TP 1 : Résolution approchée (solutions)

Pierre.Saramito@imag.fr
http://www-lmc.imag.fr/lmc-edp/Pierre.Saramito

9 octobre 2000

1 Action 1

Fichier 'init_matrice.m'

```
function a = init_matrice(n)

  for i = 1:n
    a(i,i) = 2;
  end
  for i = 1:n-1
    a(i,i+1) = -1;
    a(i+1,i) = -1;
  end
```

Fichier 'init_second_membre.m'

```
function b = init_second_membre(n)

  h = 1.0/(n+1);
  for i = 1:n
    b(i,1) = h*h*f(i*h);
  end
```

Fichier 'f.m'

```
function fx = f(x)
```

```
  fx = pi*pi*sin(pi*x);
```

2 Action 2

Fichier 'resoudre.m'

```
function u = resoudre(n)

  a = init_matrice(n);
  b = init_second_membre(n);
  u = a\b;
```

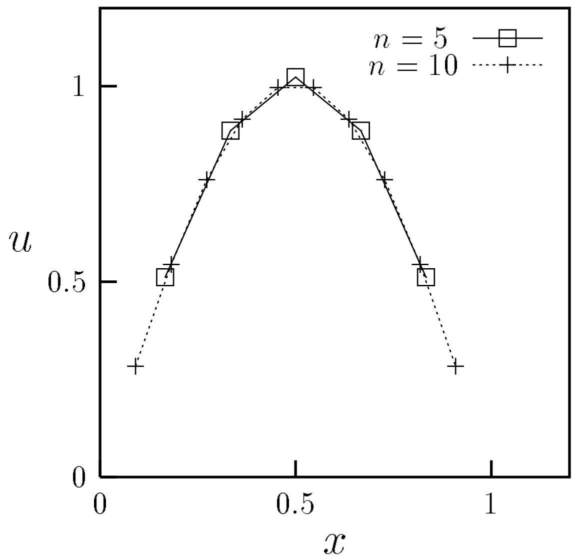

FIG. 1 – *Comparaison des solutions approchées pour* $n = 5$ *et* $n = 10$.

Fichier 'abscisse.m'

```
function x = abscisse(n)

  h = 1.0/(n+1);
  for i = 1:n
    x(i) = i*h;
```

```
  end
```

La commande `matlab` suivante :

```
  plot (abscisse(5), resoudre(5), abscisse(10), resoudre(10))
```

produit un graphique du type de la figure 1.

3 Action 3

La solution exacte est $u(x) = \sin(\pi x)$.

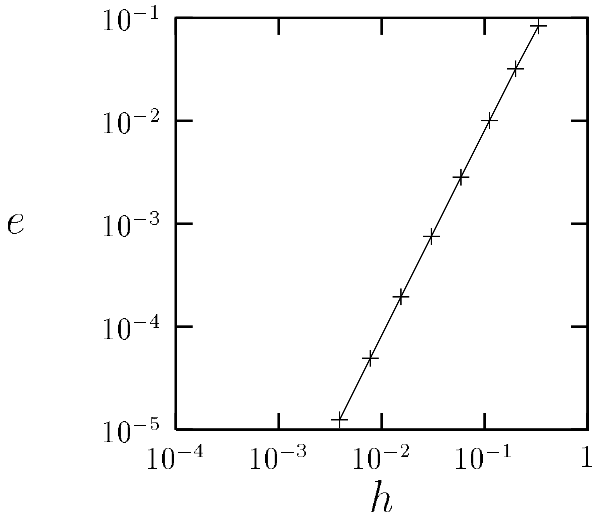

FIG. 2 – *Test de convergence : log de l'erreur maximum en fonction de log(h).*

La commande `matlab` suivante :

```
  loglog (pas(8), erreur_max(8))
```

produit un graphique du type de la figure 2. L'ordre de l'approximation peut être estimé par la pente sur cette figure: $\alpha \approx 2$.

Fichier 'pas.m'

```
function h = pas(k_max)

  for k = 1:k_max
    n    = 2^k;
```

```
    h(k) = 1.0/(n+1);
  end
```

Fichier ‘erreur_max.m’

```
function e = erreur_max(k_max)

  for k = 1:k_max
    n = 2^k;
    h = 1.0/(n+1);
    u = resoudre(n);
    for i = 1:n
      u_ex(i,1) = u_exact(i*h);
    end
    e(k) = max(abs(u-u_ex));
  end
```

Fichier ‘u_exact.m’

```
function ux = u_exact(x)

  ux = sin(pi*x);
```