

Steven R. Dunbar  
Department of Mathematics  
203 Avery Hall  
University of Nebraska-Lincoln  
Lincoln, NE 68588-0130  
<http://www.math.unl.edu>  
Voice: 402-472-3731  
Fax: 402-472-8466

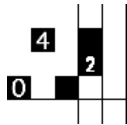
## Stochastic Processes and Advanced Mathematical Finance

---

Transformations of the Wiener Process

---

---



### Rating

Mathematically Mature: may contain mathematics beyond calculus with proofs.

---



## Section Starter Question

Suppose you know the graph  $y = f(x)$  of the function  $f(x)$ . What is the effect on the graph of the transformation  $f(x + h) - f(h)$ ? What is the effect on the graph of the transformation  $f(1/x)$ ? Consider the function  $f(x) = \sin(x)$  as an example.

---



## Key Concepts

1. Three transformations of the Wiener process produce another Wiener process. The transformations are scaling, inversion and translation. These results are especially helpful when studying the properties of the sample paths of Brownian motion.
- 



## Vocabulary

1. **Scaling**, also called **re-scaling**, is the transformation of  $f(t)$  to  $bf(t/a)$  that expands or contracts the time axis (as  $a > 1$  or  $a < 1$ ) and expands or contracts the dependent variable scale (as  $b > 1$  or  $b < 1$ ).
2. **Translation**, also called **shifting** is the transformation of  $f(t)$  to  $f(t + h)$  or sometimes  $f(t)$  to  $f(t + h) - f(h)$ .
3. **Inversion** is the transformation of  $f(t)$  to  $f(1/t)$ . It “flips” the independent variable axis about 1, so that the interval  $(0, 1)$  is “inverted” to the interval  $(1, \infty)$ .



## Mathematical Ideas

### Transformations of the Wiener Process

A set of transformations of the Wiener process produce the Wiener process again. Since these transformations result in the Wiener process, each tells us something about the “shape” and “characteristics” of the Wiener process. These results are especially helpful when studying the properties of the Wiener process sample paths. The first of these transformations is a time homogeneity that says the Wiener process can be re-started anywhere. The second says that the Wiener process may be rescaled in time and space. The third is an inversion. Roughly, each of these says the Wiener process is self-similar in various ways. See the comments after the proof for more detail.

**Theorem 1.** *Let*

1.  $W_{\text{shift}}(t) = W(t + h) - W(h)$ , for fixed  $h > 0$ .

2.  $W_{\text{scale}}(t) = cW(t/c^2)$ , for fixed  $c > 0$ .

*Then each of  $W_{\text{shift}}(t)$  and  $W_{\text{scale}}(t)$  are a version of the Standard Wiener Process.*

*Proof.* We have to systematically check each of the defining properties of the Wiener process in turn for each of the transformed processes.

1.

$$W_{\text{shift}}(t) = W(t + h) - W(h).$$

(a) The increment is

$$\begin{aligned} W_{\text{shift}}(t + s) - W_{\text{shift}}(s) &= [W(t + s + h) - W(h)] - [W(s + h) - W(h)] \\ &= W(t + s + h) - W(s + h) \end{aligned}$$

which is by definition normally distributed with mean 0 and variance  $t$ .

(b) The increment

$$W_{\text{shift}}(t_4) - W_{\text{shift}}(t_3) = W(t_4 + h) - W(t_3 + h)$$

is independent from

$$W_{\text{shift}}(t_2) - W_{\text{shift}}(t_1) = W(t_2 + h) - W(t_1 + h)$$

by the property of independence of disjoint increments of  $W(t)$ .

(c)

$$W_{\text{shift}}(0) = W(0 + h) - W(h) = 0.$$

(d) As the composition and difference of continuous functions,  $W_{\text{shift}}$  is continuous.

2.

$$W_{\text{scale}}(t) = cW(t/c^2)$$

(a) The increment

$$\begin{aligned} W_{\text{scale}}(t) - W_{\text{scale}}(s) &= cW(t/c^2) - cW(s/c^2) \\ &= c(W(t/c^2) - W(s/c^2)) \end{aligned}$$

is normally distributed because it is a multiple of a normally distributed random variable. Since the increment  $W(t/c^2) - W(s/c^2)$  has mean zero, then

$$W_{\text{scale}}(t) - W_{\text{scale}}(s) = c(W(t/c^2) - W(s/c^2))$$

must have mean zero. The variance is

$$\begin{aligned} \mathbb{E} [(W_{\text{scale}}(t) - W_{\text{scale}}(s))^2] &= \mathbb{E} [(cW(t/c^2) - cW(s/c^2))^2] \\ &= c^2 \mathbb{E} [(W(t/c^2) - W(s/c^2))^2] \\ &= c^2(t/c^2 - s/c^2) = t - s. \end{aligned}$$

(b) Note that if  $t_1 < t_2 \leq t_3 < t_4$ , then  $t_1/c^2 < t_2/c^2 \leq t_3/c^2 < t_4/c^2$ , and the corresponding increments  $W(t_4/c^2) - W(t_3/c^2)$  and  $W(t_2/c^2) - W(t_1/c^2)$  are independent. Then the multiples of each by  $c$  are independent and so  $W_{\text{scale}}(t_4) - W_{\text{scale}}(t_3)$  and  $W_{\text{scale}}(t_2) - W_{\text{scale}}(t_1)$  are independent.

(c)  $W_{\text{scale}}(0) = cW(0/c^2) = cW(0) = 0$ .

(d) As the composition of continuous functions,  $W_{\text{scale}}$  is continuous. □

**Theorem 2.** *Suppose  $W(t)$  is a Standard Wiener Process. Then the transformed processes  $W_{\text{inv}}(t) = tW(1/t)$  for  $t > 0$ ,  $W_{\text{inv}}(t) = 0$  for  $t = 0$  is a version of the Standard Wiener Process.*

*Proof.* To show that

$$W_{\text{inv}}(t) = tW(1/t)$$

is a Wiener process by the four defining properties requires another fact which is outside the scope of the text. The fact is that any Gaussian process  $X(t)$  with mean 0 and  $\text{Cov}[X(s), X(t)] = \min(s, t)$  must be the Wiener process. See the references and outside links for more information. Using this information, a partial proof follows:

1.

$$W_{\text{inv}}(t) - W_{\text{inv}}(s) = tW(1/t) - sW(1/s)$$

is the difference of normally distributed random variables each with mean 0, so the difference will be normal with mean 0. It remains to check that the normal random variable has the correct variance.

$$\begin{aligned} & \mathbb{E} [(W_{\text{inv}}(t) - W_{\text{inv}}(s))^2] \\ &= \mathbb{E} [(sW(1/s) - tW(1/t))^2] \\ &= \mathbb{E} [(sW(1/s) - sW(1/t) \\ &\quad + sW(1/t) - tW(1/t) - (s-t)W(0))^2] \\ &= s^2 \mathbb{E} [(W(1/s) - W(1/t))^2] \\ &\quad + s(s-t) \mathbb{E} [(W(1/s) - W(1/t))(W(1/t) - W(0))] \\ &\quad + (s-t)^2 \mathbb{E} [(W(1/t) - W(0))^2] \\ &= s^2 \mathbb{E} [(W(1/s) - W(1/t))^2] + (s-t)^2 \mathbb{E} [(W(1/t) - W(0))^2] \\ &= s^2(1/s - 1/t) + (s-t)^2(1/t) \\ &= t - s. \end{aligned}$$

Note the use of independence of  $W(1/s) - W(1/t)$  from  $W(1/t) - W(0)$  at the third equality.

2. It is hard to show the independence of increments directly. Instead rely on the fact that a Gaussian process with mean 0 and covariance function  $\min(s, t)$  is a Wiener process, and thus prove it indirectly.

Note that

$$\text{Cov}[W_{\text{inv}}(s), W_{\text{inv}}(t), =] st \min(1/s, 1/t) = \min(s, t).$$

3. By definition,  $W_{\text{inv}}(0) = 0$ .
4. The argument that  $\lim_{t \rightarrow 0} W_{\text{inv}}(t) = 0$  is equivalent to showing that  $\lim_{t \rightarrow \infty} W(t)/t = 0$ . To show this requires use of Kolmogorov's inequality for the Wiener process and clever use of the Borel-Cantelli lemma and is beyond the scope of this course. Use the translation property in the third statement of this theorem to prove continuity at every value of  $t$ .

□

The following comments are adapted from *Stochastic Calculus and Financial Applications* by J. Michael Steele. Springer, New York, 2001, page 40. These laws tie the Wiener process to three important groups of transformations on  $[0, \infty)$ , and a basic lesson from the theory of differential equations is that such symmetries can be extremely useful. On a second level, the laws also capture the somewhat magical fractal nature of the Wiener process. The scaling law tells us that if we had even one-billionth of a second of a Wiener process path, we could expand it to a billions years' worth of an equally valid Wiener process path! The translation symmetry is not quite so startling, it merely says that Wiener process can be restarted anywhere. That is, any part of a Wiener process captures the same behavior as at the origin. The inversion law is perhaps most impressive, it tells us that the first second of the life of a Wiener process path is rich enough to capture the behavior of a Wiener process path from the end of the first second until the end of time.



# Algorithms, Scripts, Simulations

## Algorithm

Let constants  $h$  and  $c$  be given. Create an approximation  $\hat{W}(t)$  of Wiener process  $W(t)$  on a domain  $[0, T]$  large enough to accommodate  $\hat{W}(1+h)$  and  $\hat{W}(1/c^2)$ . Using the approximation  $\hat{W}(t)$  create approximations to  $W_{\text{shift}}(t)$  and  $W_{\text{scale}}(t)$  with functions  $\hat{W}(t+h) - \hat{W}(h)$  and  $c\hat{W}(t/c^2)$ . Plot all functions on the same set of axes.

## Scripts

**Geogebra** GeoGebra

**R** R script for transformations.

```
1 p <- 0.5
2 N <- 400
3
4 T <- 2
5 h <- 0.25
6 c <- 2.0
7
8 S <- array(0, c(N+1))
9 rw <- cumsum( 2 * ( runif(N) <= p)-1 )
10 S[2:(N+1)] <- rw
11
12 WcaretN <- function(x) {
13   Delta <- T/N
14
15   # add 1 since arrays are 1-based
16   prior = floor(x/Delta) + 1
17   subsequent = ceiling(x/Delta) + 1
18
19   retval <- sqrt(Delta)*(S[prior] + ((x/Delta+1) -
20     prior)*(S[subsequent] - S[prior]))
21 }
22
23 Wshift <- function(x) {
24   retval <- WcaretN(x+h) - WcaretN(h)
25 }
26
27 Wscale <- function(x) {
28   retval <- c*WcaretN(x/c^2)
```

```

28 }
29
30 curve(WcaretN, 0,1, n=400, col = "black")
31 curve(Wshift, 0,1, n=400, add = TRUE, col = "blue")
32 curve(Wscale, 0,1, n= 400, add = TRUE, col = "red")

```

## Octave Octave script for transformations

```

1 p = 0.5;
2
3 global N = 400;
4 global T = 2;
5
6 global h = 0.25;
7 global c = 2.0;
8
9 global S
10 S = zeros(N+1, 1);
11 S(2:N+1) = cumsum( 2 * (rand(N,1)<=p) - 1);
12
13 function retval = WcaretN(x)
14     global N;
15     global T;
16     global S;
17     Delta = T/N;
18
19     # add 1 since arrays are 1-based
20     prior = floor(x/Delta) + 1;
21     subsequent = ceil(x/Delta) + 1;
22
23     retval = sqrt(Delta)*(S(prior) + ((x/Delta+1) - prior)
24         .* (S(subsequent)-S(prior)));
25 endfunction
26
27 function retval = Wshift(x)
28     global h;
29     retval = WcaretN(x + h) - WcaretN(h);
30 endfunction
31
32 function retval = Wscale(x)
33     global c;
34     retval = c*WcaretN(x/c^2);
35 endfunction

```



```

36
37 fplot("[WcaretN(x), Wshift(x), Wscale(x)]", [0,1])

```

## Perl Perl PDL script for transformations

```

1 use PDL::NiceSlice;
2
3 $p = 0.5;
4
5 $N = 400;
6 $T = 2;
7 $h = 0.25;
8 $c = 2.0;
9
10 # the random walk
11 $S = zeros( $N + 1 );
12 $S ( 1 : $N ) .= cumsumover( 2 * ( random($N) <= $p ) -
13     1 );
14 # function WcaretN interpolating random walk
15 sub WcaretN {
16     my $x = shift @_;
17     $Delta = $T / $N;
18
19     $prior      = floor( $x / $Delta );
20     $subsequent = ceil( $x / $Delta );
21
22     $retval =
23         sqrt($Delta)
24         * ( $S ($prior)
25             + ( ( $x / $Delta ) - $prior )
26             * ( $S ($subsequent) - $S ($prior) ) );
27 }
28
29 sub Wshift {
30     my $x = shift @_;
31     $retval = WcaretN( $x + $h ) - WcaretN($h);
32 }
33
34 sub Wscale {
35     my $x = shift @_;
36     $retval = $c * WcaretN( $x / ( $c * $c ) );
37 }
38

```

```

39 # file output to use with external plotting programming
40 # such as gnuplot
41 # Start gnuplot, then from gnuplot prompt
42 # plot "wienerprocess.dat" using 1:2 with lines title '
    WcaretN', "wienerprocess.dat" using 1:3 with lines
    title 'Wshift', "wienerprocess.dat" using 1:4 with
    lines title 'Wscale'
43 $M      = 300;
44 $tgrid = zeros( $M + 1 )->xlinvals( 0, 1 );
45 $W      = WcaretN($tgrid);
46 $Wsh   = Wshift($tgrid);
47 $Wsc   = Wscale($tgrid);
48
49 open( F, ">wienerprocess.dat" ) || die "cannot write: $!
    ";
50 foreach $j ( 0 .. $M ) {
51     print F $tgrid->range( [$j] ), " ", $W->range(    [$j]
    ), " ",
52         $Wsh->range(      [$j] ), " ", $Wsc->range( [$j]
    ), "\n";
53 }
54 close(F);

```

## SciPy Scientific Python script for transformations

```

1
2 import scipy
3
4 p = 0.5
5
6 N = 400
7 T = 2.
8 h = 0.25
9 c = 2.0
10
11 # the random walk
12 S = scipy.zeros(N+1)
13 S[1:N+1] = scipy.cumsum( 2*( scipy.random.random(N) <= p
    ) - 1 )
14
15 def WcaretN(x):
16     Delta = T/N
17     prior = scipy.floor(x/Delta).astype(int)
18     subsequent = scipy.ceil(x/Delta).astype(int)

```

```

19     return scipy.sqrt(Delta)*(S[prior] + (x/Delta - prior
20     )*(S[subsequent] - S[prior]))
21
22 def Wshift(x):
23     return WcaretN(x + h) - WcaretN(h)
24
25 def Wscale(x):
26     return c*WcaretN(x/c**2.)
27
28 M = 300
29 tgrid = scipy.linspace(0, 1, M+1)
30 W = WcaretN(tgrid)
31 Wsh = Wshift(tgrid)
32 Wsc = Wscale(tgrid)
33
34 # optional file output to use with external plotting
35 # programming
36 # such as gnuplot, R, octave, etc.
37 # Start gnuplot, then from gnuplot prompt
38 # plot "wienerprocess.dat" using 1:2 with lines title '
39 # WcaretN', "wienerprocess.dat" using 1:3 with lines
40 # title 'Wshift', "wienerprocess.dat" using 1:4 with
41 # lines title 'Wscale'
42
43 f = open('wienerprocess.dat', 'w')
44 for j in range(0,M+1):
45     f.write( str(tgrid[j])+ ' '+str(W[j])+ ' '+str(Wsh[j])+
46     ' '+str(Wsc[j])+ '\n');
47
48 f.close()

```

---

## Sources

This section is adapted from: *A First Course in Stochastic Processes* by S. Karlin, and H. Taylor, Academic Press, 1975, pages 351–353 and *Financial Derivatives in Theory and Practice* by P. J. Hunt and J. E. Kennedy, John Wiley and Sons, 2000, pages 23–24.

---



## Problems to Work for Understanding

1. Explain why there is no script for simulation or approximation of the inversion transformation of the Wiener process, or if possible provide such a script.
2. Given the piecewise linear approximation  $\hat{W}_N(t)$ , what are the slopes of the piecewise linear segments of the scaling transformation  $c\hat{W}_N(t/c^2)$ ?
3. Modify the scripts to plot an approximation of  $W_{\text{scale}}(t)$  on  $[0, 1]$  with the same degree of approximation as  $\hat{W}_N(t)$  for some  $N$ . Plot both on the same set of axes.
4. Show that  $st \min(1/s, 1/t) = \min(s, t)$



## Reading Suggestion:

## References

- [1] P. J. Hunt and J. E. Kennedy. *Financial Derivatives in Theory and Practice*. John Wiley and Sons, 2000. HG 6024 A3H86 2000.
- [2] S. Karlin and H. Taylor. *A Second Course in Stochastic Processes*. Academic Press, 1981.
- [3] J. Michael Steele. *Stochastic Calculus and Financial Applications*. Springer-Verlag, 2001. QA 274.2 S 74.



## Outside Readings and Links:

1. Russell Gerrard, City University, London, Stochastic Modeling Notes for the MSc in Actuarial Science, 2003-2004. Contributed by S. Dunbar October 30, 2005.

---

I check all the information on each page for correctness and typographical errors. Nevertheless, some errors may occur and I would be grateful if you would alert me to such errors. I make every reasonable effort to present current and accurate information for public use, however I do not guarantee the accuracy or timeliness of information on this website. Your use of the information from this website is strictly voluntary and at your risk.

I have checked the links to external sites for usefulness. Links to external websites are provided as a convenience. I do not endorse, control, monitor, or guarantee the information contained in any external website. I don't guarantee that the links are active at all times. Use the links here with the same caution as you would all information on the Internet. This website reflects the thoughts, interests and opinions of its author. They do not explicitly represent official positions or policies of my employer.

Information on this website is subject to change without notice.

Steve Dunbar's Home Page, <http://www.math.unl.edu/~sdunbar1>

Email to Steve Dunbar, `sdunbar1 at unl dot edu`

Last modified: Processed from L<sup>A</sup>T<sub>E</sub>X source on July 28, 2016