N}^*$ and $k_j = k_1$, with

$$k_1 \in \{1, \ldots, \min_{k=1,\ldots,N} M\ell_k/c_k\}.$$

We take

$$\Delta t = \frac{k_1}{M} = \frac{\ell_j k_j}{c_j N_j},\ j = 1, \ldots, N.$$

and there is no interpolation: we get the exact solution.

- Low computational cost (and acceleration possible, when no inteprolation is needed)
- Low memory (treat outflow first)
- Good candidate for numerical optimization of parameters for more complex cases as those currently solved (as optimal decay rates)

Link with Lattice Boltzmann method

A BGK model Let $\varepsilon > 0$ and $c \geq 0$. We consider the model

$$\partial_t f + v\partial_x f = -\frac{1}{\varepsilon}(f - M(f)).$$

where $f = f(t, x, v)$, with $v \in \{\pm 1, 0\}$ and the *maxwellian* $M(f)$ is defined through the relations:

$$\begin{cases} \rho(f) = \int f dv = f(\cdot,\cdot,-1) + f(\cdot,\cdot,0) + f(\cdot,\cdot,1), \\ \rho(f)u(f) = \int fv dv = f(\cdot,\cdot,1) - f(\cdot,\cdot,-1), \end{cases} \tag{1}$$

and

$$\begin{cases} \rho(f) = \int M(f) dv = M(f)(\cdot,\cdot,-1) + M(f)(\cdot,\cdot,0) + M(f)(\cdot,\cdot,1), \\ \rho(f)u(f) = \int M(f) v dv = M(f)(\cdot,\cdot,1) - M(f)(\cdot,\cdot,-1), \\ \rho(f)(u(f)^2 + c^2) = \int M(f) v^2 dv = M(f)(\cdot,\cdot,1) + M(f)(\cdot,\cdot,-1). \end{cases} \tag{2}$$

Conclusion/Perspectives

- Conclusion
  - Stabilization for a tree-shaped network with equal lengths
  - Numerical results are coherent with theoretical ones
  - Design of a semi-Lagrangian scheme
- Perspectives
  - Validation of the scheme w.r.t. theoretical results
  - Numerical analysis of the scheme
  - Numerical study for other configurations