Generación automática de ejercicios para la enseñanza de control*

Luis A. Márquez-Martínez y Ricardo Cuesta
CICESE, Depto. de Electrónica y Telecomunicaciones
Km. 107 Carr. Tij - Ens. C.P. 22860, Ensenada, B.C. México
{lmarquez, jcuesta}@cicese.mx
tel. +52 (646) 175-0500

Resumen

: en este trabajo, se presenta una manera práctica de generar ejercicios individuales para estudiantes. La idea se ejemplifica mediante la presentación de un programa para generar un número arbitrario de exámenes distintos sobre temas de control lineal, aunque puede ser aplicado en muchas otras áreas del conocimiento.

1. Introducción

Una parte importante del proceso de aprendizaje es la realización de ejercicios que obliguen al estudiante a aplicar los conocimientos adquiridos. Sin embargo, es práctica común que los estudiantes intercambien las soluciones entre sí, por lo que idealmente se debería dejar problemas distintos a cada uno. Sin embargo, la preparación y evaluación implican un trabajo enorme para el instructor.

Una solución práctica es a través de un software denominado WIMS [XIA04],el cual toma un problema genérico, le varía algunos parámetros de manera aleatoria dentro de rangos adecuados, y finalmente lo presenta al estudiante. Después, solicita la respuesta (ya sea en formato libre o como opción múltiple), la evalúa, y presenta la calificación al estudiante y al instructor, todo a través de internet (o una red local).

Esto permite que el alumno pueda ejercitarse cuantas veces lo desee (siempre con problemas distintos) y que al momento de examinarlo, el sistema le dé su calificación al término de la sesión. Sin embargo, este enfoque presenta algunos problemas. El más importante es que no permite ver el planteamiento del problema y su desarrollo por parte del estudiante, lo cual puede ser importante para el profesor.

Por ello, en algunos casos conviene tener una versión en papel de dichos examenes. A pesar de que existen paquetes que pueden generar examenes distintos de manera automática, estos lo hacen tomando aleatoriamente *problemas* de un conjunto preparado de antemano. Sin embargo, a nuestro mejor conocimiento, no existe software capaz de generar exámenes usando parámetros aleatorios.

La contribución de este documento es precisamente en esa dirección. En él presentamos un método para generar por computadora tantos documentos (sean exámenes o tareas) diferentes como se desee, así como con las respuestas correspondientes. Dicho método puede ser aplicado a otras áreas del conocimiento, en particular las que hacen uso de las matemáticas.

El material se presenta en dos partes. Primero se explica el enfoque utilizado, ilustrándolo con un ejemplo de un examen sobre sistemas lineales. Después, se detalla un programa que genera los exámenes y se muestra un examen típico, junto con la hoja de respuestas correspondiente.

Se concluye con algunas discusiones sobre el tema.

2. Enfoque utilizado

En esta sección se presentan algunas directrices para poder desarrollar programas que generen los exámenes personalizados. A fin de facilitar su comprensión, se utilizará como ejemplo un examen hipotético sobre sistemas lineales.

Básicamente, el enfoque consiste en los siguientes pasos:

1. Definir los problemas a resolver, dejando algunos datos como parámetros, y estableciendo los rangos de valores aceptables para cada parámetro.

2. Escribir un programa que genere aleatoriamente un conjunto de valores adecuados y escriba la información en un formato particular.

3. Generar los exámenes usando el programa de preparación de documentos LaTeX [lat].

*Parcialmente apoyado por CONACyT, México.

A continuación se detalla cada uno de estos pasos.

2.1. Elaboración de problemas genéricos

2.1.1. Concepción.

El diseño de buenos problemas genéricos es una labor que requiere un buen conocimiento del tema y algo de creatividad.
El primer paso es definir en términos generales un ejercicio que obligue al estudiante a mostrar su grado de comprensión del tema. Como ejemplo, considere los dos problemas siguientes.

P1 Dada la gráfica de Bode de un sistema, encontrar la función de transferencia que la genera.

P2 Dadas las matrices A, B, C de un sistema lineal, calcular sus matrices de controlabilidad y observabilidad, así como determinar si es controlable y/u observable.

El primer problema requiere un buen grado de dominio conceptual. El segundo, se trata más bien de seguir un procedimiento.

2.1.2. Rangos parámetricos.

El siguiente paso consiste en determinar los rangos aceptables para los valores de los parámetros. Esta es una de las partes más delicadas del proceso de diseño, ya que una mala elección de ellos puede llevar a generar problemas con valores ilógicos (e.g. resistencias o frecuencias negativas), o en los cuales sea imposible generar una respuesta correcta. Es al reflexionar en estos detalles que se puede afinar la formulación del problema.
Por ejemplo, en el problema P1, se hicieron tres consideraciones.

1. El rango de graficado debe incluir las frecuencias de corte.
2. Los polos deben ser fácilmentes distinguibles en la gráfica.
3. El rango para los polos debe ser suficientemente grande para generar un número importante de problemas distintos, y suficientemente pequeño como para que no queden dos polos extremadamente alejados.

A partir de ahí, y tras algunos ensayos, se decidió que se propondrían sistemas de segundo orden cuyos polos serían de la forma $p_i = a\,10^b$, con a, b enteros, $a \in [1, 9]$, $b \in [-2, 2]$. Estas restricciones permiten distinguir razonablemente los polos en las gráficas, a la vez que se obtiene un buen número de exámenes distintos posibles ($(9 \cdot 5)2 = 2025$).

2.2. Programación

2.2.1. Selección de paquetes de cómputo.

Una vez definido el problema y los rangos paramétricos, se procede a su programación. Para *plantear* el problema P1 al estudiante, se requiere generar graficas semilogarítmicas, mientras que para el problema P2 es necesario presentar ecuaciones matemáticas. En todos los casos, es necesario generar valores aleatorios para los parámetros.
Para llevar a cabo estas tareas se puede usar cualquier lenguaje en el que se puedan desarrollar las tareas requeridas por los problemas. En este documento se hizo uso de tres paquetes (bajo el sistema operativo Linux):

MAXIMA 5.9.0 [Sch]. Paquete de cálculo simbólico usado para generar los parámetros aleatorios. Tiene un lenguaje propio de programación, que incluye la posibilidad de llamar a otros programas, por lo que fue usado para escribir el programa general.

GNUPLOT 3.7 [WK04]. Programa de graficado Es llamado desde el programa general para generar gráficas en formato adecuado para ser procesadas.

LaTeX 2ε [lat]. Procesador de documentos. Toma la información generada por MAXIMA y GNUPLOT y genera un archivo en formato PDF (Portable Document Format) con los exámenes, y otro con las respuestas.

La interacción de estos paquetes se muestra en el diagrama siguiente.

$$\begin{array}{ccccccc}
\textbf{maxima} & \longrightarrow & \texttt{*.tex} & \longrightarrow & \LaTeX & \longrightarrow & \texttt{*.pdf} \\
\downarrow & & & & \uparrow & & \\
\texttt{*.out} & \longrightarrow & \textbf{gnuplot} & \longrightarrow & \texttt{*.eps} & &
\end{array}$$

Todo este software esta disponible gratuitamente para diversas plataformas (windows, mac y linux, entre otras). Una ventaja de esto es que es posible proporcionar a los estudiantes copias de dicho software junto con los programas desarrollados para que practiquen si así lo desean.

2.2.2. Enlace.

A fin de que se puedan comunicar los diversos paquetes utilizados, es necesario convertir información de un formato a otro. Por ejemplo, $x2$ se representa en MAXIMA como `x2`, mientras que en GNUPLOT

sería `x**2`. Sin embargo, son pocas las diferencias, por lo que esta labor no representa mayor problema en general.
Otro problema es cómo pasar la información a LaTeX para que éste sea capaz de insertarla de manera adecuada. Esto se resuelve gracias a las capacidades de generación de *macros* de LaTeX a través del comando `\newcommand`, tal como se ilustra en la siguiente sección.

3. Ejemplo práctico

En esta sección se presenta un ejemplo que muestra como generar `n` examenes diferentes, basados en los problemas P1 y P2. La forma de generar el examen es primeramente cargar el archivo `bode.mc` en MAXIMA y escribir `examen(3)` (si se desean 3 examenes diferentes). Posteriormente, en LaTeX se compilan los archivos `examenes.tex` y `soluciones.tex` para obtener los examenes y las soluciones, respectivamente.

3.1. Programas

Las explicaciones y descripciones del programa `bode.mc` escritos en MAXIMA siguen el siguiente formato:

`función(variables)` L.#líneas
Descripción de la función.

`gensys(n,v)` L.1-7
Genera las matrices A, B y C, de un sistema lineal de la forma

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx. \end{aligned} \quad (1)$$

Las matrices A, B y C son de dimensión $\mathtt{n} \times \mathtt{n}$, $\mathtt{n} \times 1$ y $1 \times \mathtt{n}$, respectivamente; mientras que sus coeficientes toman valores enteros aleatorios desde $-\mathtt{v}$ hasta `v`.

`matcontrol(A,B)` L.8-12
Obtiene la matriz de controlabilidad a partir del sistema (1).

`observa(A,C)` L.13-17
Obtiene la matriz de observabilidad a partir del sistema (1).

`genran(m)` 18
Genera un número entero aleatorio entre `-m` y `m`.

`randdec(m)` L.19-21
Genera un número entero aleatorio entre -10 y 10, distinto de cero y lo multiplica por 10^n, donde n es un número entero entre `-m` y `m`.

`logfix(n)` L.22
Obtiene la parte entera de $\log_{10}(\mathtt{n}) - 0{,}2$.

`bode(i,m)` L.23-52
Genera las gráficas de Bode para la magnitud (magi.eps) y fase (phai.eps) a partir de un sistema de segundo orden generado aleatoriamente.

Primeramente (L.23-27), se obtiene `a`, `p1` y `p2` (números aleatorios en décadas), los cuales corresponden a la ganancia y polos del sistema:

$$\begin{aligned} f(s) &= \frac{\mathtt{a}}{(s+\mathtt{p1})(s+\mathtt{p2})} \\ &= \frac{\mathtt{a}}{s2+\mathtt{b1}s+\mathtt{b2}}. \end{aligned} \quad (2)$$

En L.29-32 se calcula los valores que tendrán las gráficas de Bode en el eje horizontal.

En L.33 se crea el archivo `temp.out` y escribe en él las instrucciones que GNUPLOT y MAXIMA realizarán. El gráfico resultante será guardado en magi.eps (L.35-36).

En L.39-41 se grafica la magnitud del sistema (2), la cual está dada por

$$|f(j\omega)| = 20\log_{10}\left(\frac{|\mathtt{a}|}{\sqrt{(\mathtt{b2}-\omega 2)2+\mathtt{b1}^2\omega 2}}\right).$$

Donde el intervalo de ω se definió en las líneas 29-32. Se ejecuta el archivo `temp.out` desde GNUPLOT (L.42).

En L.43 se crea el archivo `temp.out` y escribe en él las instrucciones que GNUPLOT y MAXIMA realizarán. El gráfico resultante será guardado en phai.eps (L.45-46).

En L.49-50 se grafica la fase del sistema (2), la cual está dada por

$$\angle f(j\omega) = \arctan 2\left(-\mathtt{b1}\omega, \mathtt{b2}-\omega^2\right).$$

Se ejecuta el archivo `temp.out` desde GNUPLOT (L.51). En L.52 se borra el archivo `temp.out`.

`examen(n)` L.53-71
Crea el archivo `probs.tex` en el cual se escriben las matrices A, B y C del sistema (1), las gráficas de Bode y sus respectivos resultados, produciendo así los `n` examenes diferentes.

Primeramente (L.54) se crea el archivo `probs .tex` y escribe en él las intrucciones que MAXIMA y LaTeX realizarán. En L.56-58 se ejecutan las funciones `gensys(n,v)` y `bode(i,m)` y se almacenan los resultados.

Se crea el macro en LaTeX: \xmen{A}{B}{C}{magi}{phai}, que contiene los datos de los n examenes (L.59-63).

Finalmente (L.65-71) se crea el macro en LaTeX: \xmens{$\mathcal{C}, \mathcal{O}$}{$f(s)$}, que contiene las soluciones de los n examenes.

bode.mc

```
gensys(n,v):=block([],kill(A,B,C),
  A[i,j]:=random(v)-random(v),
  A:genmatrix(A,n,n),
  B[i,j]:=random(v)-random(v),
5  B:genmatrix(B,n,1),
  C[i,j]:=random(v)-random(v),
  C:genmatrix(C,1,n));
matcontrol(A,B):=block([r,la],kill(Q),
  la:length(A), Q:zeromatrix(1,la),
10  for n:0 thru la-1 do(
    Q:addrow(Q,(A^^n).B)),
  Q:transpose(submatrix(1,Q)));
observa(A,C):=block([la,r],kill(P),
  la:length(A),P:zeromatrix(la,1),
15  for n:0 thru la-1 do(
    P:addcol(P,C.(A^^n))),
  P:submatrix(P,1));
genran(m):=random(m+1)-random(m+1);
randdec(m):=block([d],
20  d:genran(10)*10^genran(m),
  if d=0 then randdec(m) else d);
logfix(n):=fix(ev(log(n)/log(10)-0.2,numer));
bode(i,m):=block([a,b1,b2],kill(sol),
  a:ev(randdec(m),numer),
25  p1:ev(randdec(m),numer),
  p2:ev(randdec(m),numer),
  b1:p1+p2, b2:p1*p2,
  sol:a/((s+p1)*(s+p2)),
  minimo:ev(10^(logfix(min(abs(p1),
30   abs(p2)))-1),numer),
  maximo:ev(10^(logfix(max(abs(p1),
   abs(p2)))+2),numer),
  (with_stdout("temp.out",
   print("set terminal postscript eps 22"),
35   print(concat("set output \"",mag,
    i,".eps\"")),
   print("set logscale x; set grid;
    set nokey"),
   sprint("plot [w=",minimo,":",maximo,"]
40    20*log10(",abs(a),"/sqrt((",b2,"-w**2
    )**2+(",b1,")**2*w**2))"))),
   system("gnuplot temp.out"),
  (with_stdout("temp.out",
   print("set terminal postscript eps 22"),
45   print(concat("set output \"",pha,i,
    ".eps\"")),
   print("set logscale x; set grid;
    set nokey; set angles degrees"),
   print("plot [w=",minimo,":",maximo,"]
50    atan2(-(",b1,")*w,(",b2,"-w**2))"))),
   system("gnuplot temp.out"),
   system("rm temp.out"));
examen(n):=block([],
  (with_stdout("probs.tex",
55  for i:1 thru n do(
    gensys(4,2),bode(i,2),
    mc:matcontrol(A,B),ob:observa(A,C),
    rc:rank(mc),ro:rank(ob),
    print("\\xmen{"),
60    tex('A=A),print("}{"),tex('B=B),
    print("}{"),tex('C=C),
    print("}",concat("{",mag,i,"}")),
    print(concat("{",pha,i,"}")),
    print("\\xmens{matrix de
65     controlabilidad: "),
    tex(mc),print("rango: "),tex(rc),
    print("matrix de observabilidad: "),
    tex(ob),print("rango: "),tex(ro),
    print("}{"),tex('f(s)=expand(sol)),
70     tex('f(s)=sol),
    print("}")))));
```

examenes.tex

```
% renombre de commando para matrices
\renewcommand\pmatrix[1]{\left(
\begin{matrix}#1\end{matrix}\right) }

% Para contar los examenes
\newcounter{conexa}
\newcommand{\tipo}{
\refstepcounter{conexa}
\textbf{E.\theconexa}}

% insertar figuras
\newcommand{\figura}[2]{\begin{figure}[htbp]
\centering \includegraphics[scale=.5]{#1.eps}
\caption{#2}\label{fg:#1}\end{figure}}

% problemas
\newcommand{\xmen}[5]{
\portada
\probuno{#1}{#2}{#3}
\probdos{#4}{#5}
\figura{#4}{Magnitud.}
\figura{#5}{Fase.}}

% soluciones
\newcommand{\xmens}[2]{}

% enunciado del primer problema
\newcommand{\probuno}[3]{\item Considere...
donde
#1 #2 #3
Determine...}

% enunciado del segundo problema
\newcommand{\probdos}[2]{\item En la figura
 \ref{fg:#1} se muestran...
Determine...}

% inicia documento
\begin{document}
```

ISBN: 970-32-2137-8

```
\begin{enumerate}
\input{probs}
\end{enumerate}
\end{document}
```

El archivo `soluciones.tex` cuenta con el mismo preámbulo y cuerpo que el archivo `examenes.tex`, lo único en que se diferencian es en los siguientes macros:

`soluciones.tex`

```
% Problemas
\newcommand{\xmen}[5]{}

% soluciones
\newcommand{\xmens}[2]{
\section*{Soluci\'on para\tipo}
\item \textbf{Controlabilidad y observabilidad:
\\} #1 \item \textbf{Funci\'on de transferencia
 y polos:} #2}
```

4. Discusión

El presente trabajo muestra la posibilidad de desarrollar exámenes y tareas personalizadas en el área de control automático de una manera práctica.
Aunque el esfuerzo inicial por parte del instructor es mayor bajo este esquema, éste se compensa con las ventajas de ir formando una fuente casi ilimitada (para fines prácticos) de problemas diferentes, junto con sus soluciones. Además, el profesor puede limitarse al desarrollo conceptual de los ejercicios, interactuar con estudiantes para la selección de rangos paramétricos, dejándoles a ellos la parte de programación. Además, los ejercicios generados pueden ser modificados para ser presentados y evaluados a través de una interfaz web.
Evidentemente, el mismo enfoque puede ser aprovechado en otras áreas del conocimiento tan diversas como son ingeniería, física, matemáticas biología, química y economía, por mencionar algunas.

Referencias

[lat] LaTeX - a document preparation system. *Online*. Available: http://www.latex-project.org/.

[Sch] W. Schelter. Maxima - a sophisticated computer algebra system. *Online*. Available: http://maxima.sourceforge.net/.

[WK04] T. Williams and C. Kelley. Gnuplot. *Online*. Available: http://www.gnuplot.info/, 1998-2004.

[XIA04] G. XIAO. Web interface mathematical server. *Online*. Available: `http://wims.unice.fr`, 2004.

A. Ejemplo de un examen generado y su solución

El siguiente examen fue generado utilizando el programa descrito en este artículo.

Examen de Control (**E.1**)

1. Considere el sistema dado por

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned}$$

donde

$$A = \begin{pmatrix} -1 & 0 \\ -1 & 1 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & -1 \end{pmatrix}$$

Determine la matrices de controlabilidad y observabilidad. Indique si el sistema es controlable y/u observable.

2. En la figura 1 se muestran los diagramas de Bode de la magnitud y fase respectivamente. Determine la función de transferencia y los polos del sistema.

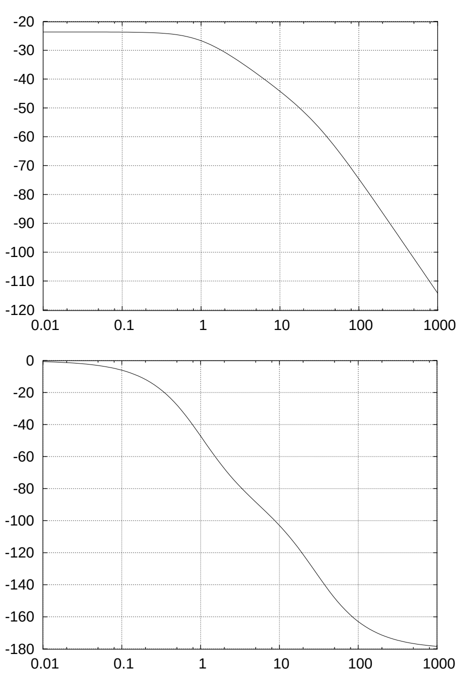

Figura 1: Magnitud y fase.

Solución para examen tipo **E.1**

3. **Controlabilidad y observabilidad:** matrix de controlabilidad:

$$\begin{pmatrix} 1 & -1 \\ 0 & -1 \end{pmatrix}$$

rango:

$$2$$

matrix de observabilidad:

$$\begin{pmatrix} 0 & 1 \\ -1 & -1 \end{pmatrix}$$

rango:

$$2$$

4. **Función de transferencia y polos:**

$$f(s) = -\frac{2}{s^2 + 31\,s + 30}$$

$$f(s) = -\frac{2}{(s+1)\,(s+30)}$$

ISBN: 970-32-2137-8