Lecture. Time integration of Nonsmooth Dynamical Systems (NSDS). Low relative degree

Vincent Acary

December 2, 2018

Difficulties and Approaches

Two major difficulties :

- Time integration of non smooth evolutions
- Solving a optimization problem together with a dynamical equilibrium constraint

Three major approaches :

- Hybrid Approach
  - Hybrid multi-modal dynamical system
  - Need to perform a decomposition of the evolution "triggering events"
  - Enumerative resolution of the mode transition process
- Event–Driven Approach
  - Two time formulations of the dynamical system (time–continuous and time–discrete)
  - Need to perform a decomposition of the evolution. "triggering events"
  - Algebraic resolution of the mode transition process
- Time–stepping approach
  - Global approach with a single formulation
  - Need to define a global formulation of the NSDS
  - Algebraic resolution of the one–step nonsmooth problem

Principle

Time-decomposition of the dynamics in

- *modes*, time-intervals in which the dynamics is smooth,
- discrete events, times where the dynamics is nonsmooth.

The following assumptions guarantee the existence and the consistency of such a decomposition

- The definition and the localization of the discrete events. The set of events is negligible with the respect to Lebesgue measure.
- The definition of time-intervals of non-zero lengths. the events are of finite number and "well-separated" in time. Problems with finite accumulations of impacts, or Zeno-state

Comments

On the numerical point of view, we need

- detect events with for instance root-finding procedure.
  - Dichotomy and interval arithmetic
  - Newton procedure for C^2 function and polynomials
- solve the non smooth dynamics at events with a reinitialization rule of the state,
- integrate the smooth dynamics between two events with any ODE solvers.

Event–detecting (Event–driven) schemes for DCS

Relative degree 0

Let us consider a LCS of relative degree 0 i.e. with D which is non singular.

$$\begin{cases} \dot{x} = Ax + B\lambda, \quad x(0) = x_0 \\ y = Cx + D\lambda \\ 0 \leqslant y \perp \lambda \geqslant 0 \end{cases} \tag{1}$$

Assumption

$B.SOL(Cx_0, D)$ is a singleton is equivalent to stating that the LCS (1) has a unique C^1 solution defined at all $t \geqslant 0$.
Denoting by $\Lambda(x) = B.SOL(Cx, D)$, the LCS can be viewed as a standard ODE with a Lipschitz r.h.s :

$$\dot{x} = Ax + \Lambda(x) = Ax + B.SOL(Cx, D) \tag{2}$$

Event–detecting (Event–driven) schemes for DCS

Definition (Active index–sets)

Let us denote by α the set of active constraints at time t

$$\alpha = \{i, y_i = C_{i\bullet}x(t) + D_i = 0, \lambda_i \geqslant 0\} \tag{3}$$

and its complementary set by

$$\beta = \{j, y_j = C_{j\bullet}x(t) + D_j \geqslant 0, \lambda_i = 0\} \tag{4}$$

LCP Solution for D a P-Matrix

Given the active index set α, the solution of the $LCP(Cx(t), D)$ is

$$\lambda(x(t)) = \begin{cases} \lambda_\alpha(x(t)) = -D^{-1}_{\alpha\alpha}(C_{\alpha\bullet}x(t)) \\ \lambda_\beta(x(t)) = 0 \end{cases} \tag{5}$$

Event–detecting (Event–driven) schemes for DCS

Smooth ODE

If the index sets α is constant on $[t_k, t_{k+1}]$, we perform the integration of

$$\dot{x} = Ax + \Lambda(x) = Ax - B_\alpha(D^{-1}_{\alpha\alpha}(C_{\alpha\bullet}x(t)) \quad (6)$$

which is a smooth ODE with a $\mathcal{C}^\infty$ right–hand–side.

Standard numerical integration with root finding

The ODE (6) can be solved with numerical methods for ODE on intervals with constant index sets *alpha* :

- One–step numerical methods : Euler, Runge–Kutta methods, Extrapolation methods
- Multi–step methods : Adams–Moulton, Adams–bashford,, BDF

A root finding procedure (Dichotomy, newton, . . .) is used to detect changes, "events" in the index sets

Event–detecting schemes for Moreau's sweeping process and DVI...

Assumptions

Let us assume that the set K is finitely represented

$$K = \{x \in \mathbb{R}^n, h(x) \leqslant 0\} \tag{7}$$

The same procedure may be performed with

$$\alpha = \{i, y_i = h_i(x(t)) = 0, \lambda_i \geqslant 0\} \tag{8}$$

Issues

1. If the set is not finitely representable, triggering events is not possible
2. If the set if defined by nonlinear constraints $h(x) \geqslant 0$, triggering events can be very difficult and not very accurate. We need a dense output of the state $x(t)$ at a given accuracy to know when a vent occurs.

Advantages and disadvantages. Event–detecting schemes

Advantages

- Seems easy to handle from the computational point of view
  - In each modes, smooth integration between two events (ODE/DAE).
  - At event, a optimization problem is solved without time evolution.

Disadvantages :

- Scability and complexity of the algorithms
- Need an accurate event detection difficult for nonlinear constraints
- Accumulation of events
- No existence or uniqueness results
- Sensitivity to accuracy thresholds. Tuning the "ε" is a hard task.

Lead to numerical schemes suitable

- Small systems with a small number of events
- High accuracy in each modes

Principle of Event–capturing scheme

- The time–step is not adapted to the time of events
- A unique formulation that contains all the modes is considered
- The time-integration is based on a consistent approximation of the differential equations according to the smoothness of the solutions ($\mathcal{C}^1$, AC, BV, measures, ...)

Event–capturing (Time-stepping) scheme for Linear Complementarity Systems (LCS)

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions

Assumptions

The solutions of the LCS

$$\begin{cases} \dot{x} = Ax + B\lambda + u \\ y = Cx + D\lambda + a \\ K^* \in y \perp \lambda \in K, \end{cases} \tag{9}$$

is assumed to be of class $\mathcal{C}^1$ solutions

The (θ, γ)– scheme

A possible scheme can be used when a solution $x(t)$ of class C^1 with $\lambda(t)$ continuous is expected

$$\begin{cases} x_{k+1} - x_k = h\left(Ax_{k+\theta} + u_{k+\theta} + B\lambda_{k+\gamma}\right), \\ y_{k+\gamma} = Cx_{k+\gamma} + D\lambda_{k+\gamma} + a_{k+\gamma}, \\ K^* \in y_{k+\gamma} \perp \lambda_{k+\gamma} \in K, \end{cases} \tag{10}$$

where $\theta \in [0, 1]$ and $\gamma \in [0, 1]$.

Notation

$x_{k+\theta} = (1 - \theta)x_k + \theta x_{k+1}$, $\lambda_{k+\gamma} = (1 - \gamma)\lambda_k + \gamma\lambda_{k+1}$

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions

The discretized system (10) amounts to solving at each time–step the following one–step problem :

$$\begin{cases} y_{k+\gamma} = M\lambda_{k+\gamma} + q \\ K^* \in y_{k+\gamma} \perp \lambda_{k+\gamma} \in K, \end{cases} \tag{11}$$

with

$$\begin{aligned} M &= D + h\gamma C(I - h\theta A)^{-1}B, \\ q &= a_{k+\gamma} + \gamma C(I - h\theta A)^{-1}\left[(I + h(1-\theta)A)x_k + hu_{k+\theta}\right] + C(1-\gamma)x_k. \end{aligned} \tag{12}$$

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions

Rule of thumb

- For $\dot{x} = Ax$, the θ–scheme

  $$x_{k+1} - x_k = hAx_{k+\theta}$$

  is of order 2 for $\theta = 1/2$.
  This requires sufficient smoothness of the solution (at least $\mathcal{C}^1$).
  $\theta = \gamma = 1/2$ can only be used if the solution is $\mathcal{C}^1$.
- For $\theta \geqslant 1/2$ and $\gamma \geqslant 1/2$, the scheme is unconditionally stable. Properties that comes with the implicit character of the scheme. The scheme dissipates also more energy.

Solutions as continuously differentiable functions (C^1 solutions)

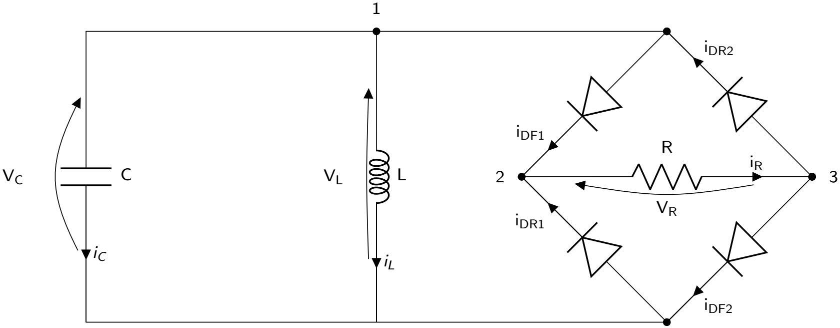
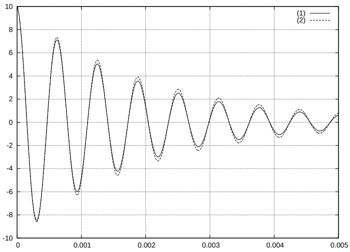
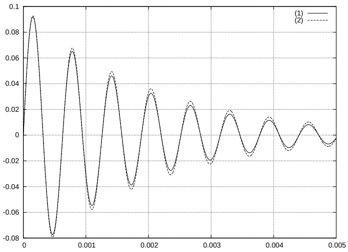


Figure: The 4-diode bridge rectifier. LC oscillator with a load resistor

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions



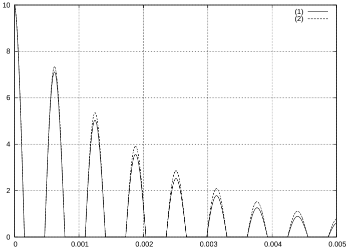
(a) Voltage across the inductor V_L versus time. State $x_{1,k}$



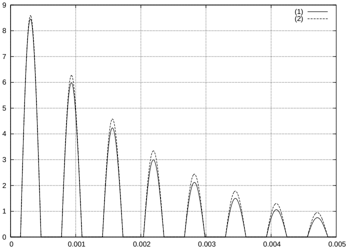
(b) Current through the inductor i_L versus time. state $x_{2,k}$

Figure: Simulation of the circuit on Fig. **??** with the scheme (10). (1) $\theta = 1, \gamma = 1$ (2) $\theta = 1/2, \gamma = 1/2$.

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions



(a) Potential at node 2 (V_2) versus time. Variable $\lambda_{1,k}$



(b) Potential at node 3 (V_3) versus time. Variable $\lambda_{2,k}$

Figure: Simulation of the circuit on Fig. **??** with the scheme (10). (1) $\theta = 1, \gamma = 1$ (2) $\theta = 1/2, \gamma = 1/2$.

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions

The discrete storage function can be defined as

$$\mathcal{V}_{k+1} = \frac{1}{2}(C\, v_{L,k+1}^2 + L\, i_{L,k+1}^2) \tag{13}$$

and the discrete dissipation function as

$$\mathcal{D}_{k+1} = h \sum_{j=1}^{k+1} R(i_{R,j})^2 \tag{14}$$

and the cumulative function $\mathcal{V}_{k+1} + \mathcal{D}_{k+1}$.
We remark that the scheme with $\theta = \gamma = 1/2$ is able to reproduce the exact energetic behaviour as in the continuous time case.

Event–capturing (Time-stepping) scheme for (LCS) with $\mathcal{C}^1$ solutions

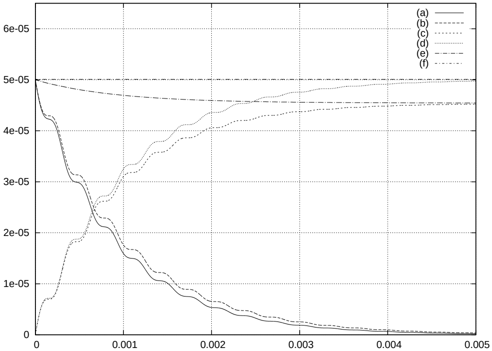


Figure: Simulation of the circuit on Fig. **??** with the scheme (10). (a) Storage function $\mathcal{V}_{k+1}$ for $\theta = 1, \gamma = 1$ (b) Storage function $\mathcal{V}_{k+1}$ for $\theta = 1/2, \gamma = 1/2$ (c) Dissipation function $\mathcal{D}_{k+1}$ for $\theta = 1, \gamma = 1$ (b) Dissipation function $\mathcal{D}_{k+1}$ for $\theta = 1/2, \gamma = 1/2$ (e) Cumulative function for $\theta = 1, \gamma = 1$ (f) Cumulative function for $\theta = 1/2, \gamma = 1/2$.

Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions

Assumptions

The solutions of the LCS

$$\begin{cases} \dot{x} = Ax + B\lambda + u \\ y = Cx + D\lambda + a \\ K^* \in y \perp \lambda \in K, \end{cases} \tag{15}$$

is assumed to be of class *AC* solutions

The (θ)– scheme

The following time–stepping scheme is used when a absolutely continuous solution $x(t)$ with $\lambda(t)$ function of Bounded Variations is expected:

$$\begin{cases} x_{k+1} - x_k = h\left(Ax_{k+\theta} + u_{k+\theta} + B\lambda_{k+1}\right), \\ \\ y_{k+1} = Cx_{k+1} + D\lambda_{k+1} + a_{k+1}, \\ \\ K^* \ni y_{k+1} \perp \lambda_{k+1} \in K, \end{cases} \tag{16}$$

with $\theta \in [0, 1]$.

Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions

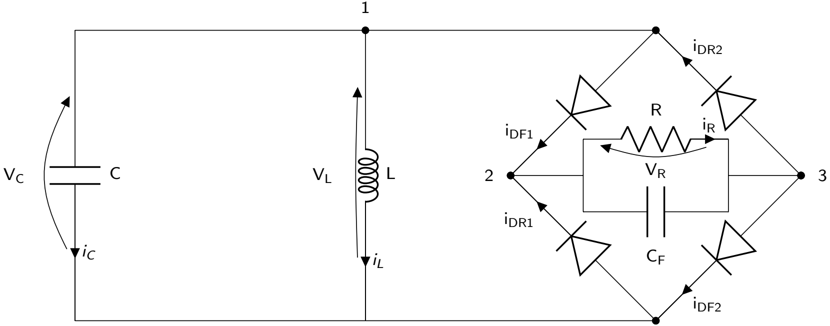
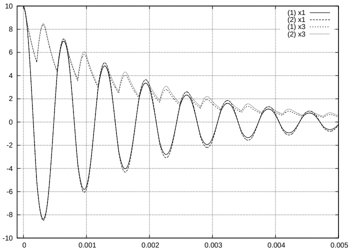
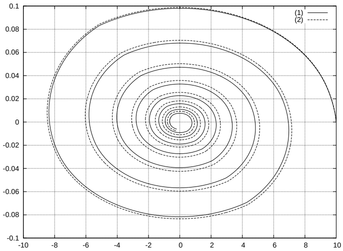


Figure: The 4-diode bridge rectifier. LC oscillator with a load resistor filtered by a capacitor

Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions



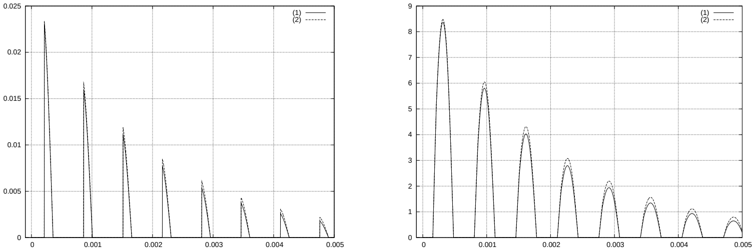
(a) state $x_{1,k}$ and $x_{3,k}$



(b) phase portrait $x_{1,k}$ vs $x_{2,k}$

Figure: Simulation of the configuration in Fig. 5 with the scheme (16). (1) $\theta = 1$ (2) $\theta = 1/2$

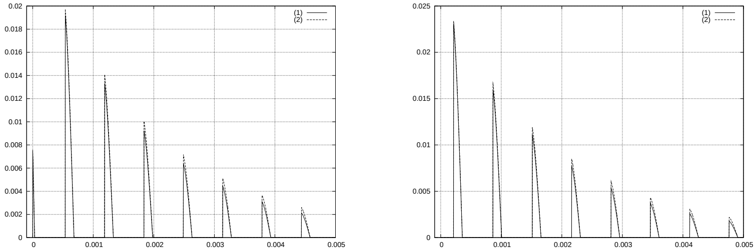
Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions



(a) variable $\lambda_{1,k}$ (b) variable $\lambda_{2,k}$

Figure: Simulation of the configuration in Fig. 5 with the scheme (16). (1) $\theta = 1$ (2) $\theta = 1/2$

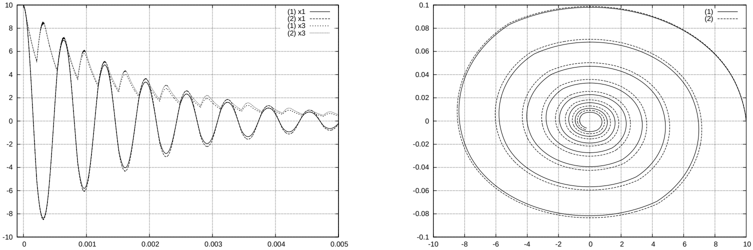
Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions



(a) variable $\lambda_{3,k}$ (b) variable $\lambda_{4,k}$

Figure: Simulation of the configuration in Fig. 5 with the scheme (16). (1) $\theta = 1$ (2) $\theta = 1/2$

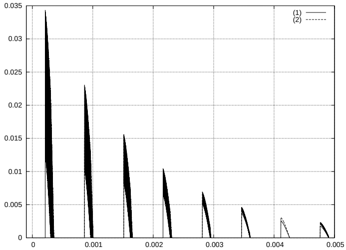
Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions



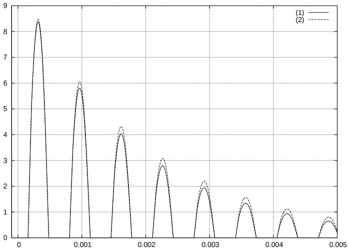
(a) state $x_{1,k}$ and $x_{3,k}$ (b) phase portrait $x_{1,k}$ vs $x_{2,k}$

Figure: Simulation of the configuration in Fig. 5 with the scheme (10). (1) $\theta = 1, \gamma = 1$ (2) $\theta = 1/2, \gamma = 1/2$

Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions



(a) variable $\lambda_{1,k}$

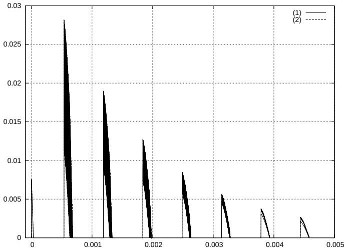


(b) variable $\lambda_{2,k}$

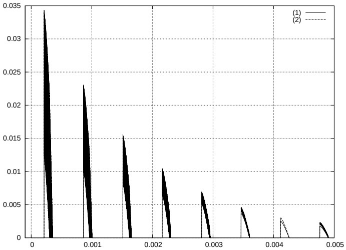
Figure: Simulation of the configuration in Fig. 5 with the scheme (10). (1) $\theta = 1, \gamma = 1$ (2) $\theta = 1/2, \gamma = 1/2$

➜ Occurrence of instabilities in the numerical response with $\gamma \neq 1$

Event–capturing (Time-stepping) scheme for (LCS) with *AC* solutions



(a) variable $\lambda_{3,k}$



(b) variable $\lambda_{4,k}$

Figure: Simulation of the configuration in Fig. 5 with the scheme (10). (1) $\theta = 1, \gamma = 1$ (2) $\theta = 1/2, \gamma = 1/2$

➜ Occurrence of instabilities in the numerical response with $\gamma \neq 1$

Event–capturing (Time-stepping) scheme for (LCS) with *BV* solutions

Assumptions

The solutions of the LCS

$$\begin{cases} \dot{x} = Ax + B\lambda + u \\ y = Cx + D\lambda + a \\ K^* \in y \perp \lambda \in K, \end{cases} \tag{17}$$

is assumed to be of class *BV* solutions

Warning

The time discretization of (**??**) has to take into account the nature of the solution to avoid point-wise evaluations of measures, which are not mathematically well–defined.

Event–capturing (Time-stepping) scheme for (LCS) with *BV* solutions

Only the measure of the time–intervals $(t_k, t_{k+1}]$ must be considered such that :

$$dx((t_k, t_{k+1}]) = \int_{t_k}^{t_{k+1}} Ax(t) + u(t)\, dt + Bdi((t_k, t_{k+1}]). \tag{18}$$

By definition of the differential measure, we get

$$dx((t_k, t_{k+1}]) = x(t_{k+1}^+) - x(t_k^+). \tag{19}$$

The measure of the time–interval by di is kept as an unknown variable denoted by

$$\sigma_{k+1} = di((t_k, t_{k+1}]). \tag{20}$$

Finally, the remaining Lebesgue integral in (18) is approximated by an implicit Euler scheme

$$\int_{t_k}^{t_{k+1}} Ax(t) + u(t)\, dt \approx h(Ax_{k+1} + u_{k+1}). \tag{21}$$

The matrix D needs to be at least rank-deficient to expect some jumps in the state.

Event–capturing (Time-stepping) scheme for (LCS) with *BV* solutions

Let us start with the simplest case of $D = 0$.
The following time–stepping scheme is used when a solution of bounded variations $x(t)$ with *di* a measure is expected and $D = 0$

$$\begin{cases} x_{k+1} - x_k = h\left(Ax_{k+\theta} + u_{k+\theta}\right) + B\sigma_{k+1}, \\ y_{k+1} = Cx_{k+1} + a_{k+1}, \\ 0 \in y_{k+1} + N_K(\sigma_{k+1}), \end{cases} \tag{22}$$

with $\theta \in [0, 1]$.
If $D \neq 0$, the second line of (22) is augmented in the following way

$$y_{k+1} = Cx_{k+1} + a_{k+1} + \frac{1}{h} D\sigma_{k+1}. \tag{23}$$

Event–capturing (Time-stepping) scheme for (LCS) with *BV* solutions

The discretized system (22) amounts to solving at each time–step the following one–step nonsmooth problem:

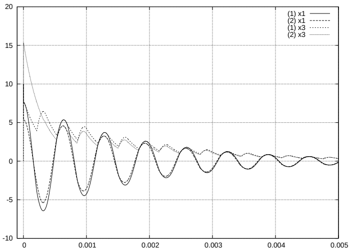
$$\begin{cases} y_{k+1} = M\sigma_{k+1} + q, \\ 0 \in y_{k+1} + N_K(\sigma_{k+1}), \end{cases} \tag{24}$$

with

$$\begin{aligned} M &= C(I - h\theta A)^{-1}B, \\ q &= a_{k+1} + C(I - h\theta A)^{-1}\left[(I + h(1-\theta)A)x_k + hu_{k+\theta}\right]. \end{aligned} \tag{25}$$

It is worth noting that the matrix M remains consistent when the time–step h vanishes if CB is assumed to be regular. This is not necessarily the case in (12).

Event–capturing (Time-stepping) scheme for (LCS) with *BV* solutions



(a) state $x_{1,k}$ and $x_{3,k}$ (b) phase portrait $x_{1,k}$ vs $x_{2,k}$

Figure: Simulation of the configuration 5. (1) scheme (24) (2) scheme (10) $\theta = 1/2, \gamma = 1/2$

➜ the state jump law is not respected when $\gamma \neq 1/2$

Event–capturing (Time-stepping) scheme for (LCS)

Backward Euler scheme

Starting from the LCS

$$\begin{cases} \dot{x} = Ax + B\lambda \\ y = Cx + D\lambda \\ 0 \leqslant y \perp \lambda \geqslant 0 \end{cases} \tag{26}$$

Camlibel et al. [1] apply a backward Euler scheme to evaluate the time derivative $\dot{x}$ leading to the following scheme:

$$\begin{cases} \dfrac{x_{k+1} - x_k}{h} = Ax_{k+1} + B\lambda_{k+1} \\ y_{k+1} = Cx_{k+1} + D\lambda_{k+1} \\ 0 \leqslant \lambda_{k+1} \perp y_{k+1} \geqslant 0 \end{cases} \tag{27}$$

which can be reduced to a LCP by a straightforward substitution:

$$0 \leqslant \lambda_{k+1} \perp C(I - hA)^{-1}x_k + (hC(I - hA)^{-1}B + D)\lambda_{k+1} \geqslant 0 \tag{28}$$

Time stepping scheme for Linear Complementarity Systems (LCS)

Convergence results

If D is nonnegative definite or that the triplet (A, B, C) is observable and controllable and (A, B, C, D) is positive real, they exhibit that some subsequences of $\{y_k\}, \{\lambda_k\}, \{x_k\}$ converge weakly to a solution y, λ, x of the LCS. [1]
Such assumptions imply that the relative degree r is less or equal to 1.

Remarks

- In the case of the relative degree 0, the LCS is equivalent to a standard system of ODE with a Lipschitz-continuous r.h.s field. The result of convergence is then similar to the standard result of convergence for the Euler backward scheme.
- In the case of a relative degree equal to 1, the initial condition must satisfy the unilateral constraints $y_0 = Cx_0 \geqslant 0$. Otherwise, the approximation $\frac{x_{k+1} - x_k}{h}$ has non chance to converge if the state possesses a jump. This situation is precluded in the result of convergence in [1].

Time stepping scheme for Linear Complementarity Systems (LCS)

Remark

Following the remark 43, we can note some similarities with the catching–up algorithm. Two main differences have however to be noted:

- the first one is that the sweeping process can be equivalent to a LCS under the condition $C = B^T$. In this way, the previous time-stepping scheme extend the catching–up algorithm to more general systems.
- The second major discrepancy is a s follows. The catching–up algorithm does not approximate directly the time-derivative $\dot{x}$ as

$$\dot{x}(t) \approx \frac{x(t+h) - x(t)}{h} \tag{29}$$

  but directly the measure of the time interval by

$$dx(]t, t+h]) = x^+(t+h) - x^+(t) \tag{30}$$

  This difference leads to a consistent time-stepping scheme if the state possesses an initial jump. A direct consequence is that the primary variable μ_{k+1} in the catching up algorithm is homogeneous to a measure of the time-interval.

The Moreau's catching–up algorithm for the first order sweeping process

Principle of Time–stepping schemes

1. A unique formulation of the dynamics is considered. For instance, for a first order sweeping process, a dynamics in terms of measures.

$$\begin{cases} -du = dr \\ dr \in N_{K(t)}(u^+(t)) \end{cases} \tag{31}$$

2. The time-integration is based on a consistent approximation of the equations in terms of measures. For instance,

$$\int_{]t_k,t_{k+1}]} du = \int_{]t_k,t_{k+1}]} du = (v^+(t_{k+1}) - v^+(t_k)) \approx (u_{k+1} - u_k) \tag{32}$$

3. Consistent approximation of measure inclusion.

$$-dr \in N_{K(t)}(u^+(t)) \tag{33}$$

$\rightarrow$

$$\begin{cases} p_{k+1} \approx \int_{]t_k,t_{k+1}]} dr \\ p_{k+1} \in N_{K(t)}(u_{k+1}) \end{cases} \tag{34}$$

The Moreau's catching–up algorithm for the first order sweeping process

Catching–up algorithm

Let us consider the first order sweeping process with a B.V. solution:

$$\begin{cases} -du \in N_{K(t)}(u^+(t)) \ \ (t \geqslant 0), \\ \ \ u(0) = u_0. \end{cases} \tag{35}$$

The so-called "Catching–up algorithm" is defined in [5]:

$$-(u_{k+1} - u_k) \in \partial\psi_{K(t_{k+1})}(u_{k+1}) \tag{36}$$

where u_k stands for the approximation of the right limit of u at t_k.
By elementary convex analysis, this is equivalent to:

$$u_{k+1} = prox(K(t_{k+1}), u_k). \tag{37}$$

The Moreau's catching–up algorithm for the first order sweeping process

Difference with an backward Euler scheme

- the catching–up algorithm is based on the evaluation of the measure du on the interval $]t_k, t_{k+1}]$, i.e. $du(]t_k, t_{k+1}]) = u^+(t_{k+1}) - u^+(t_k)$.
- the backward Euler scheme is based on the approximation of $\dot{u}(t)$ which is not defined in a classical sense for our case.

When the time step vanishes, the approximation of the measure du tends to a finite value corresponding to the jump of u. Particularly, this fact ensures that we handle only finite values.

Higher order approximation

Higher order schemes are meant to approximate the n-th derivative of the discretized function. Non sense for a non smooth solution.

Mathematical results

For Lipschitz and RCBV sweeping processes, convergence and consistency results are based on the catching–up algorithm.
[4, 3]

The Moreau's catching–up algorithm for the first order sweeping process

Time-independent convex set K

Let us recall now the UDI

$$-(\dot{x}(t) + f(x(t)) + g(t)) \in \mathbb{N}_K(x(t)), \quad x(0) = x_0 \tag{38}$$

In the same way, the inclusion can be discretized by

$$-(x_{k+1} - x_k) + h(f(x_{k+1}) + g(t_{k+1})) = \mu_{k+1} \in \mathbb{N}_K(x_{k+1}), \tag{39}$$

- In this discretization, an evaluation of the measure dx by the approximates value μ_{k+1}.
- If the initial condition does not satisfy the inclusion at the initial time, the jump in the state can be treated in a consistent way.

The Moreau's catching–up algorithm for the first order sweeping process

Time-independent convex set $K = \mathbb{R}^n_+$

The previous problem can be written as a special non linear complementarity problem:

$$\begin{cases} (x_{k+1} - x_k) - h(f(x_{k+1}) + g(t_{k+1})) = \mu_{k+1} \\ 0 \leqslant x_{k+1} \perp \mu_{k+1} \geqslant 0 \end{cases} \tag{40}$$

If $f(x) = Ax$ we obtain the following LCP(q,M):

$$\begin{cases} (I - hA)x_{k+1} - (x_k + hg(t_{k+1})) = \mu_{k+1} \\ 0 \leqslant x_{k+1} \perp \mu_{k+1} \geqslant 0 \end{cases} \tag{41}$$

with $M = (I - hA)$ and $q = -(x_k + hg(t_{k+1}))$.

Remark

It is noteworthy that the value μ_{k+1} approximates the measure $d\lambda$ on the time interval rather than directly the value of λ.

The Moreau's catching–up algorithm for the first order sweeping process

Remark

Particularly, if the set K is polyhedral by :

$$K = \{x, Cx \geqslant 0\} \tag{42}$$

If a constraint qualification holds, the DI (38) in the linear case $f(x) = -Ax$ is equivalent the the following LCS:

$$\begin{cases} \dot{x} = Ax + C^T\lambda \\ y = Cx \\ 0 \leqslant y \perp \lambda \geqslant 0 \end{cases} \tag{43}$$

In this case, the catching–up algorithms yields:

$$\begin{cases} x_{k+1} - x_k = hAx_{k+1} + C^T\mu^{k+1} \\ y_{k+1} = Cx_{k+1} \\ 0 \leqslant y_{k+1} \perp \mu_{k+1} \geqslant 0 \end{cases} \tag{44}$$

We will see later in Section 2 that this discretization is very similar to the discretization proposed by [1] for LCS.

Time stepping scheme for Differential Variational Inequalities (DVI)

Time stepping scheme for Differential Variational Inequalities (DVI)

In [6], several time-stepping schemes are designed for DVI which are separable in u,

$$\dot{x}(t) = f(t, x(t)) + B(x(t), t)u(t) \quad (45)$$

$$u(t) = \mathrm{SOL}(K, G(t, x(t)) + F(\cdot)) \quad (46)$$

We recall that the second equation means that $u(t) \in K$ is the solution of the following VI

$$(v - u)^T.(G(t, x(t)) + F(u(t))) \geqslant 0, \forall v \in K \quad (47)$$

Two cases are treated with a time-stepping scheme: the Initial Value Problem(IVP) and the Boundary Value Problem(BVP).

Time stepping scheme for DVI. IVP case.

IVP case.

$$\dot{x}(t) = f(t, x(t)) + B(x(t), t)u(t) \quad (48)$$
$$u(t) = \mathrm{SOL}(K, G(t, x(t)) + F(\cdot)) \quad (49)$$
$$x(0) = x_0 \quad (50)$$

The proposed time-stepping method is given as follows

$$x_{k+1} - x_k = h\,[f(t_k, \theta x_{k+1} + (1-\theta)x_k) + B(x_k, t_k)u_{k+1}] \quad (51)$$
$$u_{k+1} = \mathrm{SOL}(K, G(t_{k+1}, x_{k+1}) + F(\cdot)) \quad (52)$$

Time stepping scheme for DVI. IVP case.

Explicit scheme $\theta = 0$

An explicit discretization of $\dot{x}$ is realized leading to the one-step non smooth problem

$$x_{k+1} = x_k + h[f(t_k, x_k) + B(x_k, t_k)u_{k+1}] \tag{53}$$

where u_{k+1} solves the $VI(K, F_{k+1})$ with

$$F_{k+1}(u) = G(t_{k+1}, h[f(t_k, x_k) + B(x_k, t_k)u]) + F(u) \tag{54}$$

Remark

- In the last VI, the value u_{k+1} can be evaluated in explicit way with respect to x_{k+1}.
- It is noteworthy that even in the explicit case, the VI is always solved in a implicit ways, i.e. for x_{k+1} and u_{k+1}.

Semi-implicit scheme

If $\theta \in]0, 1]$, the pair u_{k+1}, x_{k+1} solves the $VI(\mathbb{R}^n \times K, F_{k+1})$ with

$$F_{k+1}(x, u) = \begin{bmatrix} x - x_k - h[f(t_k, \theta x + (1-\theta)x_k) + B(x_k, t_k)u] \\ G(t_{k+1}, x) + F(u) \end{bmatrix} \tag{55}$$

Time stepping scheme for DVI. IVP case.

Convergence results

In [6], the convergence of the semi-implicit case is proved. For that, a continuous piecewise linear function, x^h is built by interpolation of the approximate values x_k,

$$x^h(t) = x_k + \frac{t - t_k}{h}(x_{k+1} - x_k), \forall t \in [t_k, t_k + 1] \tag{56}$$

and a piecewise constant function u^h is build such that

$$u^h(t) = u_{k+1}, \forall t \in]t_k, t_k + 1] \tag{57}$$

It is noteworthy that the approximation x^h is constructed as a continuous function rather than u^h may be discontinuous.

Time stepping scheme for DVI. IVP case.

Convergence results

The existence of a subsequence of u_h, x_h denoted by u^{h_ν}, x^{h_ν} such that

- x^{h_ν} converges uniformly to $\hat{x}$ on $[0, T]$
- u^{h_ν} converges weakly to $\hat{u}$ in $\mathcal{L}^2(0, T)$

under the following assumptions:

1. f and G are Lipschitz continuous on $\Omega = [0, T] \times \mathbb{R}^n$,
2. B is a continuous bounded matrix-valued function on Ω,
3. K is closed and convex (not necessarily bounded)
4. F is continuous
5. $SOL(K, q + F) \neq \emptyset$ and convex such that $\forall q \in G(\Omega)$, the following growth condition holds

$$\exists \rho > 0, \sup\{\|u\|, u \in \mathrm{SOL}(K, q + F)\} \leqslant \rho(1 + \|q\|) \tag{56}$$

   This assumption is used to prove that a pair u_{k+1}, x_{k+1} exists for the VI (55). This assumption of the type "growth condition" is quite usual to prove existence of solution of VI through fixed-point theorem (see [2]).

Time stepping scheme for DVI. IVP case.

Convergence results

Furthermore, under either one of the following two conditions:

- $F(u) = Du$ (i.e. linear VI) for some positive semidefinite matrix, D
- $F(u) = \Psi(Eu)$, where Ψ is Lipschitz continuous and $\exists c > 0$ such that

$$\|Eu_{k+1} - E_k\| \leqslant ch \tag{56}$$

all limits $(\hat{x}, \hat{u})$ are weak solutions of the initial-value DVI.
➜ This proof convergence provide us with an existence result for such DVI with a separable in u.
The linear growth condition which is strong assumption in most of practical case can be dropped. In this case, some monotonicity assumption has to be made on F and strong monotonicity assumption on the map $u \mapsto G(t,x) \circ (r + B(t,x)u)$ for all $t \in [0,T], x \in \mathbb{R}^n, r \in \mathbb{R}^n$. We refer to [6] for more details. If $G(x,t) = Cx$, the last assumption means that CB is positive definite.

Time stepping scheme for DVI. BVP case

BVP case

Let us consider now the Boundary value problem with linear boundary function

$$\begin{aligned} \dot{x}(t) &= f(t, x(t)) + B(x(t), t)u(t) && (57)\\ u(t) &= \mathrm{SOL}(K, G(t, x(t)) + F(\cdot)) && (58)\\ b &= Mx(0) + Nx(T) && (59) \end{aligned}$$

The time-stepping proposed by [6] is as follows :

$$\begin{aligned} x_{k+1} - x_k &= h\left[f(t_k, \theta x_{k+1} + (1-\theta)x_k) + B(x_k, t_k)u_{k+1}\right], \quad k \in \{0, \ldots, N-1\} && (60)\\ u_{k+1} &= \mathrm{SOL}(K, G(t_{k+1}, x_{k+1}) + F(\cdot)), \quad k \in \{0, \ldots, N-1\} && (61)\\ & && (62) \end{aligned}$$

plus the boundary condition

$$b = Mx_0 + Nx_N \quad (63)$$

Comments

The system is henceforth a coupled and large VI for which the numerical solution is not trivial.

Time stepping scheme for DVI. BVP case

Convergence results

The existence of the discrete time-trajectory is ensured under the following assumption :

1. F monotone and VI solutions have linear growth
2. the map $u \mapsto G(t,x) \circ (r + B(t,x)u)$ is strongly monotone
3. $M + N$ is non singular and satisfies

$$\exp(T\psi_x) < 1 + \frac{1}{\|(M+N)^{-1}N\|}$$

where x ¿ 0 is a constant derived from problem data.

The convergence of the discrete time trajectory is proved if F is linear.

Time stepping scheme for Differential Variational Inequalities (DVI)

General remarks

- The time–stepping scheme can be viewed as extension of the DCS, the UDI and the Moreau's catching up algorithm.
- But, the scheme is more a mathematical discretization rather a numerical method. In practice, the numerical solution of a VI is difficult to obtain when the set K is unstructured.
- The case K is polyhedral is equivalent to a DCS.

Time stepping scheme for Higher order Moreau's sweeping process

Time stepping scheme for Higher order Moreau's sweeping process

...

Advantages and disadvantages. Time-stepping

Advantages

- Compact formulation which allow existence and uniqueness results
- Dissipativity and monotonicity properties

Disadvantages :

- More difficult mathematical framework
- Low order accuracy

Lead to Time–stepping integration schemes (without event-handling) suitable :

- Large systems with a large number of events
- Accumulation of events in finite time
- Convergence results and Existence proofs

Thank you for your attention.

[1] K. Camlibel, W.P.M.H. Heemels, and J.M. Schumacher. Consistency of a time-stepping method for a class of piecewise-linear networks. *IEEE Transactions on Circuits and Systems I*, 49:349–357, 2002.

[2] F. Facchinei and J. S. Pang. *Finite-dimensional Variational Inequalities and Complementarity Problems*, volume I & II of *Springer Series in Operations Research*. Springer Verlag NY. Inc., 2003.

[3] M. Kunze and M.D.P. Monteiro Marquès. An introduction to Moreau's sweeping process. In B. Brogliato, editor, *Impact in Mechanical systems: Analysis and Modelling*, volume 551 of *Lecture Notes in Physics*, pages 1–60. Springer, 2000.

[4] M.D.P. Monteiro Marques. *Differential Inclusions in Nonsmooth Mechanical Problems. Shocks and Dry Friction*. Progress in Nonlinear Differential Equations and their Applications, vol.9. Birkhauser, Basel, 1993.

[5] J.J. Moreau. Evolution problem associated with a moving convex set in a Hilbert space. *Journal of Differential Equations*, 26:347–374, 1977.

[6] D. Pang, J.-S. an Stewart. Differential variational inequalities. *Mathematical Programming A.*, 2006. submitted, preprint available at http://www.cis.upenn.edu/davinci/publications/pang-stewart03.pdf.