

Ateliers _ Compte-rendus

Claude Frasson, Jean-Pierre Pécuchet

▶ **To cite this version:**

Claude Frasson, Jean-Pierre Pécuchet. Ateliers _ Compte-rendus. Technologies de l'Information et de la Communication dans les Enseignements d'ingénieurs et dans l'industrie, Nov 2002, Villeurbanne, France. pp.315-320. edutice-00000672

HAL Id: edutice-00000672
https://edutice.hal.science/edutice-00000672

Submitted on 4 Nov 2004

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Réalisation de documents interactifs pour l'enseignement du TS avec génération automatique d'appliquettes JAVA

J.Liénard, G.Lejeune
LIS/ENSIEG BP 46 38402 St Martin d' Hères
joel.lienard@.inpg.fr, gerard.lejeune@.inpg.fr

Résumé

On présente la technique utilisée pour illustrer un cours de traitement du signal à l'aide de simulations accessibles via Internet. On expose les avantages apportés par le logiciel Mustig pour réaliser des simulations interactives et on décrit son transcodeur qui permet de les transformer en appliquettes Java. On décrit deux expériences de réalisation de cours interactifs ainsi réalisés.

Abstract

The technique used to illustrate a signal processing course with simulations accessed through Internet is presented.
The advantages brought by the Mustig software to make the interactive simulations are exposed, and the Mustig transcoder allowing to create the corresponding Java applets is described. Then two examples of interactive course realizations are shown.

Introduction

Le traitement du signal est une discipline qui, pour son enseignement, gagne beaucoup à être illustrée de simulations pour rendre plus concret des concepts qui, sans cela, ne sont que des mathématiques.
Les résultats des simulations sont, la plupart du temps, affichés sous forme de tracé de courbe. Pour faire comprendre l'influence des paramètres de ce traitement, il est souhaitable que l'étudiant puisse lui-même modifier ces paramètres : on obtient alors ce que nous appelons une figure interactive.
A l'époque des "Campus virtuels", il est souhaitable que ces simulations soient accessibles par Internet.

Outils pour réaliser des figures interactives

Logiciels multimédias

Le plus naturel pour réaliser un document interactif est peut-être d'utiliser un logiciel d'édition de CD-ROM, (par exemple le logiciel Director de Macromedia). Ce type de logiciel est surtout destiné à afficher des images, il n'a pas vocation à effectuer les calculs intensifs nécessaires au traitement du signal. Il est donc indispensable de calculer à l'avance toutes les courbes susceptibles d'être visualisées.
Or, si l'on souhaite faire varier seulement deux paramètres avec seulement 10 valeurs possibles, on obtient 100 combinaisons possibles. Calculer à l'avance 100 courbes pour une seule figure interactive est difficilement envisageable. Cela conduirait à un volume important de données à stocker ou à transférer sur le réseau, la préparation de toutes ces images serait fastidieuse, et surtout leur gestion serait une source d'erreur.

Programmation traditionnelle

Pour éviter la préparation et la mémorisation d'un grand nombre d'images, il faut les calculer à chaque changement de paramètre. Il faut donc le décrire par un programme informatique. Ce programme peut être écrit dans un langage traditionnel comme "C", Fortran. On préférera Java en raison de sa portabilité entre plates-formes et de son aptitude à la transmission par Internet.
La réalisation avec cette technique de documents comme ceux que nous présentons, comportant près de 100 figures interactives, serait un travail très important puisqu'il faudrait réaliser un programme par figure.

Logiciel de simulation haut niveau

Pour accélérer la réalisation, il faut programmer avec un logiciel de plus haut niveau. L'utilisation d'un langage vectoriel interprété répond à cet objectif. Cela d'autant plus qu'il offre la possibilité de réaliser également les dialogues de saisie de paramètres.
L'utilisation des simulations ainsi réalisées nécessite généralement de disposer du logiciel sur le poste de l'étudiant. Cela peut être une contrainte importante en raison, d'une part, du temps l'opération d'installation qui ne se justifie que pour une utilisation de longue durée et d'autre part, des coûts de licences. Une solution peut être, si le logiciel le permet, de réaliser des applications compilées indépendantes. On retrouve alors l'inconvénient de spécificité à une plate-forme (machine et système d'exploitation).

Solution retenue

Si l'on prend comme critère la portabilité et la facilité d'accès via Internet, la meilleure solution est celle d'appliquette JAVA.
On peut pourtant se poser également la question de la crédibilité de la démonstration pour l'utilisateur étudiant. Il est censé vérifier expérimentalement une propriété

mathématique, et il ne fait que visualiser des résultats de simulation dont il ne peut pas vérifier la véracité. La seule façon d'atténuer cet inconvénient est l'utilisation des langages de haut niveau, à condition que le code source soit consultable et facilement lisible. Un langage graphique comme MUSTIG présenté ci-dessous répond bien à cette exigence. Et comme il possède un transcodeur JAVA, il permet facilement le passage d'un document en mode natif, consultable par les utilisateurs disposant du logiciel, à l'appliquette Java, pour les autres.

Le logiciel MUSTIG

Le logiciel MUSTIG [1] est un logiciel spécialisé pour le traitement du signal. Son utilisation apporte une grande rapidité de réalisation en raison de sa très grande interactivité et de sa spécification "signal".
La réalisation de l'interface de saisie des paramètres de présentation de paramètres est particulièrement simple. La nature du langage graphique de MUSTIG (graphe de dépendance) assure la prise en compte des modifications de paramètres sans aucune programmation spécifique.
De plus, le "programme" est facilement compréhensible, car il se présente sous la forme d'un graphe.

Le transcodeur JAVA de MUSTIG

Ainsi qu'il a été dit ci-dessus, l'utilisation d'un logiciel de haut niveau impose à l'utilisateur l'installation du logiciel sur sa machine. Pour l'éviter, un transcodeur JAVA a été développé dans le logiciel MUSTIG.
Le transcodeur respecte le style de l'interface MUSTIG : il n'y a pas de bouton de demande de calcul, chaque changement de paramètre lance automatiquement les calculs nécessaires à la mise à jour des affichages de résultats. Les résultats sont ainsi toujours en concordance avec les paramètres affichés.
Les fichiers "class" générés sont de taille raisonnable, compatible avec un accès par modem 56 kb/s.

Expériences d'utilisation

Nous avons l'expérience de deux réalisations significatives d'ouvrages interactifs utilisant MUSTIG pour construire les figures animées. Chacune d'entre elles a été complétée par une variante accessible par Internet en utilisant le transcodage JAVA.
La première [1] est un CD-ROM complémentaire d'un ouvrage imprimé. L'appel des figures est réalisé par un logiciel spécifique, qui donne également accès à des textes explicatifs et à un formulaire. Une version "Run-Time" de MUSTIG est prête sur le CD, sans nécessiter de procédure d'installation. La seule contrainte est celle de la plate-forme (PC sous Window). La version JAVA de ce cours reprend la même structure, l'appel des figures transcodées en JAVA est fait par un programme JAVA. Il a été ajouté un système d'auto-évaluation par questions à choix multiples.
La deuxième réalisation [2] est un cours disponible sous forme HTML. Les illustrations sont initialement visibles sous forme d'images fixes traditionnelles. Ces images sont supports de liens vers les illustrations interactives qui s'ouvrent dans une nouvelle fenêtre. Chaque image interactive possède un bouton pour ouvrir une fenêtre d'aide. Cette aide comprend un descriptif de la simulation et une liste d'expériences proposées à l'étudiant. Dans ces expériences, chaque portion du texte proposant des conditions de réglage de paramètre est liée à un point d'entrée (également généré automatiquement) de l'appliquette qui réalise ces réglages automatiquement. La lecture de ce texte en cliquant sur chaque lien permet ainsi à l'étudiant de faire vivre la figure comme le ferait l'enseignant pendant une démonstration.

Retour de la part des enseignants

L'ampleur du travail réalisé dans chacune des réalisations (respectivement 100 et 50 figures interactives) avec des moyens humains très restreints suffit à démontrer l'efficacité de la méthode de réalisation. Les enseignants se sont concentrés sur l'objectif pédagogique, sans perdre trop de temps en technique de programmation ou de présentation.
La disponibilité de ces supports de cours a permis de modifier la forme de l'enseignement. Ils sont, tout d'abord, utilisés pour illustrer le cours magistral. Ensuite, le volume de ce cours a été diminué au profit d'un travail plus personnel des étudiants.

Retour de la part des étudiants

Il est indéniable que la manipulation de ces figures plait aux étudiants, qui s'amusent à faire varier les paramètres. Il faut pourtant éviter que cette manipulation ne soit justement qu'un jeu. Il faut pour cela compléter les figures, qui ne sont que des outils pour aider à comprendre, par des questions précises qui les obligent à un effort de réflexion.

Conclusion

L'expérience de réalisation de figures interactives à l'aide de MUSTIG a démontré la grande efficacité du procédé. La vitesse de réalisation est telle que nous avons pu, à l'occasion d'un nouveau cours dans le domaine des communications numériques, préparer des simulations pour chaque séance de cours, les montrer aux étudiants sous forme native MUSTIG, puis leur mettre aussitôt à disposition sur Internet sous forme JAVA.

[1] "Logiciel MUSTIG" Gresilog (hppt://www.gresilog.com)
[2] " Traitement du Signal pour géologues et géophysiciens" F.Glangeaud, J-L.Mari, F.Coppens, Editions TECHNIP
[3] "Traitement Numérique du Signal" J.Liénard, Média - tèque du Club EEA, section "signal et image" (http://www.clubeea.org/enseign/mediatheque_sommaire.htm)