

```

[ > restart:
[ #Options and Definitions for later use
[ > with (plots) :
[ > axesfonts:=axesfont=[TIMES,ROMAN,6]:
[ > labelfonts:=labelfont=[TIMES,ROMAN,12]:
[ > lightblue:=COLOR(RGB,0.196,0.6,0.8):
[ #Parameter Assignment
[ > s:=1/1: y:=1: rho:=0.05: r:=0.04: lambda:=0.025: g:=0.02:
[ > m:=s*rho+(1-s)*r:
[ #Dynamic System
[ > cdot:=diff(c(t),t)=c(t)*s*(r-g/s-rho-lambda+(lambda/(m)^(1/s))*c(t)/k(t)^(1/s));
[
[ 
$$c\dot{d}ot := \frac{d}{dt}c(t) = c(t) \left( -0.055 + \frac{0.5000000000 c(t)}{k(t)} \right)$$

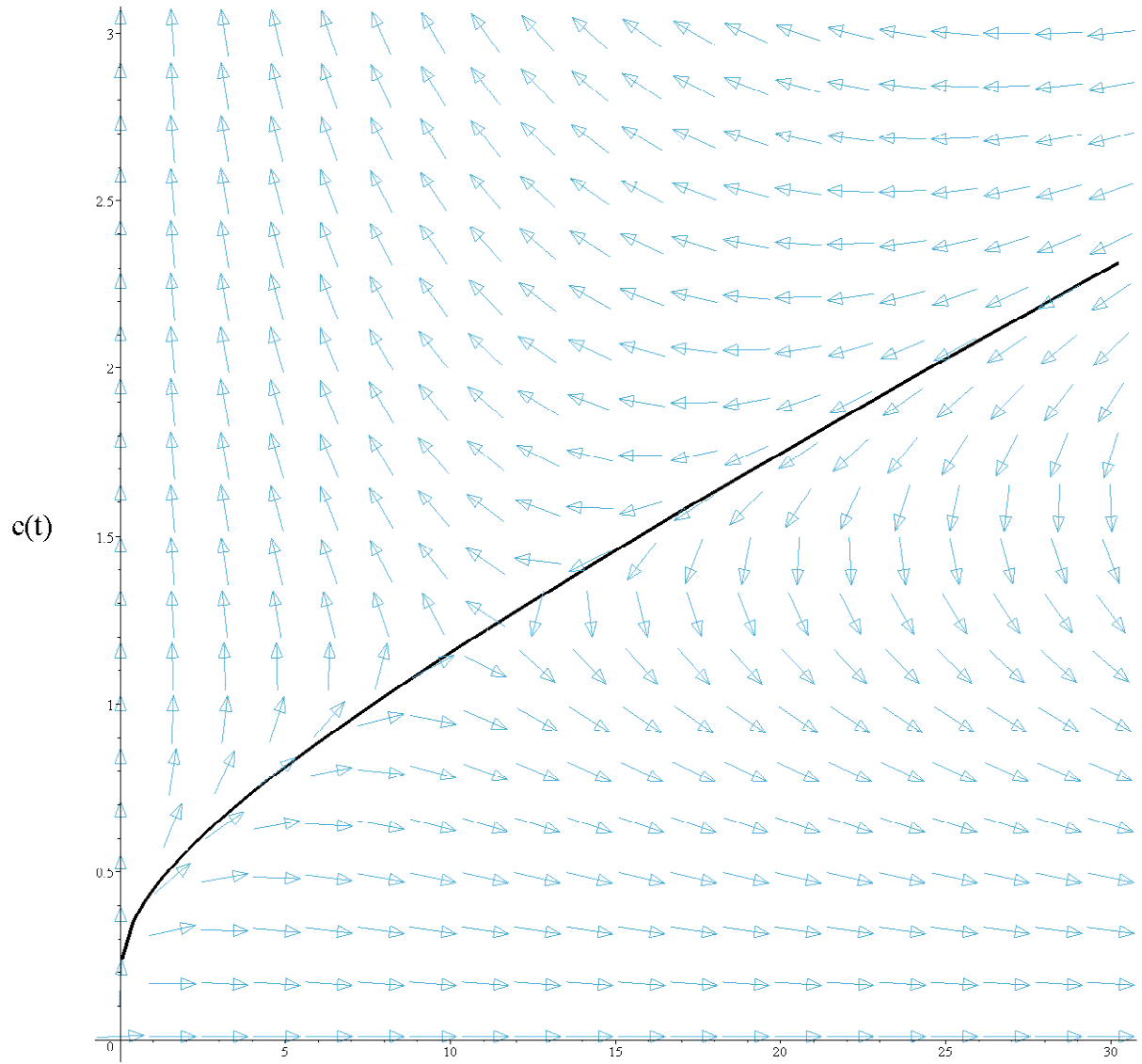
[ > kdot:=diff(k(t),t)=(r-g)*k(t)+y-c(t);
[
[ 
$$k\dot{d}ot := \frac{d}{dt}k(t) = 0.02 k(t) + 1 - c(t)$$

[ #Solving for a Stationary State (SS)
[ > eqc:=eval(cdot,{c(t)=c,k(t)=k}):
[ > eqk:=eval(kdot,{c(t)=c,k(t)=k}):
[ > systemss:=fsolve({eqc,eqk},{c,k},k=0.01..100);
[
[ 
$$systemss := \{c = 1.222222222, k = 11.11111111\}$$

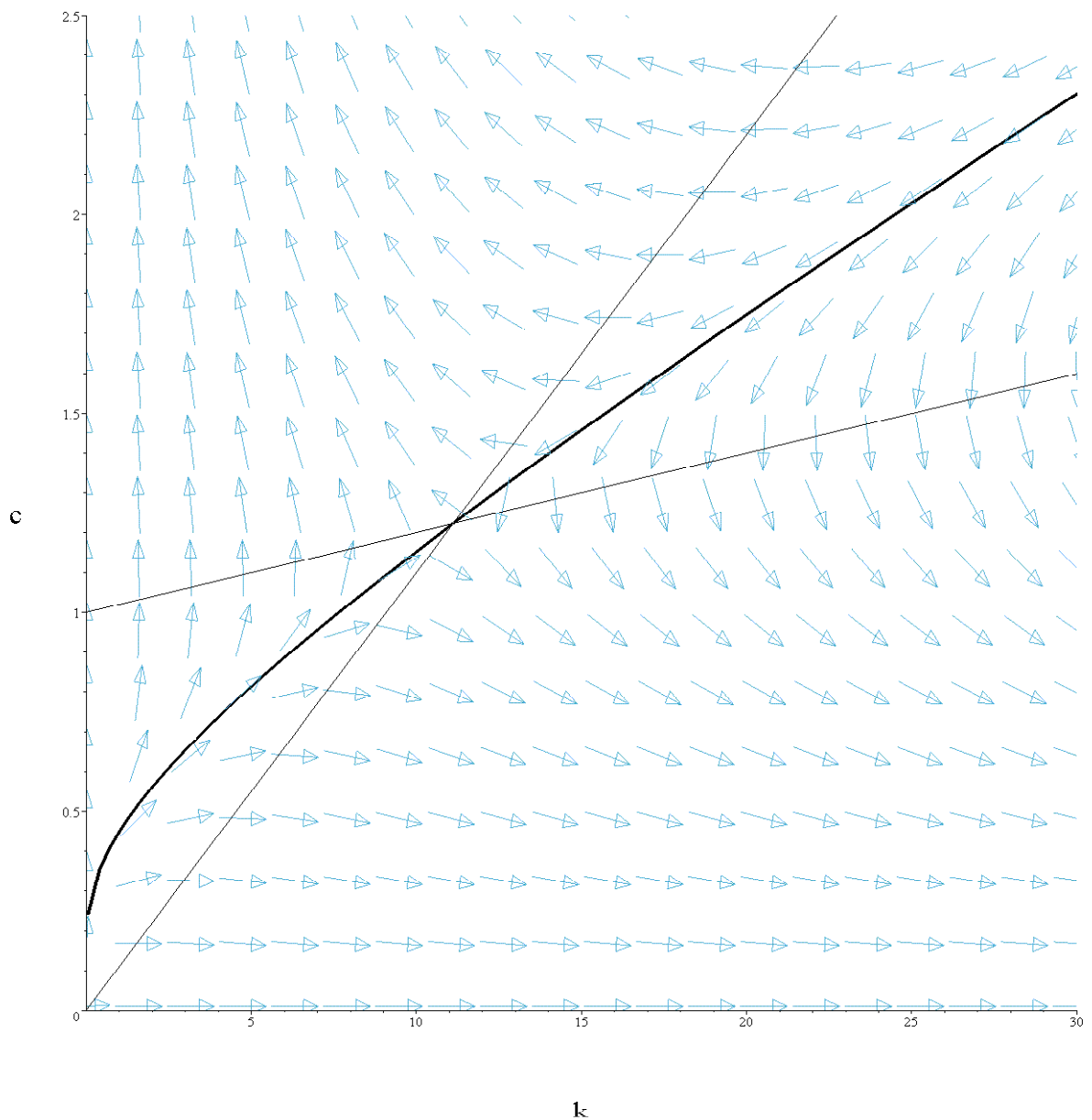
[ > css:=subs(systemss,c):
[ > kss:=subs(systemss,k):
[ #Plotting the Null-Clines (cdot=0 and kdot=0)
[ > nullcline_consumption:=m*((rho+lambda+g/s-r)/lambda)^(s)*k(t):
[ > nullcline_capital:=(r-g)*k(t)+y:
[ > nullclines:=plot({nullcline_consumption,nullcline_capital},k=0..40,c=0..3.0,colour=[black,black]):
[ #Plotting the Phase Diagram and stable manifold
[ > FNS:={c(t),k(t)}:
[ > SYS:={cdot,kdot}:
[ > DEtools[dfieldplot](SYS,FNS,t=0..50,scene=[k(t),c(t)],c=0.01..3,k=0.01..30,
[ colour=lightblue,arrows=slim):
[ > INITS:=[k(0)=0.99999*kss,c(0)=0.99999*css],[k(0)=1.0001*kss,c(0)=1.0001*css]:
[ > phase:=DEtools[phaseportrait](SYS,FNS,t=-261.9..0,INITS,scene=[k(t),c(t)],
[ k=0.01..30,c=0.01..3,stepsize=0.5,colour=lightblue,arrows=slim,linestyle=1,
[ linecolour=black,thickness=4,axesfonts,labelfonts):display(phase,title=`The
[ stable manifold`);

```

The stable manifold



```
> display({phase,nullclines},view=[0..30,0..2.5]);
```

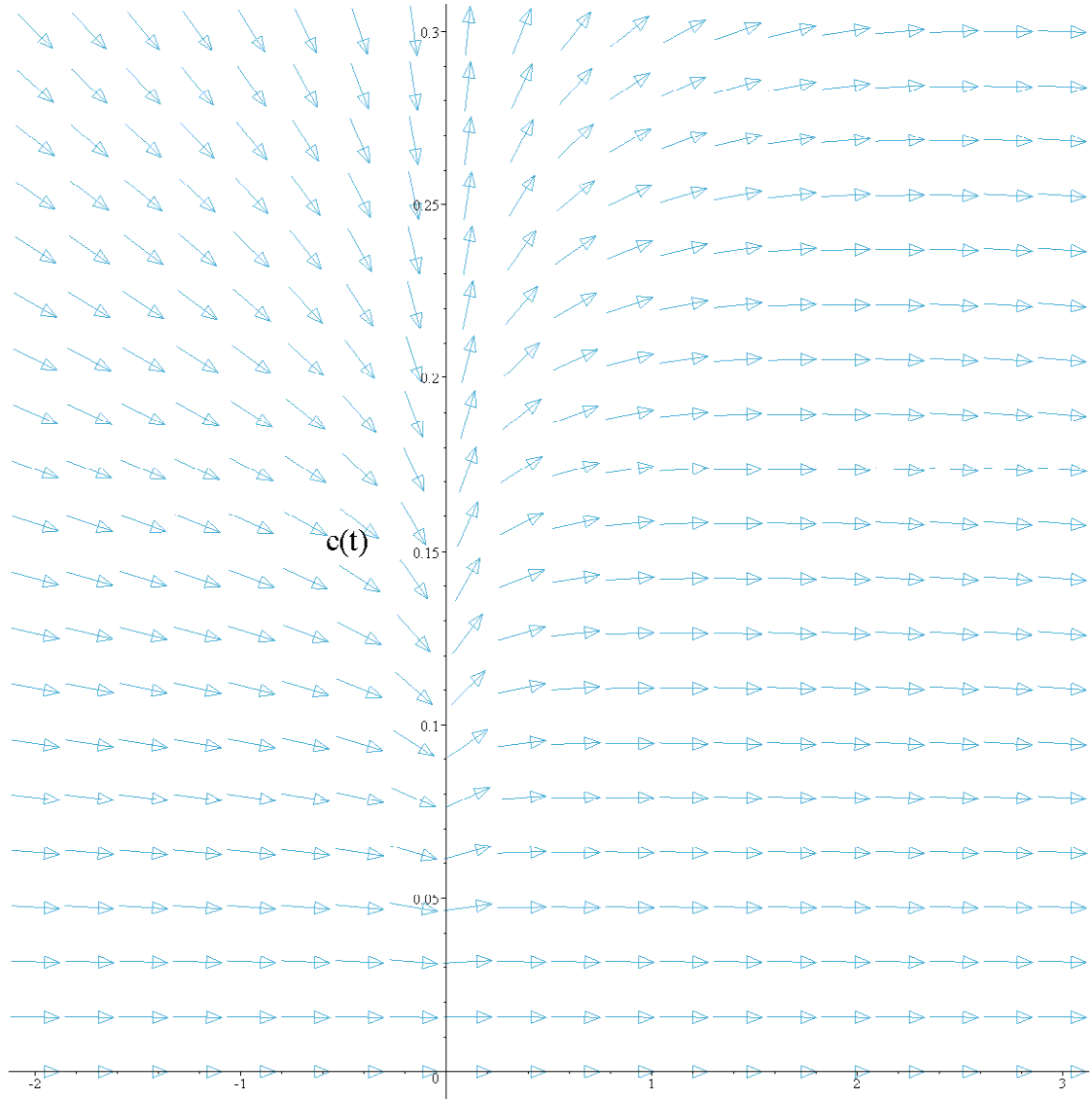


```

#Exporting the figure as a postscript figure
> plotsetup(ps,plotoutput=`Phase.ps`,plotoptions=`nocolor,portrait,noborder,axiswidth=50
0pt,axisheight=500`):
  display({phase,nullclines},view=[0..30,0..2.5],labels=[``,``]); plotsetup(default):
#Zooming on the origin to determine the nature of the intercept of the consumption function
> phase:=DEtools[phaseportrait](SYS, FNS, t=-300..0, INITS, scene=[k(t),c(t)], k=-2..3,
c=0..0.3, stepsize=0.5, colour=lightblue, arrows=slim, linestyle=1, linecolour=black,
thickness=4, axesfonts, labelfonts): display(phase, title=`Zooming on the intercept of
the "consumption function"`);

```

Zooming on the intercept of the "consumption function"



k