Examen du cours de λ-calcul *

C.Berline

22 Avril 2003

Vous disposez de 3 heures; les notes du cours sont autorisées.

[De fait ce temps a été étendu dès le début à 4 heures]

Les parties I, II et III sont indépendantes.

Les parties IV et V sont indépendantes et ne dépendent que de III; on traitera l'une ou l'autre suivant que l'on souhaite plutôt travailler avec des extensions du système F ou du système E.

Bien traiter 4 des 5 premières parties rapporte déja plus que le maximum.

La partie VI est facultative et dépend de IV; ne l'aborder que si vous avez eu le temps de traiter 4 parties.

Dans les questions qui demandent d'étendre une propriété d'un système traité dans le cours (E, F, ou $AF2$) à un sur-système on demande évidemment seulement de donner les arguments mathématiques supplémentaires nécessaires à l'extension du résultat, pas de refaire les démonstrations données dans le cours ou le poly.

Notations. Dans la résolution de certaines parties du problème on pourra trouver commode d'utiliser l'inclusion $A \subseteq A'$ entre deux types en tant qu'abbréviation du schéma de règle:

"Pour tous Γ et $t \in \Lambda$ on a: $\Gamma \vdash t : A$ implique $\Gamma \vdash t : A'$."

Dans ce cas on pourra aussi se permettre d'utiliser des expressions de la forme: $\Gamma \vdash t : A \subseteq A'$. Finalement $A = A'$ abbrègera l'ensemble des deux contraintes $A \subseteq A'$ et $A' \subseteq A$.

I. Programmation dans $AF2K$.

Le système $AF2K$ est obtenu en ajoutant à $AF2$ un symbole de prédicat unaire K, qui jouera le rôle du type des entiers, le symbole de constante d'individu 0 et le symbole de fonction unaire s, et les deux règles suivantes (l'inférence dans $AF2K$ étant notée simplement $\vdash$).

$\Gamma \vdash t : Kx$ si et seulement si $\Gamma \vdash t : \forall^1 X\, [\forall y\, (Ky \to Xsy),\, X0 \to Xx]$.

*DEA de Logique et Fondements de l'Informatique (2002-2003), UFR de Mathématiques, Université Paris 7.

On admettra que la méthode de programmation par preuves s'étend à $AF2K$, relativement au type K : pour programmer une fonction f il suffira donc de trouver un terme clos de type $\forall x\,(Kx \to Kfx)$, en utilisant les règles, relativement à un système d'équations définissant f.

On définit la suite de λ-termes clos suivante: $\bar{0} =_{def} \lambda f.\lambda a.a$ et $\overline{n+1} =_{def} \lambda f.\lambda a.f\bar{n}$, qui représentera la suite des entiers au niveau de Λ. C'est sur ces entiers-là que l'on calculera dans $AF2K$.

1. Montrer que $\vdash \bar{n} : Ks^n0$ pour tout n.
2. Programmer la fonction successeur dans $AF2K$.
3. Programmer la fonction prédécesseur dans $AF2K$.

II. Programmation dans $AF2PQ$.

Ici on va programmer *sur les entiers de Church* $\hat{n}$ mais en utilisant des règles supplémentaires qui vont nous permettre d'obtenir des programmes plus efficaces que ceux qu'on pourrait avoir dans le système $AF2$, en particulier on pourra programmer la fonction charactéristique de $\leq$, notée ici cmp par le "terme de Maurey". Pour programmer cette fonction on commence par ajouter au langage des spécifications (=types) les symboles $0, s, V, F, cmp$, où $0, V, F$ sont des constantes pour 0 et pour les booléens, s un symbole de function unaire et cmp un symbole de fonction binaire. Pour programmer la fonction cmp on utilisera les équations $cmp(0,y) = V$, $cmp(sx,0) = F$ et $cmp(sx,sy) = cmp(x,y)$. Dans la suite N et B sont respectivement le type usuel des entiers de Church et des booléens de $AF2$.

Le système $AF2PQ$ est obtenu en ajoutant à $AF2$ les deux symboles de prédicats P et Q et les quatre règles:

$\Gamma \vdash t : Px$ si et seulement si $\Gamma \vdash t : \forall y\,(Qy \to Bcmp(x,y))$.

$\Gamma \vdash t : Qy$ si et seulement si $\Gamma \vdash t : \forall x\,(Px \to Bcmp(sx,y)$.

A partir de maintenant on travaille dans $AF2PQ$.

1. Trouver deux termes clos de types respectifs $P\hat{0}$ et $Q\hat{0}$.
2. Montrer que $A =_{def} \lambda f.\lambda g.gf$ est de type $\forall x\,(Px \to Psx)$ et aussi de type $\forall y\,(Qy \to Qsy)$.
3. Dans le contexte $\mu : Nx, \nu : Ny$ trouver deux termes de types respectifs: Px et Qy.
4. En déduire un terme M tel que: $\vdash M : \forall x \forall y\,(Nx, Ny \to Bcmp(x,y))$.
5. Montrer que la méthode s'étend à toute fonction g binaire sur les entiers et à valeurs dans un type de données D quelconque qui peut être définie à partir d'une fonction $f : D \to D$ et de deux éléments $a, b \in D$ par les équations $g(0,y) = a$, $g(sx,0) = b$ et $g(sx,sy) = f(g(x,y))$, en supposant qu'il existe des termes u, α, β de Λ tels que: $\vdash u : \forall z\,(Dz \to Dfz)$, $\vdash \alpha : Da$ et $\vdash \beta : Db$ (en gardant la même notation pour le type de données concret D et la formule qui le représente).
6. En déduire un terme calculant l'inf de deux entiers.
7. (facultatif) Est-ce qu'on peut en faire autant pour le *sup*? justifier brièvement.

III. Opérateurs monotones et points fixes. .

1. On rappelle qu'un treillis complet $(T, \leq)$ est un ensemble partiellement ordonné tel que tout sous-ensemble admet un plus petit majorant (sup) (et donc un plus grand minorant (inf)). Montrer que si h est croissante sur T et si $T' \subseteq T$ est clos par h et par $infs$ quelconques, alors $z_0 =_{def} \wedge\{z \in T' \,/\, h(z) \leq z\}$ est un point fixe de h dans T', et par ailleurs le plus petit possible.

2. Montrer que, si $\mathcal{S} \subseteq P(\Lambda)$ est clos par $\Rightarrow$ et par intersections quelconques, alors la fonction H_K définie sur $P(\Lambda)$ par $H_K(Z) = \cap_{S \in \mathcal{S}}((Z \Rightarrow S), S \Rightarrow S)$, a un point fixe dans $\mathcal{S}$.

3. On considère la fonction L définie sur $P(\Lambda)^2$ par $L(P, Q) = (Q \Rightarrow R,\, P \Rightarrow R)$ où R est un sous-ensemble fixé quelconque de Λ. Montrer que L a un point fixe dans $P(\Lambda)^2$ et que si $\mathcal{S}$ est clos par $\Rightarrow$, par intersections et unions quelconques, et contient R alors L a un point fixe dans $\mathcal{S}^2$. Pour ce faire on munira $P(\Lambda)^2$ de l'ordre défini par: $(P, Q) \leq (P', Q')$ si et seulement si $P \subseteq P'$ et $Q' \subseteq Q$.

IV. Extension du lemme d'adéquation à FK et FPQ. .

Le système FK est obtenu en ajoutant à F le symbole de proposition K et les deux règles:

$\Gamma \vdash t : K$ si et seulement si $\Gamma \vdash t : \forall X\,[(K \to X),\, X \to X]$.

On dira que $\mathcal{S} \subseteq P(\Lambda)$ est *adéquat pour FK* si il est adéquat pour F.

1. Donner deux exemples de $\mathcal{S}$ adéquats pour FK.

2. Donner un argument permettant d'étendre le Lemme d'adéquation de F à FK en utilisant $III.2$. Pour ce faire on sera amené à préciser la notion d'interprétation des types de FK (à valeurs dans un $\mathcal{S}$ adéquat).

Le système FPQ est obtenu en ajoutant à F les deux constantes de proposition P et Q et les quatre règles:

$\Gamma \vdash t : P$ si et seulement si $\Gamma \vdash t : Q \to B$.

$\Gamma \vdash t : Q$ si et seulement si $\Gamma \vdash t : P \to B$

où $B \equiv \forall X\,(X, X \to X)$ désigne le type usuel des Booléens dans F.

3. Montrer qu'on peut étendre le Lemme d'adéquation de F à FPQ.

4. En déduire la normalisation forte des termes typables dans $FK, FPQ, AF2K, AF2PQ$.

V. Extensions du Système E .

On considère les extensions suivantes du système E. Les formules sont les mêmes que pour le système E, l'ensemble P de constantes (ou variables) propositionnelles contient au moins trois éléments distincts p, q, r.

1. Le système E_1 est obtenu en ajoutant aux règles de E les quatre règles:

$\Gamma \vdash t : p$ ssi $\Gamma \vdash t : \{q\} \to r$

$\Gamma \vdash t : q$ ssi $\Gamma \vdash t : \{p\} \to r$

Un ensemble $\mathcal{S} \subseteq P(\Lambda)$ est adéquat pour E_1 si il est adéquat pour E.$\wedge$ clos par intersections et unions quelconques. Donner un exemple de $\mathcal{S}$ adéquat pour E_1.

2. Montrer en utilisant III.3. qu'on peut étendre le Lemme d'adéquation de E à E_1 (on sera amené à préciser les notions d'environnement et d'interprétation qui sont judicieuses pour les types de E_1).

3. En déduire que $t \in \Lambda$ est typable dans E_1 si et seulement si il est résoluble.
4. Est–il possible d'employer cette méthode pour les systèmes:
E_2 correspondant au couple d'équations $p = \{q\} \to p$ et $q = \{p\} \to q$,
E_3 engendré par la seule équation: $p = \{r\} \to p$
E_4 par $p = \{p\} \to r$ et enfin
E_5 par $\emptyset \to p = p$?
(On justifiera rapidement).
(Tous ces systèmes correspondent à des modèles de graphes non isomorphes du λ-calcul non typé, mais ceci est une autre histoire).

VI. (Facultative) Correction des programmes obtenus dans IV. .

1. Déduire de IV (et de la correction des données dans F), la correction des données dans FPQ pour le type N (c'est-à-dire que $\vdash t : N$ implique que t est $\beta\eta$-équivalent à un entier de Church).
2. Que faudrait-il encore montrer pour assurer la correction des programmes calculant des fonctions de N^n dans N dans $AF2PQ$?

Barême.

I (4): 1:1,5, 2:1, 3:1,5.

II (7): 1:0,5, 2:0,5+1,5, 3:1, 4:1 (+1 à répartir entre 1 et 3?), 5:1,5, 6:1, 7:bonus.

III (5): 1:2,5, 2:1,5, 3:1.

IV (6): 1:0,5 ou 1, 2:1,5, 3: 3 (et bonus si bien fait), 4:1.

V (6): 1:0,5, 2:2, 3:1,5, 4:1,5 à 2..

VI Bonus

Correction du Problème (esquisse).

I.1. Par induction sur n.

I.2. En méditant un court instant sur la définition de $\overline{n+1}$ il est facile de trouver deux termes calculant le successeur et le prédecesseur. Il ne reste plus qu'à vérifier qu'ils sont du bon type dans $AF2K$.

On peut aussi effectuer une recherche plus systématique. Pour le successeur ça revient à un terme de type Ksx dans le contexte $\mu : Kx$. Et analogue pour le prédécesseur.

La recherche d'un terme pour le successeur est équivalente à la recherche d'un terme de type Xsx dans le contexte $\mu : Kx$, $f : \forall y(Ky \to Xsy)$, $a : X0$, et il est facile de vérifier que $f\mu$ répond à la question. D'où le terme:

$suc =_{def} \lambda\mu.\lambda f.\lambda a.f\mu$

I.3. Pour le prédécesseur il faut trouver un terme de type Kpx dans le contexte $\mu : Kx$. Or $\mu : Kx \vdash \mu : (\forall y(Ky \to Kp(sy)), Kp0 \to Kpx$ (instancier $X-$ par $K(p(-))$). Comme $\vdash \lambda z.z : \forall y(Ky \to Kp(sy)) = \forall y(Ky \to Ky)$ et $\vdash \bar{0} : Kp0 = K0$, on voit sans peine que

$pred =_{def} \lambda\mu.\mu\, id\, \bar{0}$ convient.

II.1 On a les équations entre types:

$Bcmp(0, y) = BV$ et $Bcmp(sx, 0) = BF$. Donc:

$\vdash \lambda d.V : P0$ et $\vdash \lambda d.F : Q0$.

II.2. On a clairement:

$\mu : Px, \nu : Qy \vdash \nu\mu : Bcmp(sx, y)$ (puisque $Qy \subseteq Px \to Bcmp(sx, y)$)

donc $\vdash \lambda\mu.\lambda\nu.\nu\mu : Px \to Psx$. Par ailleurs:

$\mu : Px, \nu : Qy \vdash \mu\nu : Bcmp(sx, sy)$ (puisque $Px \subseteq Qy \to Bcmp(x, y) = Bcmp(sx, sy)$)

donc $\vdash \lambda\nu.\lambda\mu.\mu\nu : Qy \to Qsy$.

On en déduit $\vdash A : \forall x\, (Px \to Psx)$ et $\vdash A : \forall y\, (Qy \to Qsy)$.

II.3. On déduit immédiatement de 2. que:

$\mu : Nx \vdash \mu\, A\, \lambda d.V : Px$ (induction sur x) et

$\nu : Ny \vdash \nu\, A\, \lambda d.F : Qy$ (induction sur y)

Ce qui implique que les deux termes de droite sont de type voulu dans l'union des contextes.

II.4. Comme $Px \subseteq Qy \to Bcmp(x, y)$ on a:

$\mu : Nx,\ \nu : Ny \vdash (\mu\, A\, \lambda d.V)\, (\nu\, A)\, \lambda d.F : Bcmp(x, y)$.

Donc $M =_{def} \lambda\mu.\lambda\nu.(\mu\, A\, \lambda d.V)\, (\nu\, A)\, \lambda d.F$ est du type voulu

(on reconnaît le terme de Maurey).
II.5. On utilise cette fois les équations:
$Px = \forall y\, (Qy \to Dg(x,y))$
$Qy = \forall x\, (Px \to Dg(sx,y))$
$Dg(0,y) = Da$
$Dg(sx,0) = Db.$
$Dg(sx,sy) = Df(g(x,y))$
On vérifie que successivement que:
$\Lambda d.\alpha : P0 = \forall y(Qy \to Da)$ et $\Lambda d.\beta : Q0 = \forall x(Px \to Db)$
$\mu : Px, \nu : Qy \vdash \nu\mu : Dg(sx,y)$, donc:
$\vdash A \equiv \lambda\mu\lambda\nu.\mu\nu : \forall x\, (Px \to Psx)$
$\mu : Px, \nu : Qy \vdash \nu\mu : Dg(x,y)$, donc
$\mu : Px, \nu : Qy \vdash u(\nu)\mu : Df(g(x,y)) = Dg(sx,sy)$, donc:
$\vdash A_u \equiv \lambda\nu\lambda\mu.u(\nu)\mu : \forall y\, (Qy \to Qsy)$, et enfin
$M_u =_{def} \lambda\mu.\lambda\nu.(\mu\, A\, \lambda d.\alpha)\, (\nu\, A_u)\, \lambda d.\beta$ répond à la question.
II.6. L'inf correspond au cas: $D =_{def} N$, $a =_{def} 0$, $b =_{def} 0$, $f =_{def} s$
(et *cmp* au cas: $D =_{def} B$, $a =_{def} V$, $b =_{def} F$, $f =_{def} id$).
II.7. Le problème dans le cas du sup c'est qu'il rentre dans le cas plus général où: $g(0,y) = a(y)$ et $g(sx,0) = b(x)$ et qu'il faudrait disposer pour démarrer de termes de type $\forall y\, (Qy \to Ny)$ et $\forall x\, (Px \to Nx)$ pour pouvoir obtenir, à partir de termes programmant les fonctions a et b, des termes de type $P0$ et $Q0$.

Remarque: inf et sup pouvaient s'obtenir directement à partir de M calculant *cmp* par: $inf = \lambda\mu\lambda\nu.M\mu\nu\mu\nu$ et $sup = \lambda\mu\lambda\nu.M\mu\nu\nu\mu$. Cette solution nétait pas dans l'esprit du problème mais elle a été acceptée aussi. Toutefois il fallait tout de même vérifier rapidement que ces deux termes étaient bien typés (ou donner un argument disant pourquoi).

III.1. Soit $\mathcal{T} =_{def} \{\, z \in T' \,/\, h(z) \leq z \,\}$. On a
(0) $z_0 =_{def} \wedge\mathcal{T}$.
On montre successivement que:
(a) pour tout $z \in \mathcal{T}$ on a $h(z_0) \leq h(z)$ (car $z_0 \leq z$ et h monotone);
(b) donc $h(z_0) \leq \wedge\{\, h(z) \,/\, z \in \mathcal{T} \,\} \leq \wedge\mathcal{T} =_{def} z_0$;
(la deuxième inégalité vient de la définition de T et des propriétés de $\wedge$);
(c) donc $z_0 \in \mathcal{T}$ (car $z_0 \in T'$ par clôture de T' par infs et (b));
(d) donc z_0 est le plus petit élément de $\mathcal{T}$ (par (c) + (0));
(e) et $h(h(z_0)) \leq h(z_0)$ (monotonicité de h et (b));
(f) $h(z_0) \in \mathcal{T}$ (par (T) + stabilité de T' par h);
(g) donc $z_0 \leq h(z_0)$ (par (f) + (0))
(h) donc $z_0 = h(z_0)$ (par (g) + (b)).
(i) z_0 est un point fixe de h dans $\mathcal{T}$ (par (h) et (c))
(j) c'est le plus petit (puisque tout point fixe de h est clairement dans $\mathcal{T}$).

III.2. Compte tenu du 1. et de ce que $(P(\Lambda), \subseteq)$ est un treillis complet il suffit de voir que H_K est croissante et que $\mathcal{S}$ est clos par H_K. La deuxième assertion suit clairement de la clôture de $\mathcal{S}$ par intersections et pour $\Rightarrow$. Supposons maintenant $Z \subseteq Z'$. Alors pour tout $S \in \mathcal{S}$ on a $Z' \Rightarrow S \subseteq Z \Rightarrow S$

(contravariance de $\Rightarrow$ en son premier argument), donc $(Z \Rightarrow S) \Rightarrow (S \Rightarrow S) \subseteq (Z' \Rightarrow S) \Rightarrow (S \Rightarrow S)$ pour la même raison. Il en résulte que $H_K(Z) \subseteq H_K(Z')$, cqfd.

III.3. Compte tenu de 1. Il suffit de vérifier que $(P(\Lambda)^2, \leq)$ est un treillis complet, que L est monotone, et que $\mathcal{S}^2$ est clos par L et par $infs$ quelconques. Il est clair que tout ensemble $\{(P_i, Q_i) \,/\, i \in I\}$ de couples de parties de Λ admet un *sup* pour $\leq$, qui n'est autre que $(\cup_{i \in I} P_i, \cap_{i \in I} Q_i)$. Le fait que $\mathcal{S}^2$ est clos par L suit immédiatement de la clôture de $\mathcal{S}$ pour $\Rightarrow$. Quant au fait que L est croissante pour $\leq$ il suit immédiatement des propriétés de co et contravariance de $\Rightarrow$ par rapport à ses arguments.

IV.1. $\mathcal{S} =_{def} \{\emptyset, \Lambda\}$,ou $P^{\beta\eta}(\Lambda)$, ou $P^{N_f sat}(N_f)$, etc.

IV.2. Une interprétation des types de FK est relative à un ensemble $\mathcal{S}$ adéquat pour FK, et à un point fixe S_0 de H_K dans $\mathcal{S}$, par lequel on interprète la constante K. Les interprétations des types sont engendrées à partir des environnements (c'est-à-dire des fonctions de $VarProp \to \mathcal{S}$ de la manière habituelle: $I(K) = S_0$, $I(X)$ est donné par l'environnement, $I(\forall X.A) =_{def} \cap_{S \in \mathcal{S}} I[X := S](A)$ et $I(A \to B) =_{def} I(A) \Rightarrow I(B)$. Noter que $I(\forall X(K, X \to K)) = H_K(I(K)) = I(K)$.

Le Lemme d'adéquation s'énonce alors comme pour F.

Pour le démontrer (par induction sur la dérivation du séquent) il faut prendre en compte les deux règles supplémentaires, mais le raisonnement de passage est trivial dans les deux cas puisque $I(K) = I(\forall X(K, X \to K))$.

IV.3. Idem mais cette fois relatif à $\mathcal{S}$ et à un point fixe de L, qui servira à interpréter les constantes P, Q. La question c'est de voir pour quel L, c'est-à-dire pour quel $R \in \mathcal{S}$. Intuitivement cet R doit coïncider avec l'interprétation $I(B)$ du type B des booléens dans un environnement I quelconque. Il suffit donc en fait de fixer $R =_{def} \cap_{S \in \mathcal{S}}(S, S \Rightarrow S)$. On pourrait étendre cette remarque à tout type clos dont la définition ne fait pas intervenir P ou Q, c'est-à-dire à tout type clos de F.

On étend ensuite l'ensemble des environnements à valeur dans $\mathcal{S}$ en des interprétations de tous les types de FPQ de la manière habituelle, et on vérifie que $I(B) = R$, ce qui est trivial.

Le Lemme d'adéquation s'énonce alors comme pour F.

Pour le démontrer (par induction sur la dérivation du séquent) il faut prendre en compte les quatre règles supplémentaires, mais le raisonnement de passage est trivial dans les quatre cas puisque $I(P) = I(Q \to B)$ et $I(Q) = I(P \to B)$.

IV. 4. Pour FK et FPQ il suffit d'appliquer les lemmes d'adéquations correspondants, relatifs à $\mathcal{S} =_{def} P^{N_f Sat}(N_f)$, avec le raisonnement habituel. Comme tout terme typable dans $AF2K$ ou $AF2PQ$ est typable dans FK ou FPQ, il en résulte que les termes typables dans ces systèmes sont également fortement normalisables.

Noter qu'il existe des extensions récursives encore plus expressives de $AF2K$ et $AF2PQ$ où la forte normalisation n'est plus vraie en générale, mais où l'on a un lemme d'adéquation qui permet de programmer encore ... et d'obtenir encore plus d'algorithmes. Ici on s'est contenté de travailler avec des petits fragments commodes.

VI. 1. N est un type clos du système F, et pour de tels types l'interprétation est la même qu'on la fasse dans F ou FPQ (si on garde le même $\mathcal{S}$). Donc la preuve faite pour F est valable sans changement.

VI.2. Il faudrait démontrer le lemme d'adéquation, ainsi que la correction fine des données telle qu'on l'a pour $AF2$. Ce qui demande déja (et uniquement) de définir l'interprétation de P et Q par des fonctions de $\mathcal{S}^M$ satisfaisant des équations nettement plus compliquées que ce qui se passe dans F. En fait la définition précise de l'interprétation précise de P, Q est sans intérêt pour ce problème précis, ce qui compte c'est qu'il en existe une [[A compléter]].

V.1. Exemples: $\{\emptyset, \Lambda\}$ ou $P^{s\,at}(\Lambda)$.

V.2. Les environnements I qui sont judicieux pour E_1 sont ceux qui interprètent les variables propositionnelles p, q, r de telle manière que l'on ait $I(p) = I(q) \Rightarrow I(r)$ et $I(q) = I(p) \Rightarrow I(r)$ (*). On obtient ces environnement en choisissant arbitrairement $I(r) = R \in \mathcal{S}$ et en prenant pour $(I(p), I(q))$, un point fixe de l'opérateur L de III.3. correspondant à ce R. Les interprétations judicieuses des types de E_1 sont les interprétations au sens de E qui sont engendrées par des environnements judicieux pour $E_{1.}$ Le lemme d'adéquation s'étend alors sans problème (les quatre règles supplémentaires donnent lieu à des cas supplémentaires qui se résolvent trivialement à cause des équations (*)).

V.3. La démonstration donnée pour E vaut alors mot pour mot.

V.4. Pour $E_2, E_{3,}$ et E_5 la réponse est "oui" car on peut associer facilement à ces systèmes des opérateurs croissants sur $P(\Lambda)^2$ ou $P(\Lambda)$, munis des ordre indiqués en III (dans le cas de E_5 ce sera $h_5(Z) =_{def} \Lambda \Rightarrow Z$). Dans le cas de E_4 c'est non car $h_4(Z) = Z \Rightarrow R$ n'est croissant pour aucun R.