A statistical inference model for the dynamic range estimation of LTI systems

Laurent-Stéphane Didier
Alexandre Chapoutot
Fanny Villers

September 24, 2012

Context

Embedded software

- numerical filter (FIR, IIR, etc.)
- control applications
- executed on limited hardware (e.g. no FPU)

Development of embedded systems

- high-level design describes algorithms with floating-point numbers
- implementation is adapted to take into account hardware limitations

Transformation of floating-point programs into fixed-point equivalent ones

- Evaluation of the range
- Evaluation of the numerical precision

Motivation

Our goal:

- Define a method to quickly evaluate the range of the variables.
- Compute the probability of overflow for tolerant applications

Our new method is:

- applied on **design level** (Simulink models)
- assumes that the studied designs are simulable
- allows "black boxes"
- based on **statistical models**
- defined to **estimate dynamic range of floating-point variables**

Previous attempts

Evaluate the behaviour of internal data using input stimuli samples and simulations

Simple method:

- Simulate the model several times with a set of simulations data
- Catch the minima and the maxima
- Keep the smallest minimum and the greatest maximum
- Dependent from the set of simulation

Kim , Kum and Sung method:

- Similar to the simple method; evaluation of statistical parameters
- Examine the mean μ, the standard deviation σ and the kurtosis k
- Statistical estimation of the radius R of the range
  - For one simulation

$$R = |\mu| + (k + 4)\sigma$$

  - For several simulations

$$R = |\mu_{max} + 0.1(\mu_{max} - \mu_{min}| + (k_{max} + 0.1(k_{max} - k_{min}) + 4) \times (\sigma_{max} + 0.1(\sigma_{max} - \sigma_{min}))$$

What is the statistical model?

Suppose we have an i.i.d. sequence of random variables, X_1, X_2, X_3, $\cdots$.
Let $M_n = max(X_1, ..., X_n)$
If there are parameters $a_n > 0$, $_n$ and a non-degenerate probability distribution $G(x)$ such that

$$P\left(\frac{M_n - b_n}{a_n} \leq x\right) \to G(x)$$

then G is the Generalized Extrem Value Distribution.

GEVD cumulative function is:

$$G(x;\mu,\sigma,\xi) = exp\left\{-\left[1+\xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\xi}\right\}$$

with,

- μ is the **location** parameter
- σ is the **scale** parameter
- ξ is the **shape** parameter

What are we looking at?

Inferential statistics: estimates numerical characteristics about a population

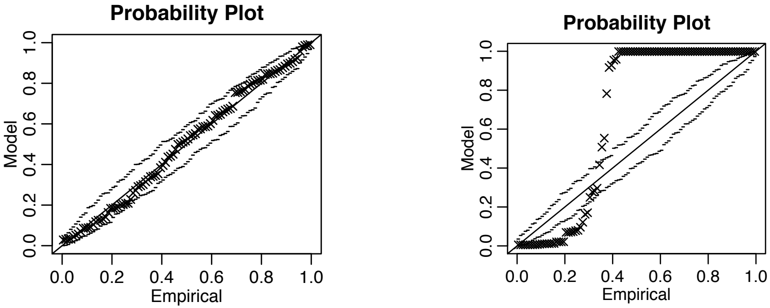
Our approach:

- perform n simulations of length d with random inputs
- collect the **maximum** of each variable from the n random simulations S_i

$$\begin{aligned} M^{(1)} &= \max\left(S_1^{(1)}, S_2^{(1)}, \dots, S_d^{(1)}\right) \\ M^{(2)} &= \max\left(S_1^{(2)}, S_2^{(2)}, \dots, S_d^{(2)}\right) \\ \vdots &= \vdots \\ M^{(n)} &= \max\left(S_1^{(n)}, S_2^{(n)}, \dots, S_d^{(n)}\right) \end{aligned}$$

How can we verify the hypothesis?

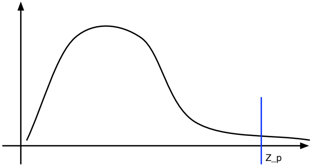
Graphical technique to assess if a data set follows a given distribution.
Here our data set is the $\left(M^{(1)}, M^{(2)}, \ldots, M^{(n)}\right)$



The data are plotted against a theoretical distribution: **the points should form a straight line**
Given by every statistical tools such as R.

How are we using the statistical model?

We want to compute z_p such that $P(M > z_p) = p$
i.e. the value z_p which will be exceeded by the maximum M with a probability p.

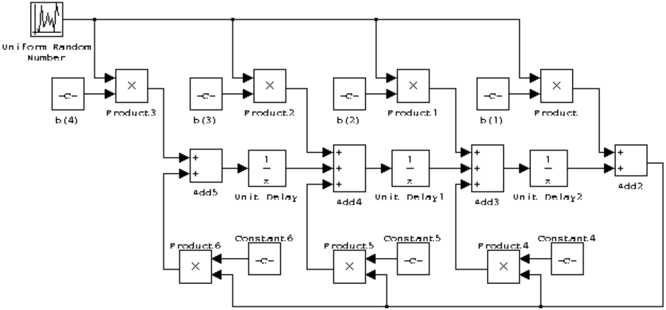


Can be computed with:

$$z_p = \begin{cases} \mu + \frac{\sigma}{\xi}\left(1 - \left(-\log(1-p)\right)^{-\xi}\right) & \text{if } \xi \neq 0 \\ \mu + \sigma \log\left(-\log(1-p)\right) & \text{otherwise} \end{cases}$$

- With $\hat{\mu}$, $\hat{\sigma}$ and $\hat{\xi}$ an estimation of the GEVD parameters, we compute an estimation $\hat{z}_p$

A typical LTI system



Uses previously computed results

$$y(n) = \frac{1}{a_0}\left(\sum_{i=0}^{P} b_i x(n-i) - \sum_{j=0}^{Q} a_j y(n-j)\right)$$

Infinite Impulse Response Filter

$$y(n) = \frac{1}{a_0}\left(\sum_{i=0}^{P} b_i x(n-i) - \sum_{j=0}^{Q} a_j y(n-j)\right)$$

Remarks

- Not really independent
- Work if the dependency is "limited"

- Evaluate the impact of the dependency
- Modelize LTI systems with ARMA stochastic process.

ARMA model

Auto Regressive/ Moving Average

A time series x_t is ARMA(p, q) if it is stationary and

$$x_t = \underbrace{\phi_1 x_{t-1} + \cdots + \phi_p x_{t-p} + \omega_t}_{\text{Autoregressive part}} + \underbrace{\theta_1 \omega_{t-1} + \cdots + \theta_q \omega_{t-q}}_{\text{Moving Average part}}$$

where θ_i and ϕ_i are constants, ω_i are white noises.

- For the range estimation, inputs $x(n)$ of the LTI system are white noises.

$$y(n) = \frac{1}{a_0} \left(\sum_{i=0}^{P} b_i x(n-i) - \sum_{j=0}^{Q} a_j y(n-j) \right)$$

- We consider the output is at least weak stationary

Weak stationarity

Definition

A time serie is weakly stationary if:

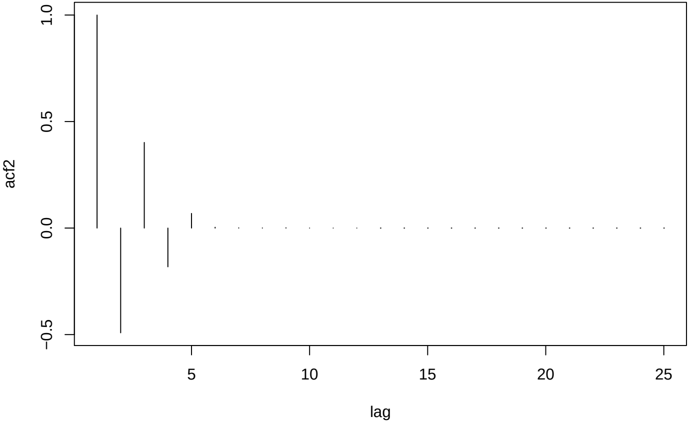
- its mean μ_t is constant and does not depend on t
- the autocovariance function $\gamma(s,t) = E[(x_s - \mu_s)(x_t - \mu_t)]$ only depends only $t - s$.

- We consider on $\gamma(t+h, t)$ and note $\gamma(h)$.
- We focus on the autocorrelation function

$$\rho(h) = \frac{\gamma(h)}{\gamma(0)}$$

- Gives an idea on what is the practical length of the dependency

Autocorrelation function



For $t > 5$, there is no impact of $x(n-t)$ on the value $x(n)$

IIR results

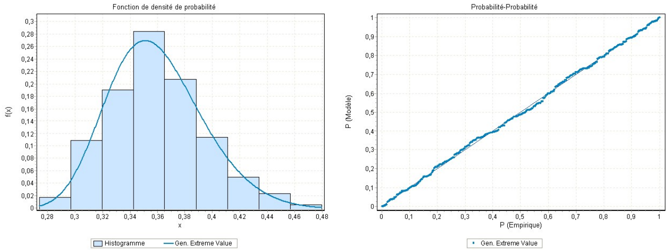


Figure : Histogram, density function and pplot at output 1

Conclusion

We have presented a new method:

- to estimate range of floating-point variables
- with a statistical models
- we showed that statistical models can be used to infer properties on LTI systems
- we show how to estimate the impact of previously computed values

Future works

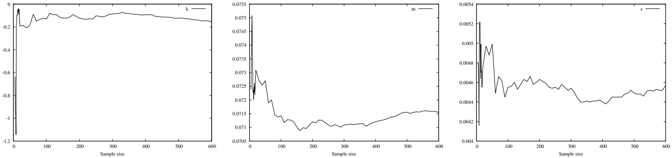
- Estimate the minimal number of simulation
- Estimate the minimal length of simulations
- A model for non LTI systems

Remark on the number of simulations

How many simulations are needed ?

Depends on what is looked for:

- The number of digits needed for coding the maximum: several tens
- The value of the maximum bound: several hundreds



IIR results

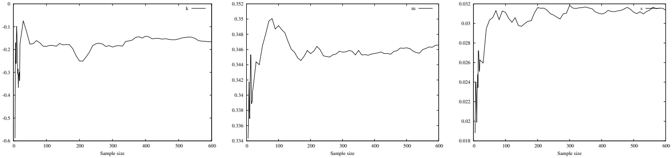


Figure : Evolution of the estimation of shape, position and scale parameters at output 1

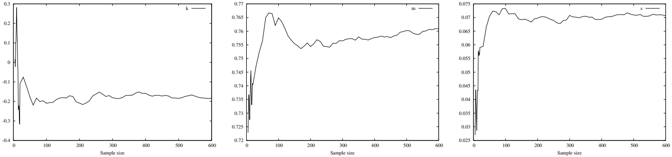


Figure : Evolution of the estimation of shape, position and scale parameters at output 3