## Discuss HW #2, solutions to problems H and I

Problem H (G&S, problem 11.2.24)

Conditions

(a) $w_x = p\, w_{x+1} + q\, w_{x-1}$ for $x = 1, 2, \ldots, T\text{-}1$ ,

(b) $w_0 = 0$

and

(c) $w_T = 1$

determine $w_x$ because condition (b) tells us that the function $w$ is harmonic at all states other than 0 and $T$, and conditions (a) and (c) tell us the values that $w$ takes at the boundary; as we saw in class (as an application of the maximum principle), a harmonic function on an absorbing Markov chain is determined by its values on the boundary.

Grinstead and Snell offer the following explanation for why conditions (a), (b), and (c) suffice to determine the values of **w**:

"Let **P** be the transition matrix for our absorbing chain. Then these equations state that $\mathbf{Pw} = \mathbf{w}$. That is, the column vector **w** is a fixed vector for **P**. Consider the transition matrix for an arbitrary Markov chain in canonical form and assume that we have a vector **w** such that $\mathbf{w} = \mathbf{Pw}$. Multiplying through by **P**, we see that $\mathbf{P}^2\mathbf{w} = \mathbf{w}$, and in general $\mathbf{P}^n\mathbf{w} = \mathbf{w}$. But

$$\mathbf{P}^n \to \begin{pmatrix} \mathbf{0} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}.$$

Thus

$$\mathbf{w} = \begin{pmatrix} \mathbf{0} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \mathbf{w} .$$

If we write

$$\mathbf{w} = \begin{pmatrix} w_T \\ w_A \end{pmatrix},$$

where $T$ is the set of transient states and $A$ is the set of absorbing states, then by the argument above we have

$$\mathbf{w} = \begin{pmatrix} w_T \\ w_A \end{pmatrix} = \begin{pmatrix} \mathbf{B}w_A \\ w_A \end{pmatrix}.$$

Thus for an absorbing Markov chain, a column vector **w** satisfying **Bw** = **w** is determined by its values on the absorbing states. Since in our example we know these values are 0 and 1, we know that **w** is completely determined. The solutions given clearly satisfy (b) and (c), and a direct calculation shows that they also satisfy (a)."

Problem I (G&S, problem 11.2.26)

Here is what Grinstead and Snell say about the uniqueness issue:

"Again, it is easy to check that the proposed solution $f(x) = x(n\text{-}x)$ satisfies conditions (a) and (b). The hard part is to prove that these equations have a unique solution. As in the case of Exercise 23, it is most instructive to consider this problem more generally. We have

a special case of the following situation. Consider an absorbing Markov chain with transition matrix **P** in canonical form and with transient states $T$ and absorbing states $A$. Let **f** and **g** be column vectors that satisfy the system of equations

$$\begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{f}_T \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \mathbf{g}_T \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_T \\ \mathbf{0} \end{pmatrix}.$$

where $\mathbf{g}_T$ is given and it is desired to determine $\mathbf{f}_T$. In our example, $\mathbf{g}_T$ has all components equal to 1. To solve for $\mathbf{f}_T$ we note that these equations are equivalent to

$$\mathbf{Q}\,\mathbf{f}_T + \mathbf{g}_T = \mathbf{f}_T\ ,$$

or

$$(\mathbf{I} - \mathbf{Q})\,\mathbf{f}_T = \mathbf{g}_T\ .$$

Solving for $\mathbf{f}_A$, we obtain

$$\mathbf{f}_T = \mathbf{N}\,\mathbf{g}_T$$

Thus $\mathbf{f}_T$ is uniquely determined by $\mathbf{g}_T$ .

**HW #3 due 2 weeks from today**

See http://jamespropp.org/584/P3.pdf .

# Rotor-routing

Disclaimer: This topic is NOT standard stochastic processes material. It's mostly my own work, and while rotor-routing is catching on in some circles, it remains to be seen whether it'll have serious applications or come to be viewed as a mere curiosity. Time will tell!

## Definition of rotor-routing

Executing an $n$-state Markov chain with transition matrix **P** by making random moves can be viewed as taking a **random walk** on the set $\{s_1, s_2, ..., s_n\}$, where the walker (or particle), when at site $s_i$, has probability $p_{ij}$ of going to site $s_j$ at the next step.

That is, each time the particle is at site $s_i$, its next step is determined by making a random selection from the probability distribution on $\{s_1, s_2, ..., s_n\}$ given by the $i$th row of **P**. For simplicity let's assume for now that all entries of **P** are equal to 0 or 1/2, so that every state has two successors. It may help to imagine a coin associated with each site $s_i$.

In contrast, with **rotor walk**, a particle leaving site $s_i$ goes to whichever successor of $s_i$ it **didn't** go to when it left $s_i$ the previous time. (When the particle is leaving $s_i$ for the first time, the rotor rule doesn't specify its behavior.)

That is, there should be a way of labelling the two successors of $s_i$ as *a* and *b* such that

the 1st time the particle leaves $s_i$,
    it goes to *a*;
the 2nd time the particles leaves $s_i$,
    it goes to *b*;
the 3rd time the particle leaves $s_i$,
    it goes to *a*;
the 4th time the particles leaves $s_i$,
    it goes to *b*;
etc.

See http://www.cs.uml.edu/~jpropp/rotor-router-model/ in its "Walk on finite graph A" mode (derandomized random walk on {0,1,2,3}).

If we have a state $s_i$ with three non-zero entries in the associated row of **P**, all equal to 1/3, then the rotor-router rule is ...

... the particle should go to whichever successor of $s_i$ it went to the least recently .

That is, there should be a way of labelling the three successors of $s_i$ as *a* and *b* and *c* such that

the 1st time the particle leaves $s_i$ ,

it goes to *a*;

the 2nd time the particles leaves $s_i$ ,

it goes to *b*;

the 3rd time the particle leaves $s_i$ ,

it goes to *c*;

the 4th time the particles leaves $s_i$ ,

it goes to *a*;

etc.

That is, the exit sequence at $s_i$ should be

*abcabcabc*...,

where the exit sequence at $s_i$ means the sequence whose $k$th term (for all $k \geq 1$) is the state that the rotor walk goes into after its $k$th transition from state $s_i$.

We may imagine an arrow at $s_i$ that successively points to *a*, to *b*, to *c*, to *a*, to *b*, to *c*, etc.; we call this arrow the rotor at $s_i$. At any instant, it points toward the successor of $s_i$ that the particle/walker went to after its most recent visit to $s_i$. Note that at the start, it points to *c*. The rule for rotor walk is "Update the arrow, then go where the arrow tells you to."

If we have a state $s_i$ with two non-zero entries in the associated row of **P**, one equal to 2/3 and the other equal to 1/3, then the rotor-router rule is ...

... the exit sequence from $s_i$ should be

*aabaabaab*... or

*abaabaaba*... or

*baabaabaa*...,

where $p_{ia} = 2/3$ and $p_{ib} = 1/3$. (See http://www.cs.uml.edu/~jpropp/rotor-router-model/ in its "Walk on finite graph B" mode (derandomized biased random walk on {0,1,2,3}).)

More generally, if all the transition probabilities $p_{i1},...,p_{in}$ are rational with common denominator $D$, we associate with state $s_i$ some specified periodic sequence with period $D$, such that the frequency of each $s_j$ in the periodic sequence is $p_{ij}$, and we decree that the exit sequence at $s_i$ should be the specified periodic sequence.

Special case: if there are only two non-zero entries in the $i$th row of **P**, then there is a natural choice for the exit sequence. Specifically, suppose $p_{ia} = p$ and $p_{ib} = 1-p$. Then we can take the $k$th term of the exit sequence to be $a$ or $b$ according to whether nint($pk$) - nint($p$-($k$-1)) is 1 or 0, where nint($x$) denotes the integer nearest to $x$. Example: With $p = 2/3$, we get the sequence $abaabaaba$... described above.

```
p = 2 / 3
```

$$\frac{2}{3}$$

```
Table[Round[p k] - Round[p (k - 1)], {k, 1, 9}]
```

{1, 0, 1, 1, 0, 1, 1, 0, 1}

For next week, read "The Goldbug Variations" by Michael Kleber (based on a puzzle I sent him and some follow-up emails between us). This example moves us in the direction of random walk on infinite graphs, and Markov chains with infinitely many states, which is one of our next topics.

Similarities between rotor-walk and random walk

## Absorption probabilities

Consider an absorbing Markov chain with a particular transient state $s_i$ and a particular absorbing state $s_j$. Assume all matrix entries are rational.

Let $b_{ij}$ be the probability that the chain will be absorbed in the absorbing state $s_j$ if it starts in the transient state $s_i$.

If we repeatedly run **rotor walk** from state $s_i$ (returning to $s_i$ after each absorption, and NOT resetting the rotors), then the asymptotic proportion of the time that the rotor-walk from $s_i$ gets absorbed at $s_j$ is equal to $b_{ij}$.

More specifically, if $N(t)$ denotes the number of times that the rotor-walk gets absorbed at $s_j$ in the first $t$ runs, then $|N(t)/t - b_{ij}| = O(1/t)$, so that in particular $N(t)/t \to b_{ij}$. (Note that if we used random walk instead of rotor-walk, we'd still have $N(t)/t \to b_{ij}$, but with $|N(t)/t - b_{ij}| = O(1/\text{sqrt}(t))$.)

Expected time until absorption

Consider an absorbing Markov chain with particular transient states $s_i$ and $s_j$.

Let $n_{ij}$ be the expected number of times that the chain will be in state $s_j$ if it starts in state $s_i$.

If we repeatedly run rotor walk from state $s_i$ (returning to $s_i$ after each absorption, and NOT reset - ting the rotors), then the asymptotic average number of times the rotor-walk from $s_i$ visits $s_j$ is equal to $n_{ij}$.

More specifically, if $N(t)$ denotes the number of times that the rotor-walk started from $s_i$ visits $s_j$ in the first $t$ runs, then $|N(t)/t - n_{ij}| = O(1/t)$, so that in particular $N(t)/t \to n_{ij}$.

Summing over all transient states $s_j$, we find that the asymptotic average number of steps the rotor-walk from $s_i$ takes until it is absorbed is equal to the expected number of steps random walk from $s_i$ takes until it is absorbed.

Stationary probabilities

Now consider an ergodic Markov chain with a particular states $s_i$. Let $w_i$ be the stationary probability of $s_i$.

If we run rotor walk (from an arbitrary initial state), then the asymptotic proportion of the time the rotor-walk visits $s_i$ is equal to $w_i$.

More specifically, if $N(t)$ denotes the number of times that the rotor-walk visits $s_i$ in the first $t$ steps, then $|N(t)/t - w_i| = O(1/t)$, so that in particular $N(t)/t \to w_i$.

Example 1: Consider the Markov chain associated with random walk on {1,2,3,4} with reflecting boundaries. Then a rotor-walk for this Markov chain eventually just goes ...,1,2,3,4,3,2,1,2,3,4,3,2,... (period 6), which visits the states 1,2,3,4 with frequency 1/6, 2/6, 2/6, and 1/6, respectively.

Example 2: Consider the Markov chain associated with random walk on {1,2,3,4} with semi-reflecting boundaries. Then a rotor-walk for this Markov chain eventually just goes ...,1,2,3,4,4,3,2,1,1,2,3,4,4,3,2,1,... (period 8), which visits the states 1,2,3,4 with frequency 2/8, 2/8, 2/8, and 2/8, respectively.

Explaining the similarities: the harmonic functions perspective

I'll explain the situation for stationary probabilities; the explanation for absorption probabilities and for expected time until absorption goes along the same lines.

For each state $s_i$, let $V_i(t)$ be the number of times that the rotor-walk has been in state $s_i$ in the first $t$ time-steps, and let $V_{ij}(t)$ be the number of times (in the first $t$ time-steps) that the rotor-walk has been in state $s_i$ and then gone immediately to state $s_j$, so that for all $j$, $V_j(t)$ differs from $\sum_i V_{ij}(t)$ by either 0 or 1 (depending on whether the first state visited by the rotor-walk was $s_j$). Also note that $V_{ij}(t)$ differs from $p_{ij}\ V_i(t)$ by at most a constant $C_{ij}$ that does not depend on $t$. Combining these constants, we have

$$V_j(t) = (\sum_i p_{ij}\ V_i(t)) \pm C$$

for some single constant $C$ that does not depend on $t$. Dividing this by $t$, we get

$$V_j(t) / t = (\sum_i p_{ij}\ V_i(t) / t) \pm C / t$$

If we take the limit as $t$ goes to infinity, the $C / t$ goes away, and we are left with

(*) $v_j = \sum_i p_{ij} v_i$

where $v_i = \lim_{t\to\infty} V_i(t) / t$ is the asymptotic proportion of time that the rotor walk is in state $v_i$ (technically, one has to show that the limit exists, but this is taken care of by an easy combinatorial side-argument that I'll omit).

However, there is only one probability vector **v** satisfying (*), and it's the stationary probability vector **w**!

(Recall: If a Markov chain is ergodic, then 1 is a simple eigenvalue, so the space of row-vectors **v** satisfying **vP** = **v** is one-dimensional, and it contains only one vector with components summing to 1, namely **w**.)

For absorption probabilities and for expected time until absorption, the arguments are similar; use the

fact that for an absorbing Markov chain, a harmonic function **f** (i.e. a function **f** satisfying **Pf** = **f**) is uniquely determined by its values on the boundary of the Markov chain.

## Your final project

In this class, you will encounter a variety of random systems whose long-term behavior is captured by various numbers (absorption probabilities, stationary probabilities, mean first passage times, etc.).

One can **sometimes** calculate these numbers exactly.

One can **always** estimate these numbers by random simulation, although the simulation will have inherent statistical error (whose magnitude may or may not be easy

to estimate or bound), assuming you have access to a source of random bits (e.g. clicks from a Geiger counter).

Likewise, one can **always** estimate these numbers by pseudorandom simulation, if one trusts one's pseudorandom number generator.

Additionally, one can **sometimes** estimate these numbers by quasirandom simulation. In the ideal case, the exact answer would be difficult to compute exactly (making some sort of simulation necessary), and quasirandom simulation would efficiently give good approximations with provably small error (smaller than the typical error for pseudorandom simulation).

In this class, the goal of your final project is to take some probabilistic problem, study it by at least **two** of

the following three sorts of methods:

A) Rigorous analysis (possibly computer-aided)

B) Pseudorandom simulation

C) Quasirandom simulation (e.g., rotor-routing or chip-firing)

So, reconsider the different sorts of discrete stochastic processes we've seen in the past few weeks (in the homework and elsewhere), and start thinking about how you'd pseudorandomly or quasirandomly simulate them.

Or, take some problem you like from the voluminous literature on fun probability problems.

Fifty Challenging Problems in Probability, by Frederick Mosteller

Duelling Idiots and Other Probability Puzzlers, by Paul

Nahin (e.g., #9 and #18)

Digital Dice: Computational Solutions to Practical Probability Problems, by Paul Nahin (e.g. #4 and #18)

## Sampling from a specified distribution

Markov chain Monte Carlo

To sample from some probability distribution, construct a Markov chain for which it is the stationary distribution, and then run it!

Typically this is done for probability distributions on huge sets, but here's a toy example: Consider the probability distribution on {1,2,3,4} with $m(1) = .1$, $m(2) = .2$, $m(3) = .3$, and $m(4) = .4$. Put

```
w = {.1, .2, .3, .4}
```

{0.1, 0.2, 0.3, 0.4}

We have $\mathbf{wP} = \mathbf{w}$ for

```
P = {{0, 0, 0, 1}, {0, 0, 1 / 2, 1 / 2}, {0, 1 / 3, 1 / 3, 1 / 3}, {1 / 4, 1 / 4, 1 / 4, 1 / 4}}
```

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}$$

Check:

```
w.P
```

{0.1, 0.2, 0.3, 0.4}

So, if we run the Markov chain associated with the matrix **P** for many steps, and see what proportion of the time the respective states are visited, we will get convergence to **w**, by the Law of Large Numbers for ergodic Markov chains. There'll be two sources of discrepancy: general statistical error, and "burn-in" or "initialization" error.

In more realistic applications, we have a probability distribution on some huge set *S*, and often it isn't even given by an exact formula; we know the ratios $w(s)/w(s')$ for so many pairs $s,s'$ that these ratios determine the vector **w** uniquely up to a normalizing constant $Z$; but we aren't given $Z$ (and often can't even compute it after the fact, or even estimate it).

## Rejection sampling

(also known as acceptance-rejection sampling)

Suppose we want to sample from a distribution $\pi$ on $\{1,2,...,n\}$ that's not too far from the uniform distribution, in the sense that $\pi(k)$ lies between $\frac{0}{n}$ and $\frac{2}{n}$ for all $1 \le k \le n$. Then we can use the following iterative procedure:

(1) Generate an element $k$ of $\{1,2,...,n\}$ uniformly (the "proposal").

(2) With probability $\frac{1}{2}\pi(k)n$, accept this element (and output it); otherwise, reject it and go back to step 1.

Each time we go back to step 1, our new proposal should be independent of all previous proposals.

(Note: elements that have been rejected can be accepted in a later iteration.)

Step 2 makes sense, since $\frac{1}{2}\pi(k)n \leq \frac{1}{2}\frac{2}{n}n = 1$.

Since our probability of accepting $k$ is proportional to $\pi(k)$ as $k$ varies from 1 to $n$, it's plausible that the outcome of this sampling scheme is an element of { 1,2,...,$n$ } distributed according to $\pi$.

More rigorously: We first argue that the probability of the algorithm succeeding on the first try is
$\frac{1}{n}(\frac{1}{2}\pi(1)n) + \frac{1}{n}(\frac{1}{2}\pi(2)n) + ... + \frac{1}{n}(\frac{1}{2}\pi(n)n)$
$= \frac{1}{2}(\pi(1) + \pi(2) + ... + \pi(n)) = \frac{1}{2}$, and that the conditional probability of $k$ being accepted on the first round is
$\frac{1}{n}(\frac{1}{2}\pi(k)n) \,/\, \frac{1}{2} = \pi(k)$.

For subsequent rounds, the distribution is exactly the same.

In particular, if the algorithm has run for $m$-1 rounds without terminating, the probability that it terminates on the next round (the $m$th) is $\frac{1}{2}$.

This is true for all $m$.

Since the the algorithm always has a probability of $\frac{1}{2}$ of terminating on the current round, the number of rounds until termination is a geometric random variable of parameter $\frac{1}{2}$, so the algorithm terminates in a finite number of rounds with probability 1, and the expected number of rounds is 2.

(If you've forgotten the formula for the expected value of a geometric random variable with parameter $p$ is $1/p$, but you remember that it's finite, here's a quick way to derive the formula: Write $E(X) = 1 + (p)(0) + (1-p)E(Y)$, where $X$ and $Y$ are both distributed accord-

ing to Geometric($p$). Since $E(X) = E(Y)$, we get $E(X) = 1 + (1-p)E(X)$, so $pE(X) = 1$, so $E(X) = 1/p$. Technically this only proves that $E(X)$ is either infinity or $1/p$, but it's still worth something.)

Furthermore, when the algorithm terminates, it is governed by the distribution $\pi$:
Let $N$ be the (random) number of rounds, and let $O$ be the (random) output returned by the procedure.

$$\begin{aligned} P(O = k) &= \sum_{m=1}^{\infty} P(O = k \text{ and } N = m) \\ &= \sum_{m=1}^{\infty} P(N = m)\, P(O = k \mid N = m) \\ &= \sum_{m=1}^{\infty} P(N = m)\, \pi(k) \\ &= \pi(k) \sum_{m=1}^{\infty} P(N = m) \\ &= \pi(k)\, P(N \text{ is finite}) \\ &= \pi(k). \end{aligned}$$

More generally, if we have the ability to sample from a certain distribution $\pi_0$ on a set $S$ (in the preceding dis-

cussion the set $S$ was $\{1,2,\ldots,n\}$ and $\pi_0$ was the uniform distribution on $S$) and we want to sample from a different distribution $\pi$ on the set $S$, with $\pi$ satisfying the bound $\pi(s) \le M\pi_0(s)$ for all $s$ in $S$ (in the preceding discussion the coefficient $M$ was 2), we can sample from $\pi$ as follows:

(1) Generate an element $s$ of $S$ governed by $\pi_0$.
(2) With probability $\frac{1}{M}\pi(s)/\pi_0(s)$, accept this element (and output it); otherwise, reject it and go back to step 1.

The number of $\pi_0$-samples required on average to obtain one $\pi$-sample is $M$, so when $\pi(s)/\pi_0(s)$ takes on large values, this method takes a prohibitively long number of iterations.

Metropolis chains

What if $\pi(s)/\pi_0(s)$ takes on very large values, but it's in some sense "continuous" as a function of $s$?

That is, suppose we have some notion of "closeness" such that if $s$ and $s'$ are close, $\pi(s')/\pi_0(s')$ can't be too much bigger than $\pi(s)/\pi_0(s)$?

Then we might combine rejection sampling with a Markov-chain-on-$S$ approach, where we only move from a state $s$ to a nearby state $s'$. We still sometimes reject such moves after considering them, and stick with $s$, but we typically reject only when $\pi(s')/\pi(s)$ is small.

If $\pi(s')/\pi(s)$ isn't too small, then we should accept the move from $s$ to $s'$, even if $\pi(s)/\pi_0(s)$ and $\pi(s')/\pi_0(s')$ are tiny.

So what we get is a new Markov chain, in which a single step consists of first proposing a step in the old Markov chain and then either accepting the proposal (and moving to a new state) or rejecting the proposal (and staying put).

If we set up the revised transition probabilities correctly, then our new Markov chain will have $\pi$ as its unique stationary measure. Even if $\pi_0$ is very different from $\pi$, so that the starting state of our old underlying Markov chain (governed by $\pi_0$) is far from equilibrium, the new derived Markov chain may be sufficiently "rapidly mixing" that the distribution governing the state of the derived Markov chain at time $n$ quickly converges toward $\pi$.

We will choose our transition probabilities so that the new Markov chain is reversible. This will give us an easy way to verify that $\pi$ is indeed the stationary distribution for the chain, but it also gives us a lot of freedom.

Recall that a Markov chain is reversible with stationary distribution $\pi$ (some say: "$\pi$ is a reversible measure for the Markov chain")
iff it satisfies the detailed balance condition

$$\pi(i)\, p_{i,j} = \pi(j)\, p_{j,i}$$

for all states $i, j$. It suffices to consider $i \neq j$.

Suppose we have some other Markov chain on $S$ with stationary distribution $\pi_0$. Just as the i.i.d. process distributed according to $\pi_0$ was a source of candidates

for rejection sampling, our Markov chain with station - ary distribution $\pi_0$ (henceforth the "$\pi_0$-chain") is a source of candidate moves $s \to s'$ for our acceptance-rejection Markov chain (henceforth the "$\pi$-chain").

Let $p^{(0)}_{i,j}$ denote the transition probability from $i$ to $j$ for the $\pi_0$-chain.

Assume that our transition rule is of the form "If a move from $i$ to $j$ is proposed, accept with probability $\alpha_{i,j}$, otherwise reject." That is,

$$p_{i,j} = p^{(0)}_{i,j}\ \alpha_{i,j} \text{ for } j \neq i,$$

where $\alpha_{i,j}$ is our acceptance threshhold for moving from $i$ to $j$. So we can write the detailed balance condition for the $\pi$-chain as

(1) $$\pi(i)\ p^{(0)}_{i,j}\ \alpha_{i,j} = \pi(j)\ p^{(0)}_{j,i}\ \alpha_{j,i}\ .$$

How about taking

$\alpha_{i,j} = \pi(j)\, p^{(0)}_{j,i}$ and $\alpha_{j,i} = \pi(i)\, p^{(0)}_{i,j}$? ...

..?..

These acceptance probabilities are way too small, since in typical applications $S$ is huge and each individual $\pi(i)$ or $\pi(j)$ is tiny.

Okay, how about taking

$\alpha_{i,j} = C\,\pi(j)\, p^{(0)}_{j,i}$ and $\alpha_{j,i} = C\,\pi(i)\, p^{(0)}_{i,j}$

with $C$ really big? ...

..?..

That's helpful; making $C$ bigger increases the probability of acceptance (and unless acceptance happens, transitions are rejected and we make no progress).

But: we'd better not take $C$ so large that the "acceptance probability" $\alpha_{i,j}$ (or $\alpha_{j,i}$ for that matter) is bigger than 1!

Note that in many applications, we don't know $\pi(i)$ and $\pi(j)$; we only know their ratio.

(Fortunately, the ratio is all we need to know, as we'll see shortly.)

Idea: We can always make $C$ bigger until one of the acceptance threshholds $\alpha_{i,j}$, $\alpha_{j,i}$ hits 1. So we can take one of them to equal 1 (and the other will be at most 1). But which?

For any choice of $C$, we have

$\alpha_{i,j} / \alpha_{j,i} = (\pi(j)/\pi(i)) (p^{(0)}_{j,i} / p^{(0)}_{i,j})$.

If the RHS is $\leq 1$, then we can put

$\alpha_{j,i} = 1$ and $\alpha_{i,j} = \text{RHS} = (\pi(j)/\pi(i)) (p^{(0)}_{j,i} / p^{(0)}_{i,j})$.

Otherwise, we put

$\alpha_{i,j} = 1$ and $\alpha_{j,i} = (\pi(i)/\pi(j)) (p^{(0)}_{i,j} / p^{(0)}_{j,i})$.

That is, we put

$$\alpha_{i,j} = \min(1, (\pi(j)/\pi(i))\ (p^{(0)}_{j,i}/p^{(0)}_{i,j})).$$

This is the Metropolis algorithm for sampling from the distribution $\pi$.

If our original Markov chain has stationary measure $\pi_0$, then the associated Metropolis algorithm Markov chain has stationary measure $\pi$, and if the ratios $(\pi(j)/\pi(i))$ and $(p^{(0)}_{j,i}/p^{(0)}_{i,j})$ stay close to 1 for all $i,j$ with $p^{(0)}_{i,j} > 0$, then a significant fraction of the proposed moves are accepted, and the Metropolis chain converges to $\pi$ almost as quickly as the original chain converges to $\pi_0$.

If $\pi_0$ is a reversible measure for the transition probabilities $p^{(0)}_{i,j}$, then $p^{(0)}_{j,i}/p^{(0)}_{i,j} = \pi_0(i)/\pi_0(j)$,
so that we can also write the formula as

$$\alpha_{i,j} = \min(1, (\pi(j)/\pi(i))\ (\pi_0(i)/\pi_0(j))).$$

Let's try this in a simple 3-state case, with S = {1,2,3}, $\pi_0$=($\frac{1}{4}$,$\frac{1}{2}$,$\frac{1}{4}$), and using simple random walk on {1,2,3} as our $\pi_0$-chain, with transition matrix

$$\begin{pmatrix} 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \end{pmatrix}$$

You can check that $\pi_0$ is not just a stationary measure for this Markov chain; it's a reversible measure.
We want to sample from $\pi$=($\frac{1}{3}$,$\frac{1}{3}$,$\frac{1}{3}$), the uniform distri - bution on {1,2,3}, by hitching a ride on the transition probabilities for simple random walk. We get
$\alpha_{1,2} = \min(1, \frac{1/3}{1/3}\frac{1/4}{1/2}) = \frac{1}{2}$ (= $\alpha_{3,2}$ by symmetry) and
$\alpha_{2,1} = \min(1, \frac{1/3}{1/3}\frac{1/2}{1/4}) = 1$ (= $\alpha_{2,3}$ by symmetry), so
$p_{1,2} = p^{(0)}_{1,2}\ \alpha_{1,2} = (1)(\frac{1}{2}) = \frac{1}{2} = p_{3,2}$ and
$p_{2,1} = p^{(0)}_{2,1}\ \alpha_{2,1} = (\frac{1}{2})(1) = \frac{1}{2} = p_{2,3}$.
We get the transition matrix

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

and we can see that $\pi$ is stationary for this matrix. (So the semi-absorbent-barrier trick is a special case of Metropolis.)

More generally, the Metropolis algorithm can modify simple random walk on any finite connected graph to achieve a uniform stationary distribution on the vertices of the graph. The acceptance probability reduces in this case to $\min(1, \deg(v_i)/\deg(v_j))$, which biases the walk against moving to high-degree vertices. Note that we need to know next to nothing about the global structure of a graph to implement this (although if the graph isn't connected, the procedure won't converge to the desired distribution, and if the graph is really big and/or loosely connected, the procedure may not converge quickly).

In many applications to physics, we have $\pi(i)$ given by $f(i)/Z$ for all $i$, where the non-negative function $f(i)$ is of the form $e^{-E(i)/T}$ for some energy function $E(i)$ and some temperature $T$, and where the partition function $Z$ equals $\sum_i f(i)$ (called a "function" because it depends on $T$). In this case, the probability ratio $\pi(j)/\pi(i)$ is of the form $e^{-\Delta E/T}$ where $\Delta E = E(j) - E(i)$.