HawkE

August 20, 2020

[1]:
```
%pylab inline
```

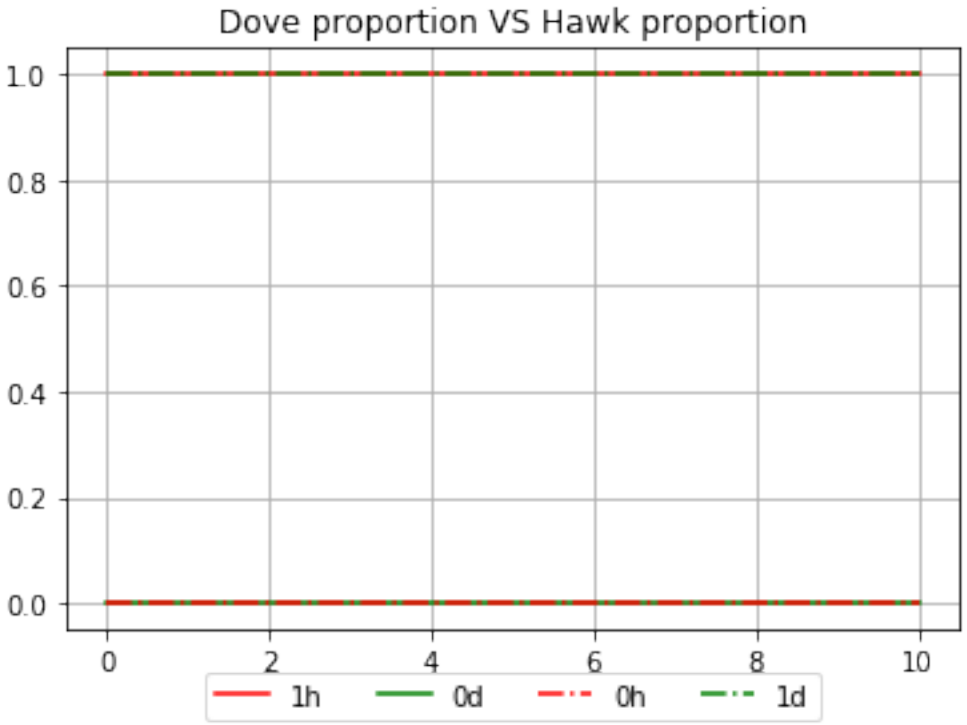
```
Populating the interactive namespace from numpy and matplotlib
```

[4]:
```
def Hawk_dove(v,c,h,d,h1,d1,g):
    #v= int(input("Enter resource: "))
    #c= int(input("Enter cost: "))
    #d= int(input("Enter number of doves: "))
    #h= int(input("Enter number of hawks: "))
    #g= int(input("How many generation: "))

    n = d+h          #Total population
    Xd= d/n          #Proportion of doves
    Xh= h/n          #Proportion of hawks
    darr=[Xd]
    harr=[Xh]

    n1 = d1+h1
    Xd1= d1/n1
    Xh1= h1/n1
    darr1=[Xd1]
    harr1=[Xh1]

    for t in range(g):
        fd = (Xd*(v/2)) + (Xh*0)              #Payoff for playing dove strategy
        fh = (Xh*((v-c)/2)) + (Xd*v)          #Payoff for plating hawk strategy
        f = (Xd*fd) + (Xh*fh)                 #Average fitness
        dchange = Xd*(fd-f)                   #Change in dove proportion
        hchange = Xh*(fh-f)                   #Change inn hawk proportion

        fd1 = (Xd1*(v/2)) + (Xh1*0)
        fh1 = (Xh1*((v-c)/2)) + (Xd1*v)
        f1 = (Xd1*fd1) + (Xh1*fh1)
        dchange1 = Xd1*(fd1-f1)
        hchange1 = Xh1*(fh1-f1)

        Xd += dchange
        Xh += hchange
```

```
        #darr.append(Xd)
        #harr.append(Xh)

        Xd1 += dchange1
        Xh1 += hchange1
        #darr1.append(Xd1)
        #harr1.append(Xh1)

        if(Xh>1):
            harr.append(1)
        elif(Xh<0):
            harr.append(0)
        else:
            harr.append(Xh)


        if(Xd>1):
            darr.append(1)      #if dove proprtion is more than 1, it takes the␣
↪value to be 1
        elif(Xd<0):
            darr.append(0)      #if it is less than 0, it takes the proportion to␣
↪be 0
        else:
            darr.append(Xd)     #Else it takes the real value, wwhich would␣
↪definitely be in between 1 and 0


        if(Xh1>1):
            harr1.append(1)
        elif(Xh1<0):
            harr1.append(0)
        else:
            harr1.append(Xh1)


        if(Xd1>1):
            darr1.append(1)      #if dove proprtion is more than 1, it takes the␣
↪value to be 1
        elif(Xd1<0):
            darr1.append(0)      #if it is less than 0, it takes the proportion␣
↪to be 0
        else:
            darr1.append(Xd1)     #Else it takes the real value, wwhich would␣
↪definitely be in between 1 and 0
```

```
    plot(harr, label = '1h' , color='r')
    plot(darr, label = '0d' , color='g')
    plt.plot(harr1, label = '0h',linestyle= '-.' ,color='r')
    plot(darr1, label = '1d',linestyle ='-.', color='g')
    plt.grid()
    title("Dove proportion VS Hawk proportion")
    legend(loc='upper center', bbox_to_anchor=(0.5, -0.05), ncol=4)
    #savefig(f"{g}.png")
    #clf()
```

[5]:
```
Hawk_dove(6,5,1,0,0,1,10)
```

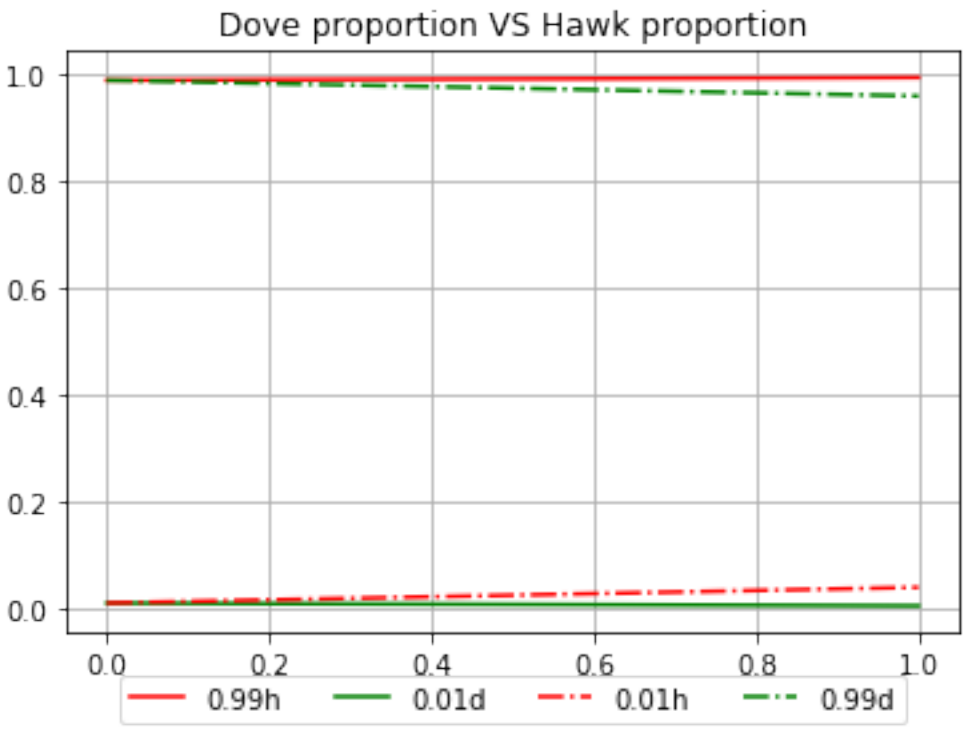


[8]:
```
def Hawk_dove(v,c,h,d,h1,d1,g):
    #v= int(input("Enter resource: "))
    #c= int(input("Enter cost: "))
    #d= int(input("Enter number of doves: "))
    #h= int(input("Enter number of hawks: "))
    #g= int(input("How many generation: "))
```

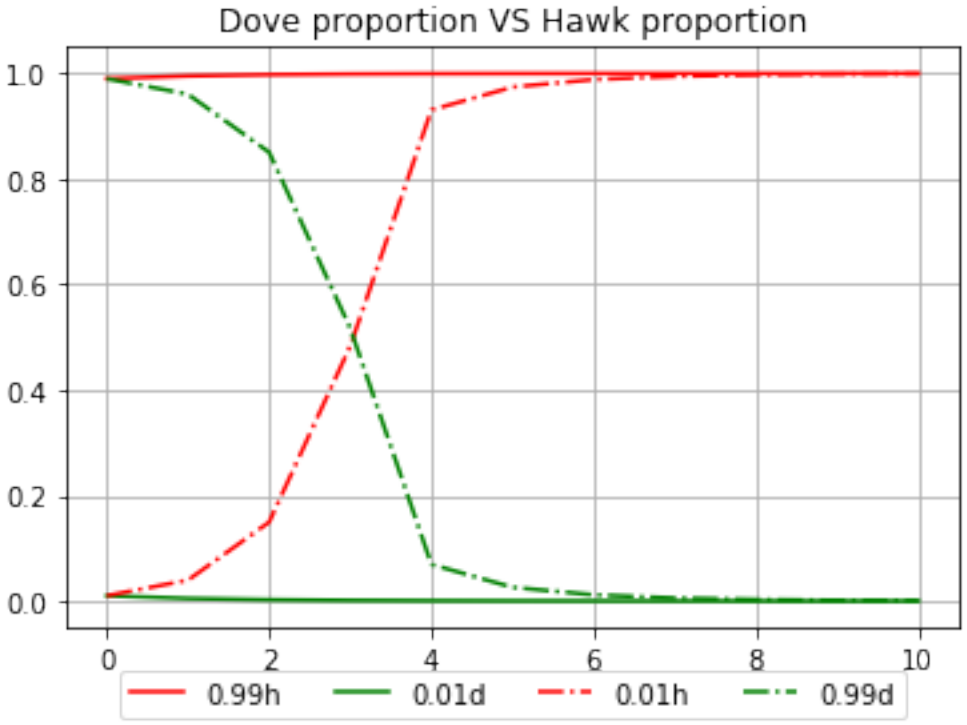
```
    n = d+h          #Total population
    Xd= d/n          #Proportion of doves
    Xh= h/n          #Proportion of hawks
    darr=[Xd]
    harr=[Xh]

    n1 = d1+h1
    Xd1= d1/n1
    Xh1= h1/n1
    darr1=[Xd1]
    harr1=[Xh1]

    for t in range(g):
        fd = (Xd*(v/2)) + (Xh*0)              #Payoff for playing dove strategy
        fh = (Xh*((v-c)/2)) + (Xd*v)          #Payoff for plating hawk strategy
        f = (Xd*fd) + (Xh*fh)                 #Average fitness
        dchange = Xd*(fd-f)                   #Change in dove proportion
        hchange = Xh*(fh-f)                   #Change inn hawk proportion

        fd1 = (Xd1*(v/2)) + (Xh1*0)
        fh1 = (Xh1*((v-c)/2)) + (Xd1*v)
        f1 = (Xd1*fd1) + (Xh1*fh1)
        dchange1 = Xd1*(fd1-f1)
        hchange1 = Xh1*(fh1-f1)

        Xd += dchange
        Xh += hchange



        Xd1 += dchange1
        Xh1 += hchange1



        if(Xh>1):
            harr.append(1)
        elif(Xh<0):
            harr.append(0)
        else:
            harr.append(Xh)


        if(Xd>1):
            darr.append(1)      #if dove proprtion is more than 1, it takes the␣
↪value to be 1
```

```
        elif(Xd1<0):
            darr.append(0)      #if it is less than 0, it takes the proportion to␣
↪be 0
        else:
            darr.append(Xd)     #Else it takes the real value, wwhich would␣
↪definitely be in between 1 and 0




        if(Xh1>1):
            harr1.append(1)
        elif(Xh1<0):
            harr1.append(0)
        else:
            harr1.append(Xh1)


        if(Xd1>1):
            darr1.append(1)      #if dove proprtion is more than 1, it takes the␣
↪value to be 1
        elif(Xd1<0):
            darr1.append(0)      #if it is less than 0, it takes the proportion␣
↪to be 0
        else:
            darr1.append(Xd1)     #Else it takes the real value, wwhich would␣
↪definitely be in between 1 and 0









    plot(harr, label = '0.99h' , color='r')
    plot(darr, label = '0.01d' , color='g')
    plt.plot(harr1, label = '0.01h',linestyle= '-.' ,color='r')
    plot(darr1, label = '0.99d',linestyle ='-.', color='g')
    plt.grid()
    title("Dove proportion VS Hawk proportion")
    legend(loc='upper center', bbox_to_anchor=(0.5, -0.05), ncol=4)
    #savefig(f"{g}.png")
    #clf()
```

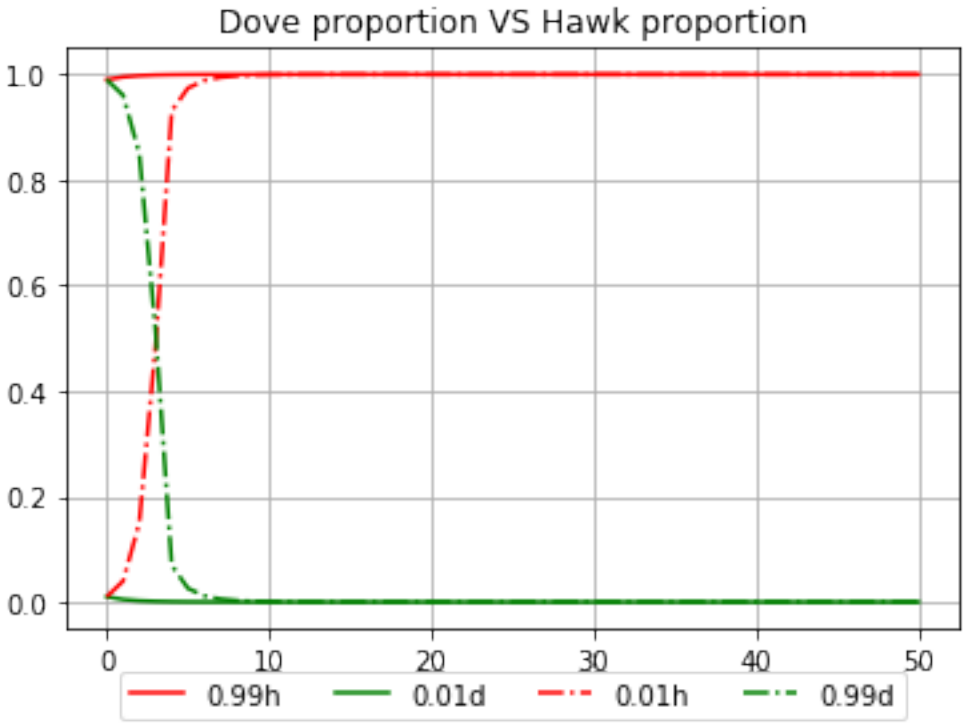
[9]:
```
Hawk_dove(6,5,0.99,0.01,0.01,0.99,1)
```



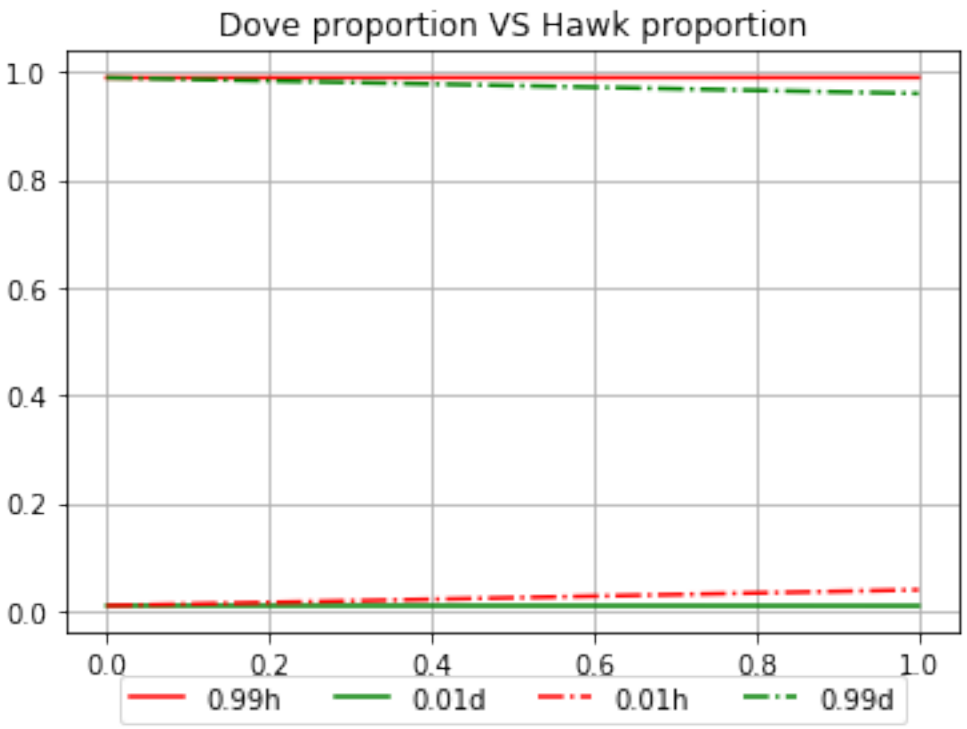
[10]:
```
Hawk_dove(6,5,0.99,0.01,0.01,0.99,10)
```



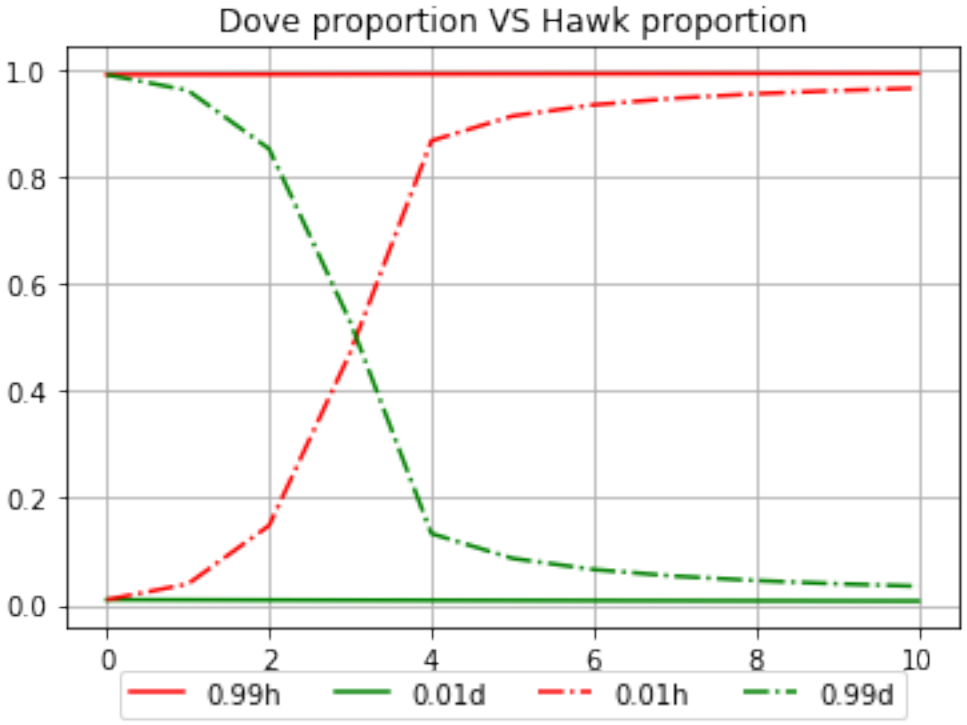
[11]:
```
Hawk_dove(6,5,0.99,0.01,0.01,0.99,50)
```



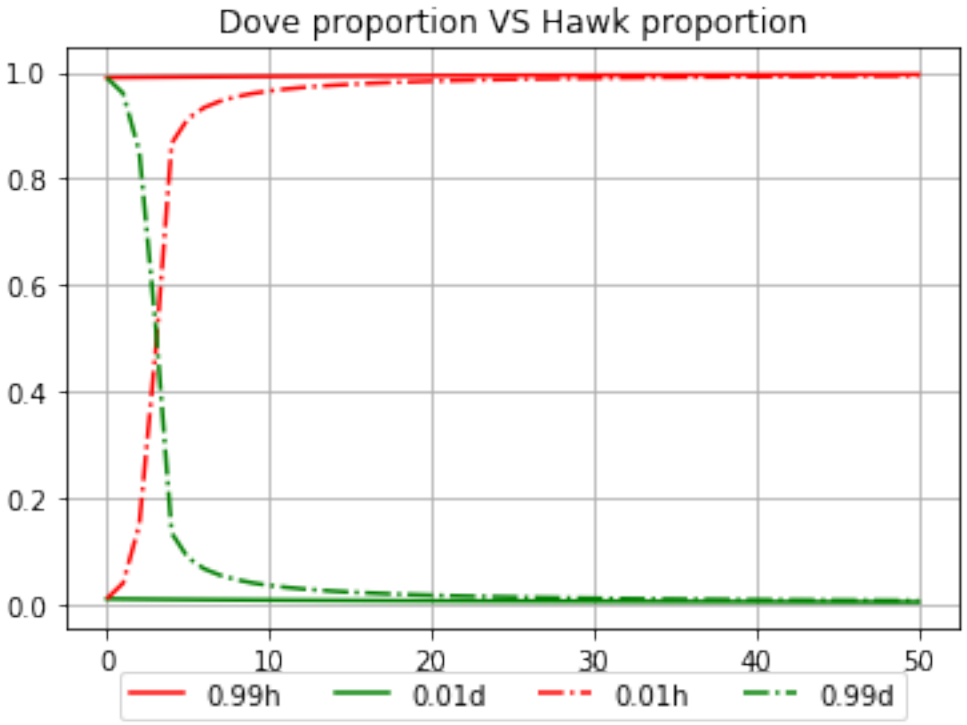
[12]:
```
Hawk_dove(6,6,0.99,0.01,0.01,0.99,1)
```



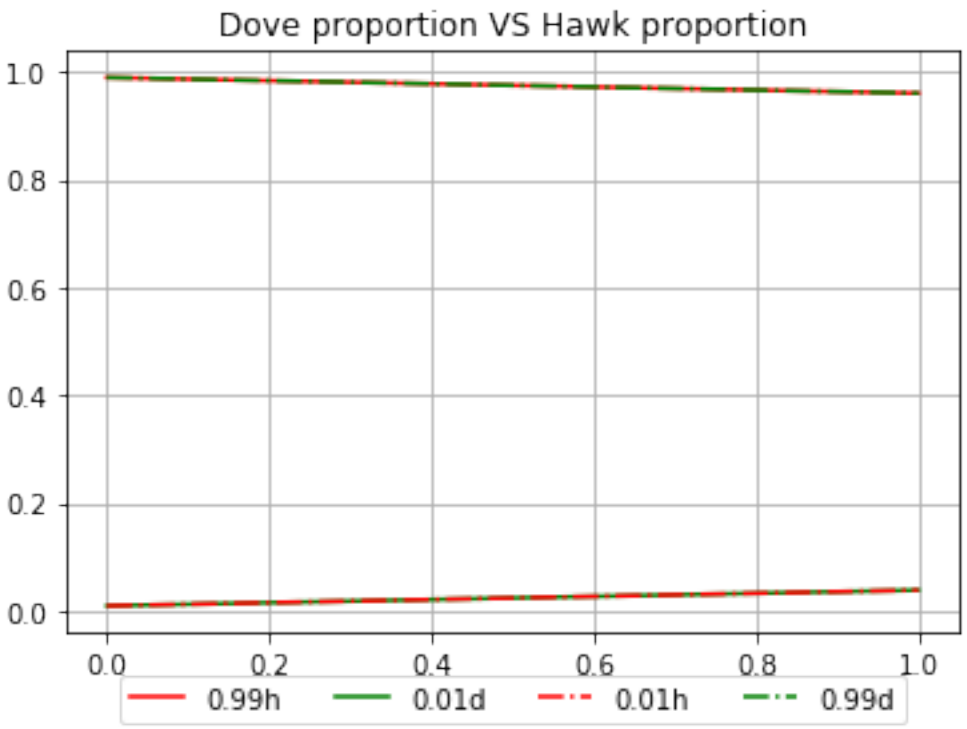
[13]:
```
Hawk_dove(6,6,0.99,0.01,0.01,0.99,10)
```



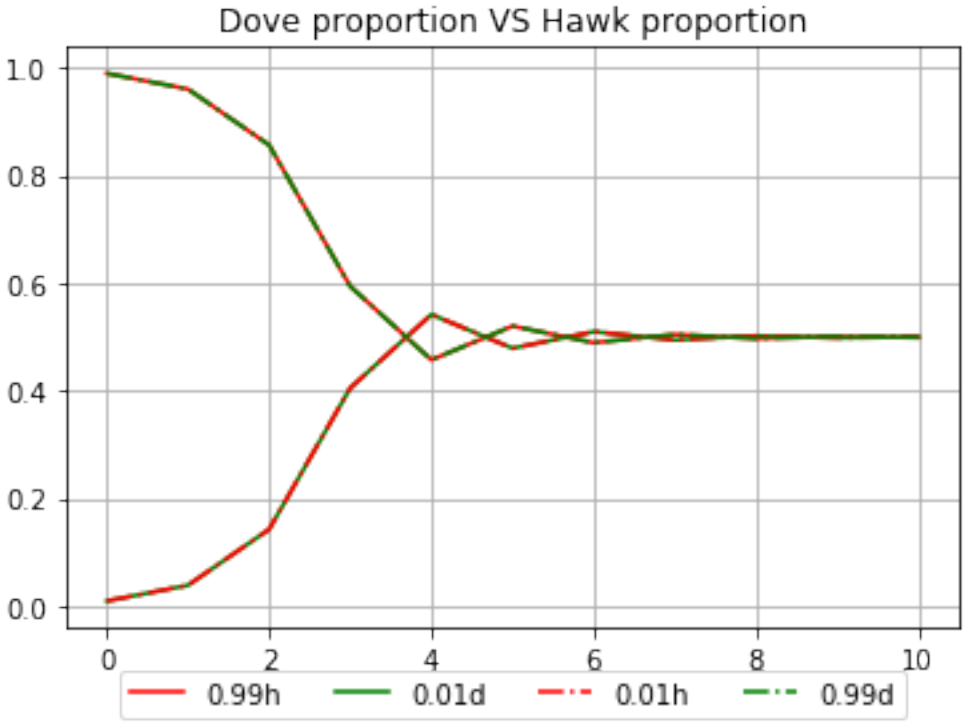
[14]:
```
Hawk_dove(6,6,0.99,0.01,0.01,0.99,50)
```



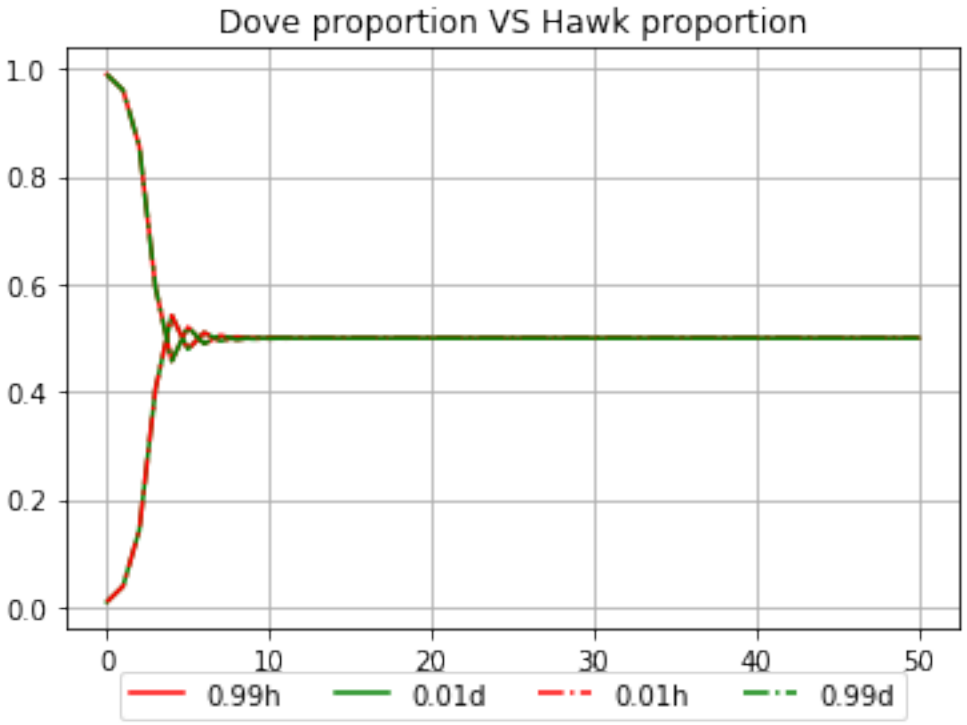
[15]:
```
Hawk_dove(6,12,0.99,0.01,0.01,0.99,1)
```



[16]:
```
Hawk_dove(6,12,0.99,0.01,0.01,0.99,10)
```



[17]:
```
Hawk_dove(6,12,0.99,0.01,0.01,0.99,50)
```



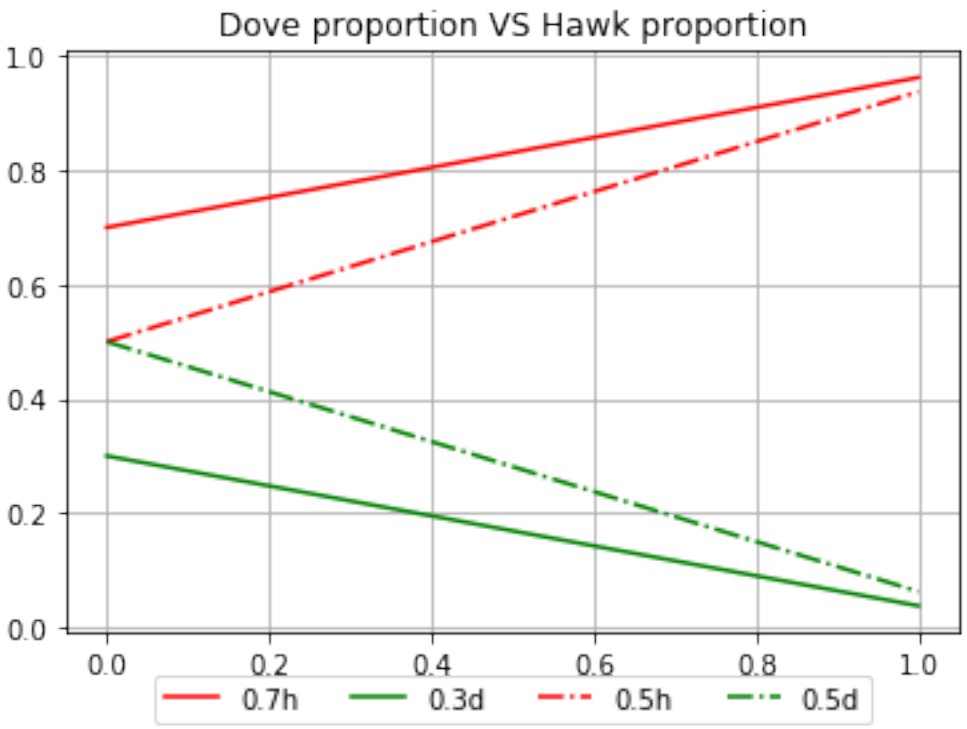
[ ]:
```
```

[18]:
```
def Hawk_dove(v,c,h,d,h1,d1,g):
    #v= int(input("Enter resource: "))
    #c= int(input("Enter cost: "))
    #d= int(input("Enter number of doves: "))
    #h= int(input("Enter number of hawks: "))
    #g= int(input("How many generation: "))

    n = d+h          #Total population
    Xd= d/n          #Proportion of doves
    Xh= h/n          #Proportion of hawks
    darr=[Xd]
    harr=[Xh]

    n1 = d1+h1
    Xd1= d1/n1
    Xh1= h1/n1
    darr1=[Xd1]
```

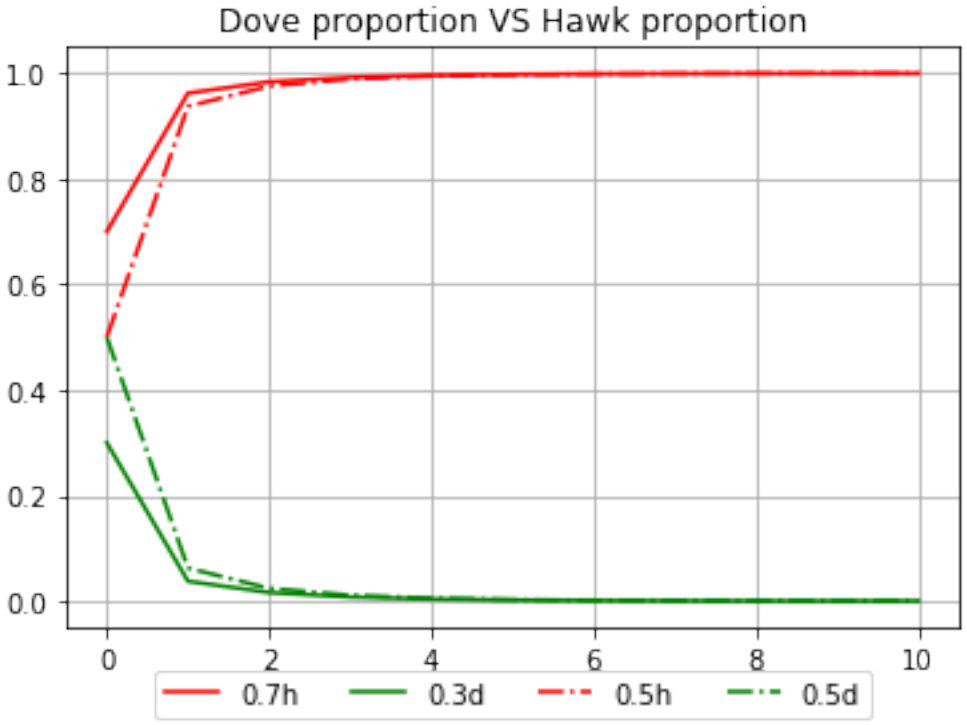
```
    harr1=[Xh1]

    for t in range(g):
        fd = (Xd*(v/2)) + (Xh*0)              #Payoff for playing dove strategy
        fh = (Xh*((v-c)/2)) + (Xd*v)          #Payoff for plating hawk strategy
        f = (Xd*fd) + (Xh*fh)                 #Average fitness
        dchange = Xd*(fd-f)                   #Change in dove proportion
        hchange = Xh*(fh-f)                   #Change inn hawk proportion

        fd1 = (Xd1*(v/2)) + (Xh1*0)
        fh1 = (Xh1*((v-c)/2)) + (Xd1*v)
        f1 = (Xd1*fd1) + (Xh1*fh1)
        dchange1 = Xd1*(fd1-f1)
        hchange1 = Xh1*(fh1-f1)

        Xd += dchange
        Xh += hchange
        #darr.append(Xd)
        #harr.append(Xh)

        Xd1 += dchange1
        Xh1 += hchange1
        #darr1.append(Xd1)
        #harr1.append(Xh1)


        if(Xh>1):
            harr.append(1)
        elif(Xh<0):
            harr.append(0)
        else:
            harr.append(Xh)


        if(Xd>1):
            darr.append(1)      #if dove proprtion is more than 1, it takes the␣
↪value to be 1
        elif(Xd1<0):
            darr.append(0)      #if it is less than 0, it takes the proportion to␣
↪be 0
        else:
            darr.append(Xd)     #Else it takes the real value, wwhich would␣
↪definitely be in between 1 and 0
```

```
        if(Xh1>1):
            harr1.append(1)
        elif(Xh1<0):
            harr1.append(0)
        else:
            harr1.append(Xh1)


        if(Xd1>1):
            darr1.append(1)      #if dove proprtion is more than 1, it takes the␣
↪value to be 1
        elif(Xd1<0):
            darr1.append(0)      #if it is less than 0, it takes the proportion␣
↪to be 0
        else:
            darr1.append(Xd1)    #Else it takes the real value, wwhich would␣
↪definitely be in between 1 and 0









    plot(harr, label = '0.7h' , color='r')
    plot(darr, label = '0.3d' , color='g')
    plt.plot(harr1, label = '0.5h',linestyle= '-.' ,color='r')
    plot(darr1, label = '0.5d',linestyle ='-.', color='g')
    plt.grid()
    title("Dove proportion VS Hawk proportion")
    legend(loc='upper center', bbox_to_anchor=(0.5, -0.05), ncol=4)
    #savefig(f"{g}.png")
    #clf()
```

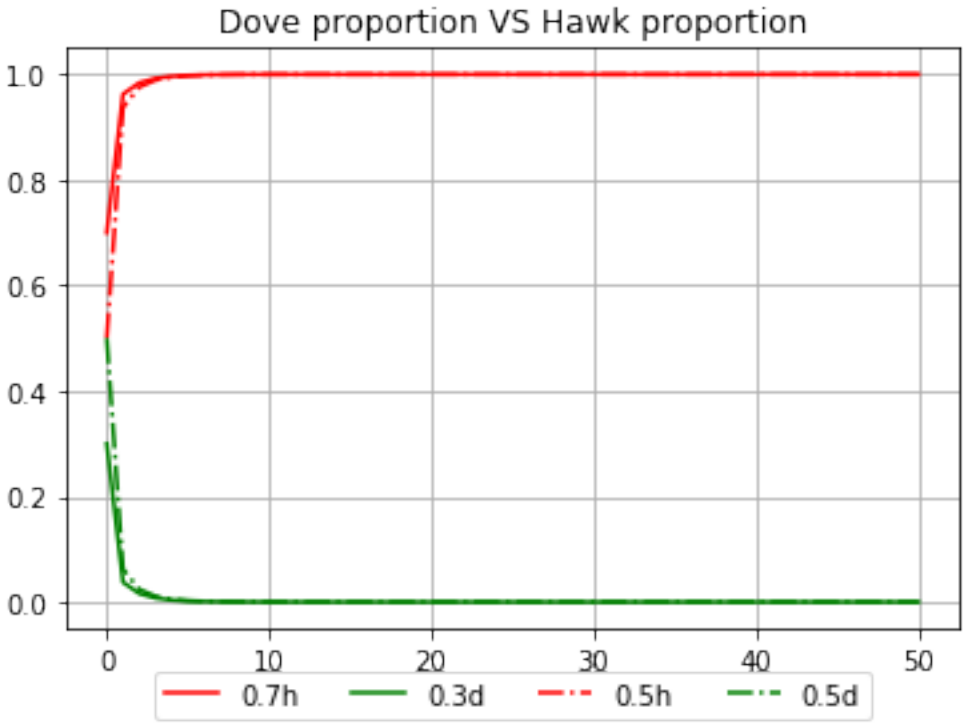
[19]:
```
Hawk_dove(6,5,0.7,0.3,0.5,0.5,1)
```



[20]:
```
Hawk_dove(6,5,0.7,0.3,0.5,0.5,10)
```

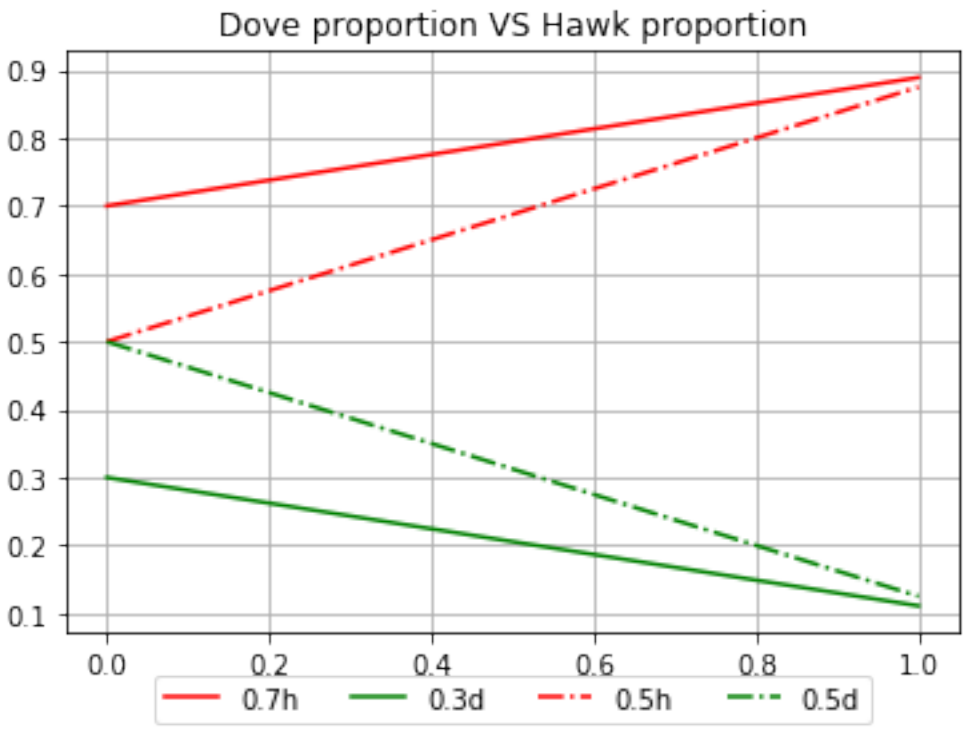


[21]:
```
Hawk_dove(6,5,0.7,0.3,0.5,0.5,50)
```

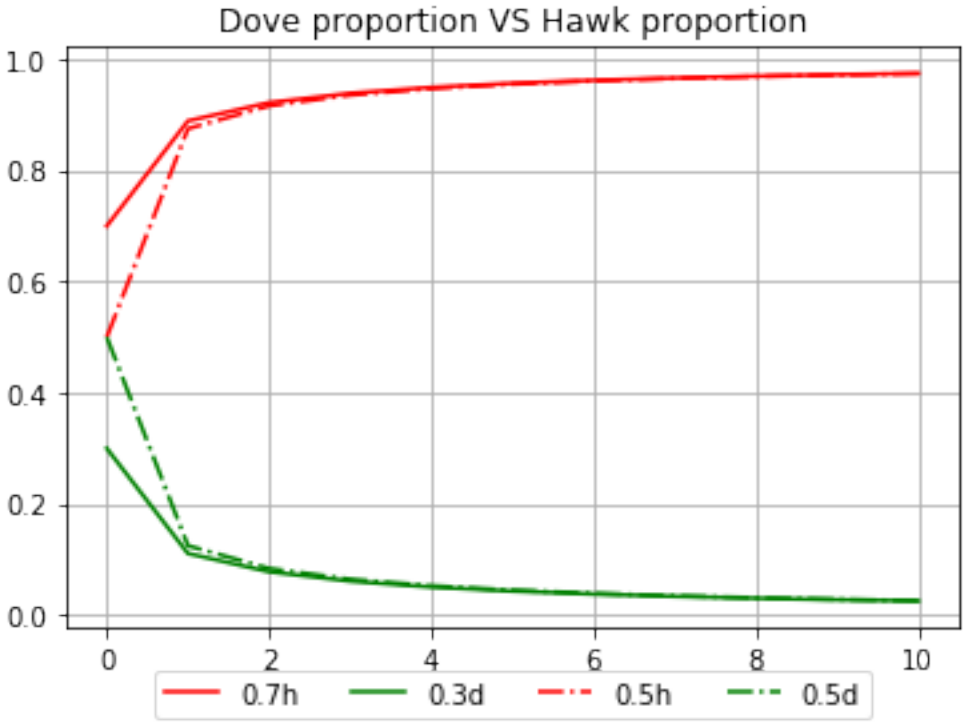


[ ]:
```

```

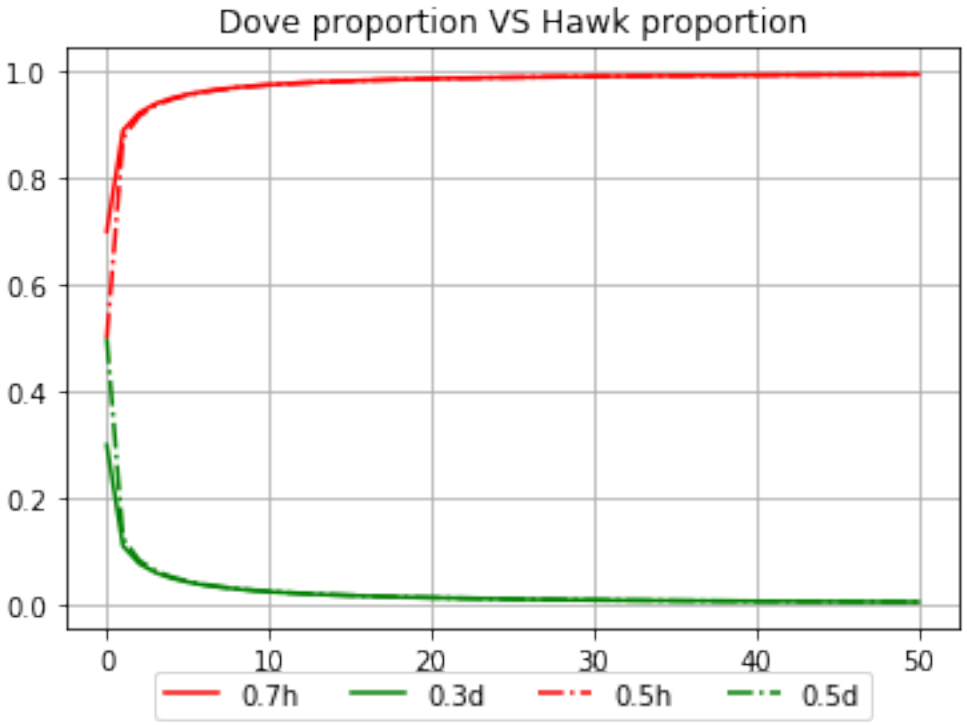
[22]:
```
Hawk_dove(6,6,0.7,0.3,0.5,0.5,1)
```



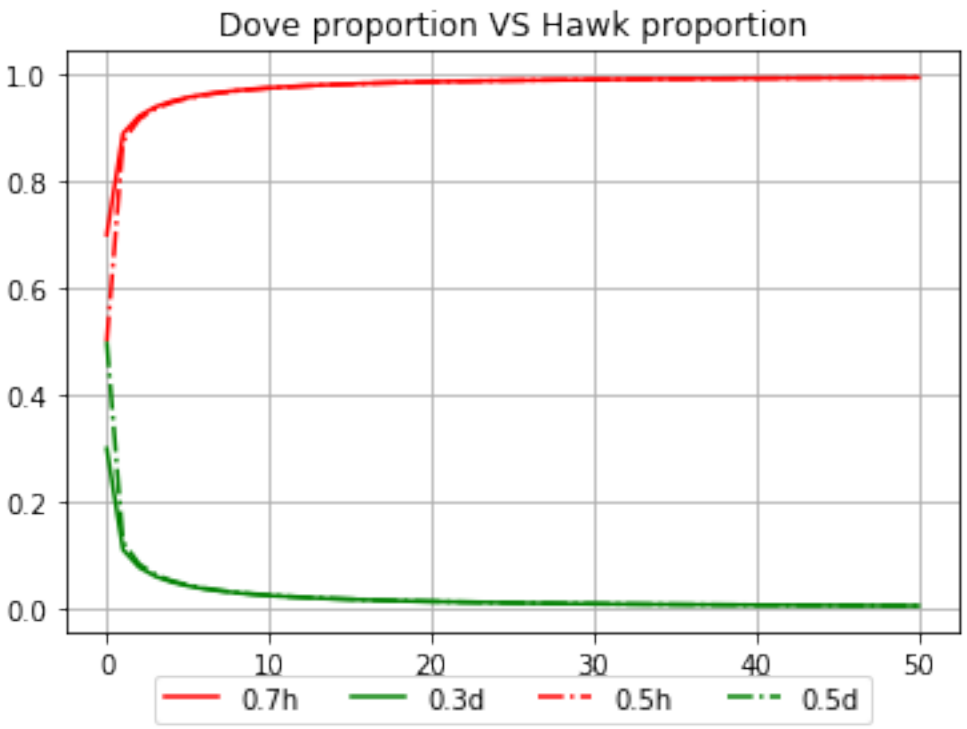
[23]:
```
Hawk_dove(6,6,0.7,0.3,0.5,0.5,10)
```



[24]:
```
Hawk_dove(6,6,0.7,0.3,0.5,0.5,50)
```

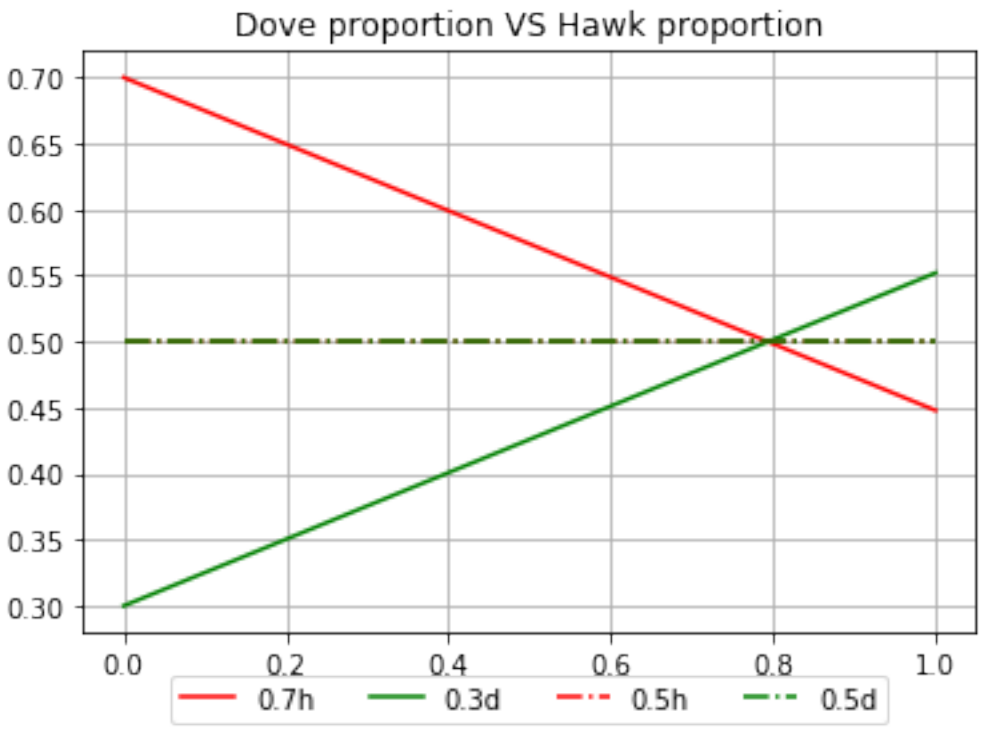


[25]:
```
Hawk_dove(6,6,0.7,0.3,0.5,0.5,50)
```

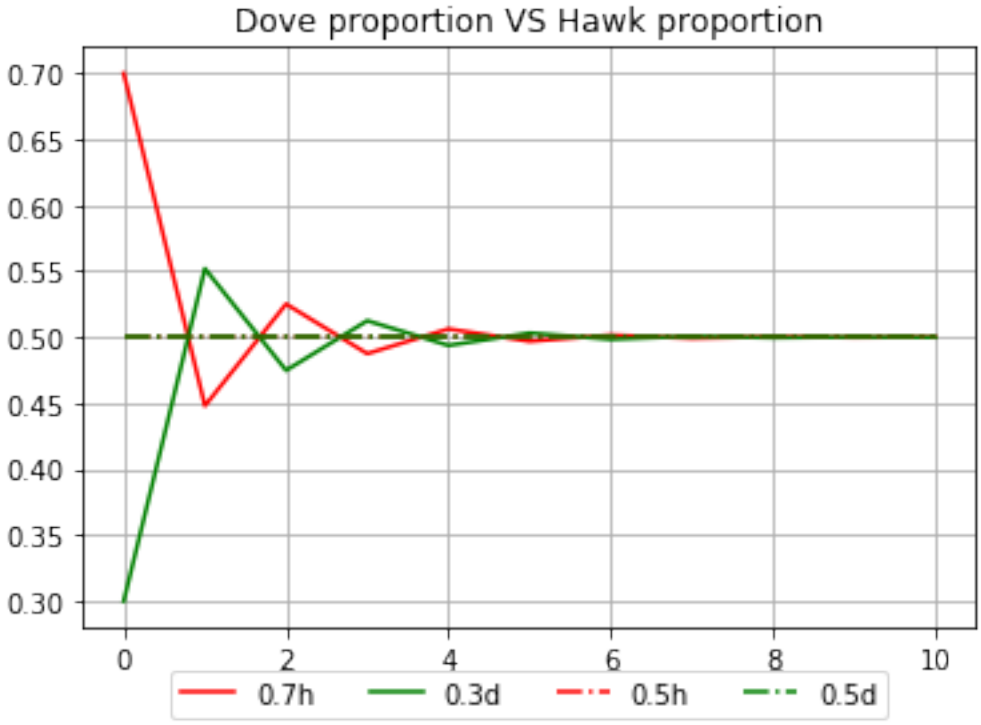


[ ]:

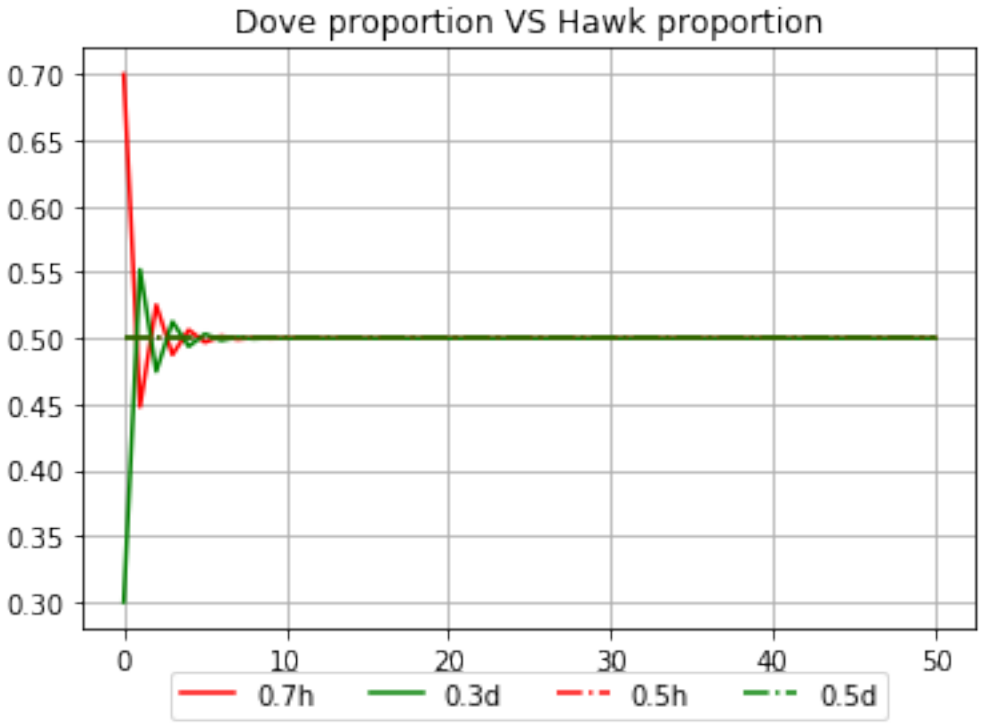
[26]:
```
Hawk_dove(6,12,0.7,0.3,0.5,0.5,1)
```



[27]:
```
Hawk_dove(6,12,0.7,0.3,0.5,0.5,10)
```



[28]:
```
Hawk_dove(6,12,0.7,0.3,0.5,0.5,50)
```



[ ]:

[ ]: