DynIBEX: a Differential Constraint Library for Studying Dynamical Systems

Julien Alexandre Dit Sandretto, Alexandre Chapoutot

DynIBEX: a Differential Constraint Library for Studying Dynamical Systems

Julien Alexandre dit Sandretto and Alexandre Chapoutot
U2IS, ENSTA ParisTech, Université Paris-Saclay, 828 bd des Maréchaux, Palaiseau, France
Contact: alexandre@ensta.fr

A free open-source library combining validated numerical integration methods with a constraint programming approach, as a plugin of IBEX library with:
- validated integration methods based on Runge-Kutta for Ordinary Differential Equations (ODEs) and Differential Algebraic Equations (DAEs)
- operators for satisfaction problems on dynamical systems
http://perso.ensta-paristech.fr/~chapoutot/dynibex/

1-Differential Equation Supported

Parametric and Constrained :

$$\begin{cases} \dot{\mathbf{y}} = F(t, \mathbf{y}, \mathbf{x}, \mathbf{p}, \mathbf{u}) \\ 0 = G(t, \mathbf{y}, \mathbf{x}, \mathbf{p}, \mathbf{u}) \\ 0 = H(t, \mathbf{y}, \mathbf{p}, \mathbf{u}) \\ \text{with } \mathbf{y}(0) \in [\mathbf{y}_0],\ \mathbf{x}(0) \in [\mathbf{x}_0],\ \mathbf{p} \in [\mathbf{p}],\ \mathbf{u} \in [\mathbf{u}] \text{ and } t \in [0, t_{\text{end}}] \ . \end{cases}$$

With:

$F : R \times R^n \times R^m \times R^r \times R^s \mapsto R^n$ (Differential Part)
$G : R \times R^n \times R^m \times R^r \times R^s \mapsto R^m$ (DAE part)
$H : R \times R^n \times R^r \times R^s \mapsto R^c$ (Additive Constraints)

Note that:

- m=0: ODE, otherwise DAE (index-1 Hessenberg)
- c>0: Constrained ODE or DAE
- r>0: Interval Parameters (uncertainties on measures, modeling, parameters)
- s>0: Embedded control

2-Existing tools in IBEX

A library for interval constraint programming:

- Interval and affine arithmetic
- Function evaluation
- Contractor programming (HC4, Newton, ...)
- Bisection (Branch & Prune)
- Set operators (inclusion, intersection, ...)

DynIBEX applications

Controller synthesis (PID, switched systems), design, viability kernel, predictive control, sensitivity analysis

3-Validated Simulation with Runge-Kutta

- Proof of existence and uniqueness of solution for ODEs and DAEs
- Local truncation error computation for any Runge-Kutta method (implicit or explicit)
- Combined with contractors (HC4)

=> Validated Runge-Kutta: Heun, RK4, Lobatto, Radau3, ...

4- Verification of temporal constraints

$$\begin{aligned} \text{Stayed in } \mathcal{A} \text{ (until } \tau) : &\quad y(t) \subset Int(\mathcal{A}), \forall t\ (\forall t < \tau) \\ \text{Included in } \mathcal{A} \text{ at } \tau : &\quad y(\tau) \subset Int(\mathcal{A}) \\ \text{Has crossed } \mathcal{A} \text{ (before } \tau) : &\quad \exists t, y(t) \cap \Box\mathcal{A} \neq \emptyset\ (\exists t < \tau) \\ \text{Gone out } \mathcal{A} \text{ (before } \tau) : &\quad \exists t, y(t) \cap \Box\mathcal{A} = \emptyset\ (\exists t < \tau) \\ \text{Has reached } \mathcal{A} : &\quad y(t_{end}) \cap \Box\mathcal{A} \neq \emptyset \\ \text{Finished in } \mathcal{A} : &\quad y(t_{end}) \subset Int(\mathcal{A}) \end{aligned}$$

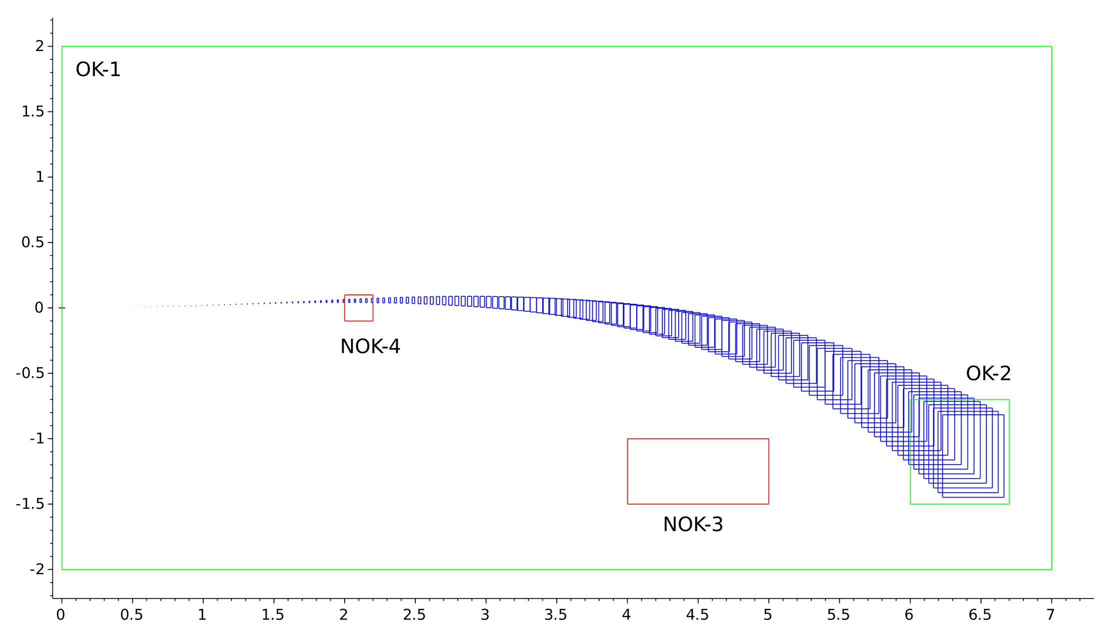
5- Example of simulation and constraints

Model of an Hovercraft:

$$\begin{cases} \dot{u} = vr + \frac{1}{m}\left(F_u - \mu_{sec} N f(u) \frac{u}{\sqrt{u^2+v^2}} - \frac{1}{2}\rho C_d S_u \sqrt{u^2+v^2}\,u - D_{11} u\right) \\ \dot{v} = -ur + \frac{1}{m}\left(\frac{1}{4A} C_L S_v F_u \delta - \mu_{sec} N f(v) \frac{v}{\sqrt{u^2+v^2}} - \frac{1}{2}\rho C_d S_v \sqrt{u^2+v^2}\,v - D_{22} v\right) \\ \dot{r} = \frac{1}{I_z}\left(-L \frac{1}{4A} C_L S_g F_u \delta - D_{33} r\right) \\ \dot{x} = cos(\psi)u - sin(\psi)v \\ \dot{y} = sin(\psi)u + cos(\psi)v \\ \dot{\psi} = r \end{cases}$$

OK-1: Stayed in safe zone
OK-2: Finished in goal
NOK-3: Avoid obstacle
NOK-4: Not in forbidden zone at τ

=> Design, control synthesis, parameter identification, model simplification, ...



[1] J.A.D. Sandretto and A. Chapoutot, Validated Explicit and Implicit Runge-Kutta Methods, Reliable Computing, 2016
[2] J.A.D. Sandretto and A. Chapoutot, Validated Simulation of Differential Algebraic Equations, Reliable Computing, 2016
[3] J.A.D. Sandretto, A. Chapoutot, O. Mullier, Tuning PI controller in non-linear uncertain closed-loop systems with interval analysis, SynCoP'2015
[4] D. Monnet, L. Jaulin, J. Ninin, A. Chapoutot, J.A.D. Sandretto, Viability kernel computation based on interval methods, SWIM'2015
[5] A. Le Coënt, J.A.D. Sandretto, A. Chapoutot, L. Fribourg, Control of Nonlinear Switched Systems Based on Validated Simulation, SNR'2016