

## IV. Les fonctions - Corrigé

### 1. Tir balistique

#### 1.1. Portée

Il y a deux façons de résoudre ce problème. La première est de résoudre l'équation à la main, puis d'en faire une fonction. La deuxième est de laisser Matlab résoudre l'équation.

##### 1.1.1. Résolution manuelle

Nous partons de cette équation:

$$z(x) = -1/2 gx^2 / (V_0 \cos \alpha)^2 + x \tan \alpha$$

Nous savons que l'objet touche le sol lorsque  $z = 0$ . Nous pouvons donc écrire :

$$-1/2 gx^2 / (V_0 \cos \alpha)^2 + x \tan \alpha = 0$$

En simplifiant le signe et un  $x$ , la portée se trouve directement :

$$x = 2 \tan \alpha (V_0 \cos \alpha)^2 / g$$

La fonction Matlab correspondante est :

```
function [portee] = tirbalistique_f1(alphadeg, V0kmh)
%TIRBALISTIQUE Tir balistique.
% TIRBALISTIQUE(alphadeg, V0kmh) calcul la portée correspondante
% à l'angle alphadeg (degrés) et à la vitesse initiale V0kmh
% (km/h)

alpha = pi*alphadeg/180;
V0 = V0kmh/3.6;
portee = 2*tan(alpha)*(V0*cos(alpha))^2 / 9.81;
```

##### 1.1.2. Résolution à l'aide de fsolve

Matlab est capable de résoudre l'équation à l'aide de la fonction fsolve.

```
function [portee] = tirbalistique_f2(alphadeg, V0kmh)
%TIRBALISTIQUE Tir balistique.
% TIRBALISTIQUE(alphadeg, V0kmh) calcul la portée correspondante
% à l'angle alphadeg (degrés) et à la vitesse initiale V0kmh
% (km/h)

alpha = pi*alphadeg/180;
V0 = V0kmh/3.6;
```

```
% Valeur de départ
x0 = 10;

% Résoudre la fonction
z = fsolve(@(x) trajectoire(V0,alpha,x,z0),x0);

portee = z;

function z = trajectoire(V0,alpha,x)

z = -(1/2)*9.81*(x^2) / (V0*cos(alpha))^2 + x*tan(alpha);
```

### 1.1.3. Script

Le script utilisant cette fonction s'écrit:

```
clear all;

prompt = {'Quel est l''angle (degrés) initial du tir ?', 'Quelle
          est la vitesse (km/h) initiale du tir ?'};
dlg_title = 'Paramètres tir balistique';

params = inputdlg(prompt, dlg_title);

alpha = str2num(params{1});
vitesse = str2num(params{2});

portee = tirbalistique_f1(alpha,vitesse);

disp(['La portée est de ', num2str(portee), ' m'])
```

En utilisant cet outil, on peut répondre aux questions suivantes :

- Quelle distance atteint un tir à la vitesse de 50 km/h avec un angle de 30° ?  
17.03 m
- Quel est l'angle optimal dans tous les cas ?  
Un angle de 45° donne la portée maximale quel que soit la vitesse.

### 1.2. Changement de la hauteur initiale du tir

La hauteur initiale peut être introduite dans la fonction :

```
function [portee] = tirbalistique_f2(alphadeg, V0kmh, z0)
%TIRBALISTIQUE Tir balistique.
% TIRBALISTIQUE(alphadeg, V0kmh) calcul la portée correspondante
% à l'angle alphadeg (degrés), à la vitesse initiale V0kmh
% (km/h) et à la hauteur initiale z0 (m)
```

```
alpha = pi*alphadeg/180;
V0 = V0kmh/3.6;

% Valeur de départ
x0 = 10;

% Résoudre la fonction
z = fsolve(@(x) trajectoire(V0,alpha,x,z0),x0);

portee = z;

function z = trajectoire(V0,alpha,x,z0)

z = -(1/2)*9.81*(x^2) / (V0*cos(alpha))^2 + x*tan(alpha) + z0;
```

Qui est ensuite appelée par le script :

```
clear all;

prompt = {'Quel est l''angle (degrés) initial du tir ?', 'Quelle
est la vitesse (km/h) initiale du tir ?', 'Quelle est la hauteur
(m) initiale ?'};
dlg_title = 'Paramètres tir balistique';

params = inputdlg(prompt, dlg_title);

alpha = str2num(params{1});
vitesse = str2num(params{2});
hauteur = str2num(params{3});

portee = tirbalistique_f2(alpha, vitesse, hauteur);

disp(['La portée est de ', num2str(portee), ' m'])
```

En utilisant cet outil, on peut répondre à la question suivante :

- Quelle est la distance atteinte par le même tir que précédemment ?

27.68 m