

# **UNIVERSITE DU LITTORAL CÔTE D'OPALE**

**MASTER Recherche  
Sciences et Technologies Mention MSPI  
Spécialité Informatique**

**Parcours Modélisation et Simulation des  
Systèmes Complexes (M2SC)**

**Master Recherche Sciences et Technologies Mention MSPI**  
**Spécialité Informatique\***  
**Parcours « Modélisation et Simulation des Systèmes Complexes »**

**Président de jury**  
Christophe RENAUD  
e-mail : renaud@lil.univ-littoral.fr

Directeur d'études 1<sup>ère</sup> année : Cyril Fonlupt (e-mail : fonlupt@lil.univ-littoral.fr)  
Directeur d'études 2<sup>ème</sup> année : Christophe Renaud (e-mail : renaud@lil.univ-littoral.fr)

### **OBJECTIFS DE LA FORMATION**

Le Master Recherche (MR) « Modélisation et Simulation des Systèmes Complexes » (M2SC) est un Master informatique : il s'appuie sur la section du CNU, la 27e (Informatique). A l'Université du Littoral Côte d'Opale (<http://www.univ-littoral.fr>), cette section est représentée par le Laboratoire d'Informatique du Littoral (<http://www-lil.univ-littoral.fr> - UPRES-EA 4029). Le LIL représente 19 enseignants-chercheurs dont 5 habilités à diriger des recherches (dont, actuellement, 3 sont professeurs).

Le thème du MR M2SC, la « Modélisation et Simulation des Systèmes Complexes », recouvre de manière unifiée les activités d'un laboratoire support. Le terme « modélisation » recouvre de nombreuses acceptions, plusieurs étant présentes dans le MR M2SC : modélisation d'applications informatiques en vue de leur maintenance et de leur qualité, modélisation des données (bases de données), modélisation de la dynamique de systèmes naturels et de systèmes de production, modélisations de scènes et d'animations et synthèse d'images.

Le MR « Modélisation et Simulation des Systèmes Complexes » a pour objectif principal de former des étudiants issus de diverses filières informatiques à la recherche et par la recherche et de détecter les meilleurs éléments susceptibles de développer une recherche de qualité dans le cadre d'une thèse de doctorat au LIL ou dans un laboratoires extérieurs (français ou étranger).

Le programme de ce parcours Recherche est basé sur les semestres 1 à 3 sur le programme du parcours ISIDIS (ou exceptionnellement I2L). Les étudiants suivent les mêmes cours. Seul le semestre 4 est différent avec 4 modules d'initiation à la recherche et un stage en laboratoire.

### **CONDITIONS D'ENTREE ET DIPLOMES DELIVRES**

Après examen du dossier, l'accès à cette formation est possible :

- en première année aux titulaires d'une Licence Informatique ou aux titulaires d'un diplôme BAC+2 (DUT Informatique, BTS Informatique, ...) et d'une troisième année à dominante Informatique ;
- en deuxième année aux titulaires d'une première année de Master à dominante Informatique et disposant de prérequis compatibles.

A la fin du cursus, l'étudiant acquiert le grade de Master en Sciences et Technologies Mention Mathématiques et Sciences pour l'Ingénieur Spécialité Informatique.

---

\* Le parcours Ingénierie du logiciel libre est sous réserve d'habilitation.

## ORGANISATION PEDAGOGIQUE DES ETUDES

Le programme proposé durant la formation reflète une progression régulière sur deux années ayant pour objectif l'acquisition des connaissances et du savoir-faire indispensables à la conduite de projets de recherche ou en entreprise. Il regroupe :

- un enseignement fondamental intégrant des matières telles que les mathématiques et les fondements théoriques de l'informatique;
- un enseignement comprenant les techniques nécessaires au métier de l'Ingénieur Informaticien, que l'on peut regrouper en trois pôles : conception et développement de logiciel; système, architecture et réseaux; ingénierie des systèmes d'information ;
- un enseignement de culture générale incluant la pratique des langues vivantes;
- une initiation à la vie d'entreprise sous forme de cours, de conférences données par des industriels, et sanctionnée par des stages obligatoires en entreprise.

## MAITRISE DE LANGUES VIVANTES

La maîtrise de l'anglais est indispensable au cursus d'un ingénieur. L'enseignement de l'anglais est présent au cours des deux années et l'obtention d'un seuil de points, fixé à 750, au TOEIC est nécessaire pour l'attribution du diplôme terminal. Les étudiants font obligatoirement de l'anglais. Les étudiants ayant le niveau requis au TOEIC peuvent décider de préparer un diplôme d'anglais (First Certificate de l'Université de Cambridge).

## VIE D'ENTREPRISE

Cet enseignement est assuré par des professionnels. Les cours sur l'ensemble des deux années apportent à l'étudiant des connaissances en communication, marketing, qualité, économie, innovation et simulation d'entretiens d'embauche, ...

Un stage en entreprise est prévu chaque année :

- A la fin de la première année : un stage de *conception-développement* de 16 semaines a pour but de faire participer l'étudiant à la conduite de projet en entreprise. Il sera admis que ce stage soit effectué dans un laboratoire de recherche. Ce stage est suivi par un enseignant. A l'issue de ce stage, un rapport est rédigé par l'étudiant et une soutenance se déroule en présence d'enseignants en informatique et des professionnels ayant encadré le stage.
- A la fin de la deuxième année : un stage se déroule à plein temps dans une des équipes d'accueil du MR M2SC sous la responsabilité d'un habilité à diriger les recherches. A l'issue de ce stage, l'étudiant rédige un mémoire. Une soutenance orale a ensuite lieu devant un jury constitué au minimum de l'enseignant-chercheur, du responsable du stage du MR et d'au moins trois habilités à diriger les recherches.

Ces stages ont un rôle déterminant pour l'attribution du Master Recherche. Tous les stages sont obligatoires.

## PROGRAMME DES ENSEIGNEMENTS

Les volumes horaires indiqués ci-dessous ne sont qu'indicatifs.

<b>1ère année - 1er semestre</b>	Total heures : 291h	Crédits : 30
----------------------------------	---------------------	--------------

**Recherche Opérationnelle** **C : 18h TD : 18h TP : 24h** **Crédits : 4**  
 Méthodes de résolution des problèmes d'optimisation : programmation linéaire, séparation et évaluation, méthodes de coupes, programmation dynamique), heuristiques stochastiques (TABU, Recuit simulé).

**Génie Logiciel II** **C : 15h TD : 15h TP : 12h** **Crédits : 4**  
 Méthodologie, Processus développement Unifié, UP, cycle de vie du logiciel, TUP, cycle Y, RUP, eXtreme Programming et UML.

**Bases de Données avancées** **C : 12h TD : 12h TP : 18h** **Crédits : 4**  
 Cheminement du cours : étude des fondements des bases de données réparties et fédérées : notion de bases de données réparties, techniques de répartition : fragmentation, répllication. Gestion des transactions et des requêtes réparties, conception d'une bd fédérée.

**Programmation fonctionnelle** **C : 12h TD : 12h TP : 18h** **Crédits : 4**  
 Introduction au  $\lambda$ -calcul, concepts d la programmation fonctionnelle et applications en *Scheme*.

**Langue vivante I - Anglais** **TD : 25h** **Crédits : 2**

**Langue vivante II – Allemand/Espagnol ou libre** **TD : 20h** **Crédits : 2**

**Vie d'entreprise** **C : 4h TD : 14h** **Crédits : 3**  
 Qualité, communication et entretien d'embauche

**Projet** **Crédits : 3**

### Parcours Ingénierie des Systèmes Informatiques Distribués

**Intelligence Artificielle II** **C : 12h TD : 12h TP : 18h** **Crédits : 4**  
 Notion d'apprentissage, notions d'espaces d'hypothèses, d'hypothèses +spécifiques/+générales, de biais inductif, de compromis biais-variance, de sur-apprentissage. Arbres de décision basé sur l'algo ID3 de Quinlan, avec critères de gain d'entropie et de gain de séparation, principe de traitement du bruit et du sur-apprentissage. Classifieurs bayésiens et à base de noyaux. Réseaux de neurones artificiels : perceptron à seuil, interprétation géométrique et problèmes (non) linéairement séparables, règle d'apprentissage du perceptron à seuil, perceptron linéaire avec mesure d'erreur et règles du gradient et gradient stochastique, perceptrons non linéaires dérivables et rétro-propagation du gradient.

### Parcours Ingénierie du Logiciel Libre

**Technologies XML** **C : 9h TD : 9h TP : 24h** **Crédits : 4**  
 Introduction à XML. Transformations de documents XML : feuilles de style (CSS, XSLT), Xpath. Liaison de documents en XML : XLink, Xpointer. Manipulations de documents XML : les API XML, SAX et DOM. XML et les bases de données (XPath, XQuery, ...).

<b>1ère année – 2<sup>ème</sup> semestre</b>	Total heures : 254h	Crédits : 30
--	---------------------	--------------

**Compilation** **C : 12h TD : 12h TP : 24h** **Crédits : 4**  
 Techniques et outils pour l'analyse d'un flot de données (expression régulière, automate, grammaire, analyseur LL, LR, SLR et LALR). Analyse lexicale, analyse syntaxique, sémantique, génération de code. Utilisation des outils Lex et Yacc.

**Systèmes Temps Réel** **C : 12h TD : 12h TP : 18h** **Crédits : 4**  
 Ce cours présente les particularités des systèmes temps réel et embarqués et les mécanismes utilisés pour réaliser ce type de système. Il se base sur l'étude de la synchronisation des processus, la communication des processus parallèles, les tâches Ada...

**Systèmes Distribués I** **C : 15h TD : 15h TP : 21h** **Crédits : 5**  
 Parallélisme, Multithreading, objet réparti, client/serveur, RPC, RMI, CORBA.

**Langue vivante I - Anglais** **TD : 25h** **Crédits : 2**

<b>Langue Vivante II – Allemand/Espagnol ou libre</b>	<b>TD : 20h</b>	<b>Crédits : 2</b>
<b>Vie d'entreprise</b> Marketing et insertion professionnelle	<b>TD : 26h</b>	<b>Crédits : 2</b>
<b>Projet</b>		<b>Crédits : 4</b>
<b>Stage (16 semaines)</b>		<b>Crédits : 3</b>

### Parcours Ingénierie des Systèmes Informatiques Distribués

<b>Informatique Théorique</b>	<b>C : 18h TD : 18h TP : 6h</b>	<b>Crédits : 4</b>
Complexité. Décidabilité. Système formel. Machine de Turing.		

### Parcours Ingénierie du Logiciel Libre

<b>Composants et architecture logicielle</b>	<b>C : 12h TD : 12h TP : 18h</b>	<b>Crédits : 4</b>
Langages de description d'architecture (ADL, ...). Introduction à MDA ( <i>Model Driven Architecture</i> ) et MDD ( <i>Model Driven Development</i> ) : les modèles (CIM, PIM, PSM et UML comme langage de modélisation), le processus de transformation et la relation MDA/code. Panorama de composants logiciels libres. Place des composants dans une architecture de logiciels libres. Spécification et mise en oeuvre de frameworks open source de composants. Conception d'une application à base de composants.		

<b>2ème année - 1<sup>er</sup> semestre</b>	<b>Total heures : 308h</b>	<b>Crédits : 30</b>
---	----------------------------	---------------------

<b>Entretien d'embauche</b>	<b>TD : 1h</b>	<b>Crédits : 1</b>
<b>Langue vivante I - Anglais</b>	<b>TD : 30h</b>	<b>Crédits : 4</b>
<b>Langue vivante II – Allemand/Espagnol ou libre</b>	<b>TD : 30h</b>	<b>Crédits : 2</b>

### Parcours Ingénierie des Systèmes Informatiques Distribués

<b>Réseaux avancés</b>	<b>C : 9h TD : 9h TP : 24h</b>	<b>Crédits : 4</b>
Rappel sur TCP/IP : routage, adressage, sockets, ... Technologies d'un Intranet (Réseaux Locaux d'Entreprise) : matériels (hub, switches, routeur, passerelle), coûts, débits et mise en place sous Linux d'un réseau local. Technologies d'Internet : connexion à Internet, matériel (routeur, modem), configuration, droit d'accès, sécurité, RNIS, PPP, Ligne Spécialisé, coûts, débits ; le protocole http, mise en oeuvre d'un serveur Web (Apache); proxy cache ; le protocole PPP : client et serveur PPP; authentification PAP ; mise en place sous Linux d'une connexion PPP à un Intranet , configuration d'une passerelle firewall. Technologies d'un Extranet : Réseaux privés, réseaux privés virtuels (VPN); cryptage des données (RSA, DES), mise en place sous Linux d'une connexion X21 Transpac, Réseaux de sauvegarde (SAN) : utilité, exemples concrets, Single Sign On : un mot de passe pour tout le réseau		
<b>Systèmes distribués II</b>	<b>C : 12h TD : 12h TP : 18h</b>	<b>Crédits : 5</b>
J2EE, serveur d'applications, JBoss, Jonas, Architecture N-tiers, JSP/Servlet, Struts, Enterprise JavaBeans, Source de données, Administration serveur, Eclipse, Introduction à .Net.		
<b>Programmation orientée objets avancée</b>	<b>C : 5h TD: 5h TP : 32h</b>	<b>Crédits : 4</b>
Méta-classe, Introspection, Réflexivité, Exécution Dynamique, Design-Pattern, Modèle-Vue-Contrôleur, Java Avancé (Model, Renderer, etc.)		
<b>Gestion de projets informatiques</b>	<b>C : 10h TP : 20h</b>	<b>Crédits : 3</b>
Méthodologies de gestion de projets et cas pratiques.		
<b>Génie logiciel III</b>	<b>C : 12h TD: 9h TP : 21h</b>	<b>Crédits : 4</b>
Problèmes de l'ingénierie des systèmes informatiques distribués. Spécification et conception distribuée. Assurance qualité des logiciels. Gestion des projets informatiques distribués		
<b>Vie d'entreprise</b>	<b>C : 25h TD : 25h</b>	<b>Crédits : 3</b>
Économie et innovation.		

## Parcours Ingénierie du Logiciel Libre

### **Système, Réseaux et Sécurité**

**C : 20h TD : 10h TP: 40h**

**Crédits : 7**

Déploiement de distributions Linux (Debian, Mandriva, RedHat, Novell Linux et Gentoo) et étude du noyau Linux (Gestion mémoire, gestion de fichiers – ext2, ext3, lufs, ... - , gestion de processus, drivers de périphériques, synchronisation, IPC, ...). Sécurisation des systèmes et des services réseaux (DNS, Web, Mail, ...). Sécurité physique, sécurité des données et sécurisation du système Linux. Filtrage, topologie, segmentation et DMZ (FireWall, IP table, Proxy/Squid, Reverse Proxy, VPN, ...). Surveillance, audit et outils de monitoring. Sécurisation des échanges (Cryptographie - SSL -, certificats de clé publique, norme X.509 et PKIX et architecture technique). Identification et SSO

### **Développement d'applications Web**

**C : 20h TP: 50h**

**Crédits : 7**

Web Services : architecture orientée services (WSDL, SOAP, XML-RPC, HTTP, ...), concepts avancés de Java (JSP, Servlets, les EJB et Enterprise Service Bus – ESB - ), déploiement et mise en oeuvre : Python/Zope, J2EE et serveurs d'applications (Tomcat, Jonas, Jboss, ...), persistance des données (Hibernate, ...), IHM Web : architecture, cinématique, transactions et outillages (Strust, ...) ; Systèmes de gestion de contenu : concepts et outils (Zope, SPIP, Mambo, OpenCMS, ...).

### **Environnements de développement libres**

**C : 30h TD : 10h TP: 30h**

**Crédits : 6**

Méthodologies et gestion de projets (méthodes agiles - RAD et eXtreme Programming -,...). Outils de gestion de suivi de projets (bugzilla, ...). Environnement de développement collaboratif : concepts, méthodologies et outillage (gforge, Eclipse, ...). Test unitaire (JUnit, cppunit, phpunit, PyUnit...). Documentation (doxygen, ...). Architecture de développement et de déploiement (autotools, gestion de paquets, ...). Gestion de versions centralisée (CVS, subversion, ...) et distribuée (arch,...). Rappels : sh, perl, python et makefile.

### **Conférences**

**C : 45h**

**Crédits : 3**

Cycle d'une quinzaine de conférences sur des thèmes relatifs aux différents modules du parcours.

**2ème année – 2<sup>ème</sup> semestre** Total heures : 72h

**Crédits : 30**

### **Projet bibliographique**

**Crédits : 6**

D'octobre à la fin décembre, chaque étudiant effectue individuellement un projet d'initiation à la recherche sous la responsabilité d'un enseignant-chercheur. Ce projet consiste à effectuer une recherche bibliographique donnée ou à effectuer l'analyse et une critique d'un logiciel existant et d'en rédiger une synthèse sur un sujet proposé. A l'issue du projet, l'étudiant présente son travail au cours d'une soutenance orale de 15 minutes. Le jury est constitué au minimum de l'enseignant-chercheur responsable du MR et d'au moins trois habilités à diriger les recherches. Le projet d'initiation à la recherche a pour objectifs :

- Apprendre à faire une recherche bibliographique sur un thème donné ou à effectuer l'analyse et une critique d'un logiciel existant (selon le type de projet choisi)
- Apprendre à en faire une synthèse écrite
- Apprendre à en faire une présentation orale synthétique (15 minutes)

### **Stage**

**Crédits : 12**

### **Informatique évolutionnaire**

**C : 18h**

**Crédits : 3**

L'évolution artificielle regroupe une classe d'algorithmes fondés sur des modèles simplifiés de l'évolution des systèmes naturels. Le principe général est de faire évoluer une population de solutions potentielles d'un problème, formulé en termes de recherche du maximum d'une fonction. Les moteurs de l'évolution sont un mécanisme de sélection (l'idée est typiquement de donner une probabilité de survie plus grande aux individus les mieux classés) et des opérateurs génétiques (mutation, croisement, etc.). On obtient ainsi des outils d'optimisation très robustes et efficaces dans de nombreux cas où les autres méthodes échouent, spécialement dans le cas de problèmes discrets, non-linéaires et de fonctions très irrégulières. Les applications au monde réel sont nombreuses et étonnamment variées : contrôle d'unités de génie chimique, conception de profils en aéronautique, commande de robots, théorie des jeux, économie, programmation automatique, traitement du signal et vision artificielle.

- introduction générale: optimisation, codage, fitness, opérateurs évolutionnaires, etc ;
- algorithmes génétiques, algorithmes d'estimation de distributions ;
- programmation génétique ;
- optimisation par colonies de fourmis.

## **Modélisation et simulation de l'éclairage et de sa dynamique**

**C : 18h**

**Crédits : 3**

Ce cours présente la problématique générale liée à la modélisation et à la simulation de l'éclairage d'une scène virtuelle. Il se focalise plus précisément sur les problèmes liés à l'illumination globale et sur les techniques mises en oeuvre pour les résoudre :

- méthodes et algorithmes
- exploitation des limites de la perception visuelle humaine
- recherche de solutions matérielles et parallèles

Il se termine par une introduction aux problèmes liés à la prise en compte de la dynamique de l'éclairage, qu'elle soit liée à l'interaction avec l'utilisateur ou à des phénomènes internes à la modélisation.

## **Modélisation et évolution des logiciels**

**C : 18h**

**Crédits : 3**

- Modélisation architecturale des logiciels distribués
- Modélisation structurelle des logiciels
- Modélisation fonctionnelle
- Modélisation comportementale
- Modélisation qualitative
- Evolution des logiciels
- Typologie des évolutions
- Analyse d'impact des modifications du logiciel
- Modélisation du développement coopératif des logiciels
- Approches et environnements du développement coopératif

## **Modélisation et simulation des systèmes naturels**

**C : 18h**

**Crédits : 3**

- Introduction à la modélisation de systèmes
- DEVS : Discrete Event Specification
- Spécification formelle de modèles
- Algorithme de simulation
- Extensions : Cell-DEVS, DS-DEVS, ...
- Couplage de modèles et multi-modélisation
- Simulation distribuée : concepts et problématiques
- Systèmes multi-agents

## **MODALITE D'INSCRIPTION**

Les dossiers de candidature sont à retirer à partir du 25 mars 2006 au CGU de Calais :

UNIVERSITE DU LITTORAL – COTE D'OPALE  
40, RUE FERDINAND BUISSON – BATIMENT A  
BP 649  
62228 CALAIS CEDEX

email : [renaud@lil.univ-littoral.fr](mailto:renaud@lil.univ-littoral.fr)  
web : <http://dpt-info.univ-littoral.fr>

Date limite de dépôt des dossiers de candidature : **vendredi 30 juin 2006**

## **CALENDRIER**

### **Pour la première année**

Rentrée : lundi 17 septembre 2006 à 10h  
Fin des enseignements : vendredi 30 mars 2007  
Début du stage : lundi 2 avril 2007  
Fin du stage : entre le 20 juillet 2007 et le 31 août 2007  
Soutenances de stage : lundi 3 septembre 2007  
Jury de première année : lundi 3 septembre 2007  
Deuxième session : à partir du mardi 4 septembre 2007

### **Pour la deuxième année**

Rentrée : lundi 10 septembre 2006 à 9h  
Fin des enseignements : vendredi 30 mars 2007  
Début du stage : lundi 2 avril 2007  
Fin du stage : entre le 17 août 2007 et le 31 août 2007  
Soutenances de stage : mercredi 5 septembre 2007  
Jury de Master : mercredi 5 septembre 2007  
Deuxième session : à partir du jeudi 6 septembre 2007