The Fountain That Math Built

Alex McCauley
Josh Michener
Jadrian Miles
North Carolina School of Science and Mathematics
Durham, NC

Advisor: Daniel J. Teague

Introduction

We are presented with a fountain in the center of a large plaza, which we wish to be as attractive as possible but not to splash passersby on windy days. Our task is to design an algorithm that controls the flow rate of the fountain, given input from a nearby anemometer.

During calm, the fountain sprays out water at a steady rate. When the wind picks up, the flow should be attenuated so as to keep the water within the fountain's pool; in this way, we strike a balance between esthetics and comfort.

We consider the water stream from the fountain as a collection of different-sized droplets that initially leave the fountain nozzle in the shape of a perfect cylinder. This cylinder is broken into its component droplets by the wind, with smaller droplets carried farther. In the reference frame of the air, a droplet is moving through stationary air and experiencing a drag force as a result; since the air is moving with a constant velocity relative to the fountain, the force on the droplet is the same in either frame of reference.

Modeling this interaction as laminar flow, we arrive at equations for the drag forces. From these equations, we derive the acceleration of the droplet, which we integrate to find the equations of motion for the droplet. These allow us to find the time when the droplet hits the ground and—assuming that it lands at the very edge of the pool—the time when it reaches its maximum range from the horizontal position equation. Equating these and solving the initial flow rate, we arrive at an equation for the optimal flow rate at a given constant wind speed. Since the wind speeds are not constant, the algorithm must make its best prediction of wind speed and use current and previous wind speed measurements to damp out transient variations.

The UMAP Journal 23 (3) (2002) 221–234.

Our final solution is an algorithm that takes as its input a series of wind speed measurements and determines in real time the optimal flow rate to maximize the attractiveness of the fountain while avoiding splashing passersby excessively. Each iteration, it adds an inputted wind speed to a buffer of previous measurements. If the wind speed is increasing sufficiently, the last 0.5 s of the buffer are considered; otherwise, the last 1 s is. The algorithm computes a weighted average of these wind speeds, weighting the most recent value slightly more than the oldest value considered. It uses this weighted velocity average in the equation that predicts the optimal flow rate under constant wind. The result is the optimal flow rate under variable wind, knowing only current and previous wind speeds.

A list of relevant variables, constants, and parameters is in **Table 1**.

Table 1.

Relevant constants, variables, and parameters.

Physical constants	Description	Value
η_a	Viscosity of air	1.849×10^{-5} kg/m·s [Lide 1999]
ρ_w	Density of water	1000 kg/m^3
ρ_a	Density of air	1.2×10^{-6} kg/m^3
Situational constants		Units
A	Cross-sectional area of fountain nozzle	m^2
$f_{\max}$	Maximum flow rate of fountain's pump	m^3/s
R_p	Radius of fountain pool	m
r	Radius of smallest uncomfortable water droplet	m
dt	Sampling interval of anemometer	s
k	$k = 9\eta_a/2\rho_w r^2$	
Situational variables		
v_a	Instantaneous wind speed	m/s
f	Instantaneous flow rate of water from the fountain	m^3/s
n	$n = g/k + f/A$	m/s
Dynamic variables		
$x(t), y(t)$	Droplet's horizontal and vertical positions	m
$v_x(t), v_y(t)$	Droplet's horizontal and vertical speeds	m/s
$a_x(t), a_y(t)$	Droplet's horizontal and vertical accelerations	m/s^2
Situational parameters		
τ_d	Default sample wind velocity buffer time	s
τ_i	Buffer time for quickly increasing sample wind velocities	s
K	Weight constant	dimensionless

Assumptions

- Passersby find a higher spray more attractive.
- Avoiding discomfort is more important to passersby than the attractiveness of the fountain.
- The water stream can be considered a collection of spherical droplets, each of which has no initial horizontal component of velocity.
- Every possible size of sufficiently small water droplet is represented in the water stream in significant numbers.
- Water droplets remain spherical.
- The interaction between the water droplets and wind can be described as non-turbulent, or "laminar," flow.
- There exists a minimum uncomfortable water droplet size; passersby find it acceptable to be hit by any droplets below this size but by none above.
- When the wind enters the plaza, its velocity is entirely horizontal.
- The wind speed is the same throughout the plaza at any given time.
- The pool and the area around it are radially symmetric, so there is no preferred radial direction.
- We can neglect any buoyant force on the water due to the air, since the error introduced by this approximation is equal to the ratio of densities of the fluids involved, on the order of 10^{-3}, which is negligible.
- The anemometer reports wind speeds at discrete time intervals dt.

Analysis of the Problem

For a water stream viewed as a collection of small water droplets blown from a core stream, the interaction between the droplets and the air moving past them can best be described in the inertial reference frame of the moving air. In this frame, the air is stationary while the droplet moves horizontally through the air with a speed equal to the relative speed of the droplet and wind, $v_r = v_a - v_x$. In the vertical direction, $v_r = v_y$, since the wind blows horizontally.

In the air's frame of reference, the water droplet experiences a drag force opposing v_r. Assuming that the air moves at a constant velocity, this force is the same in both frames of reference. In the frame of the fountain, then, the droplet is being blown in the direction of the wind. The smaller water droplets are carried farther, so we need only consider the motion of the smallest uncomfortable water droplets, knowing that bigger droplets do not travel as far.

The water droplet initially has a vertical velocity $v_y(0)$ that is directly related to the flow rate of water through the nozzle of the fountain. This initial vertical velocity component can be controlled by changing the flow rate. The droplet's motion causes vertical air resistance, slowing the droplet and affecting how long (t_w) the droplet is in the air.

Since the vertical and horizontal components of a water droplet's motion are independent, t_w is determined solely by the vertical motion. Knowing this time allows us to find the horizontal distance traveled, which we wish to constrain to the radius of the pool.

When the wind is variable, however, we cannot determine exactly the ideal flow rate for any given time. We must instead act on the current reading but also rely on previous measurements of wind speed in order to restrain the model from reacting too severely to wind fluctuations. We need to react fast to increases in wind speed, since they result in splashing which is weighted more heavily.

Design of the Model

For our initial model, we assume that v_a is constant for time intervals on the order of t_w, so that any given droplet experiences a constant wind speed.

We model the water stream as a collection of droplets that are initially cohesive but are carried away at varying velocities by the wind. The distances that they travel depend on the wind speed v_a and the initial vertical velocity of the water stream through the nozzle, $v_y(0)$. Since the amount of water flowing through the nozzle per unit time is $f = v_y(0)A$, we have $v_y(0) = f/A$. The dynamics of the system, then, is fully determined by f and v_a. First, we find the equations of motion for the droplet.

Equations of Motion for a Droplet

For laminar flow, a spherical particle of radius r traveling with speed v through a fluid medium of viscosity η experiences a drag force F_D such that

$$F_D = (6\pi\eta r)v$$ [Winters 2002].

Since a spherical water droplet has a mass given by

$$m = \rho_w \left(\tfrac{4}{3}\pi r^3\right),$$

the acceleration felt by the droplet is given by Newton's Second Law as the total force over mass. Since there are no other forces acting in the horizontal direction, the horizontal acceleration a_x is given by:

$$a_x(t) = \frac{d^2x}{dt^2} = \left(\frac{9\eta_a}{2\rho_w r^2}\right) v_r = k(v_a - v_x), \tag{1}$$

where $k = 9\eta_a/2\rho_w r^2$.

The droplet experiences both air drag and gravity in the vertical direction, so the vertical acceleration is

$$a_y(t) = -\left[\left(\frac{9\eta_a}{2\rho_w r^2}\right) v_y + g\right] = -k\left(v_y + \frac{g}{k}\right).$$

With constant v_a, we use separation of variables and integrate to find $v_x(t)$ and $v_y(t)$, using the facts that $v_x(0) = 0$ and $v_y(0) = f/A$. The results are

$$v_x(t) = v_a\left(1 - e^{-kt}\right), \qquad v_y(t) = ne^{-kt} - \frac{g}{k},$$

where $n = g/k + f/A$.

Integrating again, and using $x(0) = y(0) = 0$, we have

$$v_x(t) = \frac{v_a}{k}\left(kt + e^{-kt} - 1\right), \qquad v_y(t) = \frac{1}{k}n\left(1 - e^{-kt}\right) - gt.$$

Determining the Flow Rate

Because f is the only parameter that the algorithm modifies, we wish to find the flow rate that would restrict the smallest uncomfortable water droplets to ranges within R_p, so that they would land in the fountain's pool.

After a time t_w, the droplet has fallen back to the ground. Thus, $y(t_w) = 0$. This equation is too difficult to solve exactly, so we use the series expansion for e^{-kt} and truncate after the quadratic term: $e^{-kt} \approx 1 - x + x^2/2$. Solving $y(t_w) = 0$, we find

$$t_w \approx \frac{2}{k}\left(1 - \frac{g}{nk}\right).$$

We know that the maximum horizontal distance $x(t_w)$ must be less than or equal to R_p, with equality holding for the smallest uncomfortable droplet. For that case, using the same expansion for e^{-kt} as above,

$$R_p = x(t_w) \approx \frac{v_a}{k}\left(kt_w - 1 + 1 - kt_w + \frac{(kt_w)^2}{2}\right) = \frac{v_a k}{2}\, t_w^2.$$

Solving for t_w and equating it to the earlier expression for t_w, we get

$$\sqrt{\frac{2R_p}{v_a k}} = t_w = \frac{2}{k}\left(1 - \frac{g}{nk}\right).$$

Recalling that in this equality only n is a function of f, we substitute for n and solve for f. The result is

$$f(v_a) = \frac{Ag}{\sqrt{\dfrac{2v_a k}{R_p} - k}}. \tag{2}$$

As $v_a \to kR_p/2$, this equation becomes singular (see **Figure 2**). At lower values of v_a, it gives a negative flow rate. These wind speeds are very small; at such speeds, the droplets would not be deflected significantly by the wind. Since **(2)** assumes that the flow rate can be made arbitrarily high, it is unrealistic and invalid in application. To make the model more reasonable, we modify **(2)** to include the maximum flow rate achievable by the pump, $f_{\max}$:

$$F(v_a) = \begin{cases} \min\left(\dfrac{Ag}{\sqrt{\dfrac{2v_a k}{R_p} - k}}, f_{\max} \right), & v_a > kR_p/2; \\ f_{\max}, & v_a \geq kR_p/2. \end{cases} \tag{3}$$

An algorithm can use the given constants and a suitable minimal droplet size to determine the appropriate flow rate for a measured v_a. However, **(3)** assumes that the wind speed is constant over the time scale t_w for any given droplet. A more realistic model must take into account variable wind speed.

Variable Wind Speed

When wind speed varies with time, the physical reasoning used above becomes invalid, since the relative velocity of the reference frames is no longer constant. Mathematically, this is manifested in the equation for velocity-dependent horizontal acceleration; integrating is now not so simple, and we must resort to numerical means to find the equations of motion. Additionally, the algorithm can rely only on past and present wind data to find the appropriate flow rate. Our model needs to incorporate these wind data to make a reasonable prediction of the wind's velocity over the next t_w and determine an appropriate flow rate using **(3)**.

A *gust* is defined to be a sudden wind speed increase on the order 5 m/s that lasts for no more than 20 s; a *squall* is a similarly sudden wind speed increase that lasts longer [Weather Glossary 2002]. Our model should account for gusts and squalls, as well as for "reverse" gusts and squalls, in which the wind speed suddenly decreases. Since wind speeds can change drastically and unpredictably over the flight time of a droplet, our model will behave badly at times and there is no way to completely avoid this—only to minimize its effects.

The model's reaction to wind speed is not fully manifested until the droplet lands, after a time t_w (approximately 2 s). By the time our model has reacted to a gust or reverse gust, therefore, the wind speed has stopped changing. Without some type of buffer, in a gust our model would react by suddenly dropping flow rate as the wind peaked and then increasing it again as the wind decreased; the fountain would virtually cut off for the duration of any gust, which would release less water and thus seem very unattractive to passersby. Additionally, the water released just before the onset of the gust would be airborne as the wind speed picked up, splashing passersby regardless of any reaction by our model.

We exhibit an algorithm for analyzing wind data that makes use of **(3)**. Because velocity now varies within times on the order of t_w, we do not want to directly input the current wind speed but rather a buffered value, so that the model does not react too sharply to transient wind changes. The model should react more quickly to sudden increases in wind than to decreases, because increases cause splashing, which we weight more heavily than attractiveness.

The model, therefore, has two separate velocity buffer times: one, τ_d, the default, and another, τ_i, for when the wind increases drastically. We also weight more-recent values in the buffer more heavily, since we want the model to react promptly to wind speed changes but not to overreact. We weight each value in the velocity buffer with a constant value K plus a weight proportional to its age: Less-recent velocities are considered but given less weight than more recent ones. The weight of the oldest value in the buffer is K and that of the most recent is $K+1$, with a linear increase between the two. With the constraint that the weights are normalized (i.e., they sum to 1), the equation for the ith weight factor is

$$w_i = \frac{\left(K + \dfrac{i}{\tau - dt}\right) dt}{\left(K + \dfrac{1}{2}\right) \tau}.$$

The speeds are multiplied by their respective normalized weights and summed. This sum, v^*, is then used in **(3)** to find the appropriate flow rate for the fountain at a given time. We use τ_i rather than τ_d when the wind speed increases sufficiently over a recent interval, but not when it increases slightly or fluctuates rapidly. We switch from τ_d to τ_i whenever the wind speed increases over two successive 0.2 s intervals and by a total of at least 1 m/s over the entire 0.4 s interval.

Our algorithm follows the flow chart in **Figure 1** in computing the current flow rate

We wrote a C++ program to compute this algorithm, the code for which is included in an appendix. [EDITOR'S NOTE: We omit the code.]

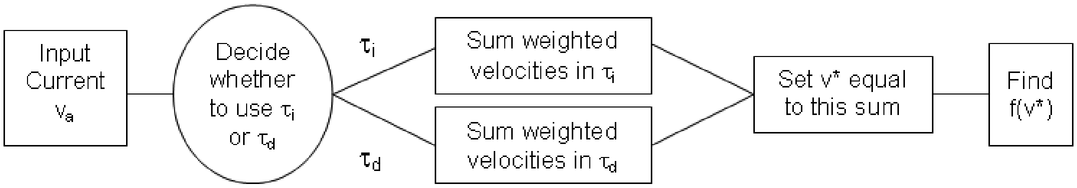
Input Current v_a
Decide whether to use τ_i or τ_d
τ_i
τ_d
Sum weighted velocities in τ_i
Sum weighted velocities in τ_d
Set v* equal to this sum
Find f(v*)

Figure 1. Flow chart for computing flow rate with variable wind speed.

Testing and Sensitivity Analysis

Sensitivity of Flow Equation

In our equation for flow rate, two variables can change: minimal droplet size and wind speed. While the minimal droplet size will not change dynamically, its value is a subjective choice that must be made by the owner of the fountain. The wind speed, however, will change dynamically throughout the problem, and the purpose of our model is to react to these changes.

We examined **(3)** for varying minimal drop sizes (**Figure 2**) and wind speeds (**Figure 3**). We used a fountain with nozzle radius 1 cm, maximum flow rate 7.5 L/s, and pool radius 1.2 m. (This maximum flow rate is chosen for illustrative purposes and is not reasonable for such a small fountain.)

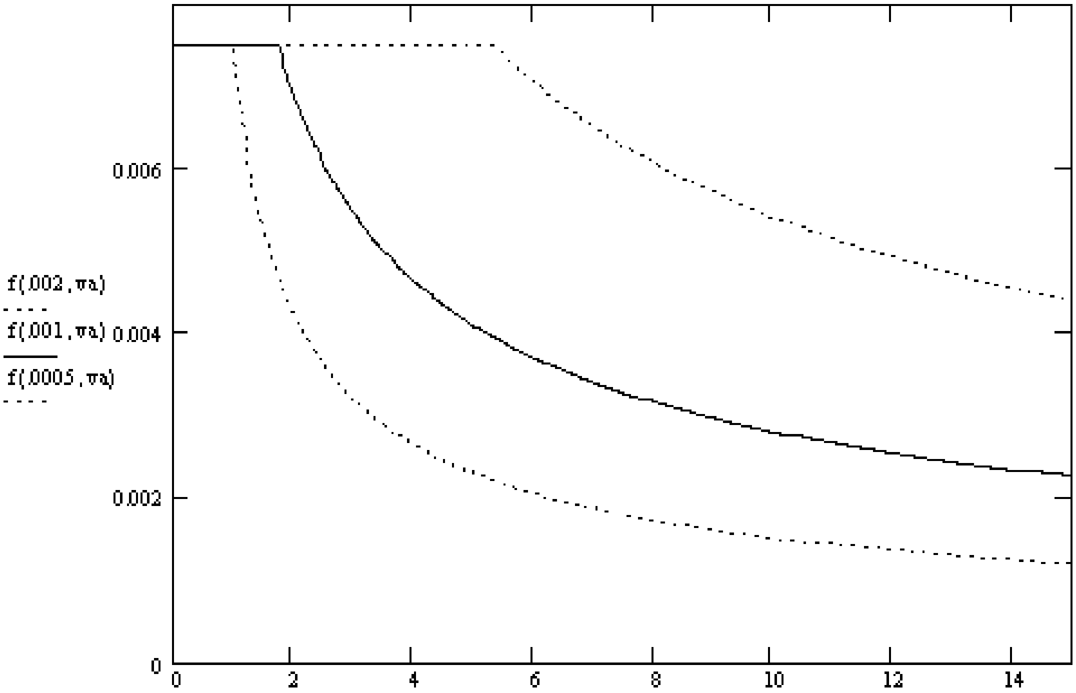

Figure 2. Graphs of flow rate f vs. wind speed v_a for several values of radius r of smallest uncomfortable droplet.

At any wind speed, as the acceptable droplet radius decreases, the flow rate o decreases. At higher wind speeds, this difference is less pronounced; but at lower speeds, acceptable size has a significant impact on the flow rate. At very low wind speeds, the fountain cannot shoot the droplets high enough to allow the wind to carry them outside the pool, regardless of drop size. Our cutoff, $f_{\max}$, reflects that the fountain pump cannot generate the extreme flow needed to get the droplets to the edge of the pool in these conditions.

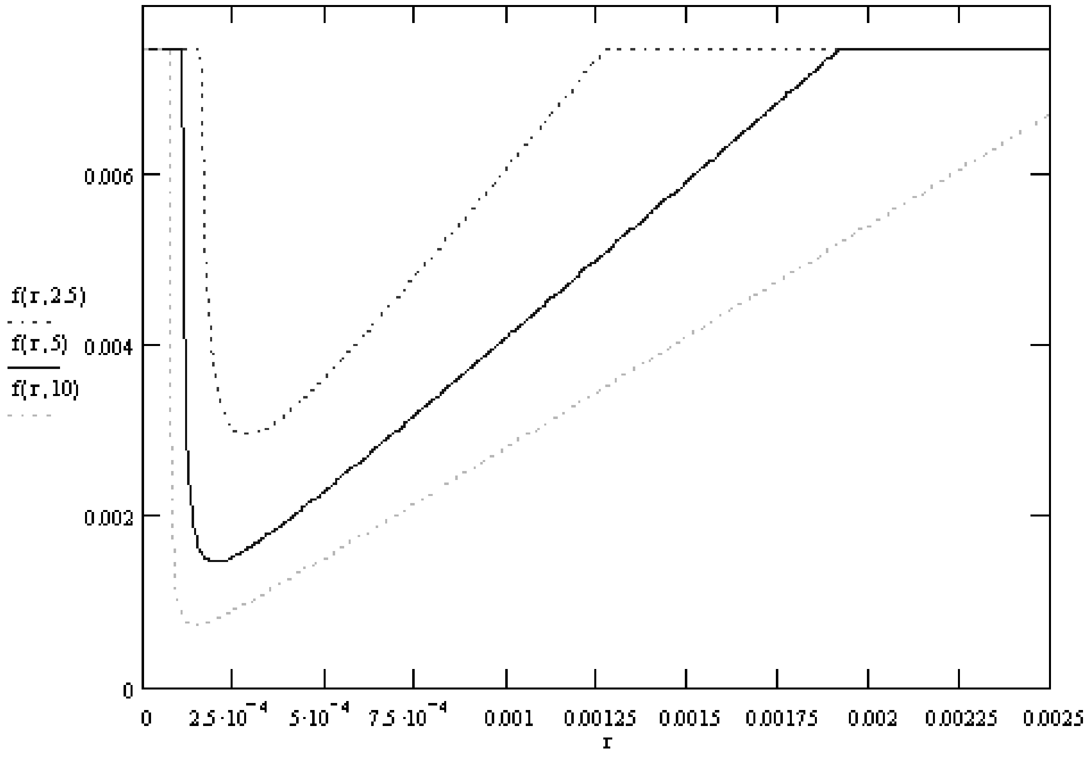

Figure 3. Graphs of flow rate f vs. radius r of smallest uncomfortable droplet for several values of wind speed v_a.

For any droplet size, as the wind speed increases, the flow rate must decrease to keep the droplets in the pool. For large r, a change in wind speed requires a greater absolute change in flow rate than for small r. For very small droplets, the drag force dominates the force of gravity, and an increase in flow also increases the drag force to such an extent that the particle spends no more time in the air. This behavior is readily apparent in **(1)** as r approaches zero. These extremely small values of r, though, describe droplets that are unlikely to discomfort passersby and thus are not significant to our model.

Sensitivity of Flow Algorithm

The results of the algorithm depend on the parameters τ_i, τ_d, and K, which determine the size of the buffer and weights of the velocities in the buffer. To test sensitivity to these parameters and to find reasonable values for them, we created the set of simulated wind speeds shown in **Figure 4**, including small random variations, on which to test our algorithm. This data set does not reflect typical wind patterns but includes a variety of extreme conditions.

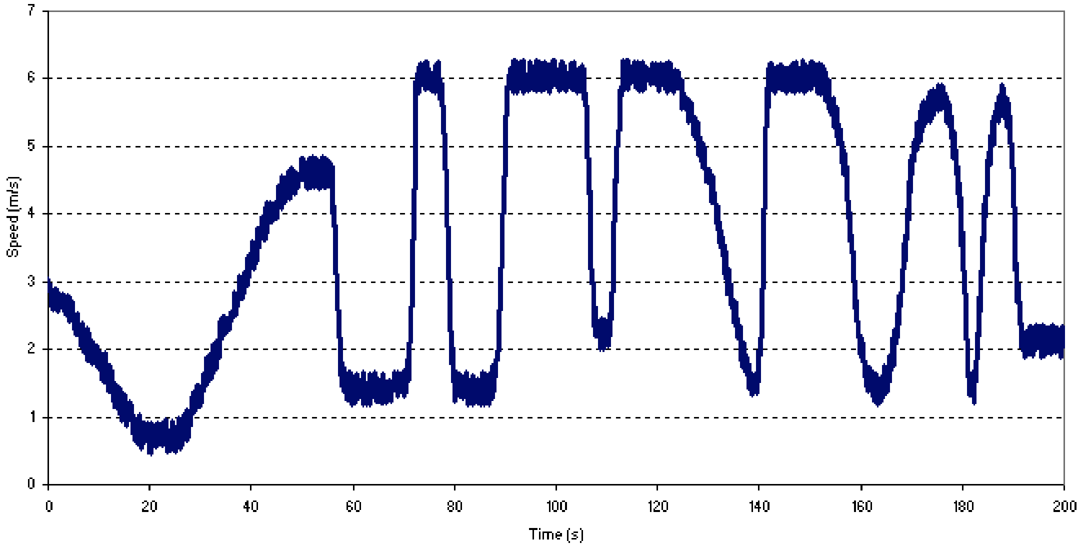

Figure 4. Simulation of wind speed for 3 min.

We wish to create a quantitative estimate of the deviation of our flow algorithm from ideal performance and then test the algorithm with different combinations of parameters to find the set that produces the smallest deviation under simulated wind conditions.

To measure how "bad" a set of flow choices is, we consider only the droplets that fall outside the pool. The "badness" is the sum over the run of the distances outside the pool at which droplets land.

To determine the distance, we need to know how droplets move through the air in varying wind speeds. Describing this motion in closed form is mathematically impossible without continuous wind data, so we approximate the equations of motion with an iterative process.

Since the time that a particle spends in the air, t_w, is not affected by the wind speed, we know t_w for each particle. We step through the time t_w in intervals of dt, computing the particle's acceleration, velocity, and position as

$$\begin{aligned} a_i &= k(v_{a,i} - v_i), & a_0 &= kv_{a,0}; \\ v_i &= v_{i-1} + a_{i-1}dt, & v_0 &= 0; \\ x_i &= x_{i-1} + v_i dt, & x_0 &= 0. \end{aligned}$$

When we reach t_w, the droplet has hit the ground, and we compare its horizontal position to the radius of the pool. We do this for each droplet and keeping track of both the largest absolute difference and the average difference.

To test the flow algorithm, we ran our program with each combination of parameters on each set of flow data. The parameter values that produced the least deviation were $\tau_i = 0.5$, $\tau_d = 1$, and $K = 10$. These values imply that only fairly recent wind speed measurements should be held in the buffer, with most recent velocity having a weight of $(K+1)/K = 1.1$ relative to the oldest. Lowering K beyond this value increases the deviation from the ideal, while increasing it further makes no difference. Similarly, increasing τ_i or τ_d increases the deviation, because the algorithm cannot respond quickly to changes in wind speed. Decreasing τ_i below 0.5 makes no difference, while decreasing τ_d would make the model too sensitive to short fluctuations in wind speed.

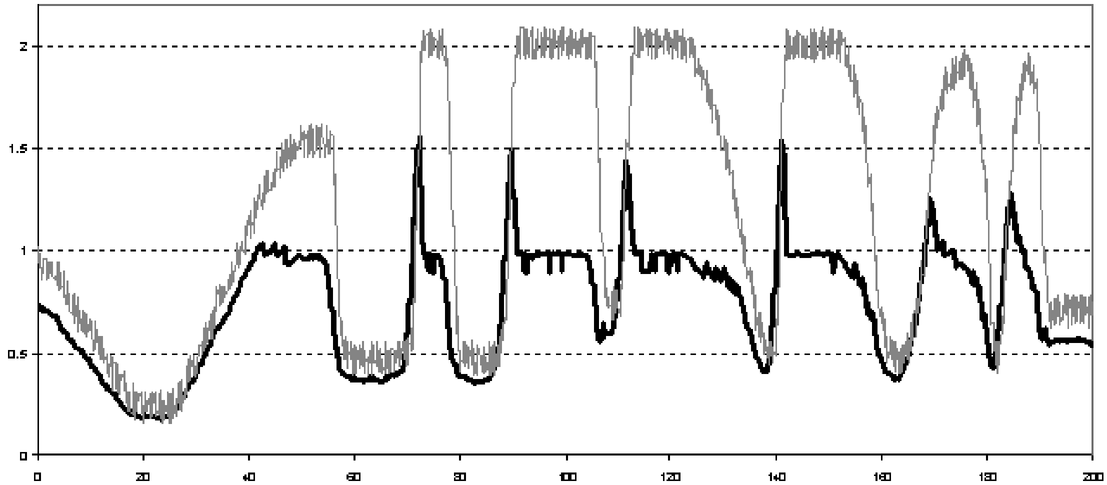

Figure 5. Range of droplets over the simulation overlaid with scaled wind speeds.

Justification

Validity of the Laminar Flow Assumption

Our model is based on a drag force proportional to v_r, which is not necessarily correct. For higher speeds or large droplet sizes, the drag becomes proportional to v_r^2. We thus need to determine whether reasonable physical scenarios allow us to model the drag force as proportional to and not v_r^2.

For a sphere of radius r moving through the air with speed v_r, the Reynolds number R is defined to be

$$R = \frac{2\rho_a v_r}{\eta_a} r \qquad \text{[Winters 2002].}$$

When $R < 10^3$, there is little turbulence and laminar flow dominates, so air resistance is roughly proportional to v_r. If $R > 10^3$, the flow is turbulent and the drag force is proportional to v_r^2 [Winters 2002]. Using a physically reasonable relative speed of 4.5 m/s (corresponding to a wind speed of roughly 10 mph), we obtain $R = (5.8 \times 10^5)r$, which gives predominantly laminar flow when $r < 1.7$ mm. Because water droplets of diameter greater than 3 mm are uncomfortable, these provide an upper limit on the droplet sizes to consider.

Because these smaller droplets bound the larger droplets in how far they go from the fountain (see below), all of our analysis is concerned with droplets whose sizes are within the allowed range for laminar flow.

Bounding the Droplet Range

For either laminar or turbulent flow, the acceleration due to drag scales with as $F/m \propto r^{-n}$, where $1 \leq n \leq 2$. Larger droplets therefore experience a lower horizontal acceleration due to drag, while acceleration in the vertical direction is dominated by gravity ($k < 0.1g$); so the time that a particle spends in the air is roughly the same for droplets of varying radius. The heavier droplets have less horizontal acceleration, so they travel a shorter horizontal distance in the same amount of time than smaller droplets. The ranges are, therefore, shorter for larger droplets, so we can bound all uncomfortably-sized droplets by the range of the smallest such droplet.

Initial Shape of the Water Stream

We assume that the water coming out of the fountain nozzle has no initial horizontal velocity; that is, the stream is a perfect cylinder with the same radius as the nozzle. In fact, the stream is closer to the shape of a steep cone and the droplets have some horizontal velocity. In the absence of wind, this assumption has a significant impact on where the droplets land, since without wind the algorithm predicts a horizontal range of zero. However, in these cases, the flow rate is bounded by $f_{\max}$ regardless of initial velocity, so the natural spread of the fountain is irrelevant. In higher wind, the initial horizontal velocity is quickly dominated by the acceleration due to the wind and thus makes a negligible contribution to the total range.

Exclusively Horizontal Wind

We assume that the wind is exclusively horizontal. Since the anemometer measures only horizontal wind speed, that is the only component that we can consider in our model. Additionally, the buildings around the plaza would tend to act as a wind tunnel and channel the wind horizontally.

Quadratic Approximation of e^{-kt}

Because the series for e^{-kt} is alternating, the error from truncating after the second term is no greater than the third term, which is $(kt)^3/6$. The relative error is $(kt)^3/e^{-kt} \approx 0.001$ for reasonable values of k and t, so our approximation introduces very little error.

Conclusions

Our final solution is an algorithm that takes as its input a series of wind speed measurements and determines in real-time the optimal flow rate to maximize the attractiveness of the fountain while avoiding splashing passersby excessively. It takes an inputted wind speed and adds it to a buffer of previous measurements. If the wind speed is increasing sufficiently, the last 0.5 s of the buffer are considered; otherwise, the last 1 s is. The algorithm computes a weighted average of these wind speeds, weighting the most recent value 10% more heavily than the oldest value considered. It then takes this weighted average and uses it in the equation that predicts the optimal flow rate under constant wind. The result is the optimal flow rate under variable wind, knowing only current and previous wind speeds.

Strengths and Weaknesses

Strengths

- Given reasonable values for the characteristics of the fountain and for wind behavior, our model returns values that satisfy the goal of maintaining an attractive fountain without excessively splashing passersby.

- The model can compute optimal flow rates in real time. Running one cycle of the algorithm takes a time on the order of 0001 s, so the fountain's pump could be adjusted as fast as physically possible.

- The constants that determine the behavior of the algorithm, τ_d, τ_i, and K, are not arbitrary but instead perform best under simulation.

- Our algorithm is very robust; it works well under extreme conditions and can be readily modified for different situations or fountains.

Weaknesses

- A primary assumptions is that the droplets coming from the fountain nozzle have no horizontal velocity. In reality, the nozzle sprays a cone of water, rather than a perfect cylinder; but this difference does not have a significant impact on the results.

- Another important assumption is laminar flow. The water droplets are of a size to experience a combination of laminar and turbulent flow, but describing such a combination of regimes is mathematically difficult and is known only through experimentation. A more rigorous representation of the drag force would increase the accuracy of our simulation, but doing so would markedly increase the complexity of the algorithm and thus make real-time computation more difficult.

- We have ignored the abundances of droplet sizes in considering discomfort. If one droplet would spray passersby, we assume that enough droplets would spray passersby to make them uncomfortable. In fact, it is only significant numbers of droplets that discomfort passersby; but we do not know how many droplets would be released nor how many would be needed to be discomforting.

References

Goldstein, S., ed. 1965. *Modern Developments in Fluid Dynamics*. Vol. 2. New York: Dover.

Hughes, W.F., and J.A. Brighton. 1967. *Fluid Dynamics*. New York: McGraw-Hill.

Lide, David R., ed. 1999. *CRC Handbook of Chemistry and Physics*. 80th ed. Boca Raton, FL: CRC Press.

Winters, L. 2002. Theory of velocity dependent drag forces. `http://courses.ncssm.edu/ph220/labs/vlab1/theory.pdf` . Accessed February 2002.

Weather Glossary. 2002. `http://www.weather.com/glossary/` . Accessed February 2002.