

## Programmation d'un phasemètre

Avec **Maple 11**, on peut aisément importer un tableau de données depuis Excel

on peut faciliter la sélection en nommant au préalable "tvv" la zone utile

puis lui attribuer ici un nom (par exemple "tvv")

$tvv := ImportData( )$

(1)

$$\left[ \begin{array}{l} 1..1024 \times 1..3 \text{ Array} \\ \text{Data Type: anything} \\ \text{Storage: rectangular} \\ \text{Order: Fortran_order} \end{array} \right]$$

$nn := ArrayTools[Size](tvv, 1)$

(2)

$$1024$$

on peut ensuite en extraire les deux signaux (ici en sautant la ligne de titre)

$ltv1 := [ ] :$

**for** ii **from** 1 **to** nn **do**  $ltv1 := [\text{op}(ltv1), [tvv[\text{ii}, 1], tvv[\text{ii}, 2]]]$  **end do:**

► **réimputation de la première série (par sécurité)**

$ltv2 := [ ] :$

**for** ii **from** 1 **to** nn **do**  $ltv2 := [\text{op}(ltv2), [tvv[\text{ii}, 1], tvv[\text{ii}, 3]]]$  **end do:**

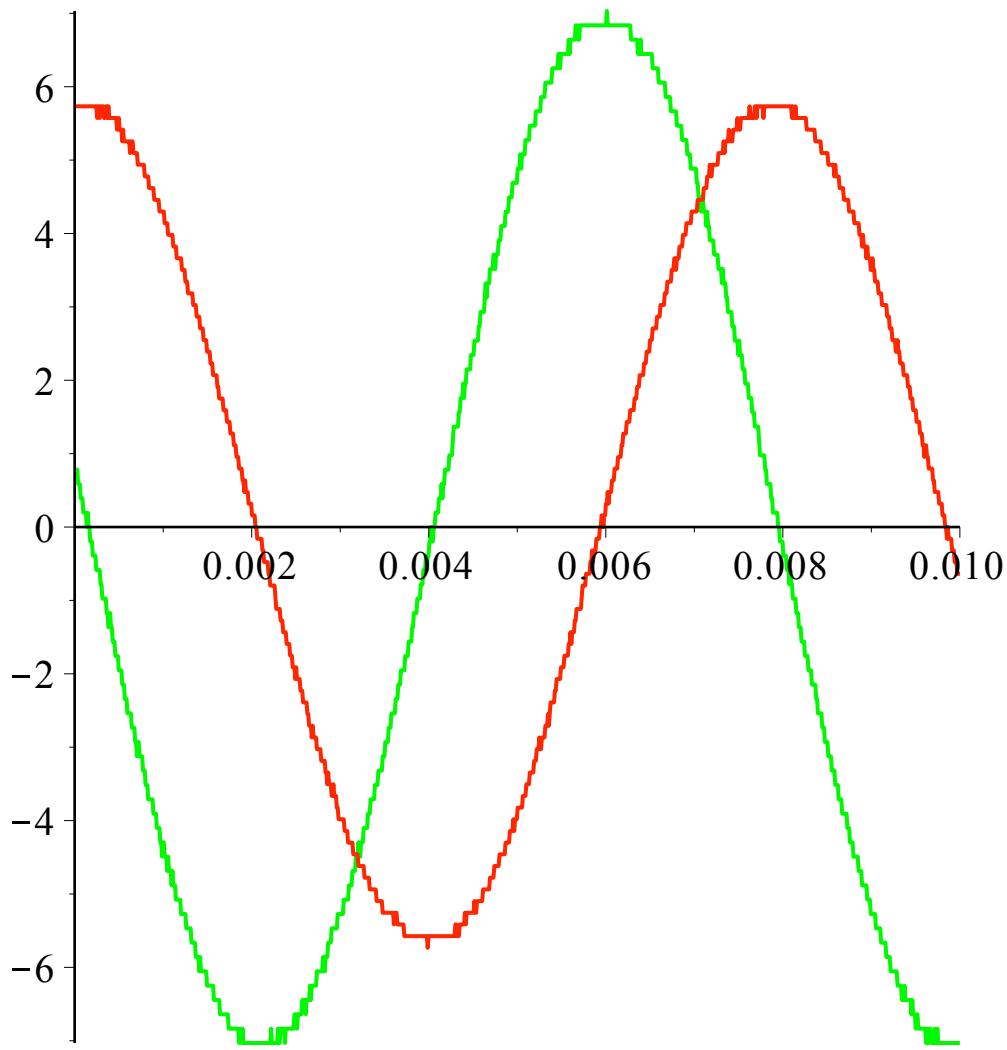
► **réimputation de la deuxième série (par sécurité)**

on peut alors vérifier les courbes (en prenant le temps d'imposer les couleurs pour identifier)

$c1 := \text{plot}(ltv1, \text{color} = \text{green}) :$

$c2 := \text{plot}(ltv2, \text{color} = \text{red}) :$

$\text{plots}[display](\{c1, c2\})$



## Programmation

On souhaite alors mettre au point une méthode de détermination automatique du déphasage  
 une première précaution consiste à compter le nombre de points  
 (si on veut pouvoir utiliser d'autres données...)

$$Np1 := \text{nops}(ltv1) \quad 1024 \quad (3)$$

$$Np2 := \text{nops}(ltv2) \quad 1024 \quad (4)$$

(il n'est pas indispensable de le limiter à  $Np1 = Np2...$ )

On peut ajouter ici un test pour refuser de calculer s'il n'y a pas assez de points  
 les données doivent comporter au moins deux périodes (sinon la mesure est douteuse)  
 chaque période doit comporter au moins 200 mesures (si on ne suppose pas une forme simple)  
 $phaseOK := \text{evalb}(\min(Np1, Np2) > 400)$

$$true \quad (5)$$

## **Les mesures comportent des fluctuations, un lissage préalable n'est pas inutile**

on suppose ici un lissage simple, considérant que les mesures sont à intervalles de temps égaux

(la méthode proposée ici suppose une moyenne d'un nombre impair de valeurs)

le lissage doit ne pas dépasser 5% d'une période... mais on ne sait pas encore leur nombre

$nliss := 7$

7

(6)

On prépare une liste filtrée

$$kliss := \frac{nliss - 1}{2}$$

3

(7)

```

lftv1 := [ ] : for k from kliss + 1 to Np1 - kliss do
tmoyen := ltv1[k, 1] :
vmoyen := ltv1[k, 2] :
for n from 1 to kliss do
tmoyen := tmoyen + ltv1[k - n, 1] + ltv1[k + n, 1] :
vmoyen := vmoyen + ltv1[k - n, 2] + ltv1[k + n, 2] :
end do:
tmoyen :=  $\frac{tmoyen}{nliss}$  :
vmoyen :=  $\frac{vmoyen}{nliss}$  :
lftv1 := [op(lftv1), [tmoyen, vmoyen]] :
end do:
Nfp1 := Np1 - nliss + 1

```

1018

(8)

```

lftv2 := [ ] : for k from kliss + 1 to Np2 - kliss do
tmoyen := ltv2[k, 1] :
vmoyen := ltv2[k, 2] :
for n from 1 to kliss do
tmoyen := tmoyen + ltv2[k - n, 1] + ltv2[k + n, 1] :
vmoyen := vmoyen + ltv2[k - n, 2] + ltv2[k + n, 2] :
end do:
tmoyen :=  $\frac{tmoyen}{nliss}$  :
vmoyen :=  $\frac{vmoyen}{nliss}$  :
lftv2 := [op(lftv2), [tmoyen, vmoyen]] :
end do:
Nfp2 := Np2 - nliss + 1

```

1018

(9)

on peut alors vérifier le lissage (en prenant le temps d'imposer les couleurs pour identifier)

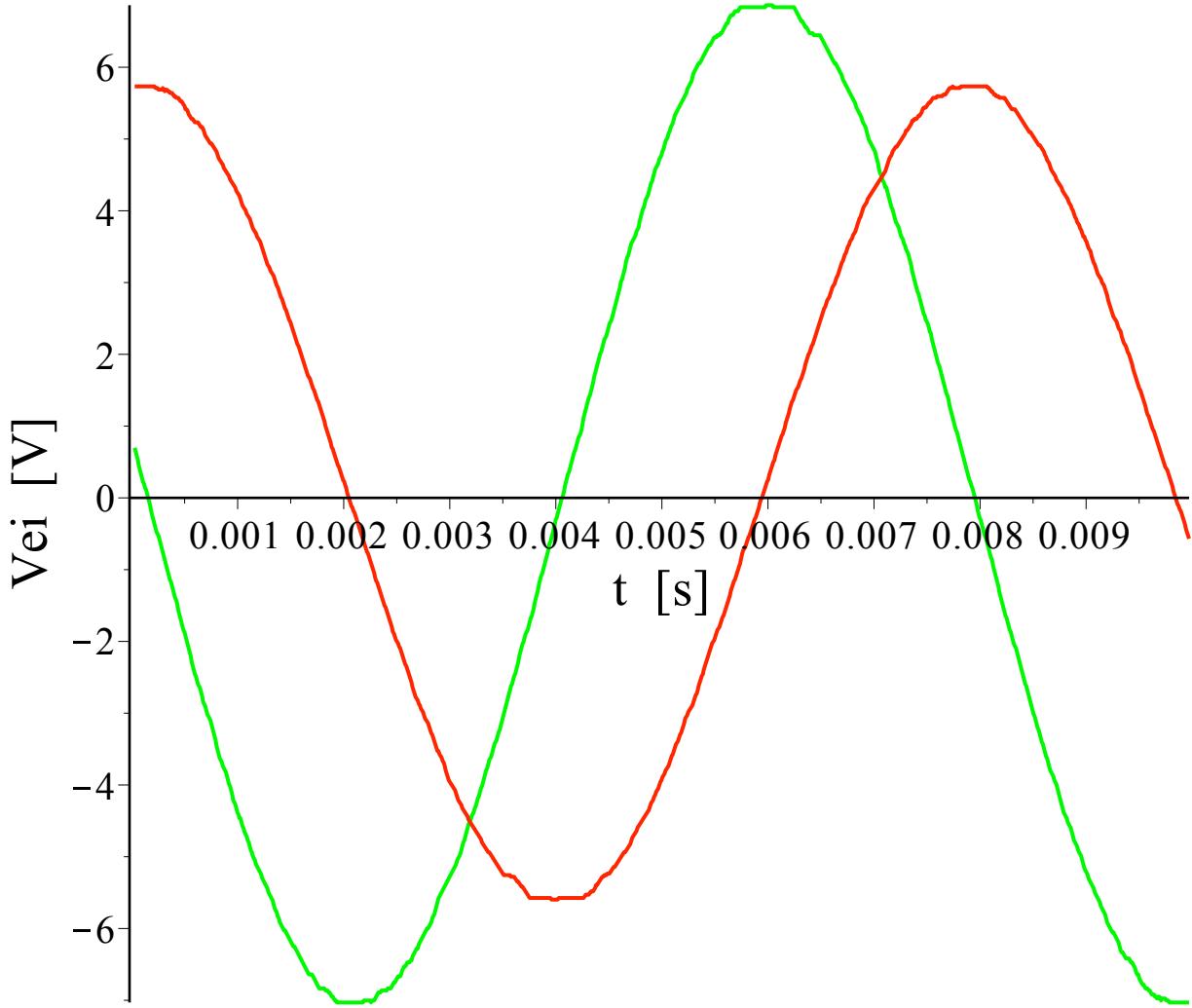
on constate qu'il subsiste quelques fluctuations au niveau des extrêums...

...mais ce n'est pas la zone critique pour le déclanchement

$c1 := plot(lftv1, color = green) :$

$c2 := plot(lftv2, color = red) :$

```
plots[display]( {c1, c2}, labels = [ "t [s]", "Vei [V]" ], labeldirections = [ horizontal, vertical], labelfont = [ "BOLD", 16 ] )
```



**Au cas où les courbes seraient décalées, on peut fixer un seuil de déclenchement**

`max1 := lftv1[1, 2]:`

`min1 := lftv1[1, 2]:`

`for ii from 2 to Nfp1 do`

`max1 := max( max1, lftv1[ii, 2] ): min1 := min( min1, lftv1[ii, 2] ) :`

`end do:`

`max1`

$$6.864411489 \quad (10)$$

`min1`

$$-7.030677797 \quad (11)$$

$$decl1 := \frac{(max1 + min1)}{2}$$

$$-0.083133154 \quad (12)$$

il est prudent de “recentrer” les deux signaux au cas où il y aurait un décalage

```

max2 := lftv2[1, 2] :
min2 := lftv2[1, 2] :
for ii from 2 to Nfp2 do
    max2 := max( max2, lftv2[ii, 2] ) : min2 := min( min2, lftv2[ii, 2] ) :
end do:
max2

```

$$5.733117269 \quad (13)$$

```

min2

```

$$-5.597536567 \quad (14)$$

```

decl2 :=  $\frac{(max2 + min2)}{2}$ 

```

$$0.067790350 \quad (15)$$

***Il faut chercher les dates de déclenchement... en choisissant le sens de variation...***

on décide de repérer le sens par  $\pm 1$

```

sensd1 := sign(decl1 - lftv1[1, 2])

```

$$-1 \quad (16)$$

```

sensd2 := sign(decl2 - lftv2[1, 2])

```

$$-1 \quad (17)$$

on cherche la première valeur dépassant le seuil dans le sens fixé

```

td1v1 := 0 :

```

```

ns1 := 1 :

```

```

while ns1 < Nfp1 and sensd1 · (decl1 - lftv1[ns1, 2]) > 0 do ns1 := ns1 + 1 end do:

```

```

td1v1 := lftv1[ns1, 1]

```

$$0.0001759530791 \quad (18)$$

on cherche à la suite le second déclenchement (pour mesurer la période)

```

td2v1 := 0 :

```

```

while ns1 < Nfp1 and sensd1 · (decl1 - lftv1[ns1, 2]) < 0 do ns1 := ns1 + 1 end do:

```

```

while ns1 < Nfp1 and sensd1 · (decl1 - lftv1[ns1, 2]) > 0 do ns1 := ns1 + 1 end do:

```

```

td2v1 := lftv1[ns1, 1]

```

$$0.007976539590 \quad (19)$$

```

period1 := td2v1 - td1v1

```

$$0.007800586511 \quad (20)$$

on procède de même pour le second signal

```

td1v2 := 0 :

```

```

ns2 := 1 :

```

```

while ns2 < Nfp2 and sensd2 · (decl2 - lftv2[ns2, 2]) > 0 do ns2 := ns2 + 1 end do:

```

```

td1v2 := lftv2[ns2, 1]

```

$$0.002043010753 \quad (21)$$

on cherche à la suite le second déclenchement (pour contrôler la période)

```

td2v2 := 0 :

```

```

while ns2 < Nfp2 and sensd2 · (decl2 - lftv2[ns2, 2]) < 0 do ns2 := ns2 + 1 end do:

```

```

while ns2 < Nfp2 and sensd2 · (decl2 - lftv2[ns2, 2]) > 0 do ns2 := ns2 + 1 end do:

```

```

td2v2 := lftv2[ns2, 1]

```

$$0.009833822091 \quad (22)$$

$$\begin{aligned} period2 &:= td2v2 - td1v2 \\ &\quad 0.007790811338 \end{aligned} \quad (23)$$

**On calcule le déphasage (en vérifiant la période pour tester s'il est plausible)**

$$\begin{aligned} test &:= \frac{period2}{period1} \\ &\quad 0.9987468669 \end{aligned} \quad (24)$$

si l'écart par rapport à 1 est assez faible, ça semble plausible

$$\begin{aligned} deltatemps &:= td1v2 - td1v1 \\ &\quad 0.001867057674 \end{aligned} \quad (25)$$

$$\begin{aligned} phase &:= 360 \cdot \frac{deltatemps}{period1} \\ &\quad 86.16541356 \end{aligned} \quad (26)$$

P.S.: si on veut inclure ce genre de calcul dans un logiciel plus complexe,  
on peut bien sûr mettre au propre des fonctions et procédures pour certaines tâches ;  
mais le but de cet exercice n'était que de réfléchir à la possibilité d'automatiser d'une mesure  
(lorsqu'on "appuie sur le bouton phasemètre" d'un oscilloscope numérique, que fait il ?)  
(pour bien utiliser un appareil "automatique", il n'est pas inutile de comprendre son fonctionnement)