Estimating π from raindrops

David Jon Furbish
Emeritus, Vanderbilt University

A delightful way to estimate the number $\pi = 3.14159...$ involves simultaneously counting raindrop impacts on a circular sensor and a square sensor during a rainstorm. Experimental demonstrations of this idea occasionally are posted on various websites. Perhaps understandably, the explanations provided with these demonstrations focus on the experimental measurements and calculations, and offer little regarding the physical basis of why the procedure leads to estimates of π. Here I fill in some of the physical details accompanied by a Monte Carlo code that illustrates the uncertainty in the procedure.

At a local scale, raindrop impacts on a surface represent a Poisson process in space and time (Uijlenhoet et al., 1999; Jameson and Kostinski, 2002; Larsen et al., 2005; Bako et al., 2017). Consider a horizontal area A impacted by drops during a rainstorm. With respect to space, the set of xy coordinate positions of impact locations after a period of time t is completely spatially random. With respect to time, let w denote the wait times (or inter-arrival time) between successive drop impacts on A. A Poisson process has the property that the probability distribution $f_w(w)$ of wait times w is exponential,

$$f_w(w) = \frac{1}{\mu_w} e^{-w/\mu_w} \,, \qquad (1)$$

where μ_w denotes the mean wait time. Moreover, successive wait times between impacts are completely independent. This independence combined with the self-similar properties of the exponential distribution (1) means that the number of impacts $N(t)$ occurring within an interval of time t is entirely independent of the number occurring within any other non-overlapping interval of time. A Poisson process therefore is memoryless.

Let R denote the radius of a circular impact sensor and let D denote the edge length of a square impact sensor. Then, let J denote the raindrop impact rate, the number of impacts per square meter per second. As a point of reference, $J = 1\,000$ m^{-2} s^{-1} represents a heavy rainstorm (Smith et al., 2009). The expected Poisson impact rate on the circular sensor is $\lambda_R = \pi R^2 J$ (s^{-1}) and the expected Poisson impact rate on the square sensor is $\lambda_D = D^2 J$ (s^{-1}). Now let $N_R(t)$ denote the number of impacts on the circular sensor at time t and let $N_D(t)$ denote the number of impacts on the square sensor with $N_R(0) = N_D(0) = 0$. These numbers at time t are described by a Poisson distribution with expected values $\mathrm{E}[N_R(t)] = \pi R^2 Jt$ and $\mathrm{E}[N_D(t)] = D^2 Jt$ and variances $\mathrm{V}[N_R(t)] = \pi R^2 Jt$ and $\mathrm{V}[N_D(t)] = D^2 Jt$. We now take the ratio of the numbers $N_R(t)$ and $N_D(t)$ to give

$$\lim_{t\to\infty} \frac{D^2}{R^2}\frac{N_R(t)}{N_D(t)} = \frac{D^2}{R^2}\frac{\mathrm{E}[N_R(t)]}{\mathrm{E}[N_D(t)]} = \pi \qquad (2)$$

This shows that as the numbers $N_R(t)$ and $N_D(t)$ increase with time t, then in the limit of $t \to \infty$ their ratio converges to the number π. Notice that this result is independent of the impact rate J.

An experiment with impact sensors collects the time stamps of impacts during a storm and successively calculates the ratio (2). As the number of impacts increases, the value of the estimate of π tends to converge to the true value. The demonstrations that I have seen usually involve one realization in which the value approached 3.14 after a few thousand drop impacts. What is not usually shown is the uncertainty in this estimate of π. Recall that at time t the num-

bers $N_R(t)$ and $N_D(t)$ possess uncertainty as reflected by their finite variances. This means that there exists a great number of possible outcomes of the ratio (2) at any time t. That is, this ratio is described by a distribution of values at time t.

Appended below is a MATLAB code that illustrates this point. If you have MATLAB or GNU Octave, just copy and paste it into a script and it should run fine. It plots estimates of π for a specified number realizations over both time t and the total number of impacts $N(t)$. If you run another programming language, then translation should not be difficult.

References

[1] Bako, A. N., Darboux, F., and Lucas, C. (2017) Raindrop interaction in interrill erosion for steady rainfalls: A probabilistic approach, *Water Resources Research*, 53, 4361–4375, doi: 10.1002/2017WR020568.

[2] Jameson, A. R. and Kostinski, A. B. (2002) When is rain steady, *Journal of Applied Meterology and Climatology*, 41, 83–90, doi: https://doi.org/10.1175/1520-0450(2002)041¡0083:WIRS¿2.0.CO;2.

[3] Larsen, M. L., Kostinski, A. B., and Tokay, A. (2005) Observations and analysis of uncorrelated rain, *Journal of the Atmospheric Sciences*, 62, 4071–4083, doi: 10.1175/JAS3583.1.

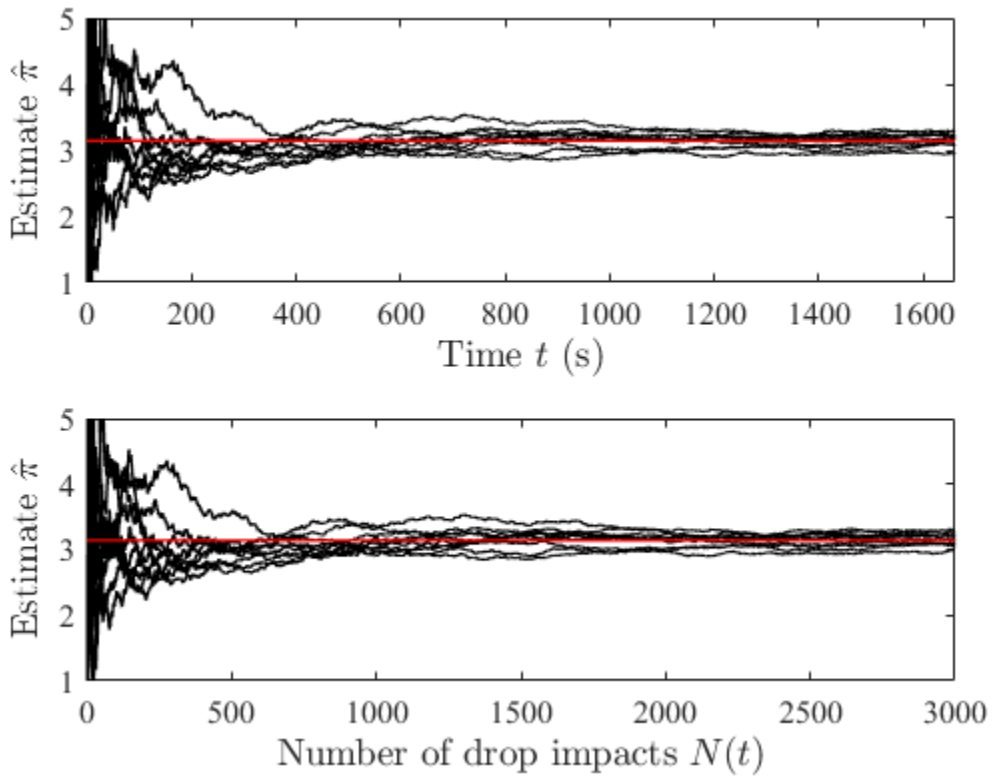
[4] Smith, J. A., Hui, E., Steiner, M., Baeck, M. L., Krajewski, W. F., and Ntelekos, A. A. (2009) Variability of rainfall rate and raindrop size distributions in heavy rain, *Water Resources Research*, 45, W04430.

[5] Uijlenhoet, R., Stricker, J. N. M, Torfs, P. J. J. F., and Creutin, J. D. (1999) Towards a stochastic model of rainfall for radar hydrology: Testing the Poisson homogeneity hypothesis, *Physics and Chemistry of the Earth*, Part B, 24, 747–755, doi: 10.1016/S1464-1909(99)00076-3.

```
clear;

R = 0.05; % circular sensor radius (m)
D = 0.1; % square sensor edge length (m)
J = 100; % raindrop rate (number of drops per square meter per second)

AR = pi*R^2;
AD = D^2;
A = AR + AD;
lambda = J*A; % Poisson impact rate
mu = 1/lambda;

RR = AR/A;
DR = D^2/R^2;
Nd = 3000; % total number of drop impacts
Nr = 10; % number of realizations

for j=1:Nr
    t(1,j) = exprnd(mu);
    N(1,j) = 1;
end
for j=1:Nr
    for k=2:Nd
        t(k,j) = t(k-1,j) + exprnd(mu);
        N(k,j) = k;
    end
end

for j=1:Nr
    lastR(j) = 0;
    lastD(j) = 0;
end

for j=1:Nr
    for k=1:Nd
        temp = rand;
        if temp <= RR
            lastR(j) = lastR(j) + 1;
        else
            lastD(j) = lastD(j) + 1;
        end
        pi_hat(k,j) = DR*lastR(j)/lastD(j);
    end
end

estimate = mean(pi_hat(Nd,:)) % average of final values of Nr realizations

X1 = [0 min(t(Nd,:))];
X2 = [0 Nd];
Y = [pi pi];

subplot(2,1,1)
```

```
plot(t,pi_hat(:,:),'k',X1,Y,'r','LineWidth',1)
axis([0 min(t(Nd,:)) 1 5])
set(gca,'FontName','Times New Roman')
set(gca,'Fontsize',12)
xlabel('Time $t$ (s)','Interpreter','Latex','Fontsize',14)
ylabel('Estimate $\hat \pi$','Interpreter','Latex','Fontsize',14)

subplot(2,1,2)
plot(N,pi_hat(:,:),'k',X2,Y,'r','LineWidth',1)
axis([0 Nd 1 5])
set(gca,'FontName','Times New Roman')
set(gca,'Fontsize',12)
xlabel('Number of drop impacts $N(t)$','Interpreter','Latex','Fontsize',14)
ylabel('Estimate $\hat \pi$','Interpreter','Latex','Fontsize',14)
```

```
estimate =

   3.166308446475466
```



Published with MATLAB® R2022b