The shift of information from robot pose links to between-feature links is a key element of the SEIF. It is a direct consequence of using the information form as a filter, for the online SLAM problem. By integrating out past pose variables, we lose those links, and they are mapped back into the between-feature elements in the information matrix. This differs from the GraphSLAM algorithm discussed in the previous chapter, which never introduced any links between pairs of features in the map.

For a pair of features to acquire a direct link in this process, both have to be active before the update, hence their corresponding elements linking them to the robot pose in the information matrix have to be non-zero. This is illustrated in Figure 12.4: A between-feature link is only introduced between features m_1 and m_2. Feature m_3, which is not active, remains untouched. This suggests that by controlling the number of active landmarks at any point in time, we can control the computational complexity of the motion update, and the number of links in the information matrix. If the number of active links remains small, so will the update complexity for the motion update, and so will the number of non-zero between-landmark elements in the information matrix.

SPARSIFICATION

SEIF therefore employs a *sparsification* step, illustrated in Figure 12.5. The sparsification involves the removal of a link between the robot and an active feature, effectively turning the active feature into a passive one. In SEIFs, this arc removal leads to a redistribution of information into neighboring links, specifically between other active features and the robot pose. The time required for sparsification is independent of the size of the map. However, it is an approximation, one that induces an information loss in the robot's posterior. The benefit of this approximation is that it induces true sparseness, and hence makes it possible to update the filter efficiently.

There exists one final step in the SEIF algorithm, which is not depicted in any of the figures. This step involves the propagation of a mean estimate through the graph. As was already discussed in Chapter 3, the extended information filter requires an estimate of the state μ_t for linearization of the motion and the measurement model. SEIFs also require a state estimate for the sparsification step.

Clearly, one could recover the state estimate through the equation $\mu = \Omega^{-1}\xi$, where Ω is the information matrix, and ξ the information state. However, this would require solving an inference problem that is too large to be run at each time step. SEIFs circumvent the step by an iterative *relaxation algorithm* that propagates state estimates through the information graph. Each local state estimate is updated based on the best estimates of its neighbors in

RELAXATION ALGORITHM