Example 6.

Sensitivity of hydraulic head at a point to Cauchy BC head under steady state flow conditions

0. Forward model

Governing equation:

$$K\,b\,\frac{d^2h}{dx^2} + R = 0 \tag{1}$$

(2)

Boundary conditions:

$$-K\,b\,\frac{dh(x)}{dx} = 0\,, \qquad x = 0 \tag{3}$$

$$-K\,b\,\frac{dh(x)}{dx} = C\,[h^* - h(x)]\,, \qquad x = L \tag{4}$$

(5)

Closed-form solution:

$$\text{Not available} \tag{6}$$

(7)

Spatial derivatives from differentiation:

$$\text{Not available} \tag{8}$$

In [52]:
```
from IPython.display import HTML, display
def set_background(color):
    script = (
        "var cell = this.closest('.code_cell');"
        "var editor = cell.querySelector('.input_area');"
        "editor.style.background='{}';"
        "this.parentNode.removeChild(this)").format(color)
    display(HTML('<img src onerror="{}">'.format(script)))
```

In [81]:
```
from warnings import filterwarnings
filterwarnings("ignore", category=DeprecationWarning)

import numpy as np

K, R, b, L, BC3h, BC3c, ocol = 10., 1e-1/1000., 10., 10000., 1., 1., 5000
X = np.arange(L)
```

1. Direct sensitivity

$$\text{Not available} \tag{9}$$

2. Perturbation sensitivity

(10)

$$\frac{\partial h(x')}{\partial h_{\Gamma_3}} \approx \frac{h(x, h_{\Gamma_3} + \Delta h_{\Gamma_3}) - h(x, h_{\Gamma_3})}{\Delta h_{\Gamma_3}} \tag{11}$$

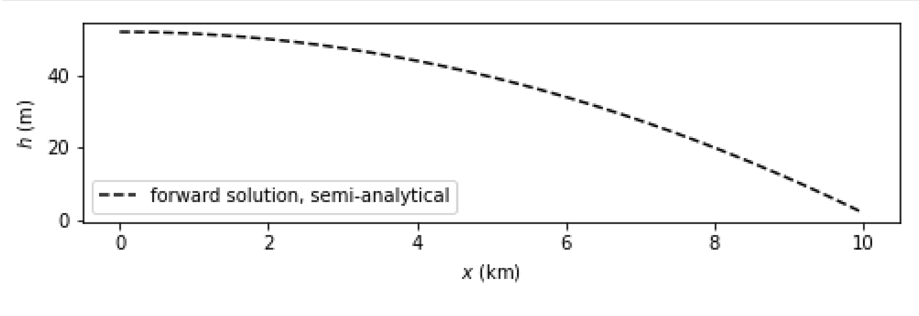
(12)

2a. Analytical

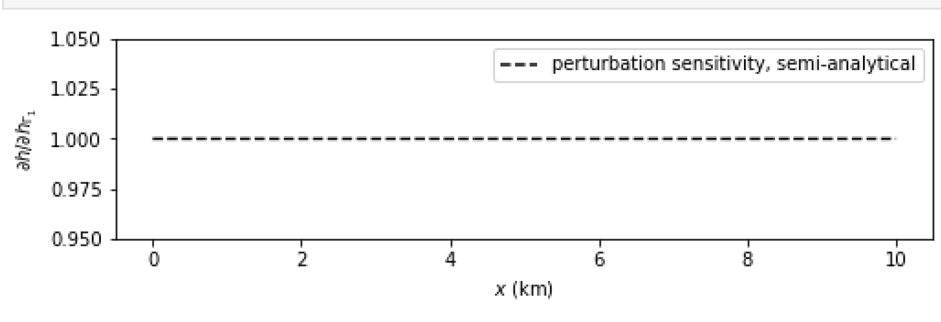
$$\text{Not available} \tag{13}$$

2b. Semi-analytical

In [101…
```
from os import getcwd, chdir
cwd = getcwd()
chdir(r'../../timml')
import timml
chdir(cwd)

M0 = timml.Model3D(kaq=K, z=[0., -b])
timml.ImpLineDoublet1D(M0, xld=0.)
timml.HeadLineSink1D(M0, xls=L, hls=BC3h, res=BC3c/100., wh=1.)
timml.StripAreaSink(M0, 0., L, R)
M0.solve(silent=True)
H0 = M0.headalongline(X, 0.).flatten()

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X/1000., H0, 'k--', mfc='none', label='forward solution, semi-analytical')
plt.xlabel('$x$ (km)')
plt.ylabel('$h$ (m)')
plt.legend(loc=3);
```



In [102…
```
M1 = timml.Model3D(kaq=K, z=[0., -b])
timml.ImpLineDoublet1D(M1, xld=0.)
timml.HeadLineSink1D(M1, xls=L, hls=BC3h+BC3h*dpar, res=BC3c/100., wh=1.)
timml.StripAreaSink(M1, 0., L, R)
M1.solve(silent=True)
H1 = M1.headalongline(X, 0.).flatten()
dhdBC3h = (H1-H0)/(BC3h*dpar)

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X/1000., dhdBC3h, 'k--', mfc='none', label='perturbation sensitivity, semi-analytical')
plt.xlabel('$x$ (km)')
plt.ylabel(r'$\partial h / \partial h_{\Gamma_1}$')
plt.legend()
plt.ylim(0.95, 1.05);
```

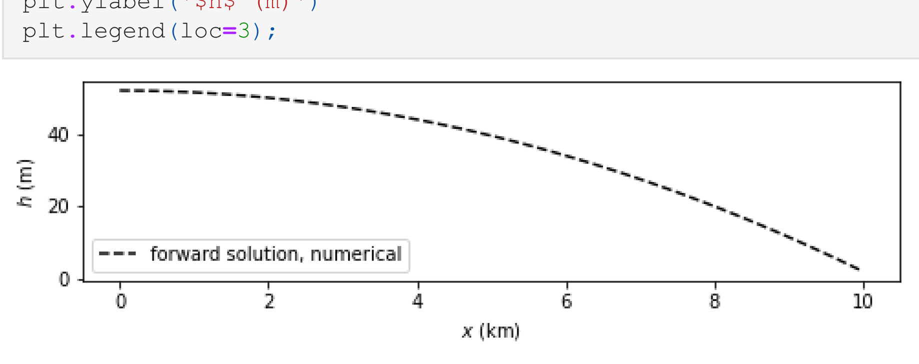


In [103…
```
set_background('rgba(0, 200, 0, 0.2)')
print('%.6f'% dhdBC3h[ocol])
```

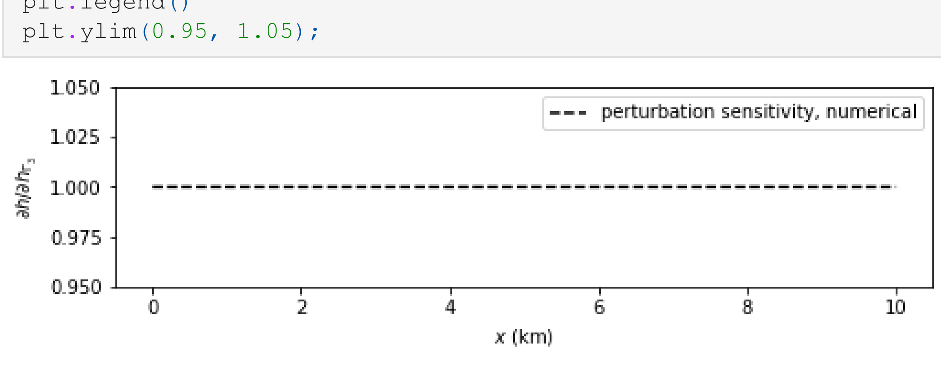
```
1.000000
```

2c. Numerical

In [56]:
```
import flopy

dpar = 1e-0
nrow, ncol = 1, int(L)
M0 = flopy.modflow.Modflow(modelname='model', exe_name='../mf2005.exe')
flopy.modflow.ModflowDis(M0, nlay=1, nrow=nrow, ncol=ncol, nper=1, delr=1., delc=1., top=0., botm=-b, steady=Tr
                         perlen=1., nstp=1)
flopy.modflow.ModflowBas(M0, ibound=np.ones([nrow, ncol]), strt=BC3h*np.ones([nrow, ncol], dtype=float))
flopy.modflow.ModflowLpf(M0, hk=K, vka=-999., ss=-999., sy=-999., ipakcb=53)
flopy.modflow.ModflowRch(M0, nrchop=1, rech=R, ipakcb=53)
flopy.modflow.ModflowGhb(M0, stress_period_data={0: [0, 0, ncol-1, BC3h, BC3c]})
flopy.modflow.ModflowPcg(M0, hclose=1e-6, rclose=1e-6)
flopy.modflow.ModflowOc(M0, stress_period_data={(0,0): ['save head', 'save budget']})
M0.write_input()
success, buff = M0.run_model(silent=True)
H0 = flopy.utils.binaryfile.HeadFile('model.hds').get_data()[0,0,:]

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X/1000., H0, 'k--', mfc='none', label='forward solution, numerical')
plt.xlabel('$x$ (km)')
plt.ylabel('$h$ (m)')
plt.legend(loc=3);
```



In [57]:
```
M1 = M0
flopy.modflow.ModflowGhb(M1, stress_period_data={0: [0, 0, ncol-1, BC3h+BC3h*dpar, BC3c]})
M1.write_input()
success, buff = M1.run_model(silent=True)
H1 = flopy.utils.binaryfile.HeadFile('model.hds').get_data()[0,0,:]
dhdBC3h = (H1-H0)/(BC3h*dpar)

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X/1000., dhdBC3h, 'k--', mfc='none', label='perturbation sensitivity, numerical')
plt.xlabel('$x$ (km)')
plt.ylabel(r'$\partial h / \partial h_{\Gamma_3}$')
plt.legend()
plt.ylim(0.95, 1.05);
```



In [58]:
```
set_background('rgba(0, 200, 0, 0.2)')
print('%.6f'% dhdBC3h[ocol])
```

```
1.000000
```

3. Adjoint sensitivity

$$\frac{\partial h(x')}{\partial h_{\Gamma_3}} = \oint_{\Gamma_3} \psi_1^*(x)\,C_{\Gamma_3}\,dx \;=\; \psi_1^*(\Gamma_3)\,C_{\Gamma_3} \tag{14}$$

Governing equation:

$$K\,b\,\frac{d\psi_1^*}{dx} + \frac{1}{2\,K\,b}\delta(x - x') = 0 \tag{15}$$

(16)

Boundary conditions:

$$\psi_1^*(x) = 0\,, \qquad x = 0 \tag{17}$$

$$\text{(undefined on } \Gamma_3\text{)} \tag{18}$$

(19)

Closed-form solution:

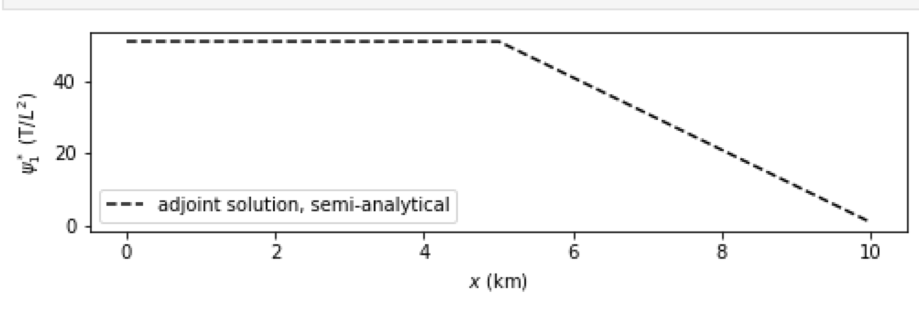
$$\text{Not available} \tag{20}$$

3a. Analytical

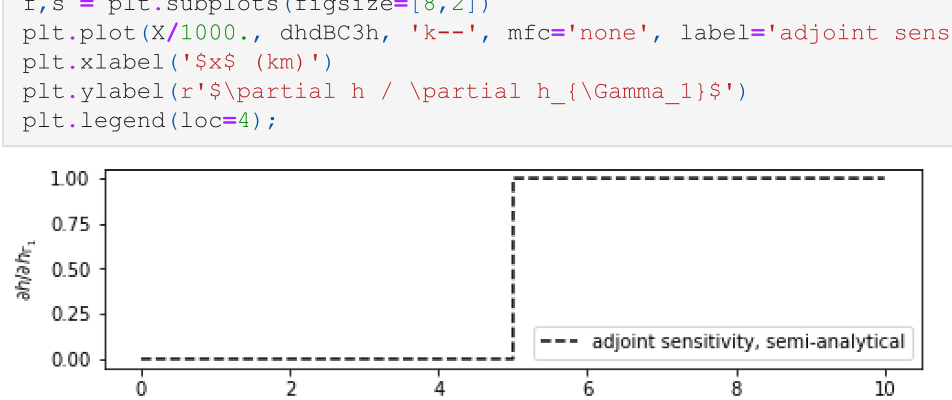
$$\text{Not available} \tag{21}$$

3b. Semi-analytical

In [104…
```
M0 = timml.Model3D(kaq=K, z=[0., -b])
timml.ImpLineDoublet1D(M0, xld=0.)
timml.HeadLineSink1D(M0, xls=L, hls=0., res=BC3c/100., wh=1.)
timml.LineSink1D(M0, xls=float(ocol), sigls=-1.)
M0.solve(silent=True)
A = M0.headalongline(X, 0.).flatten()

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X/1000., A, 'k--', mfc='none', label='adjoint solution, semi-analytical')
plt.xlabel('$x$ (km)')
plt.ylabel(r'$\psi^*_1$ (T/$L^2$)')
plt.legend(loc=3);
```



In [105…
```
dhdBC3h = np.array([M0.disvec(x,0) for x in X])[:,0].flatten()

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
f,s = plt.subplots(figsize=[8,2])
plt.plot(X/1000., dhdBC3h, 'k--', mfc='none', label='adjoint sensitivity, semi-analytical')
plt.xlabel('$x$ (km)')
plt.ylabel(r'$\partial h / \partial h_{\Gamma_1}$')
plt.legend(loc=4);
```

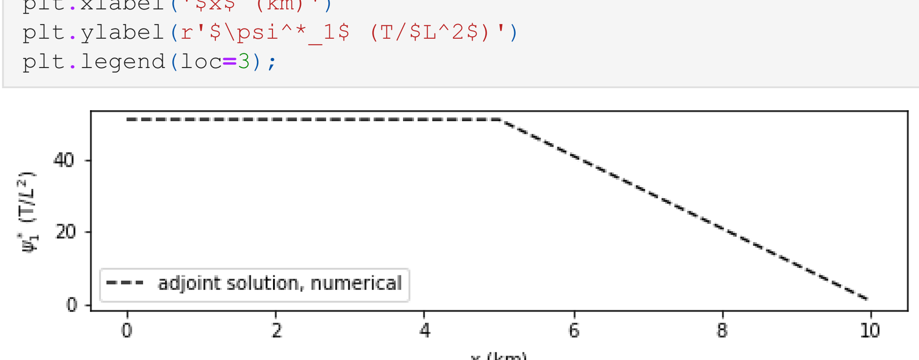


In [106…
```
set_background('rgba(0, 200, 0, 0.2)')
print('%.6f'% dhdBC1h[-1])
```

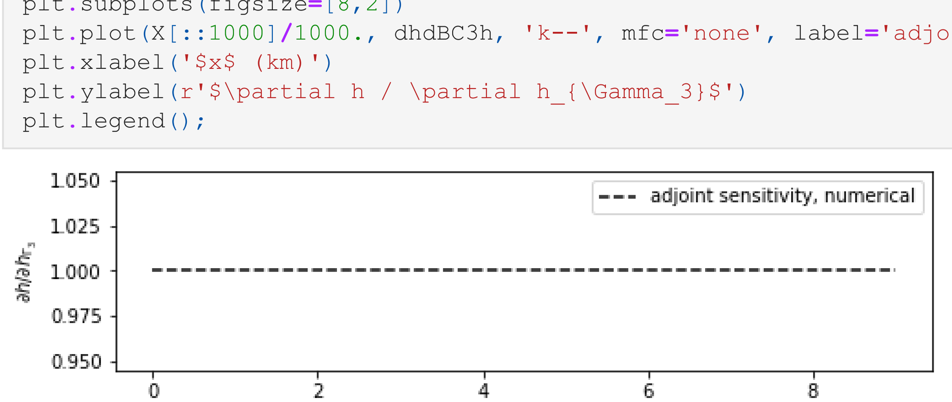
```
1.000000
```

3c. Numerical

In [63]:
```
M0 = flopy.modflow.Modflow(modelname='model', exe_name='../mf2005.exe')
flopy.modflow.ModflowDis(M0, nlay=1, nrow=1, ncol=ncol, nper=1, delr=1., delc=1., top=0., botm=-b, steady=True,
                         perlen=1., nstp=1)
flopy.modflow.ModflowBas(M0, ibound=np.ones([nrow, ncol]), strt=BC3h*np.ones([nrow, ncol], dtype=float))
flopy.modflow.ModflowLpf(M0, hk=K, vka=-999., ss=-999., sy=-999., ipakcb=53)
flopy.modflow.ModflowGhb(M0, stress_period_data={0: [0, 0, ncol-1, 0., BC3c]})
flopy.modflow.ModflowWel(M0, stress_period_data={0: [0, 0, ocol, 1.]})
flopy.modflow.ModflowPcg(M0, hclose=1e-6, rclose=1e-6)
flopy.modflow.ModflowOc(M0, stress_period_data={(0,0): ['save head', 'save budget']})
M0.write_input()
success, buff = M0.run_model(silent=True)
A = flopy.utils.binaryfile.HeadFile('model.hds').get_data()[0,0,:]

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X/1000., A, 'k--', mfc='none', label='adjoint solution, numerical')
plt.xlabel('$x$ (km)')
plt.ylabel(r'$\psi^*_1$ (T/$L^2$)')
plt.legend(loc=3);
```



In [62]:
```
dhdBC3h = np.empty(0)
for oc,xp in enumerate(X[::1000]):
    M1 = M0
    flopy.modflow.ModflowWel(M1, stress_period_data={0: [0, 0, oc, 1.]})
    M1.write_input()
    success, buff = M1.run_model(silent=True)
    A = flopy.utils.binaryfile.HeadFile('model.hds').get_data()[0,0,:]
    dhdBC3h = np.append(dhdBC3h, A[-1]*BC3c)

%matplotlib inline
import matplotlib as mpl
import matplotlib.pyplot as plt
plt.subplots(figsize=[8,2])
plt.plot(X[::1000]/1000., dhdBC3h, 'k--', mfc='none', label='adjoint sensitivity, numerical')
plt.xlabel('$x$ (km)')
plt.ylabel(r'$\partial h / \partial h_{\Gamma_3}$')
plt.legend();
```



In [64]:
```
set_background('rgba(0, 200, 0, 0.2)')
print('%.6f'% dhdBC3h[int(np.where(X[::1000]==float(ocol))[0])])
```

```
1.000000
```

In [ ]: