Algorithm Grid_localization($\{p_{k,t-1}\}, u_t, z_t, m$**):**
  for all k *do*
    $\bar{p}_{k,t} = \sum_i p_{i,t-1}$ **motion_model**$(\text{mean}(\mathbf{x}_k), u_t, \text{mean}(\mathbf{x}_i))$
    $p_{k,t} = \eta\ \bar{p}_{k,t}$ **measurement_model**$(z_t, \text{mean}(\mathbf{x}_k), m)$
  endfor
  return $\{p_{k,t}\}$

Table 8.1 Grid localization, a variant of the discrete Bayes filter. The function **motion_model** implements one of the motion models, and **measurement_model** a sensor model. The function "mean" returns the center-of-mass of a grid cell $\mathbf{x}_k$.

The second approach is the Monte Carlo localization (MCL) algorithm, arguably the most popular localization algorithm to date. It uses particle filters to estimate posteriors over robot poses. A number of shortcomings of the MCL are discussed, and techniques for applying it to the kidnapped robot problem and to dynamic environments are presented.

8.2 Grid Localization

8.2.1 Basic Algorithm

Grid localization approximates the posterior using a *histogram filter* over a grid decomposition of the pose space. The discrete Bayes filter was already extensively discussed in Chapter 4.1 and is depicted in Table 4.1. It maintains as posterior a collection of discrete probability values

$$bel(x_t) \quad = \quad \{p_{k,t}\} \tag{8.1}$$

where each probability $p_{k,t}$ is defined over a grid cell $\mathbf{x}_k$. The set of all grid cells forms a partition of the space of all legitimate poses:

$$\text{domain}(X_t) \quad = \quad \mathbf{x}_{1,t} \cup \mathbf{x}_{2,t} \cup \ldots \mathbf{x}_{K,t} \tag{8.2}$$

In the most basic version of grid localization, the partitioning of the space of all poses is time-invariant, and each grid cell is of the same size. A common granularity used in many of the indoor environments is 15 centimeters for