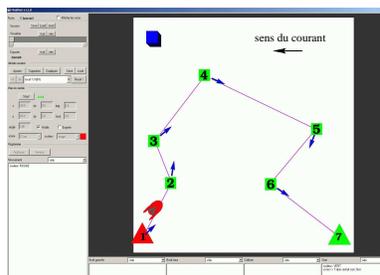


Rapport d'atelier "MobiNet"

Conception d'un TP de mathématiques interactif en lycée



Moniteurs :

Peggy Provent

Maud Marchal

Frédéric Ruyer

Tuteur :

Fabrice Neyret

2004-2005

<http://mobinet.imag.fr>

Table des matières

1	Introduction	4
2	Le projet MobiNet et le monitorat	4
2.1	MobiNet : un nouvel outil pédagogique pour l'enseignement des mathématiques et de la physique	4
2.1.1	Historique et motivations	4
2.1.2	Présentation du logiciel	5
2.2	L'informatique dans l'éducation	7
2.2.1	Modèles d'apprentissage	7
2.2.2	Quelques autres logiciels utilisés en pédagogie	8
2.3	Monitorat et Enseignement secondaire	10
2.3.1	Intérêt des contacts avec le lycée	10
2.3.2	Ouverture vers la recherche	11
3	Expérimentations aux Semaines de l'Ingénieur 2004	11
4	Réalisation d'un TP au lycée Europole	12
4.1	Contexte et premiers contacts	12
4.2	Cheminement vers un sujet	12
4.3	Élaboration du sujet	13
4.3.1	Pourquoi une course de bateaux ?	13
4.3.2	Réalisation et contraintes	13
4.4	Résultat : Le sujet de TP proposé	14
4.5	Réalisation en classe	16
4.5.1	Déroulement	16
4.5.2	Évaluation des résultats	17
4.6	Travaux futurs et perspectives	18
5	Conclusions et perspectives	18
5.1	Évolutions du projet	18
5.2	Apport de l'atelier	19
A	Documentation du logiciel	20
A.1	Interface	20
A.2	Langage de programmation de MobiNet	24
B	Un exemple de progression en classe de Seconde	28
C	Notre sujet de TP "Vecteurs et Courses de bateaux"	29
D	Fiche d'évaluation "premières impressions"	34
E	Evaluation du TP par Pirouz Djoharian	35

1 Introduction

L'objectif du monitorat d'initiation à l'enseignement supérieur est de permettre à des étudiants inscrits en thèse de se préparer aux fonctions d'enseignant-chercheur. Dans cette perspective, outre des formations et théoriques (stages) et une initiation pratique (enseignement), il est demandé aux moniteurs de troisième année de s'investir dans des ateliers dont le but est d'étudier un sujet en rapport avec la transmission du savoir (éducation, vulgarisation, communication . . .). L'atelier MobiNet, ouvert en 2002 par Sylvain Lefebvre (sous le tutorat de Fabrice Neyret), propose de s'intéresser aux applications d'un nouveau logiciel d'enseignement à la croisée des mathématiques, de la physique, de l'algorithmique, et des jeux vidéos.

Nous présentons dans ce rapport notre contribution à la réflexion et à la pratique autour de ce logiciel, en insistant sur deux points : l'expérimentation concrète, et l'élaboration de séances de travaux pratiques, débouchant sur une expérimentation au lycée Europole, sur deux groupes d'une classe de seconde dans le cadre d'un TD de mathématiques.

2 Le projet MobiNet et le monitorat

2.1 MobiNet : un nouvel outil pédagogique pour l'enseignement des mathématiques et de la physique

2.1.1 Historique et motivations

Le projet MobiNet a vu le jour en 2002 dans le cadre de la Semaine Découverte Ingénieur, événement organisé par l'Institut National Polytechnique de Grenoble dans le but de faire découvrir à des lycéens (de niveau 1ère et 2de) les métiers de l'ingénieur et le monde de la recherche scientifique, et s'est ensuite étendu hors de cette opération, notamment du fait de sa mise à disposition en ligne comme logiciel libre. Il a par exemple été utilisé par des enseignants, et dans le cadre des TPE. Il a été réalisé par Joëlle Thollot (Maître de Conférence à l'ENSIMAG), Fabrice Neyret (Chargé de recherche au CNRS), Samuel Hornus (Doctorant UJF, moniteur UJF), et Sylvain Lefebvre (Doctorant UJF, moniteur à l'INPG), au sein de l'équipe iMAGIS du laboratoire GRAVIR (INRIA Rhône-Alpes).

Lors des semaines ingénieur, le "prétexte" avancé pour MobiNet est de faire découvrir aux lycéens la conception des jeux vidéos. En effet, ce domaine possède deux grandes qualités pour l'amélioration de la perception des liens entre modélisation et réalité :

- c'est un très bon terrain d'application de nombreuses notions de physique, de mathématiques, et d'algorithmique.
- de part la nécessité qu'il impose de "déconstruire" le réel, il conduit à l'élaboration de modèles, afin de réaliser le "monde virtuel" envisagé.

Ainsi, il permet aux élèves d'une part de mieux s'appropriier les notions abstraites vues en classe, et d'autre part de développer chez eux une introduction à la démarche scientifique.

Une autre qualité, et non des moindres, de ce domaine, est son attractivité, qui permet d'accrocher des publics à priori peu portés vers les domaines scientifiques.

Sur un plan plus pédagogique, les mécanismes d'apprentissage que l'on souhaite mettre en oeuvre en utilisant MobiNet sont :

- l'appropriation des savoirs par le fait de manipuler concrètement les concepts.
- guider l'intuition en liant les paramètres numériques à un comportement mathématique ou physique.
- apprendre par essai-erreur : le processus d'apprentissage n'est pas "linéaire". Formuler des hypothèses et les confirmer ou les infirmer, et les reformuler, par l'expérience, est une des bases de la démarche scientifique.

Nous verrons plus loin, après une brève description des fonctionnalités du logiciel, quels sont plus précisément les modèles d'apprentissage en jeu, et enfin donnerons quelques pistes vers d'autres logiciels à visée pédagogique.

2.1.2 Présentation du logiciel

MobiNet permet de créer des applications graphiques complexes en combinant des objets dont le comportement et l'apparence peuvent être déterminés soit directement (variable d'état), soit par un ensemble de formules. Ces objets sont appelés *mobiles*. Chaque mobile est décrit par un ensemble de *variables d'état* (forme, couleur, position, angle ...). La figure 1 présente l'interface du logiciel. On y distingue :

- La zone graphique où apparaissent les mobiles
- Les zones de sélection de mobiles et de variables d'états qui donnent les informations relatives au mobile sélectionné (position, angle, etc ...)
- Les zones de programme qui spécifient le comportement du mobile.

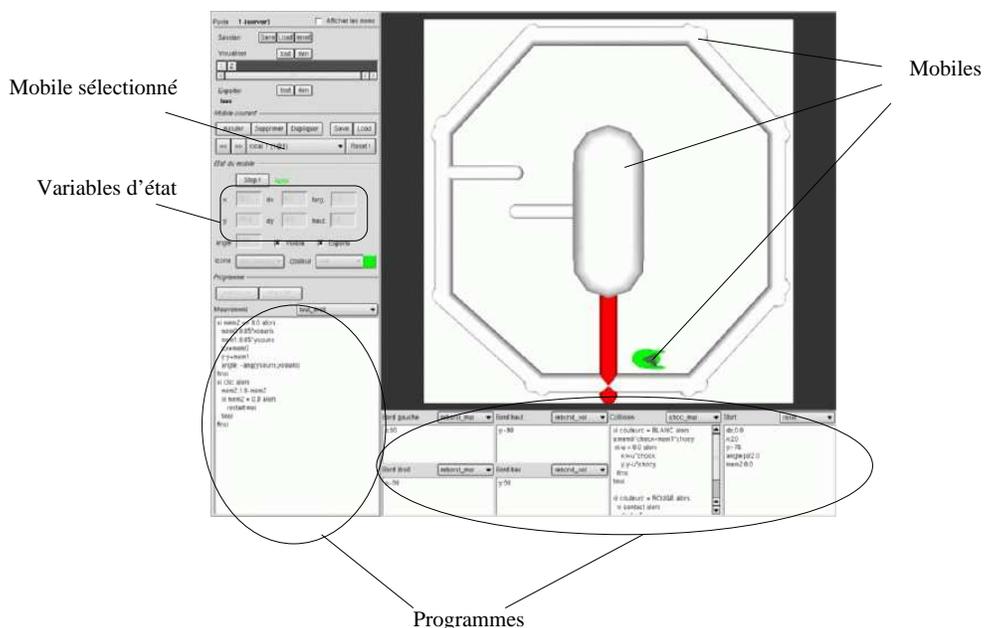


FIG. 1 – L'interface de MobiNet

Le langage de programmation, très intuitif, permet de décrire *l'évolution* des variables d'état. Il a été conçu en collaboration avec des professeurs de lycée afin de faciliter la prise en main du logiciel par les lycéens (par exemple quant à la cohérence des notations vis à vis de leurs habitudes). Le mode d'emploi détaillé est fourni en annexe A.

Un des atouts de MobiNet est de permettre la visualisation et l'interaction avec les mobiles d'un *autre* ordinateur via le réseau. Il est ainsi possible de créer des travaux pratiques basés sur le travail collaboratif entre les divers postes.

MobiNet permet donc de créer une large gamme d'applications. La figure 2 présente quelques captures d'écrans issues de sessions fournies avec MobiNet :

- Système solaire animé
- Jeux vidéos simples
- Figures géométriques animées
- Simulation bio-mécanique de jambe
- ...

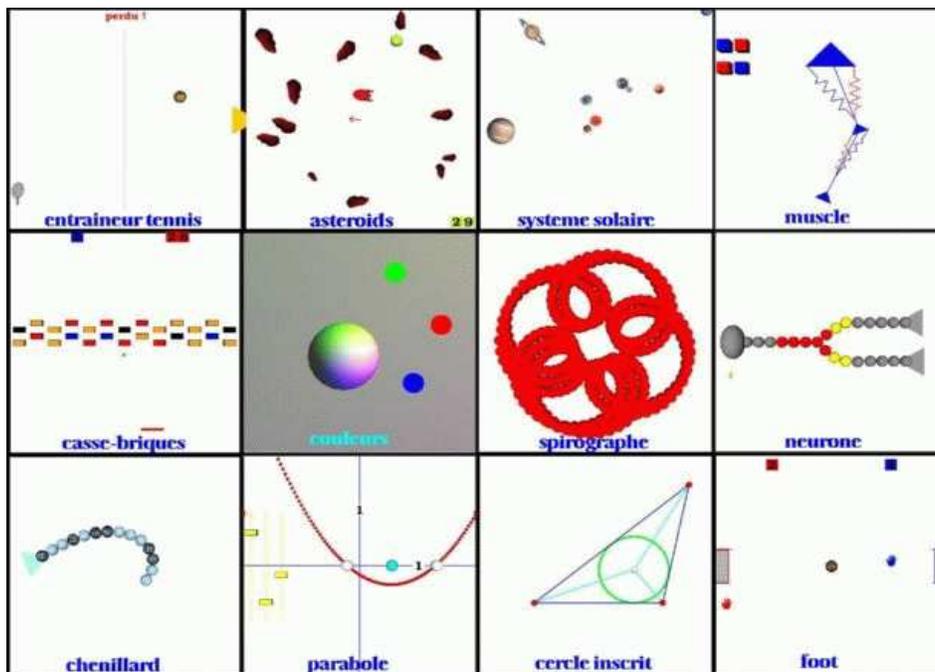


FIG. 2 – Exemples d'applications

Chacun de ces exemples a été produit en combinant des mobiles et en programmant leur comportement.

On voit à travers ces exemples quelle variété de situation peuvent être conçues avec MobiNet, et que de nombreuses situations d'enseignement peuvent être imaginées.

2.2 L'informatique dans l'éducation

Afin de mieux se fixer les idées sur les modalités de conception et de mise en oeuvre de séances avec des élèves, nous présentons dans cette section quelques concepts issus des écoles de pensée liées à la pédagogie, puis présentons d'autres logiciels d'enseignement assisté par ordinateur.

2.2.1 Modèles d'apprentissage

Le XX^{ème} siècle a vu de nombreux courants issus de la psychologie se développer et influencer l'enseignement, développant considérablement le domaine des sciences de l'éducation. Nous faisons ici un bref tour d'horizon des principales écoles de pensée et voyons comment elles sont liées au domaine de l'enseignement assisté par ordinateur.

Le béhaviorisme

La théorie comportementaliste, ou béhaviorisme (transposition directe du terme anglo-saxon) a été fondée par le psychologue John Broadus Watson (1878-1958). Cette théorie part du principe que le fonctionnement du cerveau et de la pensée sont en eux-mêmes inaccessibles et que l'analyse psychologique doit se fonder sur la seule observation des comportements individuels. De cette base est né l'enseignement programmé de Skinner (1904-1990), qui propose d'appliquer à l'homme des techniques d'apprentissage fondées sur le phénomène du conditionnement mis en évidence chez l'animal par Pavlov en 1930. Le dispositif imaginé par Skinner, d'abord réalisé sur support papier puis sur ordinateur, consiste à organiser la progression de l'apprentissage, en contrôlant le comportement de l'élève par des questions dont la correction peut être automatisée par un programme informatique. Suivant que le choix de l'élève correspond ou non à la bonne réponse, et en fonction des performances précédentes, l'élève subit un renforcement positif (message de félicitations, augmentation du score, nouvelle question plus difficile) ou négatif (message d'erreur, retour en arrière forcé, score diminué). Les apports principaux de ce courant à la pédagogie sont l'individualisation du rythme d'apprentissage, la correction immédiate des erreurs et la valorisation pédagogique de l'erreur de l'élève. Cependant, on perçoit aisément que le modèle comportementaliste, en ignorant délibérément ce qui se passe "dans" l'individu, ne peut suffire à modéliser le processus d'apprentissage.

Le cognitivisme

Le cognitivisme est l'étude scientifique de la cognition ; c'est-à-dire l'ensemble de processus mentaux tels la perception, la mémorisation, le raisonnement et la résolution de problèmes. Le psychologue Jean Piaget (1896-1980) a beaucoup contribué à ce courant en étudiant le développement de l'intelligence chez les enfants. Pour Piaget, l'intelligence n'est pas une faculté mentale parmi d'autres, mais une modalité d'une fonction plus générale : "L'adaptation", qu'il définit comme l'état d'équilibre maximum entre un organisme vivant et le milieu. Cette adaptation s'acquiert selon différentes formes ou structures : ainsi, l'adaptation mentale est un prolongement de l'adaptation biologique. On peut dire, grossièrement, que l'adaptation à l'environnement est l'équilibre entre assimilation (incorporation des expériences nouvelles dans des structures existantes) et accommodation (modification des structures existantes, provoquée par les expériences nouvelles), la régulation entre sujet et milieu, que ce soit d'ordre biologique, affectif, mental ou

social. Plus généralement, les sciences cognitives choisissent d'ouvrir la "boîte noire" des mécanismes de l'intelligence et de s'attaquer directement à la description de l'activité mentale, en s'appuyant principalement sur les acquis de la neurobiologie, de la cybernétique et de l'informatique. Dans cette perspective, Allen Newell (1927-1992) et Herbert Simon (1916-2001) encouragés par les progrès fulgurants de l'informatique, introduisent dans les années quatre-vingts l'intelligence artificielle dans les sciences cognitives. Pour eux, le cerveau fonctionne comme un système de traitement de l'information, c'est-à-dire comme un ordinateur. Dans l'éducation, la perspective d'un ordinateur enfin "intelligent" ranime "l'espoir" de l'automate enseignant et tuteur capable d'analyser les réponses et les difficultés de l'élève et de s'adapter en conséquence.

Ainsi, dans les modèles cognitivistes, l'apprentissage est perçu comme un processus individuel de construction de connaissances, principalement fondé sur les interactions de l'élève avec un ensemble d'informations organisées, documents bruts, cours, ressources éducatives diverses, livres, logiciels, ou services télématiques.

Autres courants

Issu de l'école cognitiviste, le psychologue Jerome Bruner (1915-) effectue une rupture théorique marquée par un recentrage sur les dimensions sociales et affectives de l'apprentissage, en fondant la "psychologie culturelle". En désignant l'éducation comme "entrée dans la culture", la relation de tutelle entre l'élève et un adulte plus expérimenté que lui est placée au fondement de toute théorie de l'apprentissage : l'essentiel de l'apprentissage réside dans les interactions qui se nouent au sein de la classe, entre enseignants et élèves, entre pairs.

On peut aussi citer l'apport de grands pédagogues tels Célestin Freinet (1896-1966), qui propose un apprentissage basé sur l'expérience tâtonnée et non sur l'application de règles et de théorèmes, ainsi que la confrontation immédiate aux problèmes posés, à partir de la vie courante et de l'imagination des enfants.

Liens avec l'enseignement assisté par ordinateur

Voyons maintenant comment toutes ces approches sont liées à l'enseignement assisté par ordinateur (EAO). Les applications directes de l'EAO apparaissent clairement dans l'approche comportementaliste, et dans celle de l'intelligence artificielle. Dans le cadre de la psychologie culturelle, qui replace les relations entre humains à une place plus importante, l'ordinateur peut être vu comme un support supplémentaire aux supports traditionnels, la dimension de réseau prenant une place plus importante dans la perspective de relier effectivement des individus, et non des machines. On voit également que dans une approche Piagetienne, l'immédiateté proposée par la machine ainsi que la simulation de la réalité qu'elle fournit peuvent être vus comme des stimulants pour l'apprentissage.

2.2.2 Quelques autres logiciels utilisés en pédagogie

Nous allons distinguer grossièrement cinq classes de logiciels : les logiciels non dédiés spécifiquement à la pédagogie, les logiciels de banque d'exercices, les logiciels de géométrie, les logiciels multi-usage, et les logiciels d'apprentissage du raisonnement.

– les logiciels non dédiés spécifiquement à la pédagogie : Les plus "primitifs" sont tous les

logiciels de type "tableur" (Gnumeric¹, Kspread², OpenofficeCalc³, Microsoft Excel, Lotus123, ...). Leurs fonctionnalités communes sont la possibilité de traiter des séries de chiffres à l'aide de fonctions mathématiques, et de tracer des graphiques sommaires. Ce sont donc des "super calculatrices" permettant d'expérimenter dans divers domaines (étude de fonctions, statistiques ...) essentiellement au niveau numérique. Ce type de logiciels est utilisé dès le niveau 3ème : l' "initiation à l'utilisation de tableurs-grapheurs en statistiques" est un des contenus définis dans le programme du bulletin officiel.

Sont également utilisés, notamment dans l'enseignement supérieur des logiciels commerciaux tels que Matlab ou Maple, qui sont également des "super calculatrices" dédiées au départ au monde de la recherche et aux ingénieurs, Maple ayant de plus des fonctionnalités avancées de calcul formel. Citons également Scilab⁴, développé par des équipes de l'INRIA et de l'ENPC, qui est analogue à Matlab (mais gratuit).

- les logiciels de banque d'exercices : Ce sont des logiciels qui proposent des exercices avec corrigés, organisés par sujet, et permettent de rejouer le même type d'exercice, et fournissent un corrigé et parfois indiquent les sources possibles d'erreurs. Citons Mathenpoche⁵, -développé par l'association Sésamath, association ayant pour vocation de diffuser gratuitement des ressources pédagogiques et des outils professionnels utilisés pour l'enseignement des Mathématiques- qui couvre le collège et la seconde.
Il existe également de nombreux logiciels commerciaux de ce type (Adi, Adibou, Lapin-Malin et autres...), qui associent souvent un côté ludique poussé (usage domestique oblige) à l'acquisition de connaissances.

- les logiciels de géométrie : Comme leurs noms l'indique, ces logiciels permettent de manipuler des objets géométriques, et d'effectuer des calculs. Citons deux logiciels développés au CNAM au sein du Centre de Recherche et d'Expérimentations sur l'Enseignement des mathématiques⁶ : GeoplanW est un logiciel qui permet de définir et de manipuler des objets géométriques et numériques (points, droites, cercles, nombres, transformations, repères, courbes, vecteurs, fonctions numériques, suites numériques, etc ...), GeospaceW permet de créer et de représenter des figures de l'espace de façon analogue, et GeoplanJ est une version (encore jeune) Java de Geoplan destinée à élargir les plates-formes sur lesquelles fonctionne Geoplan.

Un autre logiciel, commercial, bien qu'initialement développé au sein du CNRS, est Cabri-géomètre.

- les logiciels multi-usage : Xcas⁷, développé à l'UJF de Grenoble , permet de faire du cal-

¹ <http://www.gnome.org/projects/gnumeric/>

² <http://www.koffice.org/kspread/>

³ <http://www.openoffice.org/product/calc.html>

⁴ <http://scilabsoft.inria.fr>

⁵ <http://www.sesamath.hautesavoie.net/mathenpoche/index.php>

⁶ <http://www2.cnam.fr/creem/>

⁷ http://www-fourier.ujf-grenoble.fr/~parisse/giac_fr.html

cul formel, de la géométrie dynamique, du tableur et de la programmation. Il mêle donc différents aspects vus précédemment.

- les logiciels d'apprentissage du raisonnement : Ils sont basés sur la théorie de la démonstration, et permettent d'apprendre à structurer un raisonnement. A l'origine dédiés à la recherche, ils commencent à poindre le nez dans le domaine de la pédagogie. Citons deux logiciels français, Coq⁸ (développé à l'INRIA), et PhoX⁹ (développé à l'Université de Savoie). Autour de Coq se développe notamment le projet GeoView¹⁰ dédié au raisonnement géométrique. PhoX a une vocation pédagogique plus directe, et a déjà été utilisé dans ce but à Chambéry et à Paris.

Après ce succinct tour d'horizon du domaine, nous pouvons tenter de situer MobiNet dans ce contexte :

- par son côté graphique, il permet de retrouver certaines fonctionnalités propres aux logiciels de géométrie, sans atteindre à la finesse de ces derniers (notamment lorsqu'ils recourent au calcul formel)
- son côté programmation le place au côté de tout logiciel possédant des possibilités analogues (tableurs, Xcas...), avec cependant des possibilités d'animation plus évoluées.
- ses fonctionnalités réseau et son interface forment sa principale originalité, du fait de sa motivation orientée jeux vidéos et simulations.

2.3 Monitorat et Enseignement secondaire

2.3.1 Intérêt des contacts avec le lycée

S'engager dans un projet qui concerne le lycée possède un double intérêt pour un futur enseignant-chercheur :

- un projet du type MobiNet a entre autre pour vocation d'attirer de futurs étudiants vers les sciences, et donc c'est une façon de s'investir dans la lutte contre la désaffection des filières scientifiques.
- les contacts noués avec les professeurs et les élèves sont fructueux sur plusieurs plans. Avec les professeurs, ils permettent de connaître la réalité des programmes et les façons pratiques avec lesquelles ils sont abordés (i.e. qu'est ce qui "marche" ou pas avec des élèves, en groupes, de niveaux divers), et permettent ainsi une meilleure vision des prérequis des étudiants. Avec les élèves, ils donnent une meilleure perception des publics à venir dans les universités.

⁸ <http://coq.inria.fr/>

⁹ http://www.lama.univ-savoie.fr/sitelama/Membres/pages_web/RAFFALLI/af2.html

¹⁰ <http://www-sop.inria.fr/lemme/geoview/geoview-fra.html>

2.3.2 Ouverture vers la recherche

Un atelier comme MobiNet oriente la curiosité du futur enseignant-chercheur vers plusieurs voies de recherche :

- la réflexion autour des nouvelles pédagogies, et notamment :
 - étudier des techniques permettant de rendre les maths et la physique accessibles aux élèves d'ordinaire rebutés par l'académisme ou l'abstraction.
 - expérimenter le montage d'un TP, servant le cours tout en le mettant en pratique selon un scénario, en s'assurant qu'il "fonctionnera" avec les élèves.
- des considérations plus informatiques autour des interactions homme-machine, et des techniques de programmation
- enfin, tel qu'évoqué dans le paragraphe précédent, l'étude des liens entre le lycée (et plus généralement l'enseignement secondaire) et l'enseignement supérieur.

3 Expérimentations aux Semaines de l'Ingénieur 2004

Les semaines ingénieurs sont organisées deux fois par an par l'INPG dans le but de faire découvrir aux élèves de lycée (seconde ou première) différentes facettes des métiers d'ingénieur. MobiNet a été initialement développé afin d'être utilisé lors de ces semaines pour faire découvrir aux élèves l'informatique et la programmation de manière ludique. Les élèves viennent de classes de seconde, des lycées de l'agglomération (deux lycées par session). Les groupes sont constitués de 15 à 20 élèves accompagnés d'un enseignant. Les séances se déroulent à l'atelier de réalité virtuelle (ARV) de l'ENSIMAG-Montbonnot. La salle est pourvue de 13 postes informatiques et d'un vidéo projecteur. Environ 4 personnes sont présentes pour encadrer la séance. Le TP est construit de manière à amener par petites étapes les élèves à construire un jeu vidéo (pong), auquel ils pourront jouer en réseau, entre 2 postes. Les premières étapes consistent à apprendre à créer et à animer un mobile puis à gérer plusieurs mobiles et les événements qui peuvent se passer. Les différentes étapes par lesquelles ils doivent passer pour atteindre leur but est l'occasion de mettre en pratique de nombreuses notions mathématiques.

Nous avons participé aux semaines ingénieurs 2004 en encadrant des groupes d'élèves. Cela nous a permis de découvrir concrètement les difficultés que nous pouvons rencontrer dans ce type de situation. Par exemple, il est très important d'avoir une idée précise du temps qui doit être consacré à chaque étape de la séance, de manière à pouvoir atteindre la fin du TP. De plus, le fait d'avoir une classe dans une salle informatique pose des problèmes de discipline. Outre la disposition de la salle qui force les élèves à se retourner pour écouter le professeur, l'excitation des élèves face à une activité d'un type nouveau pour eux rend la classe plus difficile à gérer. Le niveau des élèves est très hétérogène. Il faut à la fois encourager et aider les élèves les plus faibles et maintenir la progression de l'ensemble de la classe. Le fort taux d'encadrement lors des semaines ingénieurs permet de bien s'occuper de tous les élèves. Nous avons aussi pu nous rendre compte du niveau global des élèves en classe de seconde, un savoir hésitant et surtout essentiellement théorique, difficile à mobiliser pour une tâche pratique pourtant très mathématisée.

4 Réalisation d'un TP au lycée Europe

4.1 Contexte et premiers contacts

Les premiers contacts ont été obtenus avec l'IUFM de Grenoble et l'IREM par l'intermédiaire de notre tuteur Fabrice Neyret. Michèle Gandit, professeur de Mathématiques au Lycée international de Europe, après avoir vu des présentations de Fabrice Neyret concernant MobiNet, nous a permis de rencontrer d'autres professeurs de ce Lycée. Le logiciel MobiNet a intéressé aussi bien des professeurs de mathématiques que de physique. Nos premières réunions avec un groupe de professeurs ont fait ressortir que le professeur de physique était plus intéressé pour utiliser MobiNet dans le cadre de TPE que dans le cadre d'un cours. Il semblait aussi préférer développer un projet par lui-même, sans avoir besoin d'aide. Au contraire, les professeurs de mathématiques semblaient prêts à utiliser MobiNet pour un TP/TD dans leur classe si nous pouvions développer un sujet. Nous avons aussi évoqué la possibilité de développer un TP commun entre les maths et la physique. Nous avons donc choisi de développer un projet de mathématiques. Les professeurs intéressés par notre projet ne s'occupent que de classe de seconde ou de terminale. L'année de terminale étant une année chargée et délicate pour les élèves, il semblait plus facile et moins déstabilisant pour les élèves de choisir les classes de seconde. Michèle Gandit a préféré rester en marge du projet car elle participe déjà à une expérimentation avec d'autres logiciels. Son collègue Pirouz Djoharian s'est lui investi à nos cotés pour nous aider à déterminer les sujets intéressants d'un point de vue pédagogique. En définitive, pour cette première expérience en lycée, nous avons opté pour la simplicité : un TP de mathématiques avec une classe de seconde. Mais l'équipe enseignante reste bien sûr intéressée pour suivre les différentes étapes de l'expérimentation, dans l'idée soit de réemployer le TP rapidement, soit de développer de nouvelles expérimentations dans l'avenir suite à ce premier essai.

4.2 Cheminement vers un sujet

Après les premiers contacts avec les enseignants du Lycée Europe, Pirouz Djoharian nous a transmis son planning de progression de sa classe de seconde et en particulier les différentes parties du programme et le découpage chronologique de son cours dans l'année (cf. annexe B). A l'aide de ce planning ainsi que des textes officiels du programme de mathématiques en classe de seconde, nous avons commencé à réfléchir à des sujets de Travaux Pratiques réalisables avec MobiNet, en particulier sur quelle partie du programme nous pouvions construire un TP où cet outil pédagogique apporterait une aide supplémentaire à l'équipe pédagogique. L'idée est que le thème du TP soit délicat pour les élèves et que l'emploi de MobiNet puisse être pertinent pour ce sujet. Dans le même temps, il fallait que le TP réalisé ne dure pas plus de temps qu'un TP classique (1h) et qu'il traite des mêmes notions, que ce soit en qualité ou en quantité. Une possibilité serait alors de pouvoir évaluer par la suite le gain d'un TP réalisé sous MobiNet avec un TP classique en ayant un groupe "témoin".

Différentes parties du programme de seconde nous ont semblé convenir : les fonctions (calcul de paraboles, résolution graphique d'inéquations par exemple), la géométrie, la trigonométrie. La géométrie a plus particulièrement retenu notre attention car d'une part, MobiNet se prête bien à

la construction d'objets géométriques de part sa configuration "plan en 2 dimensions", et d'autre part parce que la géométrie occasionne assez fréquemment pour les lycéens des difficultés de représentation dans l'espace des objets qu'ils manipulent.

Il est à noter que les quelques idées de projets sur les fonctions (calcul de paraboles, résolution graphique d'inéquations par exemple) ont également suscité l'intérêt des enseignants mais nous avons choisi de nous orienter vers la géométrie, et plus particulièrement les vecteurs car cette partie du programme coïncidait dans le déroulement du cours de Pirouz Djoharian avec la période où nous pouvions intervenir au lycée Europole.

L'idée finalement retenue a été de construire un TP MobiNet sur les vecteurs, en le présentant sous la forme d'une course de bateaux.

4.3 Élaboration du sujet

4.3.1 Pourquoi une course de bateaux ?

Le principe est de considérer le plan représenté sous MobiNet comme un plan d'eau où va se dérouler la course. On ajoute à ce plan des bouées numérotées. L'objectif est alors de construire la trajectoire du bateau de la bouée de départ à la bouée d'arrivée en passant par toutes les bouées dans l'ordre. La trajectoire est représentée à l'écran comme une suite de vecteurs, un vecteur allant d'une bouée à l'autre. Au cours du TP, on perturbera cette trajectoire par des courants et les élèves auront donc des modifications à effectuer sur la trajectoire.

La motivation de l'élève se trouve ici dans le fait qu'une course de bateaux est une situation concrète, et que la simulation qu'il va pouvoir produire avec MobiNet sera similaire à cette situation réelle : un navigateur va d'une bouée à une autre en suivant un certain cap (ou "direction") et en parcourant une certaine distance (notion de norme du vecteur).

4.3.2 Réalisation et contraintes

Pour réaliser le TP, nous avons tout d'abord pris en compte les notions que l'enseignant désirait transmettre. Le TP devait se dérouler environ lorsque la moitié du cours avait été effectuée et devait ainsi permettre aux élèves de manipuler les vecteurs. Les objectifs pédagogiques du TP sont : calcul de coordonnées d'un vecteur et visualisation du résultat graphiquement, calcul de la norme d'un vecteur, addition et soustraction de vecteurs, construction de vecteurs par combinaison, multiplication d'un vecteur par un scalaire.

Nous avons donc à inclure ces notions en les formulant dans une situation concrète qu'est une course de bateaux. Nous avons choisi de ne pas introduire la notion de navigation à voile qui est trop complexe par rapport au temps imparti pour le TP. La course de bateaux modélisée est donc celle de bateaux à moteur. Dans un premier temps, nous avons simplement donné comme instruction de suivre un cap (calcul des coordonnées d'un vecteur) puis nous avons dans un second temps introduit des courants afin de faire effectuer à l'élève des opérations sur les vecteurs (addition, soustraction, multiplication par un scalaire).

Les contraintes sont également temporelles et matérielles. Nous disposons d'une heure pour le TP en demi-classe (environ 15 élèves). Le matériel à notre disposition au lycée Europole est

d'environ une dizaine d'ordinateurs sous Windows (donc presque un ordinateur par élève).

La réalisation du TP passe aussi par la conception et l'impression de la fiche de TP ainsi que quelques tâches "d'intendance" : installer le logiciel MobiNet sur les différentes machines disponibles au lycée Europole, installer les comptes des élèves et prévoir les sauvegardes des sessions, demander aux élèves d'amener leur calculatrice et de quoi prendre des notes (en sachant qu'ils ont un compte-rendu à effectuer à l'issue de ce TP).

4.4 Résultat : Le sujet de TP proposé

Le TP se compose de 3 parties : nous ajoutons de nouvelles notions au fur et à mesure de l'avancement des exercices. Nous invitons le lecteur à consulter en annexe C le document de TP distribué en début de séance, afin qu'il puisse suivre le cheminement que nous proposons aux élèves.

Première partie : "Suivre un cap"

Le premier exercice permet aux élèves d'appréhender le but du TP et de prendre en main le logiciel. L'objectif pédagogique est de définir un vecteur à partir des coordonnées de 2 points. La situation exposée est simple : le plan d'eau est calme et le bateau va toujours à la même vitesse. On veut que le bateau passe par toutes les bouées dans l'ordre. L'objectif de l'exercice est alors de programmer les caps successifs que le pilote automatique du bateau devra suivre et de calculer la longueur du trajet, pour chaque morceau de trajectoire. L'idée derrière cet exercice est de faire calculer aux élèves le vecteur représentant chaque segment de trajectoire.

Concrètement, les élèves ont à remplir la case "mouvement" de chaque mobile trajectoire en déterminant deux variables correspondant aux deux composantes du vecteur "trajectoire", et ceci avec des valeurs numériques (les élèves disposeront de calculatrice et de brouillons).

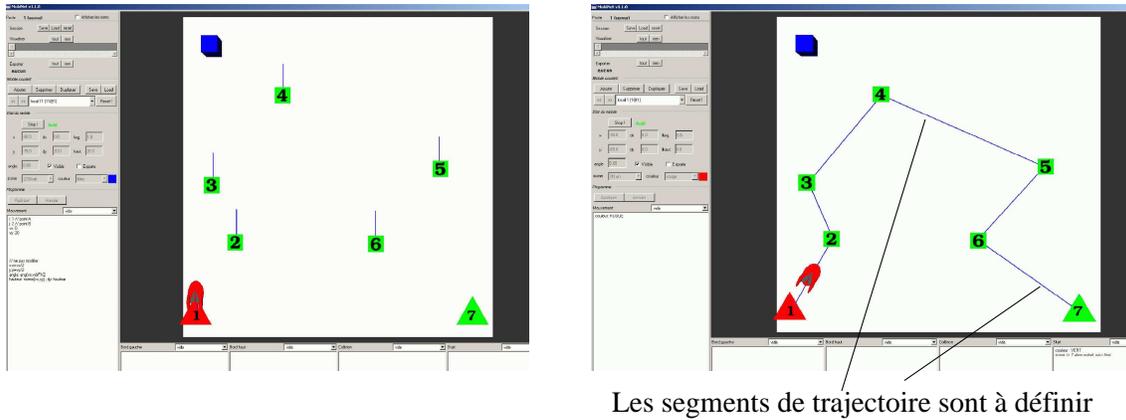


FIG. 3 – Situation initiale et situation finale de la Première partie

Deuxième partie : Introduction de courants

Dans le deuxième exercice, on introduit une donnée supplémentaire : les courants. Ceux-ci sont matérialisés de façon simplifiée par le fait que l'on rajoute une déviation au trajet réel représenté sous forme de vecteurs. L'objectif pédagogique est d'additionner les vecteurs. Les élèves ont donc de nouvelles trajectoires à calculer en additionnant les vecteurs correspondant au trajet sans courant et à la déviation du courant. Le vecteur résultat est appliqué graphiquement à la bouée de départ. Le but ici est donc de trouver la nouvelle succession de caps à prendre pour le bateau afin de compenser la déviation totale due au courant.

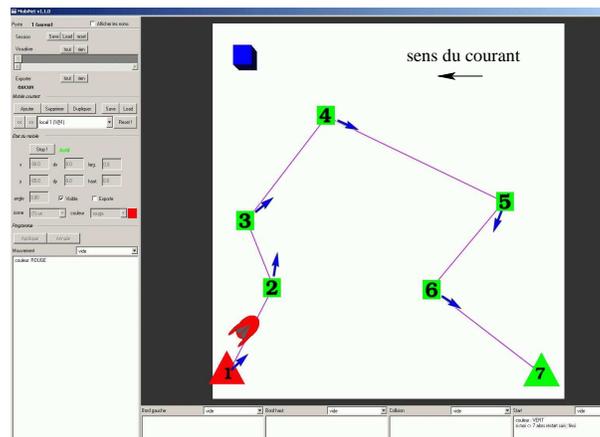


FIG. 4 – Situation finale avec les courants

Troisième partie : Vrais effets de courant

Dans l'exercice précédent, le courant était formalisé comme un vecteur s'additionnant au vecteur représentant la trajectoire, et ceci indépendamment de la distance parcourue par le bateau entre deux bouées. Dans cette exercice, on adopte un modèle plus réaliste en introduisant le fait que la déviation totale appliquée sur le bateau est en fait fonction de la distance parcourue.

Le but pédagogique de l'exercice est de multiplier un vecteur par un scalaire (la distance) et combiner plusieurs vecteurs. Les élèves ont donc d'une part à calculer la distance à parcourir, ce qui correspond à calculer la norme du vecteur trajectoire, et d'autre part à calculer le cap à suivre, celui étant fonction de la déviation totale appliquée au bateau. Cette déviation totale est fonction de la distance et de la "force" du courant par mètre. Pour évaluer la dérive totale entre deux bouées, les élèves ont donc à multiplier la force du courant par la distance puis à calculer le nouveau cap à viser afin que le bateau atteigne les bouées successives malgré la dérive.

4.5 Réalisation en classe

Le TP s'est déroulé au début du mois d'avril dans une classe de seconde du lycée Europole sous la forme de 2 demi-classes d'environ 15 élèves. Le document de TP a été validé par le professeur avant le début des séances.

Tous les moniteurs participant à l'atelier étaient présents. L'un d'entre nous a "joué" le rôle de professeur, un autre s'est occupé des démonstrations sur le vidéo-projecteur en parallèle des explications sur le TP. Le dernier moniteur avait plus un rôle d'assistant (sur le même principe que dans les ateliers des semaines ingénieur INPG). Le TP nécessite un temps de prise en main du logiciel que nous avons souhaité le plus court possible pour laisser l'essentiel du temps au TP lui-même. Pour cela, la prise en main s'est effectuée avec tous les élèves ensemble. Puis, nous avons laissé plus de liberté aux élèves pour la réalisation du TP en lui-même, notamment avec une aide plus personnalisée lors de la recherche et de la validation des résultats des exercices ; cependant, la session MobiNet que nous avons programmée pour ce TP se charge de valider au fur et à mesure les réponses correctes (en changeant la couleur des bouées correctement atteintes) : cela permet aux élèves de valider par eux-mêmes leurs résultats intermédiaires.

4.5.1 Déroulement

Ci-dessous, nous détaillons le déroulement d'une séance :

- Préparation : avant la séance de TP, nous avons installé le logiciel sur les machines, et validé le TP à l'aide de répétitions, ce qui nous a amené à quelques remaniements du sujet ou du discours.
- Séance : Le déroulement type d'une séance est le suivant :
 - Entrée des élèves et installation devant les machines.
 - Distribution des énoncés, et ouverture des sessions de chaque élève.
 - Présentation par le professeur du contexte du TP et des moniteurs qui vont l'animer.
 - Brève présentation au tableau de la situation-problème par le moniteur "professeur" : "comment programmer un pilote automatique de bateau pour qu'il réalise un parcours prédéfini ?".

- Présentation du logiciel MobiNet par le moniteur "démon" à l'aide d'un vidéo-projecteur.
- Explication sur l'objectif du premier exercice et sur les sauvegardes des fichiers utilisés.
- Suivi individuel des élèves pour le premier exercice par tous les moniteurs. Passage individuel aux exercices suivants si le premier exercice a été réalisé correctement.
- Explications des exercices 2 et 3 lorsqu'environ deux tiers des élèves ont fini l'exercice précédent.
- 5 minutes avant la fin : bilan sur les notions de calcul vectoriel vues à travers le TP.
- Distribution de la fiche d'évaluation "premières impressions" et remise aux moniteurs à la fin du TP.
- Fermeture de session, fin de séance.

A noter que nous avons prévu, en cas de grand retard, de reporter l'exercice 3 sur une séance supplémentaire. Il s'est en fait avéré que nous avons eu pour chaque séance 45 minutes effectives de TP et que l'exercice 3 n'a pas été réalisé par beaucoup d'élèves. Pirouz Djoharian a par ailleurs demandé à ses élèves de réaliser un compte-rendu écrit à faire chez eux en parallèle et nous avons donc néanmoins dû expliquer les notions introduites dans l'exercice 3 à l'ensemble de la classe ainsi que quelques pistes pour réaliser cet exercice.

La mise en place des élèves et l'ouverture des sessions a pris beaucoup plus de temps que prévu (oubli de mot de passe par les élèves par exemple). Les explications du logiciel MobiNet et de l'objectif du TP se sont relativement bien déroulées. La manipulation des fichiers .session et notamment le rapatriement depuis le serveur sur le compte local de chaque élève ont été bien compris et nous avons personnellement aidé les quelques élèves déboussolés.

Le temps de réalisation du premier exercice a beaucoup varié en fonction des élèves. La connaissance de l'outil informatique n'a pas été un facteur déterminant entre les élèves. Les différences se sont plutôt faites sentir sur les bases mathématiques sur les vecteurs et notamment les calculs à effectuer pour trouver les trajectoires. Une fois le fonctionnement du logiciel et le calcul à effectuer compris, la réalisation de l'exercice a été assez rapide. Certains élèves ont ainsi fini le TP en 30 minutes alors que d'autres n'ont pas pu terminer le premier exercice (erreur d'écriture des coordonnées des vecteurs, compréhension des notions à utiliser (ou à découvrir) pour effectuer un exercice donné).

4.5.2 Évaluation des résultats

Nous avons identifié trois éléments pour évaluer le TP :

1. Nous avons ainsi évalué la *perception* qu'ont eu les élèves du TP à l'aide d'une fiche d'évaluation (cf. annexe D). Celle-ci nous a indiqué comment les élèves ont ressenti l'utilisation du logiciel, s'ils ont perçu une amélioration de leur compréhension, et s'ils souhaitent recommencer ce genre d'expérience.
2. Nous avons également souhaité mesurer le ressenti du professeur dans la classe duquel nous intervenons, en lui demandant son point de vue sur :
 - l'organisation du projet
 - le sujet

- la gestion du TP
- 3. Enfin, nous nous sommes intéressés aux *résultats* en eux mêmes. Le professeur a en parallèle une autre classe de seconde qui a réalisé un TP plus traditionnel sur le même sujet (calculs algébriques sur les vecteurs). Une évaluation commune a été mise en place (sous forme d'un devoir surveillé et d'un devoir à la maison), l'autre classe de seconde jouant pour nous le rôle de "classe-témoin". Les résultats de cette évaluation nous ont permis d'examiner dans quelle mesure notre travail aura été utile aux élèves et d'essayer d'analyser ces résultats en regard de la conception du TP.

Bien que portant sur un échantillon restreint d'élèves, nos analyses nous ont permis de tirer quelques conclusions quant à l'élaboration d'un sujet de TP sous MobiNet. L'évaluation de Pirouz Djoharian se trouve en annexe E de ce document.

4.6 Travaux futurs et perspectives

En préparant ce TP, nous avons eu d'autres idées d'exercices sur le même sujet mais les élèves n'ont qu'une heure pour réaliser ce TP et nous n'avons donc pas pu exploiter les autres scénari que nous avons imaginés. Ainsi, le TP pourrait se prolonger en imaginant que l'on introduise des courants variables dans le plan d'eau : selon la localisation du bateau dans le plan, la force du courant serait différente et l'élève aurait à calculer différentes trajectoires en prenant en compte ces différences (ceci mobilise une interpolation des vecteurs courants à l'aide de coordonnées barycentriques). En outre, pour ce TP nous demandons aux élèves des valeurs numériques mais il serait possible de leur faire trouver et entrer les formules elles-mêmes dans le programme (ce qui leur demanderait plus de réflexion, mais simplifierait grandement ensuite le traitement par bouée).

Une autre idée serait de rendre le TP réalisable en réseau afin de permettre aux élèves de véritablement réaliser une course de bateaux : chaque élève aurait un bateau dont il aurait calculé la trajectoire et nous pourrions mesurer la distance parcourue. Dans le même ordre d'idée, nous pourrions laisser plus de liberté à l'élève sur la trajectoire de son bateau : il pourrait ainsi "contourner" la bouée au lieu d'aller d'une bouée à l'autre. Du coup, la notion de distance parcourue par chaque bateau prendrait encore plus de valeur.

5 Conclusions et perspectives

5.1 Évolutions du projet

Les séances de MobiNet continueront à faire partie des semaines ingénieur INPG bisanuelles. La diffusion de MobiNet auprès des institutions de l'éducation est à poursuivre auprès des enseignants et étudiants du lycée et de l'enseignement supérieur. Il serait intéressant également d'essayer de le diffuser vers d'autres académies. Les liens avec les autres organismes ou associations de l'éducation (IUFM, IREM, APMEP, SésaMath . . .) sont également à approfondir. La réflexion que nous avons faite cette année est appelée à se poursuivre, afin de réaliser par

exemple de nouvelles séances de TP pour enrichir le fond disponible (banque de TPs), multiplier les expérimentations, et de développer la communication du projet (banque de TPs en ligne, réseau entre les enseignants participants ou intéressés, à l'image de Sésamath, ...). Notre travail et son évaluation seront disponibles sur le site internet de MobiNet. De futurs ateliers de monitorat seront particulièrement bien adaptés pour continuer le travail !

5.2 Apport de l'atelier

Nous espérons avoir montré dans ce rapport en quoi l'atelier MobiNet a contribué à notre formation d'enseignant-chercheur : Comme nous l'avons vu, il nous a aidé à améliorer notre pratique de l'enseignement et à approfondir notre réflexion pédagogique, par les questions qu'il suscite (pourquoi, pour qui, et comment utiliser un tel outil ?), et par les nombreux échanges que nous avons eus.

Il est également fructueux pour notre contribution à la transmission de l'intérêt des matières scientifiques auprès des plus jeunes, contribution certes modeste, mais le goût que nous y avons pris ainsi que la conscience de notre rôle dans ce domaine, forment une base sur laquelle s'appuieront nos futures actions dans ce domaine.

A Documentation du logiciel

A.1 Interface

MobiNet - l'interface

Plateforme de programmation de mobiles en réseau.

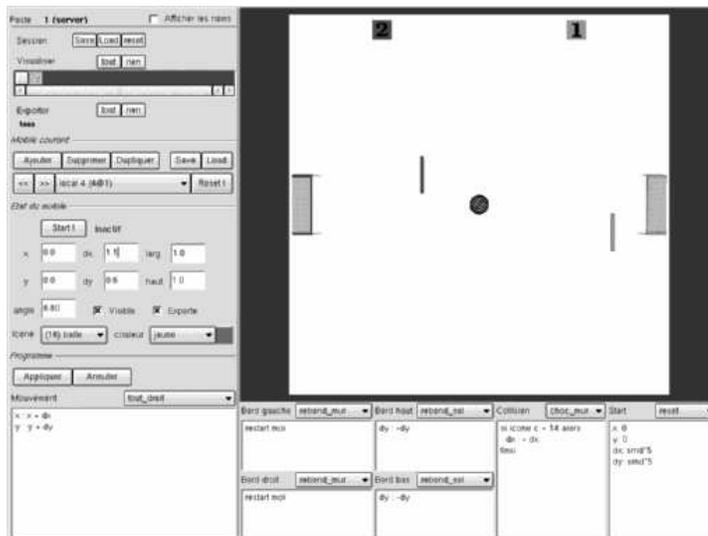
► Retour à la page de départ de Mobinet.

Dans MobiNet, on programme le comportement de différents **mobiles**: ce sont les différents objets visibles (qu'ils bougent ou non!). L'interface permet de régler tous les comportements du mobile courant: aspect, mouvement, règles de collision avec les bords ou les autres mobiles... (on commute ensuite d'un mobile à l'autre).

◆ Aspect général de l'écran (cliquer pour agrandir). Les différentes zones sont détaillées un peu plus loin.

Ici on a programmé 7 mobiles: la balle, les raquettes, les buts, les compteurs.

- La balle suit une trajectoire rectiligne, et rebondit sur les bords et les raquettes.
- Les raquettes suivent la souris verticalement. L'une des raquettes (ainsi que le but et le compteur correspondant) est en fait gérée sur une **autre machine**: deux utilisateurs partagent ici leurs mobiles en réseaux. L'un gère les rouges et la balle, l'autre les bleus.
- Les buts enregistrent les collisions avec la balle, et déclenchent les compteurs adverses.
- Les compteurs s'incrémentent quand les buts le leur signalent.



◆ En cliquant sur 'afficher les noms', on voit les numéros des mobiles (*local n* s'il est sur la machine, *n@m* s'il est géré par la machine *m*), ainsi que le système de coordonnées (de -100 à 100 pour les *x* et les *y*).



◆ Paramètres représentant le mobile courant (*ici, la balle*): position, dimension, icône, couleur, orientation... (de plus on va utiliser *dx,dy* pour indiquer la direction du mouvement.)
NB: en réalité tout est numérique (e.g. l'icône de la balle est le numéro 16), ce qui permettra de faire des opérations.

Etat du mobile

Start ! Inactif

x: 0.0 dx: 1.1 larg.: 1.0

y: 0.0 dy: 0.6 haut.: 1.0

angle: 8.80 Visible Exporte

icone: (16) balle couleur: jaune

◆ Programmation du mouvement: consiste à décrire les changements à faire entre 2 images successives (env. tous les 25ème de seconde), par "tel_paramètre: nouvelle_valeur". Voir le manuel du langage pour le choix des **commandes** utilisables dans les zones de programmation.

On peut faire des petits mouvement successifs ('dynamique'), décrire une fonction du temps ('cinématique'), ou de la souris, voire dépendant de la position des autres mobiles (e.g. poursuite), ou toute combinaison programmée selon l'inspiration.

*(NB: on dispose d'un choix de comportements prédéfinis où piocher si on le souhaite. Ces **presets** sont stockés dans un fichier texte, un animateur peut donc facilement les modifier.)*

Programme

Appliquer Annuler

Mouvement: tout_droit

x: x + dx
y: y + dy

◆ Quoi faire quand on touche un bord.

Bord gauche	rebond_mur ▼	Bord haut	rebond_sol ▼
restart moi		dy : -dy	
Bord droit	rebond_mur ▼	Bord bas	rebond_sol ▼
restart moi		dy : -dy	

◆ Quoi faire quand on collisionne un autre mobile.

Si nécessaire, on peut choisir un comportement différent selon l'identité du collisionneur.
(traduction: *si l'icône du collisionneur est une raquette, alors...*)

Collision	choc_mur ▼
si icône c = 14 alors	
dx : - dx	
fini	

◆ Valeur des paramètres au (re)démarrage.

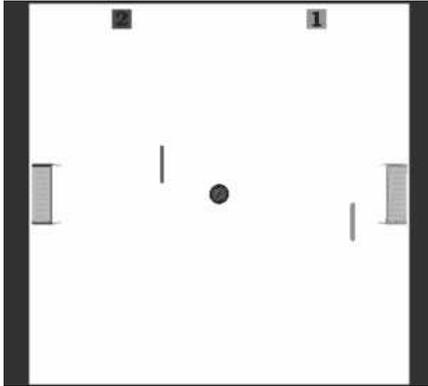
NB: on peut dire de se redémarrer (e.g. en cas de collision) ou de redémarrer un autre mobile (e.g. pour déclencher un compteur, un tir, remettre la balle en jeu...) avec l'instruction `restart num`. C'est une façon d'envoyer des **messages** entre mobiles.
(`srnd` donne un nombre au hasard entre -1 et 1.)

Start	reset ▼
x: 0	
y: 0	
dx: srnd*5	
dy: srnd*5	

◆ Créer un nouveau mobile, choisir sur lequel on travaille, etc.

Mobile courant				
Ajouter	Supprimer	Dupliquer	Save	Load
<<	>>	local 4 (4@1)	▼	Reset !

◆ Affichage du résultat.

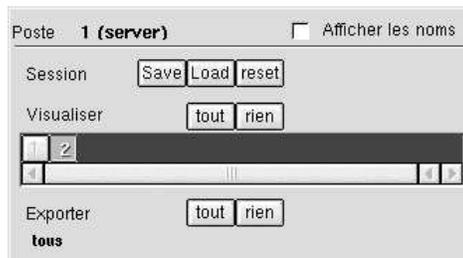


◆ Lire, écrire, interagir avec d'autres machines en réseau...

On choisit d'une part les mobiles que l'on **exporte** (i.e. que l'on rend visible sur le réseau), et d'autre part les machines dont on importe les mobiles.

On dispose donc de plusieurs modes de travail:

- travail indépendant sur chaque poste.
- visionnage de n'importe quel poste depuis le poste maître.
- visionnage superposé de tous les postes (e.g. pour projection murale).
- travail en binômes ou en trinômes (ou autre).
- travail collectif sur N postes (pas forcément sur le réseau local).



◆ Et voilà ! plutôt facile, non ?

A.2 Langage de programmation de MobiNet

MobiNet - langage des mobiles

<http://mobinet.imag.fr>

Variables d'état (ou attributs) d'un mobile :

x	l'abscisse de sa position
y	l'ordonnée de sa position
visible	indique s'il est visible
angle	son orientation (en radians)
icone	son icône
largeur	sa largeur
hauteur	sa hauteur
zoom	(pour régler les deux à la fois)
couleur	sa couleur
rouge	la composante rouge de la couleur
vert	la composante verte de la couleur
bleu	la composante bleu de la couleur
gris	la composante grise de la couleur
dx	(libre ; par ex : vx)
dy	(libre ; par ex : vy)
mem0..4	(libre)

Programme

Un programme de mobile est une suite d'*instructions* permettant de modifier les attributs. Ces instructions peuvent être entrées dans diverses zones (voir le *guide de l'interface*), qui seront exécutées selon les circonstances (tout le temps, en cas de collision aux bords ou avec un autre mobile, ou juste au démarrage). Le programme est pris en compte une fois que l'on a cliqué sur "Appliquer". S'il comporte des erreurs, la zone s'affiche en rouge, et un message apparaît en bas d'écran. (Notez que le mobile n'exécutera le programme que si on l'a mis en marche, en cliquant sur "Start"). Voici par exemple 4 instructions qui modifient différents attributs du mobile courant :

```
x : 50
couleur : ROUGE
dy : cos(t)
y : y + dy
```

Remarquons que la dernière instruction, $y : y + dy$, indique que l'abscisse du mobile courant doit être augmentée de la valeur dy . Dans la 3^{ème} instruction, t représente le temps.

Voici un récapitulatif des instructions que l'on peut utiliser :

Fonctions :

+	-	*	/	carre	racine	log
sin	cos	tan	asn	acs	ang	exp
norme	dist	rnd	srnd	min	max	
abs	ent	frac	sgn	mod	et	ou
=	<	>	<=	>=	<>	

Variables et constantes :

t	dt	PI	clic	cliquee	de clic	contact
moi	suiv	prec	souris	lui	camera	chocx
NOIR	BLANC	ROUGE	VERT	BLEU	lampe (, 2, 3)	chocy
GRIS	CYAN	ORANGE	JAUNE	VIOLET		VIDE
plusproche...	, 2, 3	_vers(, 2, 3)	_gauche	_droite	_haut	_bas
touche...		_espace	_gauche	_droite	_haut	_bas

Commandes :

:	stop	start	restart	comme	met_en	si	alors	sinon	finsi	trace :
---	------	-------	---------	-------	--------	----	-------	-------	-------	---------

```

Exemple :
y : 80*cos(t)
meml : 80*sin(t + PI/3)
si y > 0 alors
  couleur : VERT
  x : meml
sinon
  couleur : ROUGE
  x : 0
finsi

```

Remarquer l'instruction `si` : entre `si` et `alors` se trouve un *test*. Si le test est vérifié, alors les instructions qui suivent jusqu'au `sinon` sont exécutées, sinon ce sont celles entre `sinon` et `finsi` qui le seront. On peut omettre le `sinon` s'il n'y a rien à y mettre. On peut également tout écrire sur une seule ligne (noter les `' ; '`) :

```
si x>0 alors couleur : ROUGE ; sinon couleur : BLEU ; finsi
```

Les principales fonctions :

<code>abs(f)</code>	calcule la valeur absolue de f
<code>ent(f)</code>	calcule la partie entière de f
<code>frac(f)</code>	calcule la partie fractionnaire de f
<code>sgn(f)</code>	calcule le signe de f
<code>a mod b</code>	calcule a modulo b
<code>racine(f)</code>	calcule la racine carrée de $ f $
<code>carre(f)</code>	calcule f^2
<code>sin(f)</code>	calcule le sinus de f (en radians)
<code>cos(f)</code>	calcule le cosinus de f (en radians)
<code>tan(f)</code>	calcule la tangente de f (en radians)
<code>ang(a, b)</code>	calcule l'angle (orientation) du vecteur (a, b)
<code>dist(m, n)</code>	calcule la distance entre les mobiles m et n
<code>norme(a, b)</code>	calcule $\sqrt{a^2 + b^2}$, norme du vecteur (a, b)
<code>rnd</code>	tire un nombre aléatoire entre 0 et 1
<code>srnd</code>	tire un nombre aléatoire entre -1 et 1
<code>min(a, b)</code>	calcule le plus petit nombre entre a et b
<code>max(a, b)</code>	calcule le plus grand nombre entre a et b

Les autres mobiles

Un programme de mobile peut modifier les attributs de son mobile, mais ne peut pas modifier les attributs d'un autre mobile. En revanche, il peut les lire. Supposons que le mobile courant soit le numéro 1. Nous voulons que le mobile 1 ait la même abscisse que le mobile 2. Nous pouvons écrire ce programme pour le mobile 1 :

```
x : x2
```

- Les numéros des mobiles situés avant et après le mobile courant dans la liste sont donnés par `prec` et `suiv`.
- Le numéro du mobile courant est donné par `moi`.
- Le numéro du mobile le plus proche (à l'écran) du mobile courant est donné par `plusproche` (on peut aussi utiliser l'abréviation `pp`).
- En cas de collision, le numéro du mobile collisionné est donné par `lui`.

Les différents types de mobiles :

<code>m</code>	<code>prec</code>	<code>moi</code>	<code>souris</code>	<code>VIDE</code>	<code>plusproche</code>
<code>m@n</code>	<code>suiv</code>	<code>lui</code>	<code>camera</code>	<code>lampe</code>	(variantes de <code>pp</code>)

La souris

C'est aussi un mobile. On ne peut cependant pas modifier ses attributs. Mais on peut facilement programmer un mobile pour qu'il suive la souris :

```

                                s : souris
y : xsouris                      ou encore  x : xs
y : ysouris                       y : ys

```

On peut également connaître sa direction et sa vitesse avec `dxsouris` et `dysouris`. D'autre part, on peut savoir si le bouton de la souris a été cliqué par `si clic alors`, s'il a été relâché par `si declic alors`, s'il est maintenu appuyé par `si cliquee alors`.

La caméra

C'est aussi un mobile ! On peut changer sa position (ex : `xcamera : xsouris`), son zoom (`zoom camera : 2`), son angle (pour tourner la vue)...

En outre, on peut désactiver l'effacement d'écran par `visible camera : 1` pour que les objets laissent une trace visible de leur trajectoire, ce qui permet de dessiner des courbes (`raccourci : trace : 1`).

Remarque : la caméra est remise à zéro quand on clique sur "reset".

Les lampes

De même, on peut changer la couleur de l'éclairage (ex : `couleur lampe : ROUGE` ou `gris lampe : (1+sin(t))/2`) et sa direction (ex : `x lampe : x1 ; y lampe : y1 ; hauteur lampe : 10`). Initialement la hauteur de la lampe est à l'infini. On dispose en fait de deux lampes supplémentaires `lampe2` et `lampe3`, initialement éteintes (i.e. couleur noire).

Lire les attributs d'un mobile sur une machine distante

Nous voulons par exemple que notre mobile prenne la couleur du mobile 3 de la machine 15. Ceci s'écrit : `couleur : couleur3@15`

`m@n` désigne le mobile numéro `m` de la machine (ou "poste") numéro `n` (un peu comme pour une adresse mail). NB : pour que ce mobile `m@n` soit accessible, il faut que la machine `n` l'ait *exportée* (en cliquant sur "Exporter"), et que vous *importiez* ses mobiles en cliquant sur son numéro `n` dans le bandeau "Visualiser". Il est alors également pris en compte par les commandes comme `plusproche`.

Variables locales

On peut utiliser des mots quelconques pour stocker une valeur (variable intermédiaire) :

```

objet : souris                // objet devient un raccourci pour souris
d : dist(moi,objet)          // d contient la distance à la souris
x : x + 3*(x-xobjet)/d      // avance de 3 dans la direction de la souris
y : y + 3*(y-yobjet)/d

```

Remarques sur les attributs

Icônes :

Noter que les icônes 0 à 9 représentent les chiffres de 0 à 9, ce qui facilite la construction de compteurs.

Largeur, hauteur :

Attention, il s'agit d'un *facteur de zoom* par rapport à la taille initiale de l'icône. La plupart des icônes ont une taille initiale d'environ 10.

Angle :

Ici aussi, il s'agit de l'angle de rotation par rapport à l'orientation initiale, qui peut être horizontale ou verticale selon les icônes. NB : tous les angles dans MobiNet sont en radians.

Les couleurs :

`couleur` permet juste d'indiquer un nom de couleur. Pour régler ou connaître précisément la teinte en RGB, on dispose des attributs dérivés `rouge`, `vert`, `bleu` (ainsi que `gris`).

Les collisions

On les traite dans la zone ‘Collision’. On dispose alors de divers variable supplémentaires : `lui` indique le mobile avec qui on est entré en collision. On peut ainsi tester cette variable s’il faut agir différemment selon le mobile. Exemple : si `lui=2`, ou si `lui=2@3`, ou si `couleur lui = BLEU`. À noter qu’il est souvent plus commode de tester l’icône ou la couleur du mobile, qui caractérisent généralement sa catégorie, plutôt que son numéro.

Pour une utilisation avancée, on dispose en outre de la variable `contact` qui indique si c’est la première fois que l’on traite cette collision, et du *vecteur de collision* `chocx, chocy` qui indique sous quel angle les mobiles se sont touchés. Voir par exemple comment le **preset** ‘`choc`’ en fait usage.

Remarque : il y a toujours un risque que l’on soit à nouveau en état de collision au pas de temps suivant. Tester `contact` permet d’éviter de traiter 2 fois la collision (on risquerait alors de re-bondir dans la mauvaise direction). Une autre technique, également utilisable pour les collisions au bords, consiste à *décollisonner* avant toute chose, en faisant par exemple `x : x-dx ; y : y-dy`. Sachez cependant que les collisions sont une tâche délicate en informatique, et qu’il est difficile d’obtenir un comportement parfait (notamment si on bouge vite).

Interaction entre mobiles

Agir sur un autre mobile :

On ne peut modifier les attributs d’un autre mobile (on peut juste le déplacer avec la commande `met_en(m, x, y)`). Par contre on peut agir sur son état avec commandes `stop m` pour l’arrêter, `start m` pour le démarrer, ou `restart m` pour le remettre en route même s’il était déjà en marche.

Quand ce mobile `m` redémarre, il commence par exécuter son programme ‘Start’ (si vous en avez entré un dans cette zone). C’est utile par exemple pour remettre en jeu une balle qui est sortie, mais cela permet également d’envoyer des signaux à des mobiles : un compteur pourra par exemple être incrémenté à distance par `restart`, en mettant `icone : icone+1` dans sa zone ‘Start’.

Imiter le comportement d’un autre mobile :

Quand plusieurs mobiles partagent le même comportement, on peut dire, pour chaque champs souhaité, de se ‘reporter’ au programme du champs correspondant du module de référence, à l’aide de la commande `comme m` (`m` étant un mobile tournant sur le même poste).

Poursuite, fuite :

`plusproche` permet de ‘voir’ quel est le mobile le plus proche, ce qui permet par exemple de s’en approcher ou de s’en éloigner. De nombreuses variantes de cette fonction permettent d’affiner ce comportement : `plusproche2` et `3` permettent de ‘voir’ qui arrive en second ou en troisième, `plusproche_gauche` permet de ne regarder que dans le cadran Ouest du champs de vision, etc. `plusproche_vers(vx, vy, ang)` ne regarde que dans la direction (vx, vy) avec un champ de vision d’ouverture `ang`. (Abréviations : `pp`, `pp2`, `pp3`, `ppg`, `ppd`, `pph`, `ppb`, `ppv`, `ppv2`, `ppv3`). ATTENTION : dans ces conditions, il peut n’y avoir aucun mobile en vue. Il faut donc prendre garde à tester si la commande a bien trouvé un mobile :

```
m : plusproche_vers(dx, dy, Pi/2)
si m <> VIDE alors
  (fuir ou attaquer)
finsi
```

Démarrage de l’application

Si l’on n’utilise pas le réseau (usage individuel, ou usage non collaboratif et sans poste maître en salle de TP), lancer simplement l’application.

Dans les autres cas, lancer simplement l’application sur la machine maître, puis lancer **ensuite** `mobile-client nom_machine_maître` sur les autres. (Pour une machine distante, le nom est l’adresse internet complète). NB : dans le cas de TP avec des élèves, on aura sans doute intérêt à ajouter (en premier) l’option `-nosave` pour interdire la sauvegarde.

B Un exemple de progression en classe de Seconde

1 PROGRESSION

1

1 Progression

	Séquence	Contenu	Nb. Semaines	Dates	DS
1	Démarche et formalisme mathématique. Géométrie plane	Vrai/Faux, Exemple/Contre-exemple, Géométrie plane : transformation-triangles isométriques. Algèbre : développement, factorisation, Équations-Inéquations	5	08/09/04 au 09/10/04	16/10/04
2	Statistiques, Vocabulaire des Fonctions	Décrire, modéliser, Échantillonner, Intervalles, Ordre de grandeur, notation scientifique, Fonctions, représentation graphique, résolution graphique	4	11/10/04 au 13/11/04	20/11/04
3	Ordre, Variation des fonctions	Ordre sur \mathbb{R} , Variation des fonctions, Inéquations, Equiv. logique, Intervalles, Vocabulaire ensembliste	5	15/11/04 au 18/12/04	08/01/05
4	Géométrie analytique, droites et fonctions affines	Vecteurs, colinéarité, Équation de droite, Systèmes linéaires à 2 inconnues, fonctions affines	5	03/01/05 au 05/02/05	12/02/05
5	Fonctions de références, Espace, trigonométrie	Variations des fonctions carré et inverse, sinus et cosinus, factorisation et (in-)équations de degré > 1	5	07/02/05 au 26/03/05	02/04/05
6	Triangles semblables, Nature des nombres, Propriétés de la moyenne	Caractérisation de la similitude, Rapport des aires, Primalité, Décomposition en facteurs premiers, les Rationnels et les nombres décimaux, linéarité de la moyenne	5	28/03/05 au 14/05/05	21/05/05

C Notre sujet de TP "Vecteurs et Courses de bateaux"

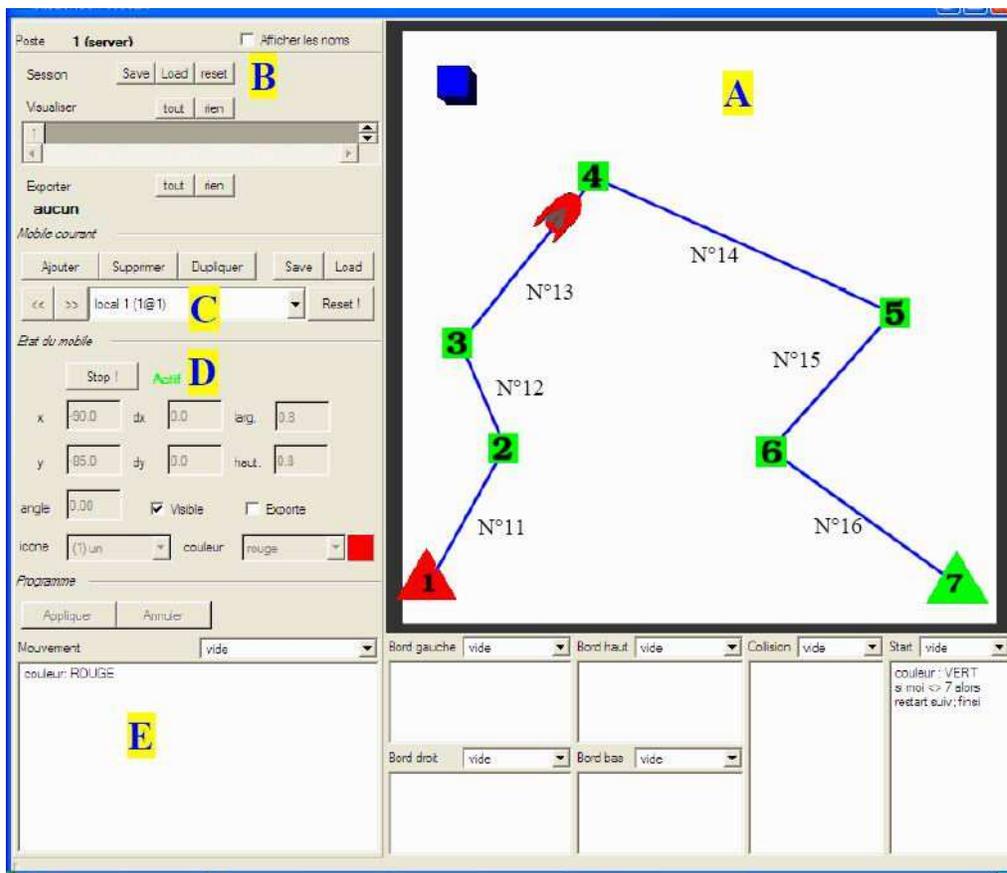
Module 15

11/04 et 14/03 2005

COURSE DE BATEAUX

Introduction

Le thème de ce TP est une course de bateaux un peu spéciale, avec des bateaux à moteur munis d'un pilote automatique. Nous utiliserons les *vecteurs* pour programmer la trajectoire des bateaux de la bouée de départ à la bouée d'arrivée. Nous commencerons par une situation simple (mer calme), qui ira en se compliquant (courants marins). La simulation de la course et la programmation du pilote automatique se feront à l'aide du logiciel **MobiNet**.



Le logiciel MobiNet

Il est déjà installé sur votre compte : loguez-vous, et cliquez sur XXX (à remplacer). MobiNet simule des objets fixes ou animés que l'on appelle des **mobiles**. L'apparence et le comportement de

ces mobiles peuvent se programmer, mais on peut aussi charger une *session* avec des mobiles tout prêts (préprogrammés). Pour les exercices qui suivent, nous partirons de mobiles déjà programmés, mais il restera à compléter les instructions de trajectoire.

L'écran de Mobinet

Pour ce TP nous aurons juste à compléter des valeurs numériques dans des programmes tout prêts. Ci-dessous, une explication des différentes zones de l'écran de **MobiNet** :

- L'affichage graphique de la simulation se fait dans la grande zone en haut à droite **A**.
 - A gauche, en haut, on peut charger une session : bouton load **B**.
 - Chargez le 1^{er} exercice (`trajectoire_init.session`) pour voir un exemple.
 - En dessous, on sélectionne le mobile à modifier : boutons `<<`, `>>`, `localN` **C**. Toutes les autres zones servent à programmer ce mobile là ; pensez-y !
 - En bas, la grande zone mouvement **E** permet de programmer le mouvement des objets en changeant leurs attributs (comme la position) à chaque instant.
- Nous n'aurons pas besoin des zones du programme situées sous l'image.

Les mobiles utilisés dans ce TP

Il y a trois types de mobiles utilisés dans ce TP :

Les bouées : Mobiles n° 1 à 7. Vous aurez besoin de les sélectionner **C** pour voir leurs coordonnées dans la zone **D**.

Les caps à suivre : Mobiles n° 11 à 16. *Ce sont les seuls mobiles qu'il faudra modifier*, en complétant les coordonnées v_x et v_y du **cap** à suivre (*ne pas modifier le reste !*).

Astuce : le **cap** à suivre en partant de la bouée n° i a pour numéro $i + 10$

Marche/Arrêt : Cliquez sur le gros carré (haut et gauche de la zone **A**) permet de démarrer et de stopper le bateau.

Remarques

- La fenêtre d'affichage **A** est muni d'un repère orthonormé qui n'est pas représenté sur l'écran. On peut par contre lire directement les coordonnées des bouées dans la zone **D**. On conviendra qu'une unité représente 1 m.
- **MobiNet** est un logiciel libre ; vous pouvez le récupérer gratuitement sur <http://www-evasion.imag.fr/mobinet/>

Travail à faire

Il y a 3 exercices à faire, chacun représentant un état différent de la mer : d'abord calme, ensuite avec des courants marins. Il est conseillé de les faire dans l'ordre, car les calculs faits dans l'exercice 1 serviront dans les suivants⁴.

Exercice 1 : mer calme

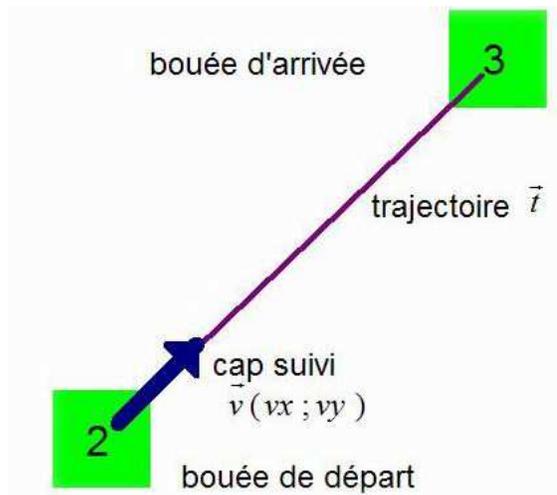
Ouvrir⁵ le fichier `trajectoire_init.session`.

On commence par une situation simple : la mer est calme, le bateau se déplace toujours à la même vitesse. On veut que le bateau passe par toutes les bouées, dans l'ordre. La seule chose qui reste à faire est de programmer le **cap** que le pilote automatique devra suivre et la longueur du trajet, pour chaque morceau de trajectoire. Cela se fait en calculant pour chaque segment de trajectoire, le vecteur **cap** \vec{v} à suivre (mobiles 11 à 16).

Pour cela, sélectionner l'un après l'autre les mobiles **cap** pour compléter le programme de mouvement. Par exemple, pour le mobile 12 (`local 12 (12@1)`), contrôlant la partie de la trajectoire entre les bouées 2 et 3, dans la zone d'édition `E`, il faut calculer et compléter les coordonnées $(v_x; v_y)$ du vecteur **cap** :

<code>i : 1 // A</code>	n° de la bouée de départ. <i>Ne pas modifier!</i>
<code>j : 2 // B</code>	n° de la bouée d'arrivée. <i>Ne pas modifier!</i>
<code>vx : ...</code>	l'abscisse du vecteur à suivre. <i>À calculer</i>
<code>vy : ...</code>	l'ordonnée du vecteur à suivre. <i>À calculer</i>

N.B. Après chaque modification d'un mobile, cliquez sur `Appliquer`, pour que la modification soit prise en compte.



⁴Voir aussi en dernière page des consignes pour la rédaction du compte-rendu

⁵Le bouton `load`, zone `B`

Utiliser le tableau ci-dessous pour inscrire les coordonnées des différentes bouées. Elles seront utilisées aussi dans les exercices suivants.

n° de bouée	1	2	3	4	5	6	7
Abscisse							
Ordonnée							

Lorsque vous avez fini de calculer toutes les trajectoires, vous pouvez activer la simulation du bateau pour la tester. Pour cela, cliquez sur le bouton Marche/Arrêt (le carré en haut à gauche de l'image). Si vous avez bien calculé les trajectoires, les bouées deviennent rouges au fur et à mesure. Sinon, modifiez la programmation, puis cliquez à nouveau sur le carré pour recommencer.

Exercice 2 : Mer agitée

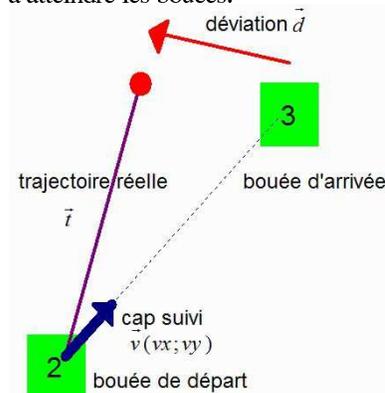
Ouvrir le fichier `trajectoire_courant_init.session`.

En réalité, en mer il y a des courants qui vont dévier votre bateau de sa trajectoire : si on ajuste le **cap** visé dans la direction d'une bouée, on est dévié un peu à côté (voir Figure 2). En présence de courant marin, la direction de la trajectoire effectuée est donc le **cap** visé \vec{v} plus la déviation \vec{d} (les deux vecteurs s'additionnent).

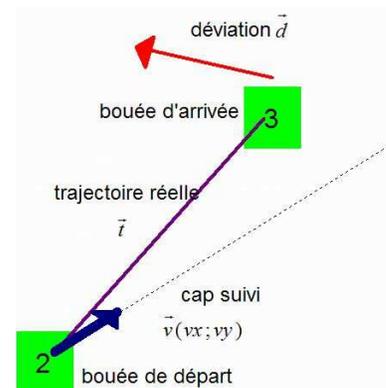
$$\vec{v} + \vec{d} = \vec{t}$$

Dans cet exercice, le vecteur déviation est le vecteur $\vec{d}(-20; 5)$.

Vous devez donc reprogrammer le pilote automatique en *compensant* la déviation, pour continuer à atteindre les bouées.



Si le cap est en direction de la bouée



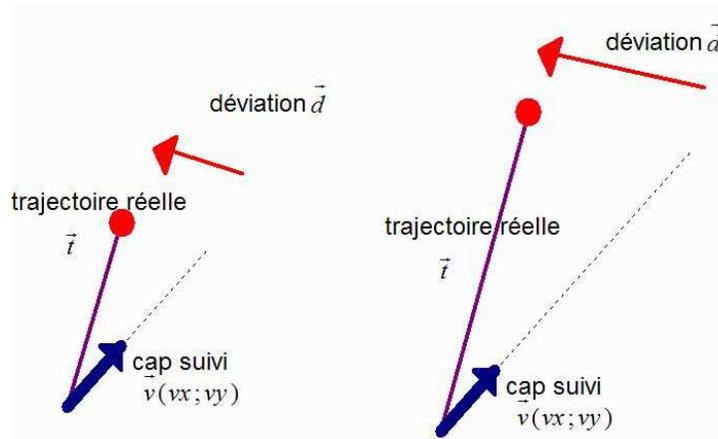
Pour compenser la déviation

Exercice 3 : les vrais courants !

Ouvrir le fichier `trajectoire_VraiCourant.session`.

En réalité, la déviation totale subie par le bateau (calculée à la bouée d'arrivée) n'est pas toujours la même, elle est **proportionnelle** à la distance parcourue.

Dans cet exercice, le vecteur déviation aura pour coordonnées $(-20; 5)$ pour une *distance parcourue* de 100 m. Donc, par exemple, si la distance parcourue est 200 m, la déviation sera le vecteur \vec{d} de coordonnées $(-40; 10)$.



Vous devez donc recalculer le cap à viser en prenant en compte cette déviation proportionnelle à la distance, et tester votre trajectoire.

Pour la rédaction du compte-rendu

Le compte-rendu du TP est à rendre pour le **jeudi 5 mai 2005**. Il doit comporter :

1. Une présentation générale du sujet :
 - Quel est le problème à résoudre ?
 - Quelles sont les notions mathématiques utilisées ?
 - Quelles sont les théorèmes ou formules utilisées, pour chaque exercice ?
2. Pour chaque exercice, expliquer la démarche et détailler un calcul.

D Fiche d'évaluation "premières impressions"

Vecteurs et course de bateaux

Fiche d'évaluation

A l'issue du TP, vous remplirez cette fiche, anonyme, qui est destinée à nous faire améliorer la qualité des séances proposées.

Pour chaque question, vous donnerez votre réponse sous la forme d'une note (de 1 à 5), la note 1 correspondant à une mauvaise appréciation, la note 3 à une appréciation moyenne, et la note 5 à une très bonne appréciation, en cochant la case correspondante.

	<i>Questions</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>
a	Le TP avec MobiNet vous a t'il intéressé?					
b	Le TP avec MobiNet vous a t'il paru utile?					
c	Avant de commencer le TP, pensiez-vous alors que ce serait facile?					
d	Avez vous trouvé MobiNet facile?					
e	Avez vous trouvé le TP facile?					
f	Ce TP vous a t'il aidé à mieux comprendre les sommes de vecteurs?					
g	Ce TP vous a t'il aidé à mieux comprendre les produits de vecteurs par un nombre?					
h	Souhaiteriez-vous faire un autre TP avec MobiNet sur une autre partie du programme?					
i	Ce TP change-t-il votre point de vue sur les maths et la physique?					
j	Utiliser MobiNet au lycée vous paraît-il utile?					

Suggestions:

E Evaluation du TP par Pirouz Djoharian

Rapport sur le TP Mobinet Classe de seconde 2. Lycée Europole

Contenu du TP

Le TP a porté sur le repérage et le calcul vectoriel. Le cours correspondant avait été vu juste la séance précédente. Aucun exercice d'application, si ce n'est que des exemples élémentaires n'avait été fait en classe. On ne pouvait raisonnablement espérer que les élèves se présentent à la séance en ayant spontanément appris le cours. Mais dans l'ensemble, les élèves se souvenaient quand même des résultats importants intervenant dans ce TP :

- Calcul des coordonnées d'un vecteur \overline{AB} à partir des coordonnées des points A et B
- Calcul des coordonnées d'une combinaison linéaire de vecteurs
- Calcul de la norme d'un vecteur

Pendant la séance, plusieurs élèves ont malgré tout demandé à ce que je leur rappelle quelques formules (addition, produit par un scalaire...). Notamment 6 élèves qui, ayant été absents une dizaine de jours en raison d'un séjour linguistique, n'avaient pas suivi le cours sur les vecteurs. Dans le premier groupe, uniquement deux ou trois élèves ont réussi à avancer significativement dans l'exercice 3, alors dans le second groupe, près d'un tiers des élèves avaient largement commencé l'exercice 3 avant la fin de l'heure.

En conclusion, le contenu et le niveau de difficulté étaient effectivement adaptés et le choix des exercices en adéquation avec le cours. Le TP était éventuellement un peu trop long, compte tenu du temps nécessaire pour la mise en route (installation des élèves, connexion à leur compte, présentation de la séance et du logiciel MobiNet).

Déroulement des séances

L'encadrement était constitué de l'enseignant et de 3 moniteurs CIES¹. Les élèves ont été invités à s'installer individuellement, chacun derrière une machine. Malgré les craintes de l'administrateur système, le lancement du logiciel et le chargement des fichiers d'exercice n'ont pas posé trop de problèmes. Après une installation parfois assez longue (mot de passe oublié ou inopérant, dysfonctionnement de certaines machines, ...), les séances ont commencé par une courte introduction, présentant le programme et les intervenants. Ensuite deux intervenants se sont chargés d'expliquer le problème proposé et le travail à faire. Cette phase introductive qui comprenait également une présentation succincte de MobiNet a pris de 15 à 20 minutes, selon la séance. À la seconde séance la présentation a été plus rapide et les élèves ont commencé à travailler plus rapidement sur les exercices.

Nous avons circulé dans la classe pour observer le travail des élèves et répondre à leurs sollicitations. Lors de la 1^{re} séance, après une vingtaine de minutes, nous avons demandé aux élèves d'interrompre leurs calculs pour passer à l'exercice suivant. Dans la 2^e séance les élèves ont avancé à des rythmes différents (avec une plus grande disparité que le 1^{er} groupe). Entre chaque exercice aucune mise en commun n'a été proposée. Uniquement à la fin de la séance, une synthèse concernant le 3^e exercice a été rapidement faite.

¹Centre d'Initiation à l'Enseignement Supérieur

Comportement, accueil et difficultés des élèves

Motivation Les élèves se sont montrés en général très motivés. Contrairement aux autres séances, où un certain nombre d'élèves ne faisait qu'attendre la fin de la séance, tous les élèves ont travaillé, à des degrés d'implication et de concentration différents, certes ! Il est intéressant de noter que les élèves généralement démissionnaires (cf. par exemple **Élève J.**²) se sont intéressés en s'appliquant de façon active et autonome (même parfois plus que les élèves considérés comme bons).

Au démarrage de la séance, un grand nombre d'élèves avait déjà ouvert l'application et même chargé des fichiers de démonstrations, pendant que le sujet du TP leur était exposé. Ce qui a été bien sûr au détriment de la concentration nécessaire pour suivre les consignes, mais témoigne par contre de l'intérêt et de la curiosité des élèves.

Interactions entre les élèves Assez rapidement plusieurs élèves se sont spontanément mis à travailler ensemble, avec un ou même parfois plusieurs voisins. Cette collaboration allait de simples explications demandées au voisin jusqu'à un réel travail de groupe (partage du travail : un qui calcule les coordonnées, l'autre rentre les valeurs dans le programme, ...). Des problèmes matériels (mot de passe inopérant, répertoire inaccessible, ...) avaient poussé dès le départ certains élèves à partager la même machine à deux.

Prise en main du logiciel La prise en main du logiciel a été assez rapide. Seul un petit nombre d'élèves a eu des difficultés à lire les coordonnées des bouées, sans doute parce que les zones correspondantes étaient grisées.

Le bouton Marche/Arrêt pour lancer le bateau n'a pas été perçu par tous les élèves et certains d'autres n'ont pas remarqué que l'on pouvait tester les trajectoires calculées étape par étape.

Aspects mathématiques Certains se contentaient d'apprécier *visuellement* l'exactitude du vecteur cap. Deux élèves (réputés bons, dont un revenant d'Allemagne) essayaient de trouver le vecteur cap par tâtonnement. D'une part parce qu'ils ne connaissaient pas le calcul des coordonnées d'un vecteur, mais aussi parce que cette approche leur paraissait plus jubilatoire³ ! De manière générale la distinction entre coordonnées du vecteur cap et les coordonnées des bouées n'a pas été facile pour tous les élèves (voir plus loin).

Les comptes rendus de TP⁴

Les élèves devaient rédiger un compte rendu, après la séance, chez eux. Les consignes de rédaction suivantes ont été données : expliquer le problème, expliquer les notions et formules mathématiques utilisées et présenter quelques exemples de calcul. Comme les deux séances de TP ont eu lieu la dernière semaine avant les vacances de Pâques, les comptes rendus ont été faits/rendus la semaine de la rentrée, c'est à dire près de 3 semaines après le TP.

Remarques générales sur le travail des élèves Une trentaine d'élèves sur 33 a rédigé un compte rendu. Ceux-ci étaient généralement faits assez sérieusement et comportaient de 2 à 6 pages. Dans la plupart des cas les élèves avaient détaillé les calculs pour chaque morceau du trajectoire, alors qu'on leur avait demandé de détailler uniquement un exemple de calcul pour chaque exercice.

L'exercice 3 qui n'avait pu être fait que par un petit nombre d'élèves pendant le TP, avait par contre été traité par une grande majorité. Les consignes de rédaction de la problématique avaient été suivies très inégalement. Concernant les notions et résultats du cours, tous les élèves ont mentionné le calcul des coordonnées d'un vecteur, le calcul de la distance et également la proportionnalité. Peu d'élèves ont pensé à mentionner le calcul des coordonnées d'une somme de vecteurs, ni celui du produit par un scalaire.

²moyenne annuelle 6,4 et 30° de la classe

³Bien sûr cette stratégie ne marche pas puisque la vérification faite par la machine, elle, est précise

⁴Une douzaine de copies scannées sont disponibles à l'adresse <http://membres.lycos.fr/math2nd/mobinet/index.html>

Beaucoup d'élèves n'ont pas fait (explicitement) le rapprochement entre $|\vec{AB}|$ et la distance AB . Certains ont mentionné le théorème de Pythagore pour le calcul de la distance AB en fonction des coordonnées. Le théorème de Pythagore avait été effectivement utilisé en cours pour la justification du calcul de la norme d'un vecteur.

La proportionnalité entre le vecteur déviation \vec{d} et la distance AB a été quelques fois présentée sous forme vectorielle (cf. par exemple **Élève K.**)

$$\frac{(x_{\vec{d}}; y_{\vec{d}})}{AB} = \frac{(-20; 5)}{100}$$

Cette proportionnalité "vectorielle" a été utilisée généralement sans problème, soit avec un tableau de proportionnalité tel que ci-dessus, soit avec un coefficient

$$q = \frac{AB}{100}$$

Ce qui est étonnant est que certains ont *divisé* le vecteur $(-20; 5)$ par $q = \frac{100}{AB}$ pour trouver le vecteur déviation !

Erreurs fréquentes et difficultés des élèves

La principale difficulté des élèves a été de comprendre et de traduire mathématiquement le traitement des courants marins et leurs compensations pour le calcul du *cap corrigé*. Ils se sont basés sur la formule présentée dans le sujet

$$\vec{\tau} = \vec{v} + \vec{d}$$

sans vraiment comprendre ce que signifie chacun des vecteurs. Beaucoup ont confondu le cap à suivre \vec{v} avec le vecteur trajectoire, ce qui les a conduits à rajouter la déviation au vecteur cap de l'exercice précédent. Cependant, avec le retour du logiciel, certains se sont-ils rendus compte que si on ajoute le vecteur \vec{d} calculé au vecteur cap de l'exercice 1, le bateau ne suit pas la direction souhaitée. Alors, pour expliquer pourquoi le vecteur déviation \vec{d} doit être soustrait au vecteur trajectoire, ils ont inventé diverses stratagèmes, allant jusqu'à l'altération des formules du cours. Par exemple, beaucoup d'élèves ont été obligés d'admettre la règle de calcul surprenante (cf. **Élève K.**) :

$$\begin{cases} x_{\vec{u}+\vec{v}} = x_{\vec{u}} - x_{\vec{v}} \\ y_{\vec{u}+\vec{v}} = y_{\vec{u}} - y_{\vec{v}} \end{cases}$$

Une explication possible de cette confusion entre la trajectoire $\vec{\tau}$ et le vecteur cap \vec{v} est que l'on demandait aux élèves de *programmer* la trajectoire du bateau, alors qu'il devait *calculer* en effet le cap visé \vec{v} (la trajectoire est la même que dans l'exercice 1 !). Le risque de confusion était d'autant plus grand que dans l'exercice 1 où $\vec{d} = \vec{0}$, les deux vecteurs \vec{v} et $\vec{\tau}$ sont identiques.

La figure ?? représente quelques extraits de copies illustrant cette difficulté. On peut noter au passage que, malgré que les élèves C. et F. ont travaillé sur la même machine, ils n'ont pas rédigé le compte rendu ensemble. Ce qui signifie qu'ils ont reconstitué l'exercice sur papier chez eux (du moins en ce qui concerne l'élève F.).

Élève F. le vecteur déviation est ajouté au vecteur cap. Il applique la formule donnée dans le sujet, le calcul est correct mais il ne résout pas le problème. Évidemment MobiNet permet de déceler et d'éviter cette erreur, mais cet élève (contrairement à son coéquipier l'élève C.) n'ayant sans doute pas pris suffisamment de note pendant la séance, a mal reconstruit la solution sur papier.

Élève K. Confusion entre les vecteurs \vec{v} et $\vec{\tau}$. Le calcul est faux mais les valeurs calculées permettent de conduire le bateau au but. L'élève, guidée par le retour perceptif du logiciel a modifié la règle de calcul des coordonnées d'une somme de vecteur pour retrouver un résultat plausible.

Élève C. Exemple d'un calcul correct accompagné d'une explication floue. Pour expliquer le calcul du cap visé, l'élève donne dans la même phrase deux interprétations différentes du mot *cap visé*.

EXERCICE 2 :
 Calcul du vecteur \vec{x} , somme des vecteurs \vec{v} (caps suivis) et \vec{d} (déviations), trajectoire des bateaux à vitesse constante et sur mer agitée (courants (\vec{d}) , mais pas encore réels):
 - On additionne les vecteurs \vec{v} et $\vec{d}(-20;5)$ grâce au théorème $\vec{x} = \vec{v} + \vec{d} = (x_p + x_d; y_p + y_d)$
 $\vec{v}_1 + \vec{d} = \vec{x}_1(25; 45) + (-20; 5) \Leftrightarrow \vec{x}_1(5; 50)$

Élève F.

exercice 2 : mer agitée
 on cherche à calculer les coordonnées du vecteur \vec{F} (à trajectoire réelle) que le bateau doit suivre pour compenser la déviation $\vec{d}(-20;5)$.
 $\vec{F} = \vec{v} + \vec{d}$ d'où $\vec{F}(x_p + x_d; y_p + y_d)$
 donc pour le segment 1, on a:
 $\vec{F}(x_p + x_d; y_p + y_d)$

Élève K.

exercice 2
 Puisque le trajet réel est le cap visé plus la déviation, alors le cap à suivre correspond au cap visé moins la déviation.
 $\vec{d} = (-20; 5)$
 $\vec{v}_1 - \vec{d} = (25; 45) - (-20; 5) = (45; 40)$

Élève C.

On est dans une mer agitée donc il y a une déviation.
 On nous donne la relation suivante: $\vec{v} + \vec{d} = \vec{F}$ où $\vec{d} = (-20; 5)$.
 On essaye avec \vec{AB} (où A=1 et B=2). $\vec{AB} + \vec{d} = \vec{F}$.
 Pour obtenir la trajectoire réelle \vec{F} , il faut "enlever" le courant de soustraire au cap suivi \vec{v} qui sera par conséquent "plus grand", c'est à dire qu'il dérivera dans le sens inverse de la déviation (s'il pouvait le faire); et lorsque le courant le poussera, il fera donc la trajectoire réelle \vec{F} .
 (on utilisera toutes les données des autres tableaux pour les calculs).

Élève E.

FIG. 1 – Exercice 2 : extraits de copies montrant les difficultés des élèves dans l'exercice 2

Élève E. Encore une mauvaise interprétation du cap visé \vec{v} , confondu dans ce cas avec le vecteur \vec{AB} défini par les bouées consécutives. L'élève, sans doute gênée par cette confusion, essaye de justifier "comme elle peut" pourquoi il faut soustraire le vecteur \vec{d} au vecteur cap précédemment calculé.

Peu d'élèves ont pensé à transposer le vecteur \vec{d} dans le membre gauche, mais même parmi ceux qui l'ont fait, les explications qui suivent montrent que la signification de cette transposition n'était pas vraiment assimilée. La figure ?? montre par exemple deux extraits de copies d'élèves où elles ont fait les calculs correctement, mais elles interprètent néanmoins le résultat du calcul comme la trajectoire du bateau entre deux bouées consécutives.

On a : $\vec{F} = \vec{v} + \vec{d}$
 donc $\vec{v} - \vec{d} = \vec{v}$
 En calculant $\vec{v}_{b1b2} = (25; 45) - (-20; 5) = (45; 40) = \vec{v}_{b1b2}$
 Soit $(20; -20); 45; 5) = \vec{v}_{b1b2}$
 $(45; 40) = \vec{v}_{b1b2}$
 la trajectoire du bateau entre b_1 et b_2 est de : $(45; 40)$.

Élève F.

On veut donc trouver la trajectoire:
 $\vec{v} + \vec{d} = \vec{F}$
 $\vec{v} - \vec{d} = \vec{v}$
 On calcule:
 $\vec{v}_{b1b2} = (25; 45) - (-20; 5) = \vec{v}_{b1b2}$
 $(20 - (-20); 45 - 5) = \vec{v}_{b1b2}$
 $(45; 40) = \vec{v}_{b1b2}$
 donc la trajectoire du bateau entre b_1 et b_2 est de $(45; 40)$.

Élève B.

FIG. 2 – Exercice 2 : Confusion (de vocabulaire ?) entre la trajectoire et le cap visé, malgré un calcul correct

Conclusion

Dans l'ensemble, l'expérience de ce TP était très positif, aussi bien dans le regard des élèves que pour les apprentissages. Il a été l'occasion de revoir et d'appliquer le cours dans un cadre différent. Le seul aspect négatif était la confusion créée par le vocabulaire *cap visé*, *trajectoire*, ... Il est en effet regrettable qu'un certain nombre d'élèves soient conduits à déformer le cours pour le mettre en concordance avec ce qu'ils perçoivent à l'écran.