

Dessiner la fusée de Tintin avec pst-solides3d

10 avril 2008

Table des matières

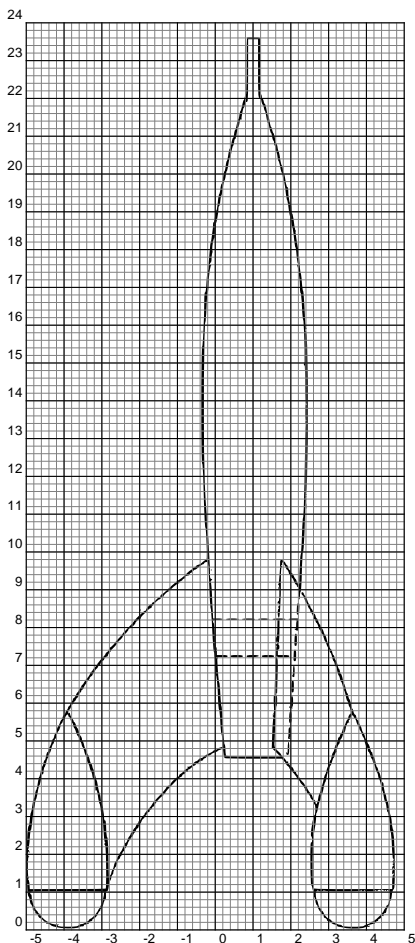
| | | |
|----------|---|----------|
| 1 | Présentation | 2 |
| 2 | Le plan de la fusée | 2 |
| 3 | La construction du corps de la fusée | 2 |
| 3.1 | Le corps de la fusée | 2 |
| 3.2 | Le coloriage du damier rouge et blanc | 3 |
| 4 | Amortisseurs, supports et empennages | 4 |
| 5 | La fusée entière | 6 |
| 6 | Utilisation des données des fichiers obtenus : fusée de base | 6 |
| 7 | Utilisation des données de la fusée de base pour d'autres situations | 7 |
| 8 | Conclusion | 7 |

1 Présentation

Ce document décrit la méthode que j'ai employée pour dessiner avec `pst-solides3d`, la célèbre fusée à damier qui emporta pour un voyage aller-retour sur la Lune Tintin, ses inséparables compagnons : le capitaine Haddock, le professeur Tournesol et bien sûr Milou avec aussi, rappelons-le, l'ingénieur Franck Wolff et comme passagers imprévus : les deux Dupondt et le méchant colonel Jorgen, mais je n'insiste pas, les amateurs de Tintin connaissent l'histoire.

2 Le plan de la fusée

Le plan de la fusée est détaillé à la page 35 du premier album *Objectif Lune*. C'est à partir de ce plan et en posant un quadrillage sur celui-ci que j'ai relevé les coordonnées des points du contour de la fusée. Le voici en réduction :



```
\psscalebox{0.5}{  
1 \begin{pspicture}(-5,0)(5,24)  
2 \rput(-0.17,11.8){\includegraphics[scale=1.1]{plan.eps}}  
3 \psgrid(-5,0)(5,24)  
4 \end{pspicture}}
```

3 La construction du corps de la fusée

3.1 Le corps de la fusée

Le profil (la section) du corps de la fusée ayant été déterminé dans l'étape précédente, dans `\psSolid`, c'est l'objet anneau qui va servir à déterminer toutes les caractéristiques du corps de la fusée grâce aux

options file et writesolid :

```
\newcommand\SectionRocket{
%y z
  0 5
-0.8 4.6
-0.85 5.6
-0.95 6.6
-1.05 7.6
-1.1 8.6
-1.2 9.6
-1.25 10.6
-1.3 11.6
-1.35 12.6
-1.4 13.6
-1.4 14.6
-1.375 15.6
-1.27 16.6
-1.2 17.6
-1.05 18.6
-0.85 19.6
-0.65 20.6
-0.35 21.6
-0.1 22.2
-0.1 23.75
  0 23.75
}
\begin{pspicture}(-5,0)(5,24)
\psSolid[object=anneau,section=\SectionRocket,file=corps-fusee,action=writesolid]
\end{pspicture}
```

La séquence suivante `LaTeX fichier.tex->dvips->GSview (Windows) ou gv (Linux)` va créer 4 fichiers :

- corps-fusee-couleurs.dat -> les couleurs des faces ;
- corps-fusee-faces.dat -> la liste des faces ;
- corps-fusee-sommets.dat -> la liste des sommets ;
- corps-fusee-io.dat -> le nombre de faces.

Par défaut, sous Windows et Linux, la protection des fichiers du disque dur est activée et ne permet donc pas l'écriture sur le disque. Pour désactiver cette protection, tout au moins temporairement, voici les deux procédures correspondantes :

Linux : le conseil de Jean-Michel Sarlat : le plus simple est donc d'utiliser ghostscript directement, en console. Comme il n'y a rien à attendre comme image :

```
$> gs -dNOSAFER fichier.ps quit.ps
```

Windows : dans le menu Options, l'option 'Protection des fichiers' ne doit pas être cochée.

3.2 Le coloriage du damier rouge et blanc

Il faut intervenir directement dans les données du fichier corps-fusee-couleurs.dat. Dans un premier temps il faut numéroter les facettes grâce à l'option numfaces=all. On lit donc les fichiers

précédents et on affiche la fusée avec ses faces numérotées.

```
\begin{pspicture}(-5,0)(5,24)
\psframe(-5,0)(5,24)
\psset{SphericalCoor,viewpoint=1000 250 0,Decran=1000}
\psset{fontsize=5,numfaces=all}
\psSolid[object=datfile,file=corps-fusee,grid](0,0,0)
\end{pspicture}
```

Il suffit maintenant dans le fichier `corps-fusee-couleurs.dat` de changer la couleur des facettes concernées de blanc en rouge :

```
(1 1 1 setrgbcolor)->(1 0 0 setrgbcolor)
```

et d'enregistrer à nouveau le fichier. Le corps de la fusée avec le célèbre damier rouge et blanc est prêt !

4 Amortisseurs, supports et empennages

Un amortisseur est constitué d'une hémisphère et le carénage des supports est un cône arrondi dont le profil de la section est un arc de cercle. Les deux vont ensemble et c'est encore l'anneau qui va permettre de tracer cet ensemble. Pour changer, le calcul se fera directement dans le `\code jps`.

```
/ammortisseur_carenage {[
-90 10 0 { % for
  /angle exch def
  angle cos
  angle sin 1 add
} for
0 2 22{
  /angle exch def
  angle cos 11.78 mul -10.78 add
  angle sin 11.78 mul 1 add
} for
0 5.75
] 24 newanneau
dup (rouge) outputcolors
{0 -5 0 translatepoint3d} solidtransform
```

Le profil de l'empennage est relevé sur le plan et il est considéré comme un objet plat à double face :

```
/empennage {
2 dict begin
  /F [
    [5 6 4] % 0
    [4 6 7] % 1
    [4 7 3] % 2
    [3 7 8] % 3
    [3 8 2] % 4
    [2 8 1] % 5
    [1 8 0] % 6
    [0 8 10] % 7
    [10 8 11] % 8
```

```

[11 8 12] % 9
[12 8 16] % 10
[12 14 13] % 11
[12 15 14] % 12
[12 16 15] % 13
[8 17 16] % 14
[8 18 17] % 15
[8 9 18] % 16
% arrière
[6 5 4] % 0
[4 7 6] % 1
[7 4 3] % 2
[7 3 8] % 3
[8 3 2] % 4
[8 2 1] % 5
[8 1 0] % 6
[8 0 10] % 7
[8 10 11] % 8
[8 11 12] % 9
[8 12 16] % 10
[14 12 13] % 11
[15 12 14] % 12
[16 12 15] % 13
[17 8 16] % 14
[18 8 17] % 15
[9 8 18] % 16
] def
%% tableau des sommets
/S [
  0 -0.8 4.6 % 0
  0 -0.85 5.6 % 1
  0 -0.95 6.6 % 2
  0 -1.05 7.6 % 3
  0 -1.1 8.6 % 4
  0 -1.2 9.6 % 5
  0 -2 9.2 % 6
  0 -3 8.3 % 7
  0 -4 7.2 % 8
  0 -5 5.75 % 9
  0 -1.6 4 % 10
  0 -2.5 3 % 11
  0 -3.2 2.2 % 12
  0 -4 1 % 13
  0 -4.0072 1.4111 % 14
  0 -4.0645 2.2313 % 15
  0 -4.1789 3.0456 % 16
  0 -4.35 3.8498 % 17
  0 -4.5765 4.6402 % 18
] def

```

```

    S F generesolid
} def

```

Ensuite empennage, carénage et amortisseur sont réunis pour ne faire qu'un objet :

```

/aile_pied1 {empennage ammortisseur_carenage solidfuz} def

```

Avant dernière étape : on crée par rotation de $+120^\circ$ et -120° les deux autres pieds et on fusionne l'ensemble.

```

/aile_pied2 { aile_pied1 {0 0 120 rotate0point3d } solidtransform } def
/aile_pied3 { aile_pied1 {0 0 -120 rotate0point3d } solidtransform } def
aile_pied1 aile_pied2 solidfuz aile_pied3 solidfuz
(base-fusee) writesolidfile

```

La dernière ligne va écrire quatre fichiers contenant toutes les caractéristiques de la base de la fusée.

5 La fusée entière

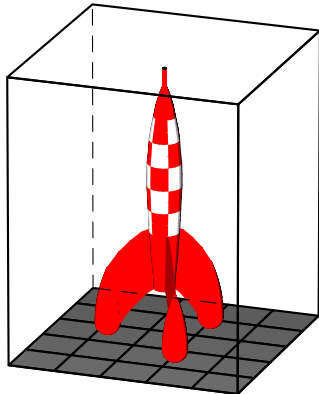
Pour enregistrer les caractéristiques complètes de la fusée il reste à fusionner le corps de la fusée avec sa base et à écrire sur le disque les fichiers :

```

\documentclass{article}
\usepackage{pst-solides3d}
\begin{document}
%% pour enregistrer les caractéristiques
%% complètes de la fusée
\codejps{
(base-fusee) readsolidfile
(corps-fusee) readsolidfile
solidfuz
(fusee-lunaire) writesolidfile
}
\composeSolid
\end{document}

```

6 Utilisation des données des fichiers obtenus : fusée de base



```

\psset{unit=0.2}
\begin{pspicture}(-5,0)(5,20)
\psset{lightsrc=100 20 20}
\psset[pst-solides3d]{SphericalCoord,
viewpoint=1000 20 20,Decran=800}
\psSolid[object=grille,base=-10 10 -10
10,ngrid=5](0,0,0)
\psSolid[object=datfile,file=fusee-
lunaire,grid](0,0,0)
\psSolid[object=parallelepiped,
a=20,b=20,c=25,action=draw
](0,0,12.5)
\end{pspicture}

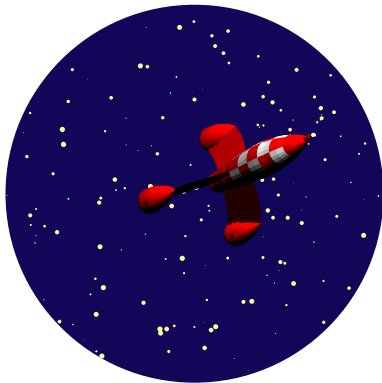
```

7 Utilisation des données de la fusée de base pour d'autres situations

Pour représenter la fusée dans d'autres positions on peut être tenté d'utiliser les paramètres classiques RotX, RotY et RotZ et le résultat obtenu, qui sera bien sûr conforme aux paramètres choisis, sera obtenu après un temps de calcul plus ou moins long. Pour remédier à cet inconvénient il est préférable de faire calculer tous les paramètres de la fusée dans la position souhaitée. Par exemple, la séquence suivante :

```
\documentclass{article}
\usepackage{pst-solides3d}
\begin{document}
\begin{pspicture}(-5,-5)(7,10)
\codejps{
(fusee-lunaire) readsolidfile
{0 60 30 rotate0point3d} solidtransform
(fusee1) writesolidfile
}
\composeSolid
\end{pspicture}
\end{document}
```

calcule toutes les caractéristiques de la fusée affichée ci-dessous :



```
1 \definecolor{BlueDark}{cmyk}{1,1,0,0.5}
2 \psset{unit=0.5}
3 \psset[pst-solides3d]{SphericalCoord,
4   viewpoint=1000 0 20,Decran=300}
5 \begin{pspicture}(-5,-5)(5,5)
6 \pscircle*[linecolor=BlueDark]{5}
7 \psRandomStar[linecolor=yellow!50,
8   randomPoints=200](-5,-5)(5,5){\
9   pscircle[linestyle=none]{5}}
10 \psset{lightsrc=100 20 100}
11 \psSolid[object=datfile,file=fusee1,
12   grid](0,0,0)
13 \end{pspicture}
```

8 Conclusion

Les quelques vignettes qui illustrent cet article ainsi que les images de la page :

<http://melusine.eu.org/lab/bpst/pst-solides3d/tintin>

sont inspirées de quelques cases de l'album de Hergé "Objectif Lune". Si vous souhaitez illustrer quelques cases du deuxième album "On a marché sur la Lune", faites-moi signe...