Real Time Aircraft Simulation using HLA standard: an overview

Simulation in Aerospace 2011, Toulouse

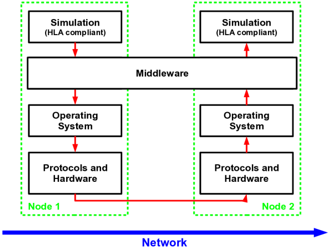
J.-B. Chaudron[1,2], D. Saussié[1], P. Siron[1,2], M. Adelantado[2]

[1]Institut Supérieur de l'Aéronautique et de l'Espace, DMIA, Toulouse
[2]ONERA, DTIM, Toulouse

Outline

Problem statement

- Modern flight simulators require a high level of computing ;
- Distributed computing paradigm proposes a high performance solution ;
- This led to development of standards (e.g. CORBA, HLA) to consistently face the problems raised by distribution ;



- Middleware : connectivity software operating as an intermediary between distributed applications.

HLA for real-time ? (1/2)

- HLA is a well-established technique used in the man-machine system area for training and research ;
- However, works to include real-time specifications and properties to HLA standard are less advanced than other ones (RT-CORBA, DDS,...) ;
  1. HLA does not provides interfaces to specify end-to-end prediction requirement for federate ;
  2. HLA does not allow management of underlying Operating(s) System(s) in terms of priority and resource ;
  3. HLA only supports two transportation types (reliable and best-effort).

HLA for real-time ? (2/2)

- Different works have been proposed to enhance services provided by the RTI (Fujimoto, Mc Lean, Boukerche) :
  1. Multi-threaded synchronous process for RTI ;
  2. Real-time Optimized RTI services (time management, data distribution, ...) ;
  3. Quality of service communication with specific protocols (RSVP, ...) ;
  4. Use of a real-time operating system to allow preemptive priority scheduling.
- However, no work proposes a complete analysis from simulation requirements to implementation ;
- Most of all, the RTI used and its associated algorithms are never clearly presented.

PRISE project

- PRISE (*Plate-forme de Recherche et d'Ingénierie des Systèmes Embarqués*) ;
- Study of new embedded system concepts and techniques through a special hardware and software environment ;
- Connecting real actors : actuators, sensors or real embedded computers in the simulation loop (*Hardware-in-the-loop*) and also human user (*Human-in-the-loop*).

Action Levels

The temporal properties of distributed real-time simulation are obtained from a complex combination of different levels :

1. ***Hardware level*** : the physical infrastructure (type of computers, type of networks and distribution topology) ;
2. ***Software level*** : the software infrastructure (operating systems and communication protocols) ;
3. ***Middleware level*** : the HLA middleware used and its specific distributed algorithms ;
4. ***Application level*** : characteristics of the HLA simulation ;
5. ***Formal level*** : formal methods applied to assess the good behavior of the whole system

PRISE : Hardware Level

The PRISE platform hardware architecture is composed of :

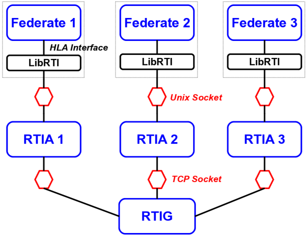
- 4 real-time nodes with Opteron 6 core processors ;
- 2 graphical HP stations with high performance GP-GPU ;
- 1 Ethernet Gigabit switch on a dedicated network ;
- 2 input organs (Yoke/Throttle/Pedal systems) ;
- A distributed clock technology allowing same clock reference to each real-time node.

PRISE : Software and Middleware Level

- Software (OS) : Each real-time node run with Linux Red Hawk Operating system (compliant with POSIX Real time standard).
- Software (Network) : We use classical UDP and TCP protocols ;
- Middleware : We will rely on and extend our Open Source RTI called CERTI because we have a complete control on its implementation.

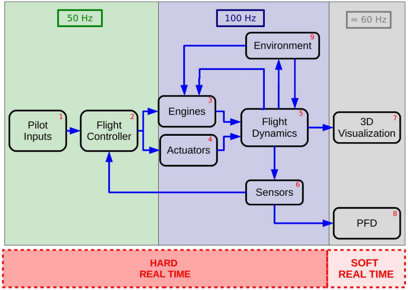
Middleware CERTI

- Open-Source middleware RTI compliant with HLA standard (DoD 1.3 and IEEE 1516) ;
- Original architecture of communicating processes ;
- Avalaible under several operating systems including Linux and Windows ;



HLA aircraft federation

- Composed of 9 federates ;
- Each representing a specific part of the aircraft or environment ;



- The federation is composed of two types of applications according to real-time requirements.

Flight Dynamics Federate

- Complete modeling of flight equations
- 12 ordinary differential equations divided in 4 groups :
  - Position (3)
  - Orientation (3)
  - Translational velocity (3)
  - Angular rate (3)
- Four numerical methods implemented for the integration (Euler, Trapèze, Adams-Bashforth 2 -3)
- Modeling of aerodynamic forces can still be enriched

Flight Dynamics Federate

- Subscribes to :
  - Actuators federate (control surface deflections)
  - Engines federate (thrust)
  - Environment federate (pressure, temperature, air density, speed of sound, Wind/Turbulence)
- Publishes for :
  - Sensors federate (position/orientation/translational velocity, angular rates & accelerations)
  - 3D Visualization federate (position/orientation)
  - Engines federate (Mach number)

Specific FOM

- Messages exchanged between federates are based upon the global federation object model (FOM) ;
- FOM provides the global definitions of all objects (and their attributes) available for all federates.

Attribute Table							
Object	Attribute	Data-type	Card.	Units	Update Type	T/A	U/R
AIRCRAFT_ POSITION	LONGITUDE	double	1	m	periodic	TA	UR
	LATITUDE	double	1	m	periodic	TA	UR
	ALTITUDE	double	1	m	periodic	TA	UR
AIRCRAFT_ ORIENTATION	PHI	double	1	rad	periodic	TA	UR
	THETA	double	1	rad	periodic	TA	UR
	PSI	double	1	rad	periodic	TA	UR
AIRCRAFT_ VELOCITY	SPEED	double	1	m/s	periodic	TA	UR
	VERTICAL_SPEED	double	1	m/s	periodic	TA	UR
	ALPHA	double	1	rad	periodic	TA	UR
	BETA	double	1	rad	periodic	TA	UR
	MACH	double	1	$\varnothing$	periodic	TA	UR
AIRCRAFT_ ANGULAR_ VELOCITY	ROLL	double	1	rad/s	periodic	TA	UR
	PITCH	double	1	rad/s	periodic	TA	UR
	YAW	double	1	rad/s	periodic	TA	UR
AIRCRAFT_ ACCELERATION	ACC_X	double	1	m/s^2	periodic	TA	UR
	ACC_Y	double	1	m/s^2	periodic	TA	UR
	ACC_Z	double	1	m/s^2	periodic	TA	UR

Execution modes

We distinguish two different run-time modes for our flight simulator :

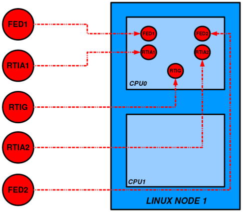
- the **Data flow** model :
  - Only scheduled by the communication flow between each federate ;
  - This approach could only be used on synchronous computers (RCIM synchronized nodes).
- the **Time management** model :
  - Scheduled by time management principles and algorithms ;
  - Ensure a consistent temporal behaviour on a common time reference : the *simulated time*.
  - This approach remains the best way to maintain consistency between federates located on asynchronous computers.

CERTI real-time analysis

- Originally, CERTI has no mechanism for taking into account quality of service or an end-to-end predictability ;
- It does not handle events differently according to a priority and it uses no predictability mechanism whatsoever at the network or the operating system ;
- However, a key benefit is to master the implementation of RTI used and thus able to incorporate changes in the source code.

CERTI updates (1/2)

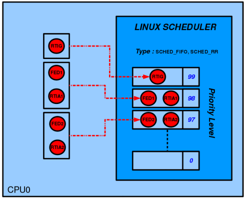
- We first implemented functions in CERTI that allow to use affinity mechanism ;
- CPU affinity is a scheduler property that assigns a process (federate, RTIA or RTIG) to a given set of CPUs on the system ;



- The Linux scheduler will honor the given CPU affinity and the process will not run on any other CPU.

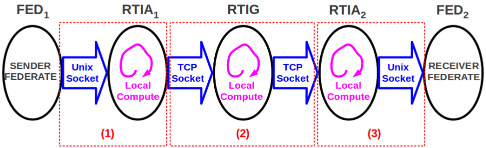
CERTI updates (2/2)

- We add another interface to allow the management of priority and scheduler for CERTI processes (i.e federates, RTIAs and RTIG) ;
- Real-time Linux operating systems allow 100 priority levels to run critical tasks ;



- We use the `mlockall` mechanism for each process to disables memory paging into the address space.

CERTI WCTT



Messages sizes	***Conf. 1***		***Conf. 2***		***Conf. 3***	
	N	***RT***	***N***	***RT***	***N***	***RT***
1000 bytes	**0.353**	**0.262**	**0.281**	**0.244**	**0.286**	**0.247**
5000 bytes	**0.406**	**0.316**	**0.411**	**0.336**	**0.422**	**0.367**
10000 bytes	**0.422**	**0.373**	**0.478**	**0.426**	**0.522**	**0.487**
50000 bytes	**1.066**	**0.952**	**1.372**	**1.224**	**1.607**	**1.536**

Scheduling Formal model (1/2)

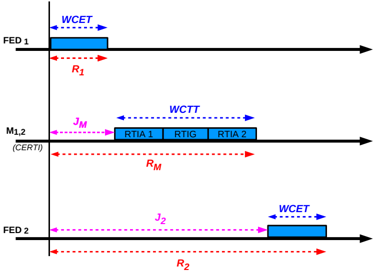
- Real-time simulations are usually validated by experiments rather than formal models and schedulability tests ;
- Formal models compliant with schedulability techniques are essential to validate hard real-time part of our simulator ;
- Federates simulate periodic processes which communicate by using HLA principles ;
- We could adapt a formal model combining tasks and messages models for Data Flow model ;
- Time management services involve more complex mechanisms and therefore are difficult to model as it is by scheduling formal models.

Scheduling Formal model (2/2)

- A periodic task τ_i (*i.e.* a federate Fed_i) is a quadruplet $< r_i, C_i, D_i, P_i >$ with :
  - r_i the time of initial activation of the task ;
  - C_i the worst case execution time (WCET) ;
  - D_i the deadline ;
  - P_i the period.
- A periodic messages $m_{i,j}$ (*i.e.* an HLA exchange between two federates Fed_i and Fed_j) is a quadruplet $< r_{m_{i,j}}, C_{m_{i,j}}, D_{m_{i,j}}, P_{m_{i,j}} >$ with :
  - $r_{m_{i,j}}$ the time of initial send of the message by a federate ;
  - $C_{m_{i,j}}$ the worst case transit time (WCTT) ;
  - $D_{m_{i,j}}$ the deadline for message transmission ;
  - $P_{m_{i,j}}$ the period of production for message.
- We use this formalism to describe our Data Flow application model by using Tindell and Clark holistic.

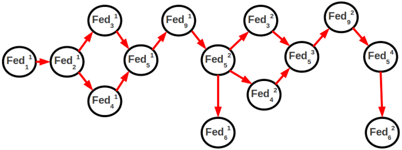
Tindell and Clark illustration

- J_i represents the delay due to data dependencies ;
- R_i represents the longest time ever taken by a task to complete its required computation ;
- Basic example with two federates



Data Flow validation

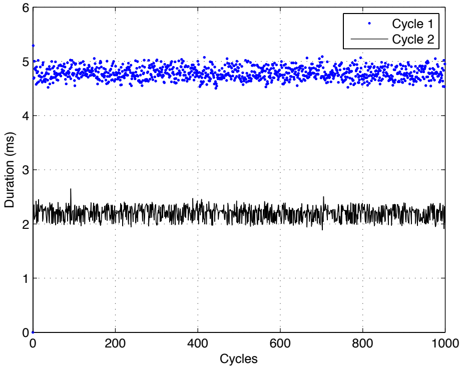
- We obtain a graph of 14 tasks and 15 messages :



- Every C_i of tasks are relevant to Federate WCET mesurements on PRISE Platform ;
- Every $C_{m_{i,j}}$ of messages are relevant to CERTI WCTT measurements on PRISE Platform ;
- We check that Response-times R_i of each task and messages are always smaller than corresponding deadlines D_i.

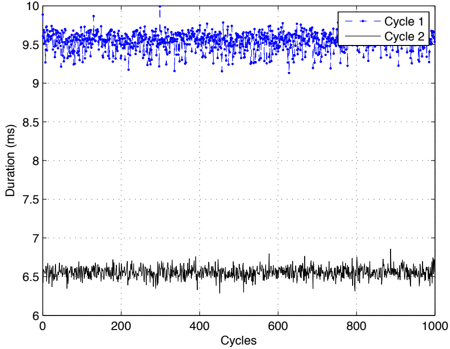
Data Flow

The data flow execution model has a good behavior for real-time purpose.



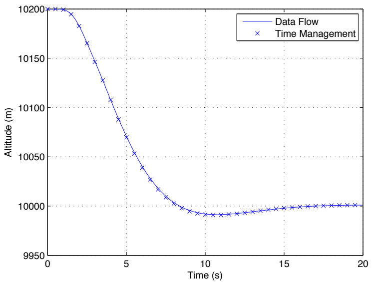
Time management

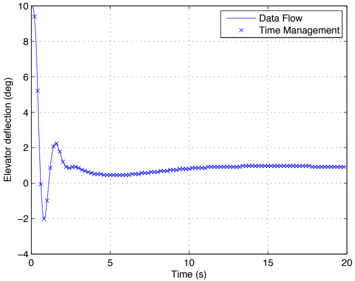
The time management services generate some overhead but are acceptable for real-time.



Simulation results

- Test of the altitude hold automatic pilot function
- Similar responses for both approaches (as expected !)





Conclusion

- First step of PRISE project completed :
- Implementation of a realistic aircraft model and its environment ;
- Extension of HLA standard to real-time world ;
- We also hope that defined FOM will become a reference FOM for this research domain.

Perpectives

- Flight formation and interaction with already existing flight simulators ;
- Real actuators and sensors integration in the loop ;
- Extend scheduling formal studies to time management model.