The results of using EO-LDAS with the Barrax data

For increasing speed of the solution for each of 32 LAI ground points 3x3 image was created. The solution was found for regularisation as well as without it.
ETM+ scene for 17.07.2004

In [67]:
```
import numpy as np
%pylab inline
plt.rcParams['figure.dpi'] = 1.2*plt.rcParams['figure.dpi']
plt.rcParams['savefig.dpi'] = 1.2*plt.rcParams['savefig.dpi']
```

```
Welcome to pylab, a matplotlib-based Python environment [backend:
module://IPython.zmq.pylab.backend_inline].
For more information, type 'help(pylab)'.
```

In [68]:
```
f=open("/home/max/Barrax/lai_points_1.txt")
tmp_str = f.read()
list_str = tmp_str.split("\n")
lai = np.zeros((6,len(list_str)-2))
for i in range(1, len(list_str)-1):
    for j in range(6):
        lai[j,i-1] = list_str[i].split()[j]
```

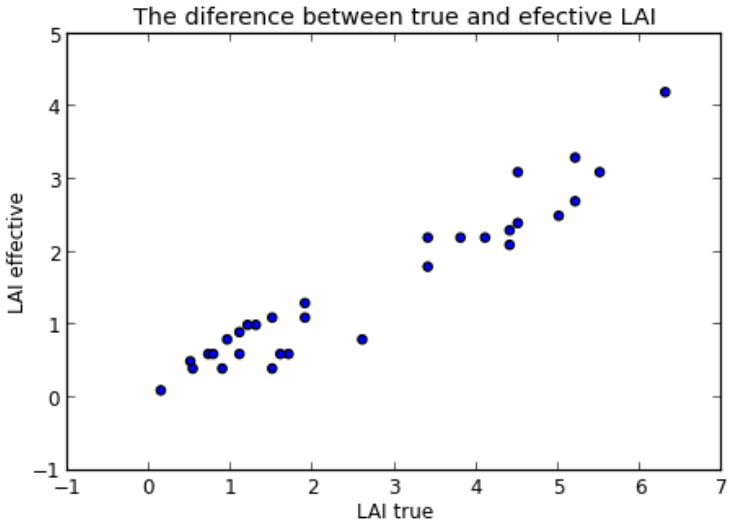
The result of the inversion by 1D model is effecive LAI. However in a field usually measured values the true LAI. In the case of Barrax there is informationabout true and efective LAI.

In [69]:
```
a,b = polyfit(lai[1,:], lai[2,:],1)
plt.scatter(lai[1,:], lai[2,:])

#plt.plot(lai[1,:], lai[2,:], lai[1,:], a*lai[2,:]+b)

plt.xlabel("LAI true")
plt.ylabel("LAI effective")
plt.title("The diference between true and efective LAI")
```

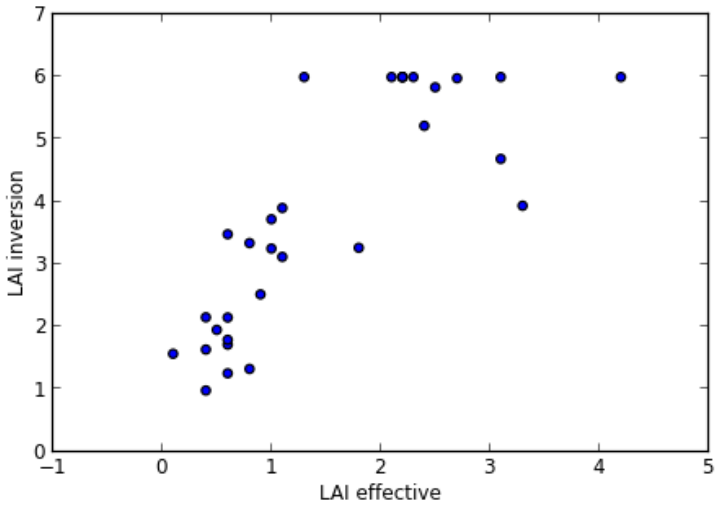
Out[69]: <matplotlib.text.Text at 0x576ad10>



No regularization

In [70]:
```
plt.scatter(lai[2,:], lai[5,:])
plt.xlabel("LAI effective")
plt.ylabel("LAI inversion")
#plt.title("LAI Inversion")
```

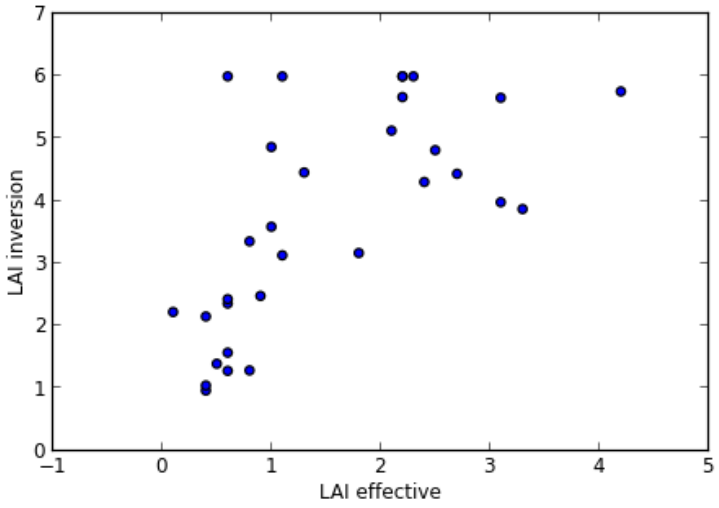
Out[70]: <matplotlib.text.Text at 0x58ae6d0>



gamma_row = gamma_col = 0.1

In [71]:
```
plt.scatter(lai[2,:], lai[3,:])
plt.xlabel("LAI effective")
plt.ylabel("LAI inversion")
#plt.title("LAI Inversion")
```

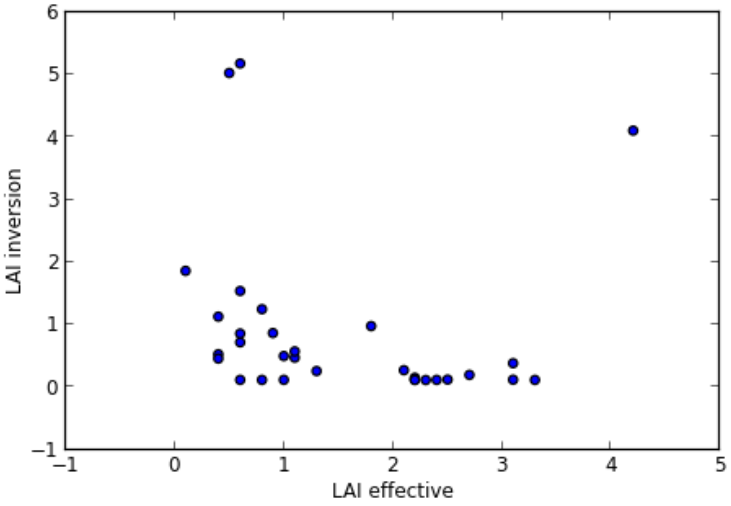
Out[71]: <matplotlib.text.Text at 0x5d09a10>



gamma_row = gamma_col = 10

In [72]:
```
plt.scatter(lai[2,:], lai[4,:])
plt.xlabel("LAI effective")
plt.ylabel("LAI inversion")
#plt.title("LAI Inversion")
```

Out[72]: <matplotlib.text.Text at 0x5d32d90>



In [72]: