New Software Dedicated to the Design and Simulation of Nonlinear Controllers Based on Feedback Linearization Technique

Azeddine Kaddouri¹, Sébastien Blais¹, Mohsen Ghribi¹, and Ouassima Akhrif²

¹ GRETER Research group, Faculty of Engineering, Université de Moncton, (NB), Canada
E1C 7A9, {kaddoua, sblais, ghribim}@umoncton.ca
² GREPCI Research group, Department of electrical engineering, École de Technologie Supérieure, Montreal, (QC), Canada, wassima@ele.etsmtl.ca

Abstract: This paper describes new software (NLSoft) developed for the design of nonlinear controllers based on feedback linearization technique. NLSoft is a software package containing several symbolic manipulation modules including differential geometric tools for the design and simulation of control systems. NLSoft presents a user-friendly graphical user interface (GUI) as well as a new and powerful module allowing the calculation time of linearizing control laws considering several Digital Signal Processors (DSPs) characteristics. These facilitate the real-time implementation of the control system. NLSoft is validated considering two illustrated examples.

1. Introduction

The nonlinearities of nonlinear dynamic systems have been traditionally handled in the literature by the use of classical linear control methods, which rely on linearized models approximating the dynamic system’s equations. These methods are just valid in a small operation range. When the required operation range has to be larger, the linear controller is likely to perform poorly. Nonlinear controllers, however, may handle the nonlinearities in large range operations directly. The nonlinear control considered here is based on the feedback linearization technique (static and dynamic) which gives a good solution for tracking control problems. Several accessible references describing its constructions are now available ([1] to [4]) as examples). This well-known nonlinear control technique has been successfully applied to the control of dynamic systems, even those with high nonlinearities in their model. However, this technique needs an extensive knowledge of differential geometry which provides a very useful framework for the analysis of nonlinear control systems. Differential geometric objects (Lie brackets, Lie derivatives, etc.) are not easily manipulated by hand. In order to facilitate this process, many new CAD tools appeared in the last decade (AISYS, AP_LIN, SCILAB, CONDENS, REDUCE, ProPAC etc). They offer the possibility of designing and/or simulating nonlinear dynamic systems. In the real-time applications, the main problem encountered with these nonlinear control techniques is that they need an important calculation time which makes implementation extremely difficult on a general use microprocessor.
Fortunately, a new class of digital signal processors (DSPs) is making this implementation possible. The software presented in this paper (NLSoft) is developed in order to design nonlinear controllers, to simulate the closed-loop systems and to give useful information for the real-time implementation considering real DSPs characteristics (TMX320F2812 and ADMC401). Furthermore, it is capable to inform us about the stability of the zero dynamics if they exist. NLSoft uses several subsystems expressed as classes which permit the design of nonlinear controllers and the plotting of the results obtained from the simulation of the closed-loop system. The block diagram of the closed-loop system to be designed and simulated using NLSoft is given in Figure 1.

2. Description of NLSoft

NLSoft consists of a design environment built in Visual C++. This design environment, hereafter referred to as "NLSoft Design Environment" or NODE, contains functions which can be grouped into five different subsystems. The first subsystem, the GUI, guides the user into making important design decisions at each step of the design process. The four remaining subsystems – Acquisition and Display, System Analysis, Storage and Symbolic Subsystem – contain several functions to adapt, save and recall the information provided by the user, perform analysis on a system, display and print results, launch symbolic algorithms and more. The NODE has the following characteristics:

- It contains several design packages which use the feedback linearization technique to design a control law that forces the output of a nonlinear system to follow some desired trajectory. It also contains a dynamic linearization package and a package for the analysis of the existence and the stability of the zero dynamics.
- The NODE also includes a new and powerful module permitting the calculation time of the designed nonlinear control laws. Two Digital Signal Processors (DSPs) characteristics are already included in NLSoft (TMX320F2812, ADMC401). The user may also consider other DSPs.

©GESTS-Oct.2005
Finally, it permits the pole placement, the simulation of the closed-loop system and the visualization of its performance. The following figure resumes the main subsystems.

![Diagram showing subsystems](image)

**Fig. 2.** Design and simulation subsystems

The NODE’s architecture is given in Figure 3.

![Diagram showing NODE architecture](image)

**Fig. 3.** NODE architecture

### 2.1 Graphical User Interface (GUI)

The GUI included in the NODE presents several controls which are available with the click of a button. The GUI is modified when results are displayed at each step of the
design process. Figure 4 shows the main controls of the NODE. The controls are grouped in three classes. The Data Input Controls allow the input of parameters characterizing either the nonlinear equations of dynamic systems or the DSP used to implement the designed nonlinear controllers. The data input controls are working closely with Acquisition and Display Functions to verify basic syntax as data is entered. These functions prevent the launch of an erroneous algorithm in a future step of the process. The Action Controls execute a different step of the nonlinear control design process. These steps include finding the linearizing control law of a nonlinear system, calculating the corresponding calculation time required for the specified DSP to implement the linearizing control law and to simulate the linearized system with several customizable input commands. The simulation results may be shown directly or stored in a Matlab m-file for the user to plot the different graphs. The Options Control allows users to modify registered variables. The Print control allows users to print previously calculated and displayed results.

![Fig. 4: Main control of the NODE](image)

The acquisition and display subsystems contain functions that can be grouped in following categories. Firstly, syntax functions are allowing the analysis of the input data entered by users. Secondly, File I/O Functions allow parameters to be saved, recalled and modified. Specifically, these File I/O functions are used for the storage of the system and DSP parameters. The System Analysis Subsystem contains the required functions to analyze the parameters of a system and to interpret results at all stages of the design process. It acquires data from the Storage Subsystem, generates symbolic algorithms and launches them. Upon completion of the algorithm, the results are interpreted, formatted and stored in the Storage Subsystem. The algorithms included in this subsystem are used to perform the following tasks: find the relative order of the nonlinear system, find the linearizing control law, transform the resulting system in a canonical form, find the calculation time on the designed controller and simulate the controlled closed-loop system under specific consideration. Finally, the

©GESTS-Oct.2005
Storage Subsystem consists of a global class used to store data required by different subsystem functions. The indexing of the stored data is made on a per function basis. The data is stored in private variables available through specific functions in order to minimize the possibility of errors.

2.2 Symbolic Subsystem
NLSoft provides a large set of symbolic functions embedded in the design environment. These functions are used by the symbolic algorithms described previously. Object-oriented programming allows the easy reuse of these functions in the different algorithms. The symbolic functions can be classified in four different groups.

- Symbolic functions performing basic mathematical operations. Amongst these are functions performing derivatives, variable replacement, parameter matching, etc.
- Symbolic functions for matrices operations and transformations. Functions finding the size of the matrix, its transpose, inverse, performing row reduction, finding the null space matrix, and more.
- Symbolic differential geometry functions. They include the Jacobian matrix, Lie Derivative etc.
- Symbolic simplification functions permitting the optimization of the output results and eliminate redundancy in the output equations.

3. Illustration examples:
In order to show the capabilities of NLSoft, we illustrate the two following examples involving SISO and MIMO systems.

3.1 SISO nonlinear system
Given the SISO nonlinear system:

\[
\begin{align*}
\dot{x}_1 &= x_1^2 x_2, \\
\dot{x}_2 &= 3x_2 + u
\end{align*}
\]

this can be written in the suitable form:

\[
\dot{x} = f(x) + g(x)u, \quad f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} = \begin{bmatrix} x_1^2 x_2 \\ 3x_2 \end{bmatrix}
\]

The output is chosen to be:

\[
y = -2x_1 - x_2
\]

Differentiating the output, one can obtain:

\[
y = L_f h(x) + L_g h(x)
\]
with:

\[ L_f h(x) = \frac{\partial h(x)}{\partial x} f(x) = \begin{bmatrix} -2 & -1 \end{bmatrix} \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} = -2x_1^2 x_2 - 3x_2 \]

\[ L_g h(x) = \frac{\partial h(x)}{\partial x} g(x) = \begin{bmatrix} -2 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = -1 \]

Therefore:

\[ \dot{y} = \dot{v} = -u - 2x_1^2 x_2 - 3x_2 \]

(6)

The linearizing control law is then:

\[ u = -(v + 2x_1^2 x_2 + 3x_2) \]

(7)

The results obtain by NLSoft are given by Figure 5.

In order to evaluate the calculation time of the linearizing control law, we consider the TMX320F2812. The calculation time (6.8 μsec) is evaluated considering floating-point operations.
The mathematical model of the salient Permanent-Magnet Synchronous Motor in the d-q synchronous reference frame [8] is given by:

$$\dot{x} = f(x) + \sum_{i=1}^{2} g_i(x)u_i = f(x) + g_1(x)u_d + g_2(x)u_q$$

(1)

With:

$$f(x) = \begin{bmatrix} f_1 & f_3 & f_3 \end{bmatrix}^T = \begin{bmatrix} -\frac{R}{L_d}i_d + \frac{L_q}{L_d}p_w r i_q \\ -\frac{R}{L_q}i_q - \frac{L_d}{L_q}p_w r i_d - \frac{\Phi_V}{L_q}p_w r \\ \frac{1}{J}(T_{em} - Bw_r) \end{bmatrix}$$

$$x = [i_d \quad i_q \quad w_r]^T \quad \text{et} \quad u = [u_d \quad u_q]^T \quad g_1(x) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

(2)

$$g_2(x) = \begin{bmatrix} 0 \\ \frac{1}{L_q} \end{bmatrix}^T \quad T_{em} = \frac{3p}{2J}(\Phi_V i_q + (L_d - L_q)i_d i_q)$$

where $u_d$ and $u_q$ are the d-q axis voltages, $i_d$ and $i_q$ the d-q axis currents, $L$ is the stator inductance, $R$ is the stator resistance and $w_r$ is the rotor speed. $\Phi_V$ is the flux linkage due to the rotor magnets, $p$ is the number of pole pairs, $B$ is the damping coefficient and $J$ is the rotor moment of inertia. $T_{em}$ is the electromagnetic torque. We choose the speed ($w_r$) and the direct current ($i_d$) as output variables, then:

$$y = [y_1 \quad y_2]^T = [i_d \quad w_r]^T$$

(3)

The linearizing control law becomes $u = D(x)^{-1} \cdot [-\zeta(x) + v]$
where:

\[
D^{-1}(x) = \begin{bmatrix}
\frac{L_d}{L_d - L_q} & 0 & 0 \\
0 & 1.5 p (L_d - L_q) y_d + \Phi_v & \frac{J L_q}{y_d}
\end{bmatrix}
\] (4)

Which is non-singular with: \( \Phi_v \neq (L_q - L_d) y_d \), with:

\[
\zeta(x) = \begin{bmatrix}
-\frac{R}{L_d} y_d i_d + \frac{L_q}{L_d} p w_d i_q \\
\lambda (L_d - L_q) y_d f_1 + \lambda (\Phi_v + (L_d - L_q) y_d) f_2 - \frac{B}{J} f_3
\end{bmatrix}
\] \( \lambda = 1.5 p / J \) (5)

The total relative degree is \( r = 1 + 2 = 3 \). Since the system order is 3, the system is then completely linearizable and there is no need to check the internal dynamics stability.

The results obtain by NLSoft are given in Figures 7 and 8.

---

**Fig. 7.** Dialog box interface for the example 2.
Fig. 8. Linearization results for the example 2

The calculation-time and the simulation results of the linearizing control law are given in Figure 9 and Figure 10 respectively. For the calculation time of the linearizing control law, we consider the TMX320F2812 DSP chip. The calculation time (90 µsec) is evaluated considering floating-point operations.

<table>
<thead>
<tr>
<th>Calculation Time</th>
<th>Number of Operations</th>
<th>Calculation Time (sec)</th>
<th>Number of Operations</th>
<th>Calculation Time (µsec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addition</td>
<td>9</td>
<td>10000</td>
<td>Subtraction</td>
<td>26</td>
</tr>
<tr>
<td>Subtraction</td>
<td>26</td>
<td>13000</td>
<td>Division</td>
<td>47</td>
</tr>
<tr>
<td>Multiplication</td>
<td>56</td>
<td>25000</td>
<td>Inverse</td>
<td>56</td>
</tr>
<tr>
<td>Inverse</td>
<td>0</td>
<td>0</td>
<td>Logarithm</td>
<td>0</td>
</tr>
<tr>
<td>Logarithm</td>
<td>0</td>
<td>0</td>
<td>Exponential</td>
<td>0</td>
</tr>
</tbody>
</table>

Total Calculation Time: 50000 µsec
4. Conclusion

The paper presented a new CAD software (NLSoft) oriented to the design and simulation of nonlinear controllers based on the feedback linearization technique. NLSoft is developed especially in order to facilitate the integration of the feedback linearization technique in R&D projects. NLSoft was developed in a compact and user-friendly GUI interfaces and gives useful information for the real-time implementation considering real DSPs characteristics. Two examples are given to validate the proposed software. We are working to integrate a dynamic linearization algorithm and the adaptive version to solve the problem of parameter uncertainties. Internet-access version of this software will be available soon.

5. Acknowledgment

This work is supported by the Natural Sciences and Engineering Council of Canada (NSERC).

6. References

Biography

▲ Name: Azeddine Kaddouri  
Address: Faculté d’Ingénierie, Université de Moncton, Moncton, E1A 3E9, New-Brunswick, Canada  
Tel: +1 506 863 2073; E-mail: kaddoua@umoncton.ca  
Other information: He received the Diplomat of Ingénieur d'état in electrical engineering from Batna University, Algeria in 1988 and the M.Sc. and the Ph.D. degrees in electrical engineering from Laval University, Canada in 1993 and 2000 respectively. From 1993 and 1999 he was a research assistant with GREPCI research group, École de Technologie Supérieure, Montréal, Canada. In 1999, he joined the University of Moncton, where he is currently an Associate professor.

▲ Name: Sébastien Blais  
Address: Faculté d’Ingénierie, Université de Moncton, Moncton, E1A 3E9, New-Brunswick, Canada  
e-mail: sblais@site.uottawa.ca  
Other information: He received the B.Eng. in electrical engineering with honors from Université de Moncton, Canada, in 2002 and the M.Sc. degree in electrical engineering from Ottawa University in 2005. Since 2001, he has been an active associate member of GRETER research group. Currently, he is working towards his Ph.D. degree in electrical engineering at Ottawa University, Ontario, Canada.

▲ Name: Mohsen Ghribi  
Address: Faculté d’Ingénierie, Université de Moncton, Moncton, E1A 3E9, New-Brunswick, Canada  
Tel: +1 506 863 2090; e-mail: ghribim@umoncton.ca  
Other information: He obtained his B.Sc. and the M.Sc. degrees from Université du Québec à Trois-Rivières, Canada and the Ph.D. degree from Université Laval, Canada respectively in 1987, 1989 and 1994. He has been a professor at École de Technologie Supérieure, Montréal and the École Polytechnique de Masuku (Gabon). Since 1997, he is a professor at Université de Moncton, Canada. Currently he is a director of GRETER research group.

▲ Name: Oussima Akhrif  
Address: Department of electrical engineering, École de Technologie Supérieure, Montreal, (QC), Canada.  
Tel: +1 514 396 8521; e-mail: wassima@ele.etsmtl.ca  
She received a Diplôme d'Ingénieur d'État from the École Mohammedia, Rabat, Morocco and the M.Sc.A. and Ph.D. degrees from the University of Maryland, College Park, in 1984, 1987, and 1989, respectively, all in electrical engineering. During 1989-1990, she was a Visiting Assistant Professor in the Systems Engineering Department, Case Western Reserve University, Cleveland, OH. In 1992, she joined the École de Technologie Supérieure, Montréal, Canada, where she currently is a Professor in the Electrical Engineering Department.