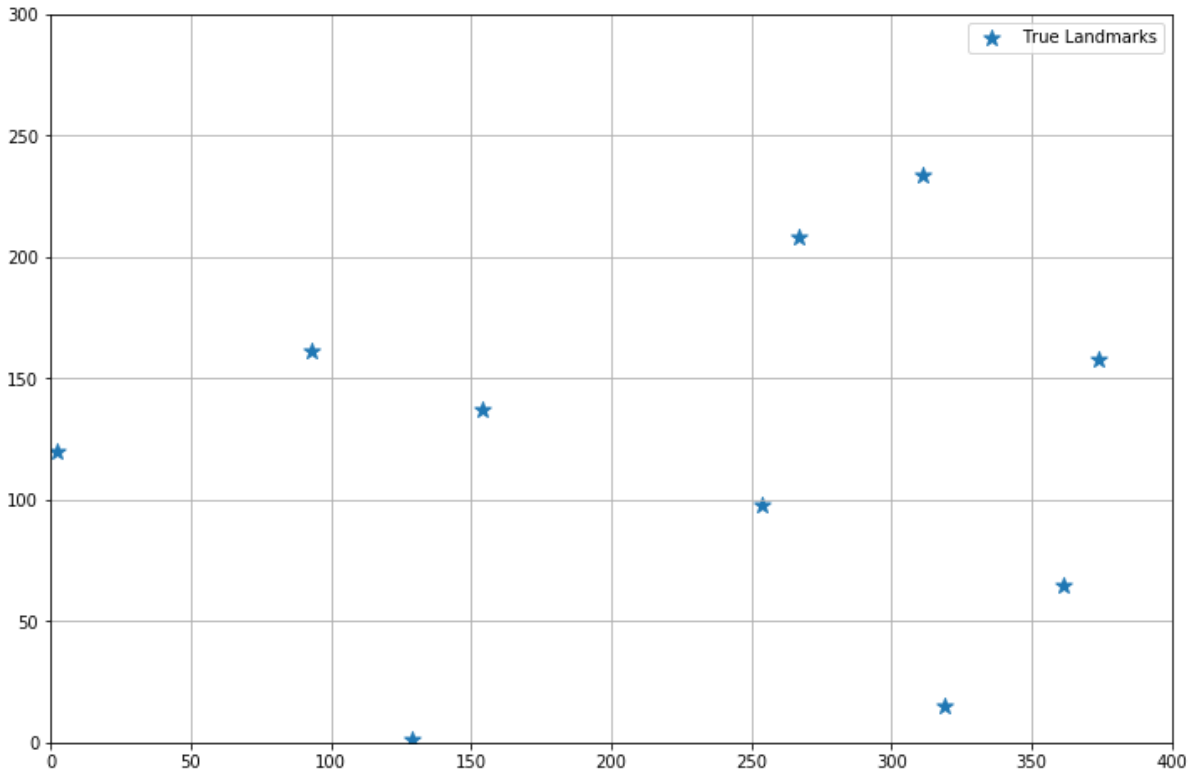
In [77]:
```
from data_utility.environment import *
from data_utility.utility import *
from slam_algorithms.robot import *
%load_ext autoreload
%autoreload 2
%matplotlib inline

plt.ion()
plt.rcParams['figure.figsize'] = (12.0, 8.0) # set default size of plots
plt.rcParams['image.interpolation'] = 'nearest'
plt.rcParams['image.cmap'] = 'gray'

env = Environment(n_landmarks=10)
fig, ax = env.plot_env()
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```



In [78]:
```
# Check Environment and robot's position

Q = np.array([[15., 0],[0, 15.]])
R = np.array([[10., 0],[0, 10.]])
P = np.identity(2)*5
s = np.array([100., 100.])
rob = MyRobot(env=env, Q=Q, R=R, s0=s, P0=P)
print("robot's initial guess of landmark: {}".format(rob.landmarks))
print("environment's landmark {}".format(rob.env.landmarks))
print("---")

ob = rob.observation_update(s=s)
s_prior = s + 50*np.random.randn(2)
for _ in range(20):
    s_true, s_prior = rob.time_update(s=s_prior, u=None)
    s_post = rob.map_construction(s=s_prior, ob=ob)
    ob = rob.observation_update(s=s_post)
    s_prior = s_post
    print("---")
rob.visualize()
```

```
robot's initial guess of landmark: {0: [205, 135], 1: [321, 158], 2: [1
83, 220], 3: [209, 233], 4: [114, 164], 5: [127, 91], 6: [27, 100], 7:
[253, 31], 8: [163, 25], 9: [7, 40]}
environment's landmark {0: [254, 98], 1: [374, 158], 2: [129, 1], 3: [3
11, 234], 4: [319, 15], 5: [361, 65], 6: [93, 161], 7: [267, 208], 8:
[2, 120], 9: [154, 137]}
---
prior state:[131.50209169 151.74397752]
post state:[133.78459933 149.45480232]
---
prior state:[133.78459933 149.45480232]
post state:[113.92315723 119.48770341]
---
prior state:[113.92315723 119.48770341]
post state:[108.52520806 112.46568346]
---
prior state:[108.52520806 112.46568346]
post state:[106.06195884 109.05620292]
---
prior state:[106.06195884 109.05620292]
post state:[104.42911172 107.02227254]
---
prior state:[104.42911172 107.02227254]
post state:[103.51165002 105.95685757]
---
prior state:[103.51165002 105.95685757]
post state:[102.94955223 105.04391724]
---
prior state:[102.94955223 105.04391724]
post state:[102.72923668 104.47573356]
---
prior state:[102.72923668 104.47573356]
post state:[102.34854645 103.84133432]
---
prior state:[102.34854645 103.84133432]
post state:[101.98073434 103.1958418 ]
---
prior state:[101.98073434 103.1958418 ]
post state:[101.82812867 102.97586086]
---
prior state:[101.82812867 102.97586086]
post state:[101.4929942  102.77069107]
---
prior state:[101.4929942  102.77069107]
post state:[101.51580317 102.51905228]
---
prior state:[101.51580317 102.51905228]
post state:[101.44026809 102.36818662]
---
prior state:[101.44026809 102.36818662]
post state:[101.33158228 102.09832847]
---
prior state:[101.33158228 102.09832847]
post state:[101.28511384 101.97025136]
---
prior state:[101.28511384 101.97025136]
post state:[101.20715231 101.73644361]
```

```
---
prior state:[101.20715231 101.73644361]
post state:[101.12128411 101.73959355]
---
prior state:[101.12128411 101.73959355]
post state:[101.15387602 101.61183102]
---
prior state:[101.15387602 101.61183102]
post state:[101.0537444  101.57750746]
---
```

Out[78]: (<Figure size 864x576 with 1 Axes>,
<matplotlib.axes._subplots.AxesSubplot at 0x1a296d5470>)