MAGIC: Model-Based Actuation for Ground Irrigation Control

Daniel A. Winkler*, Robert Wang*, Francois Blanchette†, Miguel Carreira-Perpiñán* and Alberto E. Cerpa*
Email: {dwinkler2, acerpa}@andes.ucmerced.edu, {rwang6, fblanchette, mcarreira-perpinan}@ucmerced.edu
*Electrical Engineering and Computer Science
†Department of Applied Mathematics
University of California, Merced

***Abstract*—Lawns make up the largest irrigated crop by surface area in North America, and carries with it a demand for over 9 billion gallons of freshwater each day. Despite recent developments in irrigation control and sprinkler technology, state-of-the-art irrigation systems do nothing to compensate for areas of turf with heterogeneous water needs. In this work, we overcome the physical limitations of the traditional irrigation system with the development of a sprinkler node that can sense the local soil moisture, communicate wirelessly, and actuate its own sprinkler based on a centrally-computed schedule. A model is then developed to compute moisture movement from runoff, absorption, and diffusion. Integrated with an optimization framework, optimal valve scheduling can be found for each node in the space. In a turf area covering over 10,000ft^2, two separate deployments spanning a total of 7 weeks show that MAGIC can reduce water consumption by 23.4% over traditional campus scheduling, and by 12.3% over state-of-the-art evapotranspiration systems, while substantially improving conditions for plant health. In addition to environmental, social, and health benefits, MAGIC is shown to return its investment in 16-18 months based on water consumption alone.**

***Index Terms*—Irrigation, Monitoring, Decentralized Control.**

I. Introduction

Only 1% of Earth's water is fresh and available for use [4]. Due to its scarcity, there is high incentive to reduce its usage across the board. In North America, turf, also known as lawn, is the largest irrigated crop by surface area, covering over 128,000 km^2 [8] and was estimated in 2015 to consume in excess of 9 billion gallons of freshwater each day [13]. With a historic drought afflicting the western United States following a similar shortage in the south-east United States, improved irrigation efficiency at this massive scale can help reduce the strain on our limited freshwater reserves.

Although we wish to reduce water consumption as much as possible, the primary goal of these irrigation systems is to maintain plant health. To keep turf healthy, a proper amount of water must be periodically applied. Providing too little water to the turf will cause it to turn brown and die. Although traditional irrigation control strategies often over-irrigate to be safe, this can cause its own problems. Excess surface water can cause further waste due to evaporation and runoff and can cause root rot, killing the plant. Furthermore, over-watering can cause erosion of the surrounding soil and even leech unsafe fertilizer chemicals beyond the root zone and into the ground water, as occurred in California's Salinas Valley and Tulare Lake Basin, investigated by [22].

Improper irrigation is the cause of these issues. Great improvements in irrigation system design have been made recently; new sprinkler heads apply water much more slowly to avoid runoff and leeching [5], and new irrigation controllers schedule irrigation using weather data to take into account the water lost each day due to evaporation and plant transpiration, known coupled as evapotranspiration. Even the best control strategies still behave as though all turf requires the same amount of water, when in fact there often exist large variations in soil type and depth, topography, and direct sunlight. If this information were utilized, every location throughout the irrigated space could be given the amount of water it needs. However, as the infrastructure of traditional irrigation systems is usually configured for each valve to actuate many sprinklers, such a system could not even make proper use of fine-grained water requirement information as all sprinklers must be actuated for the same amount of time.

Our contributions addresses both of these limitations. First, we develop a computationally-light model that uses characteristics of the irrigated space to analyze the fundamental causes of fluid movement. This model is then integrated into an optimization framework to allow for optimal valve scheduling to be computed. The second contribution is the development of the MAGIC sprinkler node, capable of actuating its attached sprinkler, sensing local soil moisture, and communicating wirelessly with its sister nodes in the environment. Through a large-scale deployment lasting 7 weeks, we demonstrate that the model-based MAGIC system can help provide more precise irrigation to turf areas, reducing water usage and substantially improving the quality of service over common-practice and state-of-the-art control strategies.

II. Related Work

To accomplish the minimization later described in Section 10, we must know how fluid moves across and through soil. This is well-studied in the field of soil physics; very accurate models including Hydrus [7] and Comsol Multiphysics' Subsurface Flow Module [1] exist, which solve PDEs for pressures that exist between soil and water particles in the porous media. Although very accurate, these models compute water flow and absorption on a much smaller scale than what

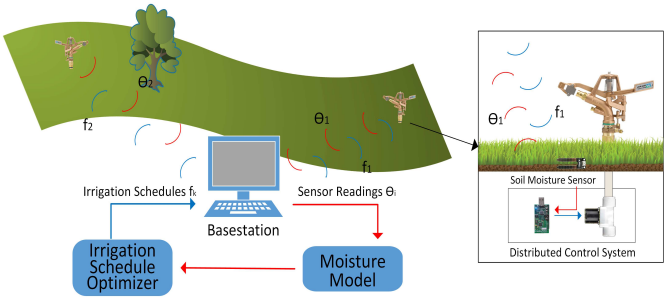


Fig. 1. MAGIC System Architecture

is necessary for irrigation control. As such, they require heavy processing, and take a tremendous amount of time to complete.

An overly-complicated fluid model makes optimization unreasonable, so alternative methods are considered for system simplification. For example, observations of an accepted fluid flow model can be incorporated into a simplified model using Data Assimilation. Used to predict states of advanced systems given a particular input such as weather forecasting [24] and large-scale hydrological patterns from satellite images [27], data assimilation is often used to create approximated system models, which can then be used for optimization. However, the immense size and discontinuity of the solution space can lead to an approximated model that is unable to reasonably predict outcomes of the system. Another option, called Lumped Element Modeling, is a method of breaking complex problems into simpler "lumped" sub-components. This method has been used to create a simplified model of fluid jets for prototype analysis [19], and to model aortic blood flow from arterial pressure in humans [30] among other applications. Although such simplifications sacrifice accuracy, the approximation can be evaluated in a short amount of time.

Researchers in [18] build a one-dimensional soil water balance model to simulate the vertical movement of moisture in an irrigated space based on soil characteristics, irrigation schedule, real-time weather data, and irrigation infrastructure. In addition, an interactive tool was designed for homeowners and landscapists to quickly evaluate an installed system. This system provides some key insight on moisture changes, such as evapotranspiration that take place on a daily timescale, but over-simplify short-term effects like runoff, that can provide key insight into the movement of water across the surface. Although it provides evaluation of a particular schedule, no strategies are offered for schedule improvement.

In [31], a model was created using a cellular automata approach that incorporated irrigation, surface, and sub-surface flow. By analyzing this model using an optimization framework, it was determined in simulation that system efficiency can be improved by increasing the granularity of control. However, due to model complexity and problem size, the optimization framework had difficulty finding a globally optimal solution with large numbers of timesteps and sprinklers.

Irrigation controllers exist that utilize one or more soil moisture sensors spread throughout the turf [12], [15], [25] to gauge need for irrigation. However, whereas systems that model the space can infer and utilize moisture status between sensor points to create schedules, these systems can only utilize the limited data from the sensor points. Furthermore, these systems still have the same physical constraints as a traditional irrigation system; they can not provide irrigation to specific regions that need more or less.

The authors of [26] identify potential improvements to agriculture irrigation systems. On-site and remote control systems are discussed, as well as their learning curves, which may prevent unskilled farmers from adopting them. Bottlenecks are recognized in system development, standardization of technology, and business models, that must advance before control system alternatives become accepted. Although a control system is not introduced, this work makes clear that control strategy improvements are highly sought-after.

Becoming more prevalently used in modern irrigation controllers, evapotranspiration technologies [14], [23] use weather stations to monitor air temperature, humidity, and other factors to estimate water lost to the environment throughout the day. Using this information, irrigation systems can use knowledge of their sprinklers' coverage to compute required on-time for water replacement. As this has become widely-adopted, it will be used as side-by-side comparison with the MAGIC system.

III. System Overview

Figure 1 shows an overview of the MAGIC system architecture. Our irrigation control system uses multiple modules to provide control to the space. To explain how these modules work together, we first describe their roles. They are described at irrigation time, when schedule generation occurs.

The MAGIC system consists of a distributed network of sensing and actuation (MAGIC) nodes, installed into the existing plumbing infrastructure of the irrigation system. Each MAGIC node is equipped with a soil moisture sensor, a solenoid to control the flow of water, and a mote to provide radio communication. The attached volumetric water content (VWC) sensor is periodically sampled from the environment by each sprinkler node in the space. This collected data is then routed through the wireless sensor network to the *Basestation*, interfacing between the 802.15.4 network and another communication medium such as an ethernet or 4G network. Once received, this data is then incorporated with the sensor readings collected from other nodes to create a "snapshot" of the soil moisture across the entire space.

The *Moisture Model*, described in Section IV, contains a mathematical formulation modeling soil and irrigation system characteristics, providing means to calculate moisture movement through and across the soil. After the current VWC from the irrigation system is integrated, the model is passed to the *Irrigation Schedule Optimizer* for schedule creation.

The *Irrigation Schedule Optimizer* sets up and solves the following constrained optimization problem. The fluid flow model is incorporated as equality constraints at each spatial location and time, which must be satisfied for an optimal solution to result in a valid flow. A goal water saturation level provides inequality constraints at each spatial location at the end of irrigation. Although in principle the PDEs defining the

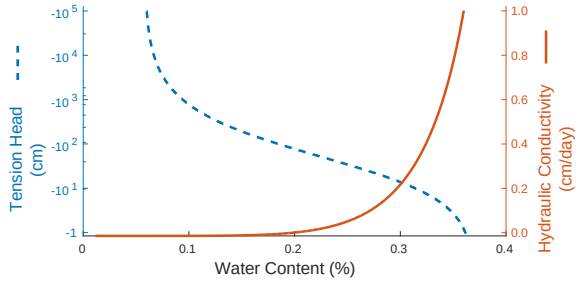


Fig. 2. Sample Retention & Hydraulic Conductivity

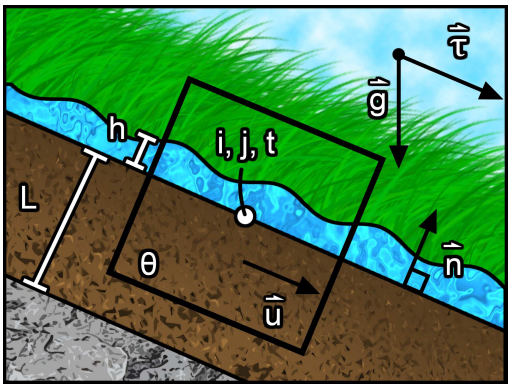


Fig. 3. Physical Model Unit Diagram

model are nonlinear, we linearize them in order to simplify the optimization (which, due to discretization, involves many variables). The final result is a linear program that, although large, can be solved accurately in a reasonable time. As the objective function minimizes the total water consumption of the irrigation system, the solution provides optimal activation schedules for each MAGIC node in the space.

Once the schedules are received by the *Basestation*, they are disseminated through the wireless sensor network to their respective MAGIC node. Upon reception of a schedule, the *Distributed Control System* routes power to the attached solenoid following the received schedule, allowing water to flow to the sprinkler.

IV. Model Development

We wish to model a particular irrigation system, and use this information to find improved control techniques. This two-dimensional model incorporates the movement of water from the sprinkler heads, water movement across the surface, the absorption into the soil, and the movement of water through the sub-surface. Although many models exist that describe one or more of these components, we present here the first model that efficiently combines them together.

A. Soil Characteristics

We first emphasize the differences between soil and typical porous media. Generally, a porous medium maintains constant characteristics across its entire range of saturation. However, in soils, two functional relationships govern the retention and movement of water through soil. First, an attraction exists between water and the soil particles, known as matric suction. When the soil is very dry, the matrix exerts a strong suction, trying to pull water from the surrounding environment. The relationship between matric suction and volumetric water content is known as the "water retention curve", a characteristic equation of the soil as defined in [20]. This relationship, shown on the left axis of Figure 2, strongly impacts the movement of water through the soil, as dry soil will prevent flow until increased saturation is reached. Second, the hydraulic conductivity of soil is dependent on the local volumetric water content. Due to the matric suction, the soil will increasingly resist the movement of moisture as the volumetric water content decreases. The hydraulic conductivity relationship is also defined in [20] and is shown on the right axis of Figure 2.

TABLE I
Model Variable Reference

Variable	Usage
f_k	Actuation function of sprinkler k
c_k	Coverage of sprinkler k
θ	Volumetric soil moisture content
h	Surface fluid height
$\vec{u}$	Velocity of water in soil
$\vec{v}$	Velocity of water on surface
κ	Soil permeability
κ_g	Grass permeability
η	Fluid viscosity
ρ	Fluid density
ζ	Sub-surface boundary constant
μ	Surface boundary constant
α_h	Surface flow parameter
$K(\theta)$	Hydraulic conductivity
$\psi(\theta)$	Matric suction
F_s	Fluid from sprinklers
$\vec{\tau}$	Tangential component of gravity
θ_{pwp}	Minimum θ for healthy plants

B. Fluid Flow Model

We model water displacement above the soil surface and through the subsurface of the soil as flow through two different porous media. Fluid flow through porous media is well-studied, dating back to the work of Henry Darcy [17], now known as Darcy's Law. To model the movement through the soil we use Darcy's Law for isotropic porous media (Eq. 1),

$$\vec{u} = \frac{\kappa}{\eta}(-\nabla P + \vec{\tau}) \tag{1}$$

where P is the pressure, $\vec{\tau}$ is the tangential component of gravity along the surface of the porous media, and other quantities as defined in Table I. This model assumes that the porous media everywhere has the same dependence on water content. As our model tracks soil moisture at small scales, such a simplification is more practical than the alternative of collecting and analyzing samples across the entire space.

Darcy's Equation requires the determination of the pressure, which is generally linearly related to the amount of water above the point in question. In our model, we compute the depth-averaged subsurface flow by considering a soil depth L. In this case the mean pressure driving flow includes the weight of water in the subsurface above a point, $\rho g L\theta$, the weight of water on the surface, $\rho g h$ where h is the height

of water above the soil surface, and the matric suction of the soil $\rho g\psi(\theta)$. We thus express the mean pressure in soil as $P = \rho g(h + L\theta + \psi(\theta))$. We can therefore express the liquid velocity in the subsurface as

$$\vec{u} = -K(\theta)\nabla h + K(\theta)\vec{\tau} - K(\theta)(L + \psi'(\theta))\nabla\theta \tag{2}$$

as depicted in Figure 3, where we defined the hydraulic conductivity as $K(\theta) = \rho g\kappa(\theta)/\eta$.

To track the time-rate of change of the volumetric soil moisture content, θ, we use the divergence of the moisture flux, $\vec{u}\theta$, and the inflow from surface water, $\zeta hK(\theta)$, where $\zeta = 1/L^2\phi_s$ is a proportionality constant mapping surface water height to volumetric content in the subsurface based on soil porosity and depth, calculated by balancing the pressure gradient with the soil permeability. We thus have

$$\frac{\partial\theta}{\partial t} = -\nabla\cdot(\theta\vec{u}) + \zeta hK(\theta) \tag{3}$$

As the velocity, $\vec{u}$, is itself the gradient of the volumetric soil moisture content, the movement of water will behave as a diffusive process, moving sub-surface moisture towards areas of lower concentration. With the inclusion of a gravity term, $\vec{\tau}$, in the velocity equation, water will tend to move in the direction of steepest descent. Lastly, by allowing water to move through the boundary from the surface into the sub-surface, we allow irrigation on the surface of the soil to positively affect the amount of water moving through the sub-surface. Together, these terms result in sub-surface water movement that realistically depends on volumetric water content, local topography, and surface conditions.

In addition to tracking the soil moisture, we need to determine the height of the surface water, h. Depending on the species of lawn chosen to model, a square inch of turf can have tens to hundreds of blades of grass, each of which will impede the movement of water across the surface of the soil. By comparing the inertia of the fluid and the drag forces caused by the blades of turf, we find that for surface water velocities less than 1cm/s, the surface flow through the turf can be modeled as fluid flow through a (very) porous medium. Additional information and the calculations that support this approach can be found in Appendix A. The velocity of water through the turf, $\vec{v}$, is therefore computed using Darcy's Equation for isotropic porous media, as was done for the sub-surface velocity in Equation 2:

$$\vec{v} = \frac{\kappa_g}{\eta}(-\rho g\nabla h - \vec{\tau}) = -\alpha_h\nabla h - \frac{\kappa_g}{\eta}\vec{\tau} \tag{4}$$

where the fluid density, permeability, and gravity terms have been absorbed into α_h. Similarly to sub-surface movement, the velocity through the layer of grass is dependent on the orientation of the surrounding topography, as well as the viscosity and density of the fluid.

The permeability used in Equation 4, κ_g, is not as well-constrained as that of the sub-surface fluid flow. The shape, size, and density of the grass layer is dependent on the species of grass, as well as its health. Research has been conducted [16], [28] for the application of filter design that defines the permeability of a fibrous media based on its overall porosity and the size of the fibers. Using the proposed method, we find an approximate permeability of 10^{-5}m , a value similar to that of well-sorted gravel. This completes the description of the velocity of surface water, and we may use this velocity to keep track of the height of water on the surface:

$$\begin{aligned}\frac{\partial h}{\partial t} &= -\nabla\cdot(h\vec{v}) + F_s - \mu hK(\theta)\\ F_s &= \sum_{k=1}^{K} c_k f_k(t)\end{aligned} \tag{5}$$

where F_s is the rate of irrigation, determined using the activation $f_k(t)$ of sprinkler k at time t and coverage c_k of sprinkler k. The amount of water lost to soil is the same as that added to soil moisture, converted from soil moisture to pure water, where $\mu = \zeta L\phi_s/\phi_g$, with ϕ_s, ϕ_g as the porosities of soil and grass and L as the soil depth. We note that evaporation and leeching terms were not included in our formulation, due to the way our case study was conducted. At the request of campus authorities, all irrigation was performed in late evening, providing ample time to absorb into the soil, and allowing only minimal evaporation to occur. As for leeching, our deployment soil sat on top of a thick layer of clay, measured [20] to have a hydraulic conductivity 60–3700$\times$ smaller than the soil found at our deployment, making leeching terms negligible. Although these terms are omitted in the present study, they could easily be introduced for an application that requires them.

C. Boundary and Initial Conditions

To complete our PDE-based model, we specify boundary conditions describing moisture movement in and out of our domain on the sides and bottom. In many irrigated spaces, the turf is bordered by natural boudaries (i.e. sidewalk) that prevents all flow, making modeling very simple. As our deployment does not border any such features, we model the boundaries on the sides of our space using a "fixed" boundary condition. On the surface, the boundary value is 0 so that water can not move in from other locations, and on the sub-surface the boundary value is θ_{pwp}, the minimum volumetric content to maintain turf health. For larger regions, an infinite domain can be modeled by enforcing the condition that all quantities have zero flux (zero normal derivatives) at the boundary. Due to the slow moisture movement through the soil, an approximate boundary condition should only have minor effects near the boundary of the domain. As the thick clay layer beneath our deployment greatly hinders moisture movement, we consider the loss through the bottom of our system to be negligible, and thus omitted.

As irrigation occurs at most once-daily, we assume that all surface moisture has had ample time to fully absorb into the soil, so the initial condition for surface moisture is set to zero. As the MAGIC node periodically samples the soil moisture sensors to monitor the state of the soil, this data is readily available for use as sub-surface initial conditions to the model. As the sensors are coarsely-distributed spatially throughout the

area, the data is upsampled to the same granularity as the optimization problem using a bilinear interpolation.

D. Fluid Flow Model Simplification

We simplify the model in two ways for computational reasons. Firstly, we linearize the model PDEs. Although this is not necessary if one only wants to solve the PDEs to obtain the flow over time given a schedule, it considerably facilitates the numerical optimization over the schedule. Nonlinear equality constraints make the feasible set nonconvex and give rise to local optima, which complicate finding a good optimum. They also require nonlinear optimization, which is slower. Second, we discretize the spatial and temporal domains and approximate the derivatives using finite differences. With these simplifications, since the objective function and inequalities are already linear in our application, the resulting optimization problem is a linear program, for which efficient solvers are available that can handle millions of variables and constraints. As seen later, this allows us to obtain a valid schedule in a relatively small amount of time.

1) Fluid Flow Model Linearization: The goal of linearization is to characterize each model equation as a linear combination of optimization variables. To remove non-linearities arising from optimization variables multiplied together, we make reasonable assumptions about the behavior of the system to substitute these non-linearities with linear counterparts.

We break each optimization variable into a base value, with subindex 0, and a small deviation, denoted with a hat. For example, the volumetric moisture content, θ, is rewritten in the form $\theta = \theta_0 + \hat{\theta}$. Each occurrence of the original four optimization variables is replaced with a similar representation, and simplified to achieve the following four linear equations, where we define a function $\varphi(\theta) = K(\theta)(L + \psi'(\theta))$ to simplify notation:

$$\frac{\partial h}{\partial t} = -\nabla\cdot(\hat{h}\vec{v}_0 + h_0\hat{\vec{v}}) + F_s - \eta\Big(h_0K(\theta_0) + h_0K'(\theta_0)\hat{\theta} + \hat{h}K(\theta_0) + \underline{\hat{h}K'(\theta_0)\hat{\theta}}\Big) \quad (6)$$

$$\frac{\partial \theta}{\partial t} = -\nabla\cdot(\hat{\theta}\vec{u}_0 + \theta_0\hat{\vec{u}}) + \zeta\Big(h_0K(\theta_0) + h_0K'(\theta_0)\hat{\theta} + \hat{h}K(\theta_0) + \underline{\hat{h}K'(\theta_0)\hat{\theta}}\Big) \quad (7)$$

$$\vec{u} = -K(\theta_0)\nabla\hat{h} - K(\theta_0)\nabla h_0 - K'(\theta_0)\hat{\theta}\nabla h_0 - \underline{K'(\theta_0)\hat{\theta}\nabla\hat{h}} + K(\theta)\vec{\tau} - \varphi(\theta_0)\nabla\theta_0 - \varphi(\theta_0)\nabla\hat{\theta} - \varphi'(\theta_0)\hat{\theta}\nabla\theta_0 - \underline{\varphi'(\theta_0)\hat{\theta}\nabla\hat{\theta}} \quad (8)$$

$$\vec{v} = -\alpha_h\nabla h + \vec{\tau} \quad (9)$$

As we assume that the deviations are small relative to the base value, the underlined terms should be much smaller than other terms appearing in the equations, and are ignored.

2) Fluid Flow Model Discretization: The derivatives appearing in our modeling equations are approximated with finite differences. We use forward differences for the time derivatives for θ and h, for example $d\theta/dt \approx \frac{\theta_{i,j,t+1}-\theta_{i,j,t}}{\Delta t}$, where Δt is the time interval size and t is the temporal index, with t = 0,...,N_t. We use centered differences for the spatial derivatives for θ and h, for example $d\theta/dx \approx \frac{\theta_{i+1,j,t}-\theta_{i-1,j,t}}{2\Delta x}$, where Δx is the spatial grid size and i,j are the indices of a spatial cell, for i=0,...,N_x and j=0,...,N_y . Consider the discretization on a continuous range [0, N_x] in the x direction. The entire range is broken into N_x segments of length Δx. As the range is inclusive and contains both 0 and N_x, the number of discrete variables is N_x + 1, corresponding to x_0,x_1,...,x_{N_x}. This discretization is performed on all variables of our linear model equations, resulting in the 6 equations found in Appendix B.

TABLE II
OPTIMIZATION VARIABLES

Variable	Description
i, j	Spatial index $\in 0 : N_x, 0 : N_y$
t	Temporal index $\in 0 : N_t$
k	Sprinkler location index $\in 1 : K$
f_{kt}	Sprinkler k actuation at time t $\in \{0, 1\}$
$h_{(ijt)}$	Height of water on surface
$\theta_{(ijt)}$	Soil volumetric water content
$u^x_{(ijt)}$ $u^y_{(ijt)}$	Soil water velocity, x, y direction
$v^x_{(ijt)}$ $v^y_{(ijt)}$	Surface water velocity, x, y direction

V. Optimization Over the Schedule

An irrigation system that is capable of sprinkler-level actuation must be tested to see if it is a viable option as a control system. To test its capabilities, we require the ability to produce optimal sprinkler scheduling for the system. The objective of this optimization is to produce a schedule that provides enough moisture at all points in the space to maintain health, while minimizing system water consumption.

Our optimization problem is the following linear program (LP): The objective function is total water spent over a period N_t of time. We define $f_k(t)$ as a binary function that equals 0 if sprinkler k is off at time t and 1 otherwise, then the water spending is proportional to $\sum_{k=1}^{K}\int_0^{N_t} f_k(t)dt$. Discretizing over time gives as objective function $\sum_{k,t=0}^{K,N_t} f_{kt}$, where f_{kt} are $K \times N_t$ optimization variables (the sprinklers' schedule).

We have as additional optimization variables the values of $h, \theta, u^x, u^y, v^x, v^y$, (6 variables) at each spatiotemporal cell, with a total of $N_t \times N_x \times N_y \times 6$ variables. So the complete set of *optimization variables* is $\{f_{kt}, h_{ijt}, \theta_{ijt}, u^x_{ijt}, u^y_{ijt}, v^x_{ijt}, v^y_{ijt}\}_{i,j,t=0}^{N_x,N_y,N_t}$.

The equality constraints arise from the necessity of the joint values of these variables to satisfy the fluid flow PDEs everywhere in time and space. There are 6 PDEs, hence we have 6 equality constraints for each spatiotemporal cell. They are given by the linearized, discretized PDEs of Appendix B. As each constraint involves only 4 variables because of the spatial neighborhood relation induced by the finite differences, the matrix of equality constraints is sparse.

The inequality constraints define the goal state of the system, namely a schedule must provide volumetric water content in the soil exceeding the minimum water content plants need, θ_{pwp}, and be within prescribed lower and upper limits θ_{l} and θ_{u}. These constraints are of the bound type, i.e., they have the form "variable $\leq$ constant" for each variable.

The optimization variables f_{kt} are binary, which makes the problem an integer linear program (ILP). ILPs are NP-complete and must be approximated in practice. Here, we simply relax them to the continuous LP by letting each variable

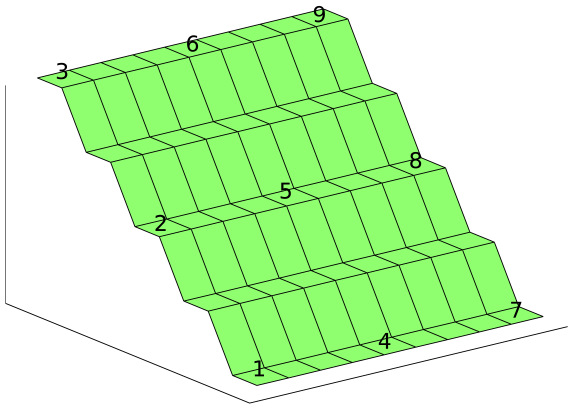
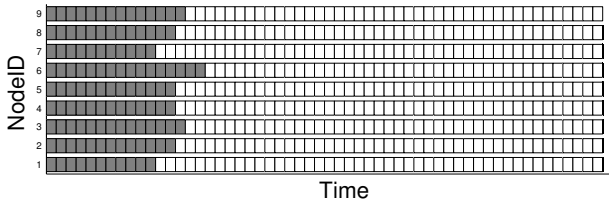
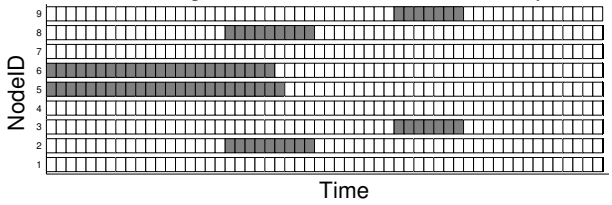


Fig. 4. MAGIC sprinkler and sensor locations



(a) Optimized Schedule without Intermittency



(b) Optimized Schedule with Intermittency

Fig. 5. Example Optimized Schedules

be real in [0,1]. Although other approaches exist that can give better approximations (such as branch-and-bound), they are impractical for the size of our problem.

The discretization in space and time results in a large number of variables and constraints. For example, using a coarse spatial grid of 10×10 with 100 timesteps results in 10000 cells and so 60000 variables (plus $100\times K$ schedule variables for K sprinklers), 60000 equality constraints and 10000 inequality constraints. Fortunately, the equality constraints are sparse and the inequality constraints are simple bounds.

We currently use Stanford's CVX convex optimization library [21] to solve the LP. CVX is convenient for our project because of its flexibility in the choice of LP solvers and its programmatic interface.

$$\min_{\{f_{kt},h_{ijt},\theta_{ijt},u^x_{ijt},u^y_{ijt},v^x_{ijt},v^y_{ijt}\}_{i,j,t=0}^{N_x,N_y,N_t}} \sum_{k=1}^{K}\sum_{t=0}^{N_t} f_{kt} \quad \text{s.t.} \tag{10}$$

$$0 \le f_{kt} \le 1 \qquad \substack{k=1,\dots,K \\ t=0,\dots,N_t} \tag{11}$$

$$\theta_{\mathrm{l}} \le \theta_{ijt} \le \theta_{\mathrm{u}} \qquad \substack{i=0,\dots,N_x \\ j=0,\dots,N_y \\ t=0,\dots,N_t} \tag{12}$$

$$\theta_{\mathrm{pwp}} \le \theta_{ijN_t} \qquad \substack{i=0,\dots,N_x \\ j=0,\dots,N_y} \tag{13}$$

$$\text{PDE model equations (16)–(21)} \tag{14}$$

The PDE model equations are as calculated in Section IV-D2, and can be found in Appendix B.

VI. Proof-of-concept Simulation

As the model is integrated into the optimization framework, we can perform a proof-of-concept experiment to ensure optimization produces schedules that follow intuition. Figure 4 shows the topography and node locations tested, made to resemble the hillside used in our case study shown in Figure 6. The hillside was modeled with soil characteristics as would be found in the deployment location, and the optimization was performed. The schedule produced can be seen in Figure 5a, where the dark blocks correspond to active sprinklers as time progresses on the x axis. As this is an example problem, the time is purposefully unitless as it does not represent a real scenario. This schedule can be observed to favor irrigation at the top of the hill, and favor Nodes 1 and 7 least, as they are located in the bottom corners. This follows intuition, as any unused water at the top of the hill will move away as runoff, and benefit the downhill turf.

The soil was then adjusted to mimic a less-absorbant clay, and its thickness was reduced. The thin soil requires less water, but as the clay is less absorbant, watering all at once would cause most water to be lost as runoff. The schedule produced in response to this environmental change is shown in Figure 5b where the dark blocks correspond to active sprinklers. The optimizer finds a solution that causes actuation to occur intermittently, making irrigation non-continuous. As the less-absorbant soil causes runoff to occur much more dramatically, the optimal solution prevents the lower sprinklers from actuating at all, allowing the runoff from above to provide adequate moisture to the region below.

VII. Case Study: Live Deployment

In many applications, it is possible to compare a newly-developed model to other accepted models. However, as an all-inclusive model for our application does not exist, we compare to reality by evaluating the performance of an irrigation system using the control modifications we have proposed.

We chose to deploy two systems side-by-side once a suitable location is found. Ideally, the two systems will cover similar soil and turf, face the same direction so that sun exposure will be equivalent, and have completely independent irrigation to avoid cross-contamination. In addition, to ensure all sprinkler coverage is the same, the same water source is used to power both systems, and actuation is provided to both sides with the installation of our MAGIC nodes. The only difference between the systems are the control schedules sent to each side.

In the beginning of our project, we intended to use an existing irrigation system to perform our deployment. In looking for a suitable location, we came to realize that the granularity of irrigation control on our University's campus was less than ideal. Locations spanning more than ten thousand square feet across heterogeneous terrain were actuated by a single control valve. Although MAGIC can simply be screwed into place at each sprinkler to retrofit an existing irrigation

Fig. 6. Deployment side-view

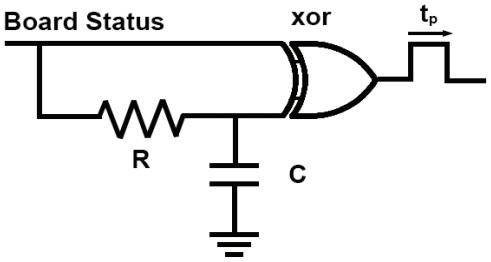


Fig. 7. Example solenoid safety mechanism

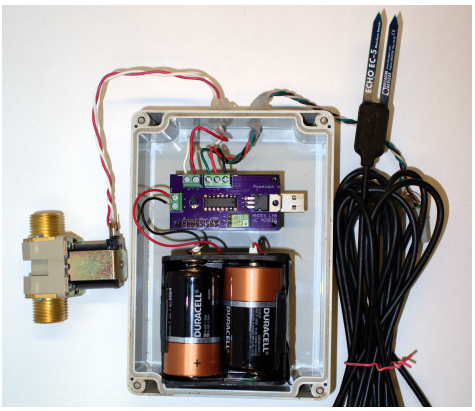


Fig. 8. Sensing/actuation (MAGIC) node

system, University officials preferred that the existing system not be touched. For this reason, we designed and deployed our own parallel irrigation system.

A. Seeking a Suitable Location

With the help of University personnel, a suitable location was found on a stepped hillside (see Figure 6) far away from the nearest foot-traffic. The hill, rising about 9 feet over a distance of 70 feet, acted as an elevated surround for a University soccer field, the hill stretching more than 200 feet. To take advantage of this topography, two irrigation systems were placed side by side, each spanning a 70'x70' area along the hillside. Between the two systems were 5 feet of nonirrigated space, to prevent any spray from one system to enter the other side. As a whole, the two irrigation systems spanned an area of 70'x145', approximately 10,150ft^2.

B. System Development and Deployment

The underground irrigation system used by groundskeepers to maintain the hillside was plumbed with high-flow water lines leading to each of their sprinklers. As campus greenskeepers preferred us not to manipulate their system, our system drew its water from nearby hose spigots. Flow limitations of the hose spigots did not allow actuation of both systems at the same time, so the two irrigation systems are irrigated sequentially during our case study. Although a commercialized version of MAGIC would be installed in the ground, our temporary system was placed on the surface for ease of access to our prototype. The nodes themselves were placed on the ground just next to the sprinklers, while the solenoid providing actuation was fixed to the sprinkler riser.

To choose the sprinkler configuration for our side-by-side deployment, we consulted sprinkler manufacturer specifications; a rule of thumb for system planning is that the coverage of one sprinkler should reach 75-100% of the distance to the next closest sprinkler, to avoid uncovered areas on the diagonal. To cover one of the two 70'×70', we would require 9 sprinklers in a 3×3 grid, each reaching more than 26'. It was found that the low-flow MP-Rotator 3000 sprinkler [5], a new rotor that is known to be remarkably efficient, could reach up to 30', 9 of which can be powered by our water source. As the MP-Rotator is quickly replacing older sprinkler technologies at our University for their slow-application and minimum runoff, we decided they would be the best choice for our deployment.

Lastly, our deployment included a central basestation fitted with power, a small plug computer, an elevated 802.15.4 mote to receive data from the sensing nodes, and a 4G hotspot, to allow us to communicate with the wireless sensor network from a remote location. Although our basestation was overbuilt to facilitate ease of debugging and close monitoring of the prototype system, a commercialized system may have only the mote to interface with the sensor network, and an interface to any external service (local, cloud, etc).

C. Node Development

A standard solenoid requires constant power to allow water to flow, making it a poor choice for a battery-powered system. However, one benefit is that a power failure on the control node will cause the solenoid to close, preventing continued irrigation and potentially serious flooding. To extend node lifetime, we chose to use a latching solenoid for sprinkler actuation, requiring only a 50ms pulse of positive (to open) or negative (to close) voltage. To prevent over-irrigation in the event of a power failure, a simple circuit can be built into the MAGIC board that sends a closing pulse to the solenoid if a loss of power is detected. As shown in Figure 7, a pin is maintained on the microcontroller that changes from 0 to 1 in event of failure (or from 1 to 0, a NOT gate can be used to invert it). Once this pin is active it immediately brings the top pin of an XOR logic gate to 1, allowing current to flow through the circuit. Once the capacitor charges, the second pin of the XOR logic gate activates, terminating the current. This solenoid-terminating pulse, generated while the capacitor charges, is for a period $t_p \propto CR$, where C and R are the capacitance and resistance values used in the circuit. With this

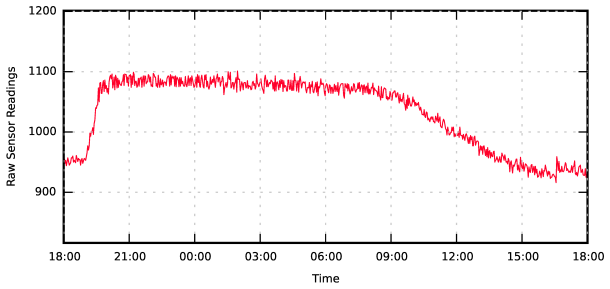


Fig. 9. Daily soil moisture cycle

configuration, the control node will benefit from the low-power operation of a latching solenoid without sacrificing safety.

To allow the MAGIC node to manage the input/output connections from the mote, we manufactured a printed circuit board to interface with these peripherals. This interface board, shown in Figure 8, has connections for battery power, sensor (right) and solenoid (left). The different voltages required by the mote and solenoid were provided by voltage regulators built into the board. As the chosen latching solenoid requires the board to produce a positive (opening) and negative (closing) voltage for operation, the board was equipped with an h-bridge, allowing us to easily produce bi-directional current.

A key feature of the MAGIC node is the ability to measure the volumetric water content in the surrounding soil. We opted to purchase research-quality Decagon EC-5 [2] sensors, with a reported accuracy of $\pm 3\%$. The Decagon sensors cost roughly $110, but data fidelity in our model verification was paramount. Raw sensor readings collected over a period of one day with a high sampling frequency can be seen in Figure 9. The sensors report the dielectric constant of the soil, an electrical property highly dependent on the volumetric water content. A linear calibration function provided by the sensor manufacturer is used to convert the raw readings to volumetric water content. An attentive reader will notice the sharp increase in the late afternoon due to irrigation, stable readings throughout the night, with sensor fall beginning as the sun comes up at 7:45am. This decrease steepens as the sun rises and faces the deployment directly after 10am. In a sensor-dense environment, a more economical alternative to the EC-5 may be chosen, such as the 35$ Watermark sensor [29], which can be calibrated to $\pm 5\%$ accuracy.

D. System Comparison

In this study, we compare the operation of the MAGIC system against two baseline control systems. The first baseline, evaluated in a 5-week deployment, employs a trial-and-error control strategy used widely in practice (including our campus). A greenskeeper will monitor an irrigation system for days or weeks; if an excess of runoff provides evidence of over-watering, or if brown patches provide evidence of under-watering, they will adjust the system accordingly. This irrigation scheduling, often remaining unchanged through entire seasons, leads to a misuse of water as it does not account for changing weather or soil requirements. We emulate this strategy by matching exactly the amount of water coverage as would be provided by the greenskeepers of our campus.

The second baseline control strategy, evaluated in a 2-week deployment, employs a state-of-the-art evapotranspiration (ET) control strategy. As described in Section II, these systems use weather forecasting to estimate the amount of water lost by the soil due to evaporation and plant transpiration. Irrigation controllers that use ET technology typically irrigate every 1-3 days, replacing water lost over that period. To emulate an ET system, we query a local weather station that computes hourly ET loss, and compute the previous day's water losses. With our sprinklers' surface coverage rate, we create daily valve schedules to do exact replacement of these losses.

VIII. Experimental Results

Through 7 weeks of deployment, we ran two irrigation systems side-by-side on identical patches of turf, periodically collecting soil moisture data from each. For the first 5 week deployment, our campus' control strategy was tested and for the second 2 week deployment, state-of-the-art evapotranspiration (ET) control was used. In both deployments, these systems were compared against schedules computed by our model-based optimization, actuated using our independent actuation/sensing (MAGIC) nodes. We focus our analysis on the comparison between MAGIC and state-of-the-art ET system, hereby named "Control". The goal of these case-studies was to determine if the MAGIC system could reduce the amount of water used without sacrificing quality of irrigation.

A. Quality of Service

The primary objective of an irrigation system is to maintain plant health. A very efficient system that is unable to meet this objective will be replaced with a less-efficient system that provides satisfactory water coverage to the turf. To reduce water consumption during irrigation, we must know the minimum level of soil moisture that is still able to keep the plant healthy. This level of volumetric water content (VWC) is known in hydrology as the permanent wilting point (θ_{pwp}), a soil characteristic where the matric suction forces binding the water to soil particles are equal to the suction exhibited by the plant. Prolonged residence in a soil with moisture beneath θ_{pwp} will cause wilting and eventual death to the plant, so we make the assumption that less time spent below this threshold will help maintain long-term plant health.

To prevent poor plant health, we want to minimize the amount of time spent with soil VWC under θ_{pwp}. The permanent wilting point for loamy soils like that found in our deployment is typically between 10-15% [9], so we assume the worst case and assign θ_{pwp} to be 15%. We expect that if MAGIC were to distribute moisture in a smarter way by targeting areas that would otherwise receive inadequate water, the MAGIC sensor readings will spend less time underneath the θ_{pwp} threshold than the Control system (ET).

The sensor data from each MAGIC node during the deployment and the minimum healthy saturation θ_{pwp}=.15 are shown in Figure 10. It can be observed that the Control system (left) spends much more time with readings underneath θ_{pwp}. An exception is seen during day 11, where seasonal

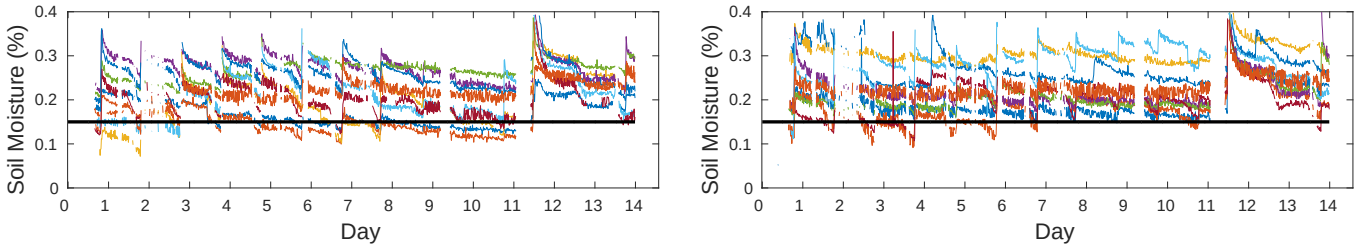


Fig. 10. Data snapshot for RMSE calculation for Control (left) and MAGIC (right) systems

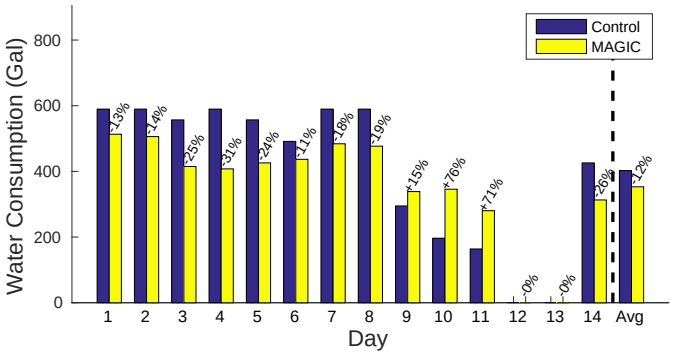


Fig. 11. Water Consumption of Parallel Systems

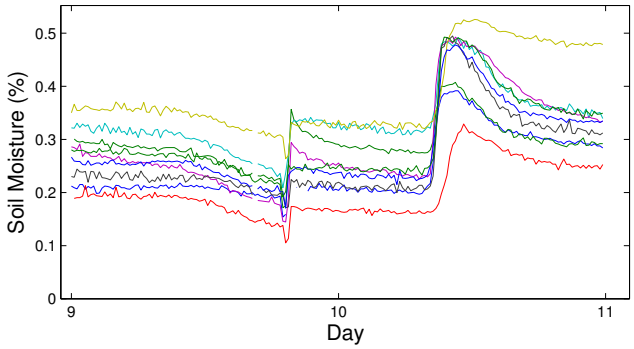
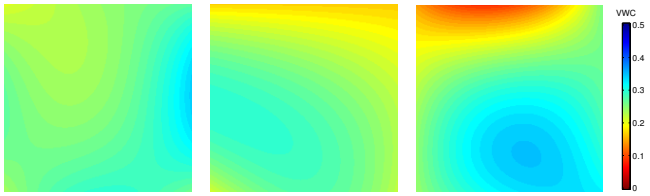


Fig. 12. MAGIC sensor readings through rain



(a) MAGIC (b) ET (c) Campus

Fig. 13. Average moisture coverage of compared systems

rain occurred. The following two days showed higher-than-average moisture for both systems. In spite of this, the Control side spends a combined 68.1 hours beneath the θ_{pwp} across the entire deployment, over 4x more than the 16.7 hours experienced by the MAGIC system. Similarly, comparison between MAGIC and our campus' irrigation control revealed a 3.23x improvement in quality. Although this shows that MAGIC is not perfect, it also demonstrates that more precise watering strategies can provide an improved quality of service, despite using less water.

B. Water Consumption Analysis

Throughout the deployment, the total on-time for each sprinkler was recorded in both systems. This actuation time is used to calculate each system's total water consumption, based on each sprinkler's regulated flow rate and coverage. A side-by-side comparison of each system's water consumption is shown for the two weeks of deployment in Figure 11. The variation of water consumption of the Control system indicates changes in local weather. Low water consumption of both systems on days 9, 10, and 11 is due to cooler local weather, where the Control side consumed significantly less water. Rain on the 12th day caused both systems to cease irrigation until two days later, when the soil became dry enough to require it.

Across the entire deployment, the MAGIC system consumed 12.3% less water than the Control (ET) system, and 23.4% less in comparison to our campus' control strategy. Schedules created by the MAGIC system were shown to consume as much as 513 gallons and as little as 280 gallons, a 233 gallon variation in response to the state of soil moisture. The reason our system is able to save water while providing a higher quality of service to the space is due to our ability to pinpoint regions within the irrigated space with varying moisture requirements. Through our model-driven approach, we can optimize actuation to send more water to areas that would otherwise receive insufficient moisture, while sending less water to the areas that would otherwise be over-watered.

Sensor readings through rain during the deployment can be seen in Figure 12. To respond to such weather events, the weather stations used for evapotranspiration monitoring are generally equipped with a precipitation sensor which deactivates the irrigation system when rain is first experienced [6], or when a measured amount of rain has fallen. In the case where rain is sufficient to irrigate the turf, this is satisfactory, but if the rain is very light, MAGIC might choose to provide additional irrigation.

C. Moisture Uniformity

An interesting pattern recognized throughout the deployments was the emergence of moisture uniformity. The optimization performed by MAGIC will create schedules that compensate for areas of high or low moisture levels using soil moisture information. However, due to lack of moisture information and the physical limitations of the irrigation system (inability to actuate individual sprinklers), the ET and our campus' systems are unable to correct for uneven moisture. These artifacts can be seen in Figure 13, generated from data collected during both the 5-week Spring deployment, and the 2-week Fall deployment. In this figure, the top of

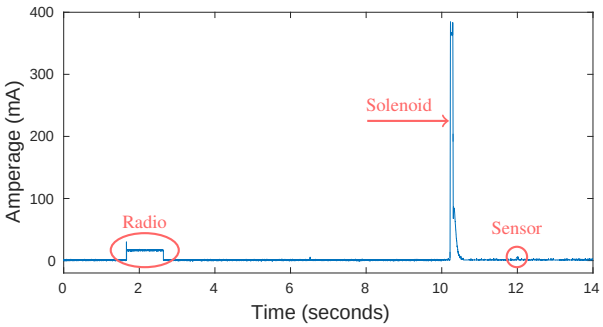


Fig. 14. Sample energy trace of the Magic Node

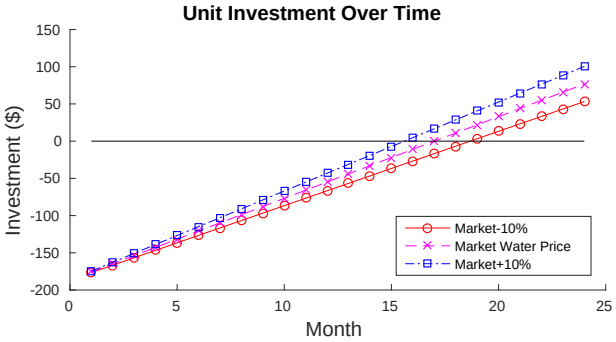


Fig. 15. Return of Investment timeline with varying water pricing

the image aligns with the uphill region depicted in Figure 4. Although variation in weather patterns between these two separate deployments makes a true side-by-side comparison difficult to make, these results demonstrate MAGIC's ability to produce schedules that correct for heterogeneities to provide homogeneous water coverage.

D. Energy Consumption Analysis

From a wireless sensor network standpoint, the ability of a system to operate for a long period of time without user intervention is fundamental. Irrigation control nodes are no different, especially if they are meant to be buried in the ground. For this reason, our prototype hardware and software were designed to consume as little energy as possible. To this end, the MAGIC nodes were fitted with a latching solenoid, allowing the flow of water to be turned on or off with a short pulse of power, rather than a constant supply. For additional energy savings, the radio in each node is duty-cycled, activating for only a 10 second period every 10 minutes. It was computed that using this duty-cycle, the 4 D cell batteries providing power to our nodes could run for over 2 years without requiring change. By performing more aggressive duty cycling, reducing radio activation to only one minute each day, our prototype irrigation system could run uninterrupted in excess of 14 years, while still performing its daily irrigation and data collection. A sample energy trace of our prototype can be seen in Figure 14, where radio, solenoid, and sensor consumption is shown over background cpu usage.

IX. Return on Investment Analysis

A primary consideration before the purchase and installation of a replacement control system is the return on investment, or the time it takes a system to save enough money to cover the cost of installation and usage. To calculate the return on investment, we must take into account the initial cost of the replacement system and the monetary savings expected from the increased efficiency of the replacement system.

Here we consider the cost to develop a single MAGIC node in bulk for return on investment analysis. The primary components can be readily found; the sensor, solenoid, batteries, and waterproof enclosure are all possible to purchase from other manufacturers. In our prototype the communication module used was a tmote sky [11]. However, as our application requires very specific circuitry to provide power to the various modules, commercialization of MAGIC would involve the manufacture of a stripped-down communication module, with the inclusion of the additional components described in Section VII-C. The pricing of a barebones tmote sky replacement and other MAGIC components are listed in Table III.

TABLE III
Sprinkler Node Manufacture Cost

Component	Price
Mote	$37.57
Moisture Sensor	$110
Batteries	$4
Solenoid	$15
Waterproof Enclosure	$10
Manufacture & Assembly	$10
	$186.57

To evaluate the expected return of investment, we compute the cost of the system and calculated the net investment as time progresses. The factor that most influences system payback is price of water. As this value is constantly changing, we perform the analysis using the current price for our campus, $5.60 per thousand gallons, and incorporate a deviation of 10% to account for changing water market prices. Although the unit itself carries a high initial cost, return of investment can be expected to occur in 16-18 months, as shown in Figure 15. Considering the immense political pressure for irrigation water to be more expensive [3], [10], it is a good bet that the savings from the MAGIC system will be on the rise in the near future. Even with the addition of a 20% business markup, return of investment can still be expected in 18-23 months.

It is difficult to directly extend the savings seen in our prototype to all irrigated space on a University's campus due to the heterogeneity of the installed system architectures. However, the MAGIC system can be easily upgraded to control sprinklers of any type, delivering site-specific actuation for systems at any scale. For example, on a campus such as ours, the majority of irrigated spaces use rotor sprinklers. As the rotors use substantially more water, independent actuation could provide an even greater positive environmental and financial impact, to be investigated in future work.

X. Discussion

Weather during the deployment were mainly hot and sunny, interrupted by 3 cool but sunny days and a day of rain,

indicative of a changing season. During the 3 days of cooler weather, the Control (ET) system is seen in Figure 11 to consume significantly less water than MAGIC. Shown in Figure 10, these 3 days are also when the Control system spends the most time underneath a healthy level of moisture, the period of worst service quality. In contrast, the MAGIC system manages to maintain a healthy level of moisture. This shows that the Control system's use of weather only ignores the specific water needs across the space. If these shoulder weather days are omitted from our analysis, MAGIC reduces water consumption by 20%, and quality of service is improved by a factor of 2.4 (in comparison to 12% reduction and over 4x quality improvement). This shows that although the evapotranspiration system is capable of saving water by correcting for weather patterns, it is not always able to do so in a way that maintains plant health.

One key benefit of the MAGIC system is its ease of installation and modification. Development of a new sprinkler like the MP Rotator, which claims a 30% efficiency improvement can easily be installed to the MAGIC node, and will benefit from MAGIC's actuation decisions.

XI. Limitations and Future Work

The MAGIC system is best suited in any irrigated space where water is expected to move due to slope (downhill movement) or a soil layer deeper than the root zone (leeching). In cases where the soil is flat and bounded below by an inpenetrable layer, MAGIC may not be substantially more efficient than ET-based systems. As soil conditions are influenced directly by recent weather effects, we expect MAGIC to perform at least as efficiently as an ET system in all scenarios. Evaluation of MAGIC effectiveness in these conditions is planned for future work.

Although MAGIC can respond to recent weather events detected by the soil moisture sensors, explicit weather forecasting would alow predictive measures to be taken, for example disabling irrigation if rain is expected in the next 48 hours. Incorporating this input into the model can be done by providing a weather term to the surface water in Eq. 5, similarly to sprinkler input F_s.

In our deployment, schedule optimization took several minutes, with the number of optimization variables growing as a function of spatial and temporal discretization, $N_t \times N_x \times N_y \times 6$. In the future, we intend to evaluate MAGIC operation in truly large-scale irrigation systems and investigate methods of improving system performance. One way to speed up the linear program solver is by finding a good initialization for the optimization variables (not to be confused with the initial conditions of the PDE model). In our case this could be provided by a multiscale approach, where we first solve a LP using a coarse grid (spatial and/or temporal), extrapolate this solution to a finer grid, and use this as initial point for the iterative LP optimizer. This reduces the number of iterations spent solving the fine grid and thus the overall runtime.

To simplify the placement of irrigation infrastructure, sprinklers are almost always installed in a grid pattern. However, it might be possible to use the developed model to compute ideal sprinkler locations to compensate for natural topological characteristics for a system that has not yet been installed. In future work, we hope to evaluate the potential for water savings by also allowing variation in sprinkler positioning.

For MAGIC to estimate water movement, it must understand the spatial characteristics of the irrigated turf. This includes the topography of the terrain and estimates of soil type and depth. Topography measurement of advanced terrain can be performed by various imaging methods by satellite, drone, or other emerging technologies, and soil characteristics may be estimated by the system installer, but they remain a burden. To lessen these requirements, future work may be directed to data-driven system identification, where soil moisture measurements throughout the space and knowledge of fluid movement can be used to build the model over time.

XII. Conclusions

Fresh water is a delicate resource, and we must find ways to use it sustainably. The largest irrigated crop by surface area in North America, turf has a demand for an estimated 9 billion gallons each day. Due to current shortages, there is strong social, environmental, and monetary incentive to shrink this enormous consumer. In this work, we seek to improve the efficiency of turf irrigation systems by analyzing heterogeneous water needs across a span of turf. To this end, we develop a computationally-light moisture movement model within an optimization framework to produce optimal valve scheduling within an irrigation system. To test its effectiveness we produce the MAGIC sprinkler node, with the ability to actuate, sense local soil conditions, and communicate wirelessly with sister nodes in the network. Through two separate deployments spanning a total of 7 weeks, we find that the MAGIC system can reduce system water consumption by 23.4% over our campus' control strategy, and by 12.3% over a state-of-the-art evapotranspiration system. Despite this reduced water usage, MAGIC was also found to reduce turf exposure to unhealthy levels of moisture by a factor of 3.23 over the campus' control, and a factor of 4.08 over the evapotranspiration control. The MAGIC system is expected to return its investment in 16-18 months based on water savings alone.

XIII. Acknowledgements

We would like to thank all the referees and our shepherd Kay Römer for their constructive comments and helpful suggestions. This material is based upon work partially supported by the National Science Foundation under grants #CNS-1254192 and #CNS-0923586, and the Center for Information Technology Research in the Interest of Society.

Appendix A
Grass as Porous Media

Darcy's law [17] is applicable to systems where the drag due to the multiple obstructions is the dominant fluid force. To determine when flow through turf may be modeled as flow through a porous medium, we estimate and compare

the drag, viscous force, and inertial force per unit volume. Denoting a typical flow velocity scale as U, a typical grass blade size as a, viscosity as μ, density as ρ, the thickness of the liquid layer as h, and the porosity as ϕ, we estimate the drag per volume, based on the low Reynolds number drag of an elongated obstruction of length h and lateral size a, as

$$D \sim \frac{4\pi\mu U h(1-\phi)}{ha^2} = \frac{4\pi\mu U(1-\phi)}{a^2}. \tag{15}$$

Within the liquid, the viscous force per volume is estimated from the viscous term in the Navier-Stokes equations, as $F_v \sim \mu U/h^2$. Finally, the inertial force per volume is estimated, also from the Navier-Stokes equations, as $F_i \sim \rho U^2/h$.

As typical values for water and grass blades, we use $\rho = 1\text{g/cm}^3$, $\mu = 0.01\text{g/cm s}$, $a = 0.1\text{cm}$, $h = 1\text{cm}$, and $\phi = 0.9$, and find $\frac{F_v}{D} \sim \frac{a^2}{4\pi h^2}\frac{1}{1-\phi} \sim 1/100$ showing that the drag is dominant over viscous forces for any velocity of the flow. The ratio of inertial forces to drag is $\frac{F_i}{D} \sim \frac{a^2\rho U}{4\pi\mu h}\frac{1}{1-\phi} \sim U$. The applicability of our model is therefore limited to systems where the water flows at velocities $U < 1\text{cm/s}$, which corresponds to most intermittent irrigation regimes.

APPENDIX B
DISCRETIZED MODEL FORMULATION

The following 6 equations hold at each spatiotemporal cell (i, j, t) and represent the discretized, linearized flow motion of Equations (6)–(9) over the variables u and v (velocity of water in soil and surface, respectively, both horizontal components x and y for each), h (surface fluid height) and theta (volumetric moisture content). They also represent the equality constraints in our optimization problem.

$$\begin{aligned}\tfrac{h_{i,j,t+1}-h_{i,j,t}}{\Delta t} = &-\tfrac{1}{2\Delta x}\Big((\hat{h}v_0^x + h_0\hat{v}^x)_{i+1,j,t} - (\hat{h}v_0^x + h_0\hat{v}^x)_{i-1,j,t} \\ &+(\hat{h}v_0^y + h_0\hat{v}^y)_{i,j+1,t} - (\hat{h}v_0^y + h_0\hat{v}^y)_{i,j-1,t}\Big) + F_s \\ &-\eta\Big(h_0K(\theta_0) + h_0K'(\theta_0)\hat{\theta} + \hat{h}K(\theta_0) + \hat{h}K'(\theta_0)\hat{\theta}\Big)\end{aligned} \tag{16}$$

$$\begin{aligned}\tfrac{\theta_{i,j,t+1}-\theta_{i,j,t}}{\Delta t} = &-\tfrac{1}{2\Delta x}\Big((\hat{\theta}u_0^x + \theta_0\hat{u}^x)_{i+1,j,t} - (\hat{\theta}u_0^x + \theta_0\hat{u}^x)_{i-1,j,t} \\ &+(\hat{\theta}u_0^y + \theta_0\hat{u}^y)_{i,j+1,t} - (\hat{\theta}u_0^y + \theta_0\hat{u}^y)_{i,j-1,t}\Big) \\ &+\zeta\Big(h_0K(\theta_0) + h_0K'(\theta_0)\hat{\theta} + \hat{h}K(\theta_0) + \hat{h}K'(\theta_0)\hat{\theta}\Big)\end{aligned} \tag{17}$$

$$\begin{aligned}u^x_{i,j,t} = &-\tfrac{K(\theta_{0i,j,t})}{2\Delta x}(\hat{h}_{i+1,j,t} - \hat{h}_{i-1,j,t} + h_{0i+1,j,t} - h_{0i-1,j,t}) \\ &-K'(\theta_{0i,j,t})\hat{\theta}_{i,j,t}(h_{0i+1,j,t} - h_{0i-1,j,t}) + K(\theta_{i,j,t})\tau^x \\ &-\tfrac{\varphi(\theta_{0i,j,t})}{2\Delta x}(\theta_{0i+1,j,t} - \theta_{0i-1,j,t} + \hat{\theta}_{i+1,j,t} - \hat{\theta}_{i-1,j,t}) \\ &-\varphi'(\theta_{0i,j,t})(\theta_{0i+1,j,t} - \theta_{0i-1,j,t})\hat{\theta}_{i,j,t}\end{aligned} \tag{18}$$

$$\begin{aligned}u^y_{i,j,t} = &-\tfrac{K(\theta_{0i,j,t})}{2\Delta x}(\hat{h}_{i,j+1,t} - \hat{h}_{i,j-1,t} + h_{0i,j+1,t} - h_{0i,j-1,t}) \\ &-K'(\theta_{0i,j,t})\hat{\theta}_{i,j,t}(h_{0i,j+1,t} - h_{0i,j-1,t}) + K(\theta_{i,j,t})\tau^y \\ &-\tfrac{\varphi(\theta_{0i,j,t})}{2\Delta x}(\theta_{0i,j+1,t} - \theta_{0i,j-1,t} + \hat{\theta}_{i,j+1,t} - \hat{\theta}_{i,j-1,t}) \\ &-\varphi'(\theta_{0i,j,t})(\theta_{0i,j+1,t} - \theta_{0i,j-1,t})\hat{\theta}_{i,j,t}\end{aligned} \tag{19}$$

$$v^x_{i,j,t} = -\alpha_h\Big(\tfrac{h_{i+1,j,t}-h_{i-1,j,t}}{2\Delta x}\Big) + \tau^x \tag{20}$$

$$v^y_{i,j,t} = -\alpha_h\Big(\tfrac{h_{i,j+1,t}-h_{i,j-1,t}}{2\Delta x}\Big) + \tau^y \tag{21}$$

REFERENCES

[1] Comsol subsurface flow module. http://www.comsol.com/subsurface-flow-module.
[2] Decagon devices. http://www.decagon.com/products/soils/.
[3] Farming irrigation price increases. http://www.bloomberg.com/news/articles/2014-07-24/california-water-prices-soar-for-farmers-as-drought-grows.
[4] Freshwater crisis. http://environment.nationalgeographic.com/environment/freshwater/freshwater-crisis/.
[5] Hunter MP rotator. http://www.hunterindustries.com/irrigation-product/nozzles/ mp-rotator.
[6] Hunter rain-clik rain detection. http://www.hunterindustries.com/irrigation-product/sensors/rain-clikr.
[7] Hydrus liquid simulator. http://www.pc-progress.com.
[8] Looking for lawns. http://earthobservatory.nasa.gov/Features/Lawn/.
[9] Permanent wilting point. http://nrcca.cals.cornell.edu/soil/CA2/CA0212.1-3.php.
[10] Price of water is too low. http://www.nytimes.com/2014/10/15/business/economy/the-price-of-water-is-too-low.html.
[11] Tmote sky. http://www.snm.ethz.ch/Projects/TmoteSky.
[12] UGMO irrigation. http://www.ugmo.com/.
[13] US outdoor water use. www3.epa.gov/watersense/pubs/outdoor.html.
[14] R. G. Allen, L. S. Pereira, D. Raes, M. Smith, et al. Crop evapotranspiration-guidelines for computing crop water requirements-fao irrigation and drainage paper 56. *FAO, Rome*, 300(9):D05109, 1998.
[15] B. Cardenas-Lailhacar, M. D. Dukes, and G. L. Miller. Sensor-based automation of irrigation on bermudagrass, during wet weather conditions. *Journal of Irrigation and Drainage Engineering*, 134(2):120–128, 2008.
[16] A. Costa. Permeability-porosity relationship: A reexamination of the Kozeny-Carman equation based on a fractal pore-space geometry assumption. *Geophysical Research Letters*, 33(2):n/a–n/a, 2006.
[17] H. Darcy. Les fontaines publiques de la ville de Dijon. Paris: Dalmont. 1856.
[18] N. Dobbs, K. Migliaccio, M. Dukes, K. Morgan, and Y. Li. Interactive irrigation tool for simulating smart irrigation technologies in lawn turf. *Journal of Irrigation and Drainage Engineering*, 2013.
[19] Q. Gallas, R. Holman, T. Nishida, B. Carroll, M. Sheplak, and L. Cattafesta. Lumped element modeling of piezoelectric-driven synthetic jet actuators. *AIAA journal*, 41(2):240–247, 2003.
[20] M. T. V. Genuchten. A closed-form equation for predicting the hydraulic conductivity of unsaturated soils. *Soil science society of America journal*, 44(5):892–898, 1980.
[21] M. Grant, S. Boyd, and Y. Ye. CVX: Matlab software for disciplined convex programming, 2008.
[22] T. Harter and J. R. Lund. Addressing nitrate in california's drinking water. Technical report, University of California, Davis, 2012.
[23] M. E. Jensen, R. D. Burman, and R. G. Allen. Evapotranspiration and irrigation water requirements. ASCE, 1990.
[24] E. Kalnay. *Atmospheric modeling, data assimilation, and predictability*. Cambridge university press, 2003.
[25] Y. Kim, R. Evans, and W. Iversen. Remote sensing and control of an irrigation system using a distributed wireless sensor network. *IEEE Transactions on Instrumentation and Measurement*, 57(7):1379–1387, July 2008.
[26] E. Playán, R. Salvador, C. López, S. Lecina, F. Dechmi, and N. Zapata. Solid-set sprinkler irrigation controllers driven by simulation models: Opportunities and bottlenecks. *Journal of Irrigation and Drainage Engineering*, 140(1):04013001, 2013.
[27] R. H. Reichle, D. B. McLaughlin, and D. Entekhabi. Hydrologic data assimilation with the ensemble kalman filter. *Monthly Weather Review*, 130(1):103–114, 2002.
[28] E. Rodriguez, F. Giacomelli, and A. Vazquez. Permeability-porosity relationship in RTM for different fiberglass and natural reinforcements. *Journal of composite materials*, 38(3):259–268, 2004.
[29] E. Spaans and J. Baker. Calibration of watermark soil moisture sensors for soil matric potential and temperature. *Plant and Soil*, 143(2):213–217, 1992.
[30] K. Wesseling, J. Jansen, J. Settels, J. Schreuder, et al. Computation of aortic flow from pressure in humans using a nonlinear, three-element model. *Journal of applied physiology*, 74:2566–2566, 1993.
[31] D. Winkler and A. E. Cerpa. Distributed independent actuation for irrigation control. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, BuildSys '14, pages 148–151, New York, NY, USA, 2014. ACM.