Exploiting Symmetries to Construct Efficient MCMC Algorithms
With an Application to SLAM

Roshan Shariff
University of Alberta

András György
University of Alberta

Csaba Szepesvári
University of Alberta

Abstract

The Metropolis-Hastings (MH) algorithm is a flexible method to generate samples from a target distribution, a key problem in probabilistic inference. In this paper we propose a variation of the MH algorithm based on group moves, where the next state is obtained by first choosing a random transformation of the state space and then applying this transformation to the current state. This adds much-needed flexibility to the “textbook” MH algorithm where all measures involved must be given in terms of densities with respect to a common reference measure. Under mild conditions, our main result extends the acceptance probability formula of the textbook algorithm to MH algorithms with group moves. We work out how the new algorithms can be used to exploit a problem’s natural symmetries and apply the technique to the simultaneous localization and mapping (SLAM) problem. New experimental results comparing our method to existing state-of-the-art specialized methods on a standard range-only SLAM benchmark problem validate the strength of the approach.

1 INTRODUCTION

Probabilistic reasoning plays a major role in state-of-the-art artificial intelligence (AI) approaches to major challenges (Korb and Nicholson, 2003; Russel and Norvig, 2009; Poole and Mackworth, 2010). In particular, probabilistic graphical models are widely used in computer vision (Prince, 2012), robotics (Thrun et al., 2005a; Ferreira and Dias, 2014), speech and natural language processing (Manning and Schuetze, 1999), machine learning (Bishop, 2006; Murphy, 2012) and agent research (Xiang, 2002). A key step of working with probabilistic graphical models is inference, that is, the computation of a posterior distribution given the model and some data. As the posterior can rarely be expressed in a closed form amenable to direct evaluation by a computer, one often must resort to approximate inference methods (Pearl, 1988; Darwiche, 2009; Koller and Friedman, 2009), amongst which in this article we focus on the Metropolis-Hastings (MH) algorithm, which is a special Markov Chain Monte Carlo (MCMC) method.

The MH algorithm takes a target distribution and transforms a user-chosen Markov kernel (the “proposal kernel”) into another one such that, under mild conditions on the proposal kernel, a Markov chain based on the new kernel will have a limiting distribution equal to the target (Metropolis et al., 1953; Hastings, 1970). While the MH algorithm gives substantial flexibility in choosing the proposal kernel, the calculations needed to implement the MH algorithm are simple only for special forms of the proposal kernel such as the textbook case when all measures involved have a density with respect to a common reference measure¹ (Tierney, 1994). In this paper we describe two new classes of proposal kernels, based on group transformations of the state space and give the corresponding MH algorithms in closed form. The algorithms require basically the same amount of computation as the textbook MH algorithm, while we will argue that they significantly expand the scope of the MH algorithm. We will illustrate the results by specializing the algorithm to the simultaneous localization and mapping (SLAM) problem in robotics (Thrun et al., 2005b) and argue that the algorithm essentially recovers the MCMC-SLAM method of Torma et al. (2010), providing much needed insight into the behavior of

¹This restriction disallows even Gibbs sampling, since the target distribution typically has a density with respect to the Lebesgue measure on \(\mathbb{R}^n\), which however is zero on the one-dimensional subspaces on which proposals are made (see Section 3.2). The target distribution must therefore be conditioned on the space of proposals, which is straightforward for Lebesgue measures and linear subspaces but requires the machinery of measure theory to be correct in general (Chang and Pollard, 1997).
Figure 1: A probability density on $\mathbb{R}^2$ (center) having two factors (left and right).

this method as well as the first fully rigorous proof of its correctness\(^2\). In fact, it was this method that served as the inspiration for the present paper. In a new set of experiments, we demonstrate that this algorithm is competitive with state-of-the-art methods of robotics.

The paper is organized as follows: In Section 2 we use an example to motivate our approach, which is described in Section 3. Section 4 expands upon the example to illustrate how our approach can exploit symmetries. Section 5 is devoted to describing the SLAM problem, its symmetries, and how the general construction of Section 3 can be instantiated in this setting. We close the paper by providing experimental results on range-only SLAM (Section 6) followed by our conclusions (Section 7).

2 MOTIVATION AND PROBLEM STATEMENT

Suppose we want to draw samples from the simple two-dimensional probability distribution $P$ of Fig. 1. Its density $p(x,y)$ has two factors: $p_1(x,y)$ and $p_2(x,y)$, which need not be probability densities themselves (i.e., $p(x,y) = cp_1(x,y)p_2(x,y)$ for some constant $c > 0$). The MCMC approach is to construct a transition probability distribution that induces a random walk over $\mathbb{R}^2$, the distribution of which converges to $P$ in the steady state. The Metropolis-Hastings (MH) algorithm allows us to specify a proposal distribution, and under mild conditions, constructs a suitable MCMC transition kernel by proposing a new state but rejecting it with some probability. With some no-reject proposal kernels the rejection probability is zero, which means the proposal kernel is itself suitable as a transition kernel. The MCMC algorithm will be efficient if the proposal kernel does not often propose low-probability states (which would increase the rejection rate) and quickly explores the high-probability states (speeding up convergence to the steady state).

Often, a proposal kernel updates the state by modifying one variable at a time (the canonical example is Gibbs sampling; some multivariate “slice sampling” kernels also do this). However, it is immediately apparent that such an update would be problematic for our example: it would be impossible to move between the $\pm X$ and $\pm Y$ modes of the distribution without transiting through a low-probability region. Another common approach is to change all the variables by a small delta, perhaps drawn from a multivariate normal distribution. However, the variance of this proposal kernel must be carefully tuned for each variable: too small and it will be confined to one mode in a multimodal distribution like ours; too large and it will often propose points in the low-probability regions. In general, this idea does not work well with multi-modal distributions.

One might argue that we have overstated the difficulty of the problem. One sees at a glance that $p_1$ is radially symmetric and $p_2$ is scale invariant: we can make sampling much easier simply by reparameterizing the state space using polar coordinates $(r, \theta)$ instead of Cartesian coordinates $(x, y)$. Updating one variable at a time is then very effective: one can draw an independent sample from $P$ just by sampling $r$ according to $p_1$ and $\theta$ according to $p_2$. Indeed, our difficulties were simply because the Cartesian representation of the state space is mismatched with the independence and symmetry structure of the problem, whereas in the polar representation the $r$ and $\theta$ variables are independent with distributions derived from $p_1$ and $p_2$, respectively.

In general, however, it is not always possible to come up with a parameterization that reflects so cleanly the symmetries of the factors. Instead, since the symmetries are more readily apparent than a suitable parameterization, we can sidestep the problem of re-parameterizing the state space and instead work directly with the known symmetries. To do this, we will use the mathematical tools of topological group theory, which have been extremely fertile in the study of continuous symmetries. As an ancillary advantage, the family of algorithms we describe will be independent of the representation of the state space, by construction. This avoids the problems noted above with algorithms that depend crucially on a favorable choice of parameterization.

The idea of using groups has been intensely studied in statistics (Eaton, 1989; Wijsman, 1990; Diaconis, 1988) and groups have also found their way to machine learning (Smola and Kondor, 2003; Kondor et al., 2007; Kondor and Dempsey, 2012). Model symmetries are also exploited in the body of work on “lifted” probabilistic inference; these symmetries can be encoded by groups (Niepert, 2012a) and the problem has been approached with MCMC techniques (Niepert, 2012b). The focus of that work is on performing inference on a reduced space that collapses equivalence classes. Thus “lifting” is not applicable when the symmetries are approximate or when different symmetries apply to each subproblem: then the representation cannot be reduced and states must be explored that are symmetric for

\(^2\)Theorem 2 of Torma et al. (2010) is not correct when, in the notation of Section 3.1, $\Delta^Q \neq 1$ or $\chi \neq 1$. However, this does not affect the special case of SLAM.
one subproblem but not for another. To the best of our
knowledge the closest work to ours is that of Liu and col-
leagues (Liu and Wu, 1999; Liu and Sabatti, 2000; Liu,
2004), where the primary concern is generalizing Gibbs
sampling so that it can work with group transformations
(the main problem being the derivation of the right “condi-
tional” distribution over the set of transformations consid-
ered). However, as in general in Gibbs sampling, it is left to
the user to implement sampling from the derived distribu-
tion. In the present paper, however, we start from the MH
algorithm, giving the user the freedom to choose an easy to
sample distribution over the transformations.

3 METROPOLIS-HASTINGS WITH
GROUP TRANSFORMATIONS

A Markov Chain Monte Carlo (MCMC) algorithm to sam-
ple from a probability distribution $P$ over a state space $W$
is specified by a transition kernel $Q(dw | w)$, which gives
rise to a Markov chain $U_0,U_1,U_2,...$ where $U_0$ is sampled
according to some initial distribution $P_0$ and each $U_i$ after
that is sampled according to $Q(\cdot | U_{i-1})$. Under appropri-
cate conditions on $Q$, the random variables $U_n$ converge in
distribution to $P$ as $n \to \infty$; $P$ is then called a steady state
distribution of the Markov chain. A convenient condition
to force $P$ to be a steady state distribution of $Q$ is that $P$
and $Q$ should satisfy detailed balance:

$$P(du)Q(dv | u) = P(dv)Q(du | v); \quad (1)$$

the Markov chain is then said to be reversible. Indeed, the
meaning of (1) is that if $(U,V)$ is sampled from the joint
in (1) then we cannot tell whether $(U,V)$ was generated by
first choosing $U$ from $P$ and then following $Q$ to generate
$V$, or whether it was generated by first choosing $V$ from $P$
and then following $Q$ to generate $U$. Under additional
conditions on $Q$, such as $Q$ being $\phi$-irreducible and aper-
iodic, $P$ is the unique steady state distribution of $Q$ and the
Markov chain $(U_i)$ sampled from $Q$ will indeed converge in
distribution to $P$ regardless of $P_0$ (see, e.g., Roberts and

The MH algorithm is one way to construct reversible trans-
ition kernels: given a proposal kernel $Q'(dw' | w)$, the MH
kernel first samples $U'_{n+1}$ according to $Q'(\cdot | U_n)$ and then
accepts $U'_{n+1}$ as $U_{n+1}$ with probability $\alpha(U_n,U'_{n+1})$; other-
wise $U_{n+1}$ is taken to be $U_n$. With an appropriate choice of
the acceptance probability function $\alpha : W \times W \to [0,1]$, the
MH transition kernel satisfies detailed balance (Tierney,
1998). However, we will call any transition kernel obtained
via the above procedure an MH transition kernel regardless of
whether it satisfies detailed balance or whether its sta-
tionary distribution matches the target distribution.

We assume that $W$ is a topological space so that we can
reason about continuous transformations of the state space.

Our MH proposal kernel $Q'$ will select a continuous trans-
formation of $W$ and apply it to the current state of the chain.
Take $G$ to be a set of such transformations: for any $g \in G$
and $w \in W$ we will write $gw$ for the state resulting from ap-
plying $g$ to $w$. The composition of any two $g_1, g_2 \in G$, writ-
ten as $g_1 g_2$ and defined by $(g_1 g_2)w := g_1(g_2w)$, is certainly
a continuous transformation of $W$. Hence, without loss of
generality, we require that $g_1 g_2 \in G$ for any $g_1, g_2 \in G$.
Finally, since we would like our Markov chain to be re-
versible, every transformation in $G$ should be invertible:
for every $g \in G$ there should be some $g^{-1} \in G$ such that
$g^{-1}(gw) = g(g^{-1}w) = w$ for any $w \in W$.

It follows that $G$ contains a unit $e$ that is the identity trans-
formation on $W$: $ew = w$ for any $w \in W$; $e$ is simply the
composition of any $g$ and $g^{-1}$. As we would like the composition
operation to be associative (for any $g_1, g_2, g_3 \in G$, $(g_1 g_2)g_3 = g_1(g_2 g_3)$), our previous condi-
tions together mean that $G$ is a group that acts on $W$ via the
action $(g,w) \mapsto gw$, with group multiplication being the
aforementioned composition operation. To capture the no-
tion of transformations that are “similar” to each other, as-
sume that $G$ is endowed with a topology and that inversion
and multiplication are continuous operations with respect to
this topology; this makes $G$ a topological group. Finally,
assume that the topology of $G$ is such that the group acts
continuously on the state space: the group action is a con-
tinuous $G \times W \to W$ map.

Working in this general setting will allow our algorithm and
its correctness results to rely only on the operational notion of
transforming the state space in certain ways, and the re-
sulting algorithms will remain unchanged under different para-
meterizations of the state space. The state representa-
tion can be chosen freely, guided only by practical imple-
mentation concerns. However, as a guide to intuition, the
reader can imagine the state space $W$ to be a subset of the
Euclidean space $\mathbb{R}^n$ using an arbitrary choice of parameter-
isation. The group $G$ can be taken to be the invertible con-
tinuous maps, or even just the invertible affine transforma-
tions. One must only keep in mind that an algorithm con-
structed under these restrictions must be explicitly proven to
be invariant under reparameterization; it is not automatic-
ally invariant by construction as in the general setting we
adopt.

3.1 Metropolis-Hastings Based on Group Moves

The proposal kernel can be defined in terms of a condi-
tional distribution $Q_G(dg | w)$ over the group $G$; it samples
g \sim Q_G(\cdot | w)$ and proposes the new state $gw$. Further, un-
der certain technical conditions on the action of $G$ on $W$
and their respective topologies, there will be natural (rel-
atively) invariant measures on $W$ and $G$ (analogous to the
Lebesgue measure on $\mathbb{R}^n$). In particular, our conditions
will allow us a (left) Haar measure $\mu$ on $G$, which is in-
variant under translation on the left and relatively invariant
under translation on the right: if \( g \in G \) and \( H \subset G \) then
\[
\mu(gH) := \mu(\{gh \mid h \in H\}) = \mu(H)
\]
and
\[
\mu(Hg) := \mu(\{hg \mid h \in H\}) = \Delta_r^G(g)\mu(H),
\]
where \( \Delta_r^G : G \to \mathbb{R}_+^\times \) is called the (right) modular char-
eracter of \( G \) and is an inherent property of the group itself, where \( \mathbb{R}_+^\times \)
denotes the group of positive real numbers under multiplication (i.e., composition of scaling factors). It
is a continuous group homomorphism from \( G \) to the multiplicative
group of positive real numbers.\(^3\) In many cases \( \Delta_r^G = 1 \) identically: if the group is discrete, or commutative
(abelian), or its topology is compact, for example. We will also have a relatively invariant measure \( \lambda \) on \( W \): if \( g \in G \) and \( V \subset W \) then
\[
\lambda(gV) := \lambda(\{gv \mid v \in V\}) = \chi(g)\lambda(V).
\]
\( \chi : G \to \mathbb{R}_+^\times \) is also a continuous group homomorphism.
In practice we will often be able to construct an invariant
\( \lambda \), so that \( \chi = 1 \) identically.

We will assume that the target distribution \( P \) and proposal \( Q_G \) are absolutely continuous with respect to \( \lambda \) and \( \mu \), re-
spectively, with densities \( p \) and \( q \):
\[
P(dw) = p(w)\lambda(dw), \quad Q(dg \mid w) = q(g \mid w)\mu(dg).
\]
We will also assume that the initial state of the Markov
chain lies within the support of \( P \). Our MH transition ker-
el based on \( Q_G \) is defined by the following procedure:

**Procedure 1.** Given the current state \( w \in W \), sample the
new state \( w' \) as follows:

1. Sample \( g \sim Q_G(\cdot \mid w) \).
2. Calculate \( \alpha := \frac{\chi(g)p(gw)q'(g^{-1}\mid gw)}{\Delta_r^G(g)p(w)q'(g \mid w)} \).
3. Accept \( w' = gw \) with probability \( \min\{1, \alpha\} \).

In the procedure we use the function \( q' \) (derived from \( q \)) to
account for the possibility that many different moves \( g \in G \)
may result in the same \( w' \). In particular, \( q' \) is defined as follows:
For \( w \in W \), let \( G_w := \{g \in G \mid gw = w\} \) be the
isotropy subgroup of \( G \) at \( w \); it measures the injectivity of
the map \( g \mapsto gw \); for any \( g \in G \), the set of all \( g' \) that
satisfy \( g'w = gw \) is exactly \( gG_w \). Under mild conditions
on \( G \) and \( W \), \( G_w \) will be seen to be compact, implying
that there exists a unique Haar measure \( \beta_w \) on \( G_w \) with
\( \beta_w(G_w) = 1 \). Then
\[
q'(g \mid w) = \int_{G_w} q(gh \mid w) \beta_w(dh).
\]

\(^3\)For any \( g, h \in G \), it satisfies \( \Delta_r^G(gh) = \Delta_r^G(g)\Delta_r^G(h) \) and \( \Delta_r^G(g^{-1}) = (\Delta_r^G(g))^{-1} \) (and hence \( \Delta_r^G(e) = 1 \)).

Remark 1. It follows from this definition that \( q'(\cdot \mid w) \) is
constant on each \( gG_w \). Moreover, if \( q \) itself has this property
then \( q' = q \).

That Procedure 1 is “correct” (in the sense that the MH
kernel it defines is in detailed balance with \( P \)) will be the
subject of Theorem 1.

We note, in closing, that Procedure 1 encompasses the stan-
dard MH algorithm defined for Euclidean spaces. Indeed, if
the state space \( W \) and group \( G \) are both \( \mathbb{R}^n \) with \( gw = g+w \),
without loss of generality one can rewrite the proposal in
terms of the move \( g = w' - w \). Then the Lebesgue measure
\( m \) serves as both \( \lambda \) and \( \mu \). Since \( m \) is invariant, \( \chi = 1 \). Fur-
thermore, since vector addition is commutative, \( \Delta_r^G = 1 \).
Finally, for any \( x, y \in \mathbb{R}^n \) there is a unique \( g \) such that \( x + g = y \), so \( q' = q \) and
\[
\alpha = \frac{p(w')q(w' - w \mid w)}{p(w)q(w' - w \mid w)}.
\]

3.2 Mixtures of Group Moves

The proposal kernel described above is often too restrictive,
in that \( Q_G \) may not have a density with respect to the Haar
measure \( \mu \) on \( G \). For example, the Gibbs sampler on \( \mathbb{R}^n \)
updates the state space by modifying one coordinate at a
time. Its proposal distribution is therefore concentrated on
the coordinate axes (which have zero Lebesgue measure on
\( \mathbb{R}^n \)) and so does not have a density with respect to that
measure.

One way to increase flexibility is to allow several differ-
ett groups \( G_1, G_2, \ldots, G_n \) to act on the state space \( W \), each
associated with a kernel \( Q_i(\cdot \mid w) \) (\( i = 1, \ldots, n \)). Each
\( Q_i \) will be assumed to have a density \( q_i \) w.r.t. the Haar
measure \( \mu_i \) on \( G_i \). We will choose \( \lambda \) to be a measure on
\( W \) that is simultaneously relatively invariant under all the
groups: \( \chi_i \)-relatively invariant under each \( G_i \), respectively.
The proposal kernel \( Q' \) will be a mixture of the \( Q_i \) with co-
efficients \( a(i \mid w) > 0 \), \( i = 1, \ldots, n \), with \( \sum_{i=1}^n a(i \mid w) = 1 \)
for all \( w \in W \). The MH transition kernel based on \( Q' \) is
defined by the following procedure:

**Procedure 2.** Given the current state \( w \in W \), sample the
new state \( w' \) as follows:

1. Sample \( i \sim a(\cdot \mid w) \) and \( g \sim Q_i(\cdot \mid w) \).
2. Calculate

\[
\alpha := \frac{\chi_i(g)a(i \mid w)p(gw)q'(g^{-1} \mid gw)}{\Delta_r^G(g)a(i \mid w)p(w)q'(g \mid w)}.
\]
3. Accept \( w' = gw \) with probability \( \min\{1, \alpha\} \).

That \( P \) and this kernel are in detailed balance will be the
subject of Theorem 2.
3.3 Correctness

We will assume that the proposals are chosen in such a way that $\phi$-irreducibility holds: in particular, this is easy to verify in the case of SLAM below. To prove that the MCMC transition kernels described in Procedures 1 and 2 satisfy detailed balance, we will require some technical conditions on the space $W$ and the groups $G$ or $G_i$.

**Assumption 1.** The state space $W$ and the groups $G$ and $G_i$ are locally compact and Hausdorff.\(^4\)

The local compactness condition on the groups $G$ and $G_i$ guarantees the existence of the Haar measures on them. The Hausdorff property implies that every compact set in a space is also closed, and thus singleton sets are also closed.

The second assumption is designed to exclude certain pathological examples of group actions:

**Assumption 2.** The action of each group $G_i$ on the state space $W$ is proper: the map $\theta : (w, g) \mapsto (w, gw)$ preserves compactness of pre-images (i.e., $\theta^{-1}(K)$ is compact in $W \times G$ for every compact $K \subset W \times W$).

A group $G$ acting properly on the space $W$ has several desirable properties. Most importantly for our immediate purposes, the isotropy subgroup $G_w$ of $G$ at $w \in W$, defined by $G_w := \{ g \in G \mid gw = w \}$, is compact and thus also locally compact. Thus there is a finite Haar measure $\beta_w$ on each isotropy subgroup $G_w$, which, without loss of generality, is normalized: $\beta_w(G_w) = 1$.

As noted earlier, for any $g \in G$, the set of all $g'$ that also satisfy $g'w = gw$ is $gG_w$. Thus, if the action of $G$ on $W$ is proper, we are assured that the structure of $G$ is not too rich in relation to the space it acts upon: $gG_w$ is compact and thus not too “large”. With this, we can state our first main result:

**Theorem 1.** If the state space $W$ and group $G$ satisfy Assumptions 1 and 2, then the Markov transition kernel defined by Procedure 1 satisfies detailed balance (1).

To show the correctness of Procedure 2, we will need to assume that the image of $w$ under any two $G_i, G_j$ overlap only negligibly. To do this, we will assume that all the $G_i$ are, in fact, subgroups of some overarching group $K$, so that we can define intersections of the $G_i$:

**Assumption 3.** Define $G_{i,j} := G_i \cap G_j$ for $1 \leq i, j \leq n$. Then for each $i \neq j$ the condition

$$p(w) \int 1\{g \in G_{i,j}G_{k,w}\} q'(g \mid w) \mu_k(dg) = 0, \quad w \in W$$

is satisfied with either $k = i$ or $k = j$, where $G_{k,w}$ is the isotropy subgroup of $G_k$ at $w \in W$.

---

\(^4\)A topological space is locally compact if every point has a compact neighbourhood; it is Hausdorff if for every pair of distinct points, there are disjoint neighbourhoods containing each point.

**Theorem 2.** If the state space $W$ and each $G_i$ $(1 \leq i \leq n)$ satisfy Assumptions 1 to 3, then the Markov transition kernel defined by Procedure 2 satisfies detailed balance (1).

4 EXPLOITING SYMMETRIES

Judiciously choosing the groups $G_i$ and proposal kernels $Q_i$ allows the MH kernel with group transformations (Procedure 2) to take advantage of symmetries of the target distribution. Consider a distribution $P$ with a density that can be factored as follows:

$$p(w) = \prod_{i=1}^{m} p_i(w),$$

where $p_i(hw) = p_i(w)$ for all $h \in H_i$;

we say that each group $H_i$ is a symmetry of the factor $p_i$, or that $p_i$ is invariant under the action of $H_i$. For concreteness, we present a variation on the example of Section 2: $p$ is a density with respect to the Lebesgue measure $\lambda$ on $W = \mathbb{R}^2 \setminus \{(0,0)\}$ with $m = 3$ factors, $p_1$ and $p_2$ are as described earlier, and we add another factor $p_3$ with no useful symmetries; thus $H_1$ and $H_2$ are, respectively, the groups that rotate and scale $\mathbb{R}^2$ around its origin, and $H_3$ is the trivial group (containing only the identity transformation).

To apply Procedure 2 to this example, take $n = 2$, $G_1 = H_2$, $G_2 = H_1$, and $a(i \mid w) = \frac{1}{2}$ for $i = 1, 2$ and all $w$. In this example, for $i = 1, 2$, $\Delta_i \equiv 1$ identically (since both groups are commutative) and $q_i = q_1$ (by Remark 1, since the isotropy subgroups are trivial). The proposed state is $w' = gw$ for some $g \in G_i$, so we see immediately that $p_i(w') \neq p_i(w)$ is only possible for $j \in \{i, 3\}$. Thus, in the $i = 1$ case, the $p_2$ factor cancels out of the acceptance probability:

$$\alpha_{i=1} = \frac{\chi_1(g)p_1(gw)p_2(gw)p_3(gw)q_1(g^{-1} \mid gw)}{p_1(w)p_2(w)p_3(w)q_1(g \mid w)}. \quad (2)$$

Next we choose $q_1$, attempting to cancel the $\chi_1$ and $p_1$ factors as well. Since $G_1$ acts by scaling $\mathbb{R}^2$, we can identify it with $\mathbb{R}^2_+ \setminus \{0\}$, the group of positive real numbers under multiplication (i.e., composition of scaling factors). Then $g \in \mathbb{R}^2_+$ acts on $\mathbb{R}^2$ by $(x, y) \mapsto (gx, gy)$, the corresponding effect on the Lebesgue measure (area) on the plane is described by $\chi_1(g) = g^2$, and $\mu_1(dg) = g^{-1}dg$ is a Haar measure on $\mathbb{R}^2_+$. The obvious choice is to set $q_1(g \mid w) = \chi_1(g)p_1(gw)$ with a normalizing constant $c_1(w)$; then for any $w \in W$, since $q_1$ must be a probability kernel, we use the definitions of $\mu_1$ and $\chi_1$ to get

$$\int_0^\infty q_1(g \mid w) g^{-1}dg = c_1(w) \int_0^\infty p_1(gw) g dg = 1. \quad (3)$$

A simple calculation using (3) yields $c_1(gw) = g^2c_1(w) = \chi_1(g)c_1(w)$, which we substitute into (2):

$$\alpha_{i=1} = \frac{\chi_1(g)p_1(gw)p_2(gw)p_3(gw)\chi_1(g^{-1})p_1(w)}{p_1(w)p_2(w)p_3(w)\chi_1(g)p_1(w)}. \quad (2)$$

870
An analogous derivation can be carried out for the $i = 2$ case, identifying $G_2$ with $[0, 2\pi)$ as the set of rotation angles under the operation of addition (mod $2\pi$). Then $\chi_2 = 1$ and $\mu_2$ is just the Lebesgue measure on $G_2$; again we get $\sigma_{1,2} = p_2(gw)/p_2(w)$. In fact, the same technique works in general for any target distribution $P$, even if $\Delta^{G_i} \neq 1$, as long as $\chi_i(g)p_i(gw)$ is $\mu_i$-integrable:

**Proposition 1.** Suppose $q_i(g \mid w) = c_i(w) \chi_i(g)p_i(gw)$ $(g \in G_i, w \in W_i)$ is a probability kernel density for some appropriately chosen normalizer $c_i$. Then $q_i = q_i$ and

$$\frac{\chi_i(g)p_i(gw)q_i(g^{-1}\mid gw)}{\Delta^{G_i}(g)p_i(w)q_i(g \mid w)} = 1.$$ 

We conclude that when Procedure 2 is applied to a target distribution having factors $p_i$ invariant under $H_i$, the proposals in the mixture should be chosen so that (a) $G_i \subset H_j$ for as many $j \neq i$ as possible, eliminating the $p_j$ terms from the acceptance probability, and (b) $q_i(g \mid w) \propto \chi_i(g)p_i(gw)$ to eliminate the $\chi_i$, $\Delta^{G_i}$, and $p_i$ terms; the constraint is that the $G_i$ transformations sampled according to $Q_i$ must collectively be able to explore the support of $P$. Indeed, ideally only the non-symmetric factors of $p$ appear in the acceptance probability, as we saw in the example. If we had $p_3 = 1$ as in Section 2, we would recover the no-reject algorithm that produces independent samples every time it performs a rotation and a scaling. The simpler acceptance probability also means that only the non-symmetric factors contribute to the time required to compute it.

5 THE SLAM PROBLEM

The SLAM problem is concerned with a robot navigating an unknown environment under the effect of sensor and control noise. The goal is to determine the robot's trajectory as well as the map of the environment based on the robot's observations. The environment comprises $N$ landmarks; the position of each is denoted by a variable $Y_i$ $(i = 1, \ldots, N)$ taking values in a space $\mathcal{Y}$. Let $X_t$ $(t = 0, \ldots, T)$ denote the *pose* (typically, position and orientation) of the robot at time step $t$ and take values in space $\mathcal{X}$. At every time step the robot can observe the landmarks, and at time step $t$ the observation of landmark $i$ is denoted by $Z_i^t$ taking values in $\mathcal{Z}$. For simplicity, we assume that all landmarks can always be observed and the robot can distinguish the landmarks. The goal of the SLAM problem is to estimate the trajectory $X = (X_0, \ldots, X_T)$ and the landmark positions $Y = (Y_1, \ldots, Y_N)$ based on the observations $Z = (Z_i^t)_{0 \leq t \leq T, 1 \leq i \leq N}$ (our notation consistently refers to time steps and landmarks with subscripted and superscripted indices, respectively).

We use the Bayesian formulation of SLAM, in which the robot’s trajectory, environment, and observations are random variables and are assumed to evolve according to the following dynamical system: (a) $X_0$ and $Y$ are independent with known densities; (b) at each time step $t = 0, 1, \ldots, \ldots$, each observation $Z_i^t$ depends only on $X_t$ and $Y_t$ via the conditional density $p_{Z_i^t \mid X_t, Y_t}$, and (c) the pose of the robot $X_t$ depends only on $X_{t-1}$ and the previous observations $Z_{<t} := (Z_0, \ldots, Z_{t-1})$ via the conditional density $p_{X_t \mid X_{t-1}, Z_{<t}}$ (where $Z_t = (Z^1_t, \ldots, Z^N_t)$). That is, we make the following Markov assumptions: (a) $Z_i^t$ is conditionally independent of $X_{<t}$ and $Y_j$ ($j \neq i$) given $X_t$ and $Y_t$, and (b) $X_t$ is conditionally independent of $X_{<t-1}$ and $Y$ given $X_{t-1}$ and $Z_{<t}$. Also, we assume throughout that conditional densities exist relative to some dominating measure, usually an appropriate Lebesgue or Haar measure.

The SLAM posterior is the conditional density $p_{X,Y,Z}(\cdot \mid z)$ over trajectories and environments given observations $Z = z$. We first factor the joint density $p_{X,Y,Z}(x,y,z)$ as $p_{Y}(y)p_{X,Y}(x,y \mid z)$. Then, under the above Markov assumptions, we obtain

$$p_{X,Y,Z}(x,y \mid z) = \prod_{t=0}^{T} p_{X_t \mid X_{t-1}, Z_{<t}}(X_t \mid X_{t-1}, Z_{<t}) \cdot p_{Z_t \mid X_t, Y}(Z_t \mid X_t, Y) \cdot p_{Y}(y)$$

$$p_{X,Y,Z}(x,y \mid z) = \frac{p_{Y}(y)p_{X,Y}(x,y \mid z)}{p_{Z}(z)} \tag{4}$$

We consider the SLAM problem in which the robot moves on a two-dimensional plane. Then its position and orientation are fully specified by the rigid (i.e., distance-preserving and non-reflecting) transformation of $\mathbb{R}^2$ from the robot’s body-local coordinate system to the global coordinates. Any rigid transformation can be decomposed into a rotation around the origin followed by a translation; the set of such transformations under composition forms the *special Euclidean group* $\text{SE}(2)$. The space of poses is therefore $\mathcal{X} := \text{SE}(2)$. The landmarks are specified by their positions on the plane, so $\mathcal{Y} := \mathbb{R}^2$.

5.1 Symmetries of SLAM

We assume that, apart from the landmarks, the environment is essentially homogeneous (we will elaborate upon this means), giving rise to certain symmetries in the factors of the SLAM posterior distribution. If a robot has pose $x \in \mathcal{X}$, in its body-local frame the coordinates of another pose $x' \in \mathcal{X}$ are $x^{-1}x'$, and those of a landmark $y \in \mathcal{Y}$ are $x^{-1}y$. One can verify that these local coordinates do not change if $x$, $x'$, and $y$ are all transformed by some $g \in \mathcal{G} := \text{SE}(2)$ to $gx$, $gx'$, and $gy$, respectively. The assumption that the environment is homogeneous means, firstly, that the motion of the robot is not affected by its location in a way undetectable to its sensors. In particular, for a given value of $Z_{<t}$, the motion model $p_{X_t \mid X_{t-1}, Z_{<t}}$ depends only on the relative movement $X_{t-1}X_t$ and not on the global coordinates. Secondly, since the sensors are fixed to the robot’s body, the observation of a landmark depends only on its local coordinates in the robot’s frame: $p_{Z_i^t \mid X_t, Y_t}$.
depends only on $Z_t^i$ and $X_t^{-1}Y_t$. Thirdly, the landmarks and the robot’s initial pose are a priori equally likely to be anywhere in the environment: $p_{X_0}$ and $p_{X_0}$ are invariant under $\mathcal{G}$. The homogeneity of the environment thus implies that no reference frame is privileged, and that being transformed by $\mathcal{G}$ does not affect the likelihood of a SLAM solution. To resolve the resulting ambiguity, without loss of generality we work in the coordinate system whose origin is the robot’s initial pose (i.e., $X_0$ is the identity transformation).

Thus, for our purposes, the SLAM posterior is a distribution over the state space $W := X^T \times \mathcal{Y}^N$ of all possible trajectories (that start at the origin) and environments. The group $K := \mathcal{G}^T \times \mathcal{G}^N$ acts on $W$, with the $g_t, g_t^T \in \mathcal{G}$ components acting on $w_t \in X$ and $w_t^T \in \mathcal{Y}$, respectively (by our convention, the subscripts and superscripts refer to the pose and landmark components, respectively). Using the terminology of Section 4, the $p_{X_t|X_{t-1},Z_{\text{ctl}}}$ factors are invariant under the subgroups $H_t := \{g \in K \mid g_{t-1} = g_t\}$ and the $p_{Z_t|X_t,Y_t}$ factors under $H_t := \{g \in K \mid g_t = g_t^T\}$.

5.2 The MCMC-SLAM Algorithm

We now specify how Procedure 2 may be applied to the problem of sampling from the SLAM posterior. First, we select a function $b : \{1, \ldots, N\} \rightarrow \{1, \ldots, T\}$, which “anchors” each landmark to one of the time steps at which it was observed. The proposal is a mixture of $T + N$ kernels, indexed with subscripts or superscripts as before. The mixture component corresponding to time step $t$ transforms $W$ by an element of $G_t := (\cap_{i \neq t} H_i) \cap (\cap_t H_{b(i)})$, which is a symmetry of the $p_{X_t|X_{t-1},Z_{\text{ctl}}}$ factors for $s \neq t$ and of the $p_{Z_t|X_t,Y_t}$ factors for $(s,i) \notin V_t$, where

$$V_t := \{(s,i) \mid s < t \leq b(i) \text{ or } b(i) < t \leq s\}.$$ 

Indeed, this is a maximal set of factors for which $G_t$ is a symmetry without being reduced to triviality. One can verify that an element of $G_t$ is determined by $g \in \mathcal{G}$ that acts on $w_s$ if $s \geq t$ and on $w'$ if $b(i) \geq t$; other components of $w \in W$ are left unchanged. The mixture components corresponding to landmark $i$ use $G_t^i := (\cap_{j \neq i} H_i) \cap (\cap_t H_{b(i)})$, which is a symmetry of all the $p_{X_t|X_{t-1},Z_{\text{ctl}}}$ factors and those $p_{Z_t|X_t,Y_t}$ factors with $j \neq i$; again, this is a maximal invariant set. The corresponding proposal kernel densities $q_t$ and $q_t^i$ are chosen to be proportional to $p_{X_t|X_{t-1},Z_{\text{ctl}}}$ and $p_{Z_t|X_{t-1},Y_t}$, respectively, following Section 4. Procedure 3 lists the resulting algorithm. Note that if the trajectory is stored in the tree structure of Fenwick (1994), modified to support non-commutative operations, the state update can be carried out in $O(\log T)$ time; the calculation of the acceptance probability then dominates, thus scaling with the number of factors whose values have changed.

---

**Procedure 3.** Given $w \in W$ consisting of a trajectory $x_1, \ldots, x_T$ and landmarks $y_1, \ldots, y_N$, propose $w'$.  

(i) Sample either a time step $t$ or a landmark $i$ from a given discrete distribution with probabilities $a_t(w)$ and $a_i(w)$, respectively (i.e., $\sum_{t=1}^T a_t(w) + \sum_{i=1}^N a_i(w) = 1$).

(ii) If the previous step sampled time step $t$:

1. Set $x'_t \sim p_{X_t|X_{t-1},Z_{\text{ctl}}}(\cdot | x_{t-1}, z_{\text{ctl}})$.
2. Set $x'_t = x_t x_{t-1}^{-1} x_s$ for $s > t$.
3. Set $y'_i = x'_t x_t^{-1} y_i$ for $m(i) \geq t$.
4. Calculate

$$\alpha := \frac{a_i(w')}{a_t(w)} \prod_{(s,i) \in V_t} p_{Z_t|X_s,Y_s}(z_{t,s}^i | x'_t, y'_i) \prod_{(s,i) \notin V_t} p_{Z_t|X_s,Y_s}(z_{t,s}^i | x_t, y_i).$$

(iii) Otherwise, if it sampled landmark $i$:

1. Set $y'_i \sim p_{Z_i|X_{b(i)},Y_{b(i)}}(z_{b(i)}^i | x_{b(i)}, \cdot)$.
2. Calculate

$$\alpha := \frac{a_i(w')}{a_t(w)} \prod_{t \geq b(i)} p_{Z_t|X_t,Y_t}(z_{t,i}^i | x_t, y_i)$$.

(iv) Accept new state $w'$ with probability $\min\{1, \alpha\}$. All unmodified variables keep their original values.

6 EXPERIMENTS

We applied the MCMC-SLAM algorithm to two publicly available datasets (Djugash, 2010) from an autonomous robot with sensors that measure range to radio beacons. In the Plaza 1 data set, the robot traveled 1.9 km over 9,657 time steps and received 3,529 range observations of four landmarks. In the Plaza 2 data set, the robot traveled 1.3 km over 4,091 time steps and received 1,816 range measurements, also of four landmarks. Highly accurate ground truth trajectories were also recorded. We compare the algorithm to the Spectral SLAM algorithm (Boots and Gordon, 2013). We found that exploiting symmetries as outlined in Section 4 was crucial: the naive MCMC kernel that updated individual components of the trajectory or environment did not make any progress in a reasonable amount of time.

Table 1 shows the RMS distance of each robot pose from the ground truth for each data set. It is averaged over 50 independent runs of the MCMC algorithm, with the interval indicating one standard deviation. Since any SLAM solution is only specified up to the choice of origin, we apply the best-fit rigid transformation between the estimated and known maps (Boots and Gordon do the same).

The MCMC $(r + s)$ algorithms incrementally extend the SLAM posterior by introducing the factors coming each
time step, in turn. They use \( b(i) = \arg \min_i Z_i \). The chain takes \( r \) steps after each extension, and \( s \) steps at the end. At each time step, newly introduced variables are initialized by sampling from the corresponding proposal kernel. MCMC (10 + 1000) took approximately 13.8 s on Plaza 1 and 2.8 s on Plaza 2; MCMC (100 + 10000) took 131.1 s and 28.1 s, respectively. The larger number of steps is required to achieve good accuracy on Plaza 2 because it is more challenging: the robot consistently turns in one direction, making the control noise biased. In comparison, Spectral SLAM took 0.73 s and 0.51 s on a similar computer. The “Spectral + Opt.” algorithm runs a final batch optimisation pass and takes several thousands of seconds.

Thus, even though the MCMC algorithm is computationally somewhat more expensive, we see that it performs competitively with Spectral SLAM and all the other methods tested by Boots and Gordon (2013). In addition, it has the advantage of easily handling missing observations, without a process of imputing them as is done by Spectral SLAM. Finally, being a Bayesian algorithm, it produces the SLAM posterior distribution rather than just a solution; indeed, we expect it to perform better if the robot noise characteristics were faithfully modelled.

7 CONCLUSIONS AND FUTURE WORK

The Metropolis-Hastings (MH) algorithm is a widely used technique to implement approximate probabilistic inference, but its “textbook version” is quite limited. To build potentially faster mixing chains, in this paper we explore the possibility of proposals where the next state is based on transforming the current one using a randomly chosen transformation. The main contribution of the paper is a formula that shows how the acceptance function can be calculated in closed form in this case. This is shown both for a single kernel, and when a mixture kernel is used. The strength of the approach is its generality: We derive the results without any differentiability requirements, making them applicable to both continuous and discrete domains. While the increased generality made the paper more technical, to enhance clarity, we used the SLAM problem to illustrate the ideas. On a challenging domain, we obtained strong experimental evidence in favor of our new approach. While it remains for future work to demonstrate the approach on a wider range of problems, we believe that the approach proposed in the paper, due to its generality and flexibility, will have a profound impact on how AI systems perform approximate inference.

Acknowledgements

This work was supported by the Alberta Innovates Technology Futures and the National Science and Engineering Research Council of Canada.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Plaza 1</th>
<th>Plaza 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectral</td>
<td>0.79 m</td>
<td>0.35 m</td>
</tr>
<tr>
<td>Spectral + Opt.</td>
<td>0.69 m</td>
<td>0.30 m</td>
</tr>
<tr>
<td>MCMC (10 + 1000)</td>
<td>0.32 ± 0.02 m</td>
<td>0.54 ± 0.06 m</td>
</tr>
<tr>
<td>MCMC (100 + 10000)</td>
<td>0.33 ± 0.04 m</td>
<td>0.36 ± 0.03 m</td>
</tr>
</tbody>
</table>
References


