

# Étude d'une fonction de deux variables

L'objectif de ce TP est d'étudier les méthodes de représentation d'une fonction numérique de deux variables réelles, et de résolution approchée d'un système d'équations à deux inconnues réelles. On pose pour  $(x, y) \in \mathbb{R}^2$  :

$$f(x, y) = x^3 + 2xy + y^2, \quad g(x, y) = x^2 + 3xy + y^3.$$

## 1) Visualisation de $f$

### a) Représentation dans $\mathbb{R}^3$

Pour se faire une idée des variations d'une fonction numérique  $f$  de deux variables on peut dessiner la surface représentative de  $f$ , c'est à dire la surface de  $\mathbb{R}^3$  d'équation  $z = f(x, y)$ . Utiliser l'instruction Maple `plot3d` :

```
plot3d(x,y,f(x,y),x=a..b,y=c..d)
```

Vous pouvez « faire tourner » la surface à la souris, ajouter des axes ou changer le mode de représentation, consulter l'aide sur `plot3d` pour voir quelles sont les options disponibles.

La fonction Maple `display3d` permet de superposer plusieurs dessins de surfaces en éliminant les parties cachées. Faire afficher la surface représentative de  $f$  et un plan parallèle à  $xOy$  (qui est la surface représentative d'une fonction constante) sur un même dessin. Ceci permet de « voir » l'intersection de ces deux surfaces qui est, à une translation verticale près, la courbe du plan d'équation  $f(x, y) = k$  où  $k$  est la cote du plan choisi (ligne de niveau  $k$  de  $f$ ).

### b) Représentation dans $\mathbb{R}^2$

Maple contient une fonction `contourplot` qui affiche directement quelques lignes de niveau de  $f$  dans une région spécifiée du plan :

```
contourplot(f(x,y),x=a..b, y=c..d, contours=n)
```

$n$  est le nombre de lignes de niveau désirées, Maple choisit automatiquement les valeurs des niveaux et trace une approximation des lignes correspondantes à partir des valeurs de  $f$  au sommets d'une subdivision rectangulaire du pavé  $[a, b] \times [c, d]$ . Pour obtenir un tracé précis, il peut être nécessaire d'imposer le nombre de subdivisions à utiliser par une option `grid=[u,v]` ( $u$  divisions en abscisse,  $v$  en ordonnée, les valeurs par défaut sont  $u = v = 25$ ). Noter que Maple doit calculer les valeurs de  $f$  en  $uv$  points pour réaliser le tracé, ne pas prendre des valeurs trop élevées pour  $u$  et  $v$ .

### c) Tracé des lignes de niveau à l'aide d'un système différentiel

On peut aussi dessiner une ou plusieurs lignes de niveau de  $f$  en les interprétant comme des courbes paramétrées vérifiant une équation différentielle et en résolvant numériquement cette équation. Soit  $k \in \mathbb{R}$  et  $C_k$  la ligne de niveau  $k$  de  $f$ . On suppose que  $C_k$  est, au moins localement, le support d'une courbe paramétrée de classe  $C^1$  :  $t \mapsto (x(t), y(t))$ . Puisque  $f(x(t), y(t))$  est constante, on a par dérivation composée :

$$x'(t) \frac{\partial f}{\partial x}(x(t), y(t)) + y'(t) \frac{\partial f}{\partial y}(x(t), y(t)) = 0$$

donc le vecteur dérivé  $(x'(t), y'(t))$  est orthogonal au gradient de  $f$  en  $(x(t), y(t))$ , soit colinéaire au vecteur  $N(t) = (-\frac{\partial f}{\partial y}, \frac{\partial f}{\partial x})(x(t), y(t))$ . Ainsi,  $x$  et  $y$  sont solutions d'un système différentiel de la forme :

$$x' = -\lambda(t) \frac{\partial f}{\partial y}(x, y), \quad y' = \lambda(t) \frac{\partial f}{\partial x}(x, y)$$

et réciproquement, pour tout choix de la fonction  $\lambda$ , les courbes intégrales du système précédent vérifient  $f(x(t), y(t)) = \text{cte} = f(x_0, y_0)$ . En général on choisit  $\lambda(t) = 1/\|N(t)\|_2$  ce qui revient à paramétrer la ligne de niveau de  $f$  passant par  $(x_0, y_0)$  par une abscisse curviligne.

Programmer cette méthode. Pour représenter graphiquement la solution approchée d'un système différentiel  $x' = \alpha(x, y)$ ,  $y' = \beta(x, y)$  on utilisera les instructions suivantes :

```

p := dsolve({diff(x(t),t) = α(x(t),y(t)),
             diff(y(t),t) = β(x(t),y(t)),
             x(0) = x0, y(0) = y0},
            {x(t),y(t)}, type=numeric);

odeplot(p, [x(t),y(t)], a..b);

```

La première instruction retourne dans la variable **p** une procédure Maple qui résout numériquement le système différentiel avec condition initiale donné, la seconde utilise cette procédure pour tracer la courbe  $t \mapsto (x(t), y(t))$ ,  $t$  variant entre  $a$  et  $b$ . Pour tracer un réseau de lignes de niveau de  $f$  il suffit de créer une liste de dessins en faisant varier le point initial  $(x_0, y_0)$  et de les afficher simultanément avec la fonction **display**.

On peut vérifier la précision de la résolution approchée en faisant tracer la courbe  $t \mapsto (t, f(x(t), y(t)))$  qui doit être un segment horizontal si  $f(x(t), y(t))$  est bien constant. Pour ce faire, remplacer  $[x(t), y(t)]$  par  $[t, f(x(t), y(t))]$  dans l'instruction **odeplot** précédente.

## 2) Résolution d'un système d'équations $f(x, y) = a, g(x, y) = b$

Il s'agit en fait de trouver les points communs aux lignes de niveau  $a$  pour  $f$  et  $b$  pour  $g$ . On prendra  $a = 2, b = 1$  pour les essais.

### a) Première tentative

Maple contient un « résolveur » d'équations, **solve**. Cette fonction n'est garantie correcte que pour un système d'équations linéaires ou pour des équations algébriques à coefficients entiers. Compte tenu des formes de  $f$  et  $g$  on peut donc taper bêtement :

```
solve({f(x,y)=2,g(x,y)=1},{x,y});
```

### b) Résolution graphique puis numérique

Faire tracer sur un même dessin les lignes de niveau dont on cherche les intersections. On trouve graphiquement trois points d'intersection dont on peut connaître approximativement les coordonnées en les désignant à la souris. Pour calculer des valeurs approchées de ces points, utilisez le « résolveur numérique » de Maple :

```
solve({f(x,y)=2,g(x,y)=1},{x,y},{x=α..β,y=γ..δ});
```

qui recherche (probablement par dichotomie) les solutions du système proposé dans le pavé  $[\alpha, \beta] \times [\gamma, \delta]$ .

### c) Méthode de Newton

En dimension 1, on construit une suite convergeant dans les bons cas vers un zéro de  $f$  à l'aide de l'itération de Newton :

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

La méthode se généralise en dimension quelconque. Ici, pour résoudre  $f(x, y) = a, g(x, y) = b$  on introduit la fonction  $\varphi : (x, y) \mapsto (f(x, y) - a, g(x, y) - b)$  et l'on construit la suite  $(x_n, y_n)$  à partir de  $(x_0, y_0)$  et de la relation de récurrence :

$$(x_{n+1}, y_{n+1}) = (x_n, y_n) - J_{\varphi}^{-1}(x_n, y_n) \cdot \varphi(x_n, y_n)$$

où  $J_{\varphi}$  désigne la matrice jacobienne de  $\varphi$ . Tester cette méthode sur le système proposé.

En dimension 1 on peut prouver que l'on a obtenu une bonne approximation d'un zéro de  $f$  en calculant les valeurs de  $f$  en deux points proches encadrant celui trouvé et en constatant que  $f$  change de signe entre ces deux points. Peut-on généraliser cette méthode de preuve en dimension 2 ?