Club MATH

Centre informatique pédagogique (CIP)
Case Postale 172
1211 GENEVE 3
Tél. (022) 318.05.30
Responsable:
Raymond Morel

Lettre nº 13

Lundi 1er juin 1992 à 17 h.

Equations différentielles

Traduction et adaptation d'un article de Troels Petersen, Differential Equations, *Mathematica Journal*, Volume 1, Issue 3, Winter 1991

Présentation: Bernard Vuilleumier

La solution d'une équation différentielle est une famille de courbes

Vous pouvez étudier des équations différentielles ordinaires à l'aide de *Mathematica*. La commande **DSolve** permet de résoudre formellement des équations qui relient une fonction inconnue d'une variable et ses dérivées:

```
DSolve[y''[x]==2x/(1+x^2),y[x],x]
```

Le premier argument de la fonction définit l'équation, le second la fonction inconnue et le troisième la variable indépendante. La solution est une famille de courbes paramétrées par les constantes d'intégration `C[1]` et `C[2]`. Elle se présente sous la forme suivante:

```
{{y[x] -> 2*ArcTan[x]+C[1]+x*C[2]+x*Log[1+x^2]}}
```

Vous pouvez fixer des conditions aux limites ou des contraintes en ajoutant des équations:

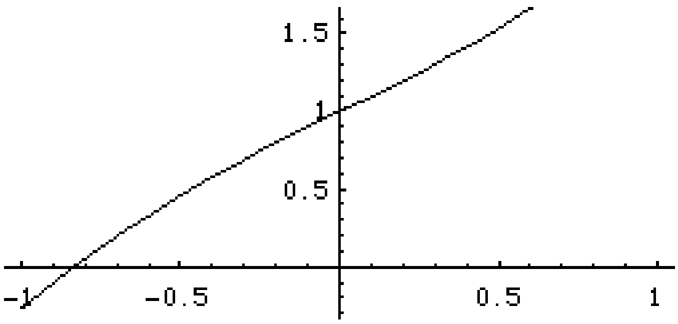
En fixant des conditions aux limites on obtient une solution particulière

```
solution = DSolve[{y''[x]==2x/(1+x^2),
                   y''[0]==0,
                   y'[0]==1,
                   y[0]==1
                   },
          y[x],x
     ]
{{y[x] -> 1-x+2*ArcTan[x]+x*Log[1+x^2]}}
```

Si vous voulez représenter graphiquement la solution ou calculer des valeurs, vous affectez le deuxième élément de la liste ci-dessus à une variable:

On peut représenter graphiquement la solution ou calculer ses valeurs en différents points

```
f=solution[[1,1,2]]
1 - x + 2*ArcTan[x] + x*Log[1 + x^2]
Plot[f,{x,-1,1}]
Table[f,{x,xmin,xmax,pas}]
```



On peut résoudre numériquement les équations qui n'ont pas de solution générale

Il existe relativement peu d'équations qui possèdent une solution explicite exprimable en termes de fonctions usuelles. Mais les autres équations peuvent généralement être résolues numériquement avec **NDSolve**.

```
solnum=NDSolve[{y''[x]+2y'[x]+16y[x]==0,
                y[0]==1,
                y'[0]==0
                },
               y,{x,0,3}
       ]
```

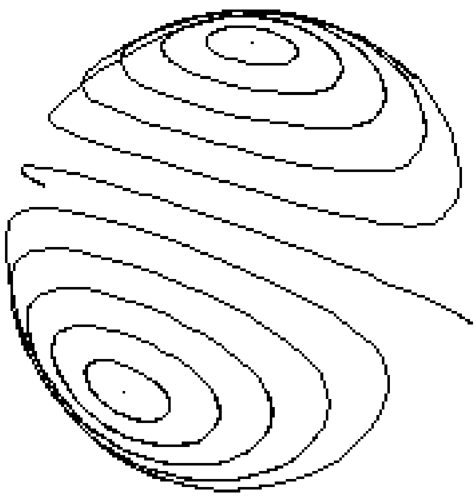
Etant donnés une équation différentielle ordinaire ou un système d'équations, des conditions initiales et un domaine de définition pour la variable indépendante, **NDSolve** génère une solution approchée sous la forme d'un objet du type **InterpolatingFunction** :

```
{y -> InterpolatingFunction[{0., 3.}, <>]}
```

Pour résoudre numériquement une équation, il faut fournir suffisamment de conditions initiales pour que la solution soit complètement déterminée

NDSolve utilise une méthode pas à pas pour approcher les valeurs de la solution aux points x_0, x_1, ... x_m. Les valeurs intermédiaires sont interpolées par une fonction cubique. Cette fonction, représentée par `InterpolatingFunction[{x`$_{min}$`, x`$_{max}$`}, <>]`, peut être évaluée en tout point x compris entre x_{min} et x_{max}. Le symbole <> représente la table des valeurs qui définissent la fonction et qui ne sont pas affichées. **NDSolve** résout des problèmes à valeurs initiales, c'est-à-dire des équations différentielles pour lesquelles les valeurs de y et de ses dérivées sont données au point de départ x_0. La solution formelle donnée par **DSolve** pouvait contenir des constantes d'intégration indéterminées, mais celle donnée par **NDSolve** ne peut pas en contenir. Par conséquent, il faut fournir suffisamment de conditions initiales pour que la solution numérique soit complètement déterminée. En calculant et en dessinant des solutions pour une liste de conditions initiales, **NDSolve** permet d'obtenir des portraits de phase de systèmes à deux ou à trois dimensions.

Il est possible d'obtenir graphiquement les solutions correspondant à différentes conditions initiales



Prochaine réunion: lundi 31 août 1992 à 17 h.