

A METHOD AND SYSTEM FOR AUTOMATED DIFFERENTIAL IRRIGATION

FIELD OF THE INVENTION

The present invention relates to the field of agricultural irrigation.

BACKGROUND OF THE INVENTION

Various systems for automated agricultural irrigation are known.

SUMMARY OF THE INVENTION

In various preferred embodiments, the present invention provides a method for reducing the amount of water required to irrigate an agriculture field, by applying different amounts of water to different parts of the field, based at least in part on an analysis of spatial soil properties of the field including topological features, and extrapolation of data from soil sensors placed in different parts of a field.

According to a preferred embodiment of the present invention provides a computerized differential irrigation system comprising:

- a computerized Topography Integrated Ground water Retention (TIGER) map generator receiving at least the following inputs:

- a topographical input describing topographical features of an area to be irrigated; and

- an electromagnetic input describing conductive features of the area to be irrigated,

and in which the computerized Topography Integrated Ground water Retention (TIGER) map generator includes:

- a computerized topographic feature processing functionality providing information relating to at least one of slope, aspect and catchment area features of said area to be irrigated; and

- a computerized topographic feature utilization functionality employing at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated; and

- a computerized computing functionality employing the Topography Integrated Ground water Retention (TIGER) map together with at least current outputs of wetness sensors located at the plurality of different regions within the area to be irrigated to generate a current irrigation plan; and

- a computerized irrigation control subsystem automatically utilizing the current irrigation map to control irrigation within the area to be irrigated based on the current irrigation instructions and to cause different amounts of water to be provided to the different regions within the area to be irrigated.

The invention also provides a computerized irrigation planning system comprising:

a computerized Topography Integrated Ground water Retention (TIGER) map generator receiving at least the following inputs:

a topographical input describing topographical features of an area to be irrigated; and

an electromagnetic input describing conductive features of the area to be irrigated,

and in which the computerized Topography Integrated Ground water Retention (TIGER) map generator includes:

a computerized topographic feature processing functionality providing information relating to at least one of slope, aspect and catchment area features of the area to be irrigated; and

a computerized topographic feature utilization functionality employing the at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated; and

a computerized computing functionality employing the Topography Integrated Ground water Retention (TIGER) map together with at least current outputs of wetness sensors located at the plurality of different regions within the area to be irrigated to generate a current irrigation plan.

The invention further provides an automated Topography Integrated Ground water Retention (TIGER) map generating system comprising:

a data input interface receiving at least the following inputs:

a topographical input describing topographical features of an area to be irrigated; and

an electromagnetic input describing conductive features of the area to be irrigated,

computerized topographic feature processing functionality automatically deriving from the inputs, information relating to at least one of slope, aspect and catchment area features of the area to be irrigated; and

computerized topographic feature utilization functionality employing the at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated.

The invention also provides an automated soil type classification system comprising:

an input interface receiving:

offline pre-existing laboratory generated soil drying curves, which indicate at least the following parameters for a plurality of different types of soils: field capacity, wilting point and refill point; and

empirical field drying curves for a field for which irrigation is to be planned;

and a computer operated automatic correlator employing the offline pre-existing laboratory generated soil drying curves and the empirical field drying curves for a field for which irrigation is to be planned to automatically provide a soil type map for the field for which irrigation is to be planned.

The invention also provides a computerized differential irrigation system comprising:

a computerized Topography Integrated Ground watEr Retention (TIGER) map generator receiving at least the following inputs:

a topographical input describing topographical features of an area to be irrigated; and

an electromagnetic input describing conductive features of the area to be irrigated,

and in which the computerized Topography Integrated Ground watEr Retention (TIGER) map generator includes:

a computerized automatic soil type analysis functionality which obviates the need for laboratory testing of soil in the area to be irrigated.

The invention also provides a computerized irrigation efficiency metric generating system comprising:

a computerized Topography Integrated Ground watEr Retention (TIGER) map generator receiving at least the following inputs:

a topographical input describing topographical features of an area to be irrigated; and

an electromagnetic input describing conductive features of the area to be irrigated,

and in which the computerized Topography Integrated Ground watEr Retention (TIGER) map generator includes:

a computerized topographic feature processing functionality providing information relating to at least one of slope, aspect and catchment area features of the area to be irrigated; and

a computerized topographic feature utilization functionality employing the at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated; and

a computing functionality employing the Topography Integrated Ground watEr Retention (TIGER) map together with at least current outputs of wetness sensors located at the plurality of different regions within the area to be irrigated to generate a current irrigation plan; and

an irrigation efficiency analyzer operative to:

ascertain an amount of water required to irrigate the area based on the current irrigation plan;

ascertain an amount of water required to irrigate the area if differential irrigation is not employed; and

calculate an irrigation efficiency metric representing a water saving produced by employing the current irrigation plan.

The invention also provides methods of using any one of the described and/or claimed systems within the body of this disclosure.

It is acknowledged that the terms "comprise", "comprises" and "comprising" may, under varying jurisdictions, be attributed with either an exclusive or an inclusive meaning. For the purpose of this specification, and unless otherwise noted, these terms are intended to have an inclusive meaning-

i.e. they will be taken to mean an inclusion of not only the listed components which the use directly references, but also to other non-specified components or elements.

This application is related to and claims priority from NZ Provisional Patent Application Serial No. NZ 603449, filed November 6 2012 and entitled Precision Irrigation Scheduling, the disclosure of which is hereby incorporated by reference.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will be understood and appreciated more fully from the following detailed description of the invention, taken in conjunction with the drawings in which:

FIG. 1 is a simplified schematic diagram, which provides an overview of a differential irrigation system constructed and operative in accordance with an embodiment of the present invention;

FIG. 2 is a simplified schematic diagram, which illustrates creation of a Topography Integrated Ground water Retention (TIGER) zone map in accordance with a preferred embodiment of the present invention;

FIG. 3 is a simplified schematic diagram, which illustrates operation of an automated soil type ascertaining process;

FIG. 4 is a simplified schematic diagram, which illustrates operation of an irrigation logic process;

FIG. 5 is a simplified schematic diagram, which illustrates an embodiment of the invention that controls a drip irrigation system; and

FIG. 6 is a simplified schematic diagram, which illustrates ascertaining an Irrigation Water Utilization Metric (IWUM) in accordance with a preferred embodiment of the present invention, which is useful in optimizing water pricing and allocation by a water provider.

FIG. 7, which is an example of the Topography Integrated Ground water Retention (TIGER) zone map 115 of FIG. 1. It is appreciated that the map comprises of three irrigation management zones. These correspond to soil physics and soil moisture data provide hereinabove, with reference to FIG. 2.

FIG. 8, which is an image of graphs of soil drying curves, illustrates results of the automated soil type ascertaining process 270 of FIG. 2. It is appreciated that the graphs depict a collection of soil drying curves; each line correlates to a specific sample (right plate). These samples are successfully trended and grouped into distinct soil class categories.

FIG. 9 is an image of screens of a mobile computing app, constructed and operated in accordance with a preferred embodiment of the present invention. The screen images of the software, demonstrate the full automation of the irrigation planning process. It is appreciated that without full automation, which is provided by the differential irrigator 100 of FIG. 1, such app and screens would not be possible. As an example, many factors, climatic, plant related, time related, and soil related, would need to be displayed to the user. The user would also need to view a much larger and more detailed map of the field 105, in order to consider how to irrigate. In contrast, the app shown provides the user with simplicity of automated use, which is similar to that of a 'television remote control', rather than that of complicated software. It is appreciated that this simplicity cannot be achieved without the automation of differential irrigation that the present invention offers.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

Reference is now made to FIG. 1, which is a simplified schematic diagram providing an overview of the present invention.

Irrigation planning for large fields, the process of deciding how much water to apply onto which part of a large field and when - is known in the art to be a complex process, and one which has never been successfully automated. The hardware required for such irrigation is available, and one example is known as Site-specific Variable Rate Irrigation (SS-VRI or VRI). But an automated process to maximize the value of such variable rate irrigation, or differential irrigation - at present doesn't exist. Much has been studied and known about the various factors affecting irrigation needs. But, the process of analyzing these various factors, for a specific field, crop and climate, and automatically transforming them into an effective automated irrigation plan, remains a process which until the present has defied automation, and requires site specific, manual, ongoing expert analysis.

A recent review Evans et. al, Review: Adoption of site-specific variable rate sprinkler irrigation systems (Irrig. Sci 2013), states, inter alia,: "The development of algorithms, sensor specifications, and placement criteria and decision support systems for SS-VRI is still in their infancy. General, broad-based, intuitive, and easily adjusted software (decision support) for implementation of prescriptions for SS-VRI systems is not available for a multitude of crops, climatic conditions, topography, and soil textures. The complexity in optimizing multi-objective, multivariate (irrigation) prescriptions for dynamically changing management zones will be a substantial challenge for researchers, industry, and growers alike".

In fact, the current process of planning differential irrigation is at present so far from automation and so dependent on skilled manual expertise, that the above review concludes, inter alia, that "specialized, continual training on the hardware, software, and advanced agronomic principles is needed now for growers, consultants, dealers, technicians, and other personnel on how to define management zones (areas), write prescriptions, and develop seasonal crop irrigation management guidelines. This has been slowed because the criteria for training individuals to develop management zones, write appropriate crop-specific prescriptions, and assist with the decision-making processes have yet to be defined."

Current irrigation logic methodology tries to assess as many of the complex factors affecting irrigation, either using sensors to measure them, or models to predict them. These include crop factors (crop type and phase), climate factors (temperature, humidity, wind, etc.) and soil factors (soil type, soil water retention capacity, and soil moisture). The complexity of this information is such, that it cannot be automatically 'resolved' into an irrigation plan. Rather, the 'raw' information is then presented to the farmer who would consult it, and then manually decides how to irrigate.

This challenge is much greater in large fields. Irrigation-logic needs of small domestic gardens or vegetable patches may be adequately addressed by relatively simple soil-moisture sensors 'closed-loop' systems. Such systems simply use a soil moisture sensor and irrigate to replenish a desired soil-moisture threshold. But extending them to large fields would require dozens of soil-sensors under a single irrigator, often hundreds across a farm, which would be both cost prohibitive as well as would interfere with field cultivation, such as plowing.

The present inventors have realized that would be very useful if there was an accurate map charting the 'water holding' properties of a field (for example, clay retains more water than sand). If such a map existed, it would be possible to divide the field into effective irrigation zones,

and monitor soil moisture in each of these zones, knowing that the same soil moisture is expected to be found everywhere within this zone. Irrigation could then be guided accordingly.

The accepted way of attempting to create such irrigation management zones, relies on Electro-Conductivity (EC) mapping, also referred to as Electro-Magnetic (EM) mapping, a procedure which measures the conductivity of soil and thereby gives an indication of its water content, and which is further described herein below.

The inventors earlier tried to develop such a reliable 'water holding' map of a field based on EC mapping, in order to guide irrigation - and they have failed. In their study (Hedley, AGWAT 2009) they created and tested the effectiveness of irrigation zones based directly on Electro-Conductivity mapping of a field, using the accepted methodologies for EC mapping and data analysis. They then installed 50 soil moisture sensors, 50 meters apart, in a grid across the 32 hectare field studied, expecting to prove that there is little variance between the soil moisture readings within each of three EC-based soil-zones. This would indicate that the zoning is effective, and mean that it is then possible to use a single sensor in a zone, and expect its measurements to reflect the soil moisture across the entire zone.

Unfortunately, the results indicated that in fact there was a significant variance between sensor readings within each of the EC-based zones, and little to no difference between the zones (mean and standard deviation (SD) were identical in two EC-based irrigation zones, and less than 1 SD different from the third zone, with % coefficient of variation (%CV) in all three zones ranging between 9% and 14%). This observation is further validated by the fact that there was little variance between multiple readings the same sensor over time, indicating that the sensors themselves are reliable.

The present invention proposes a different method of producing a novel, reliable water retention potential map, referred to here as a Topography Integrated Ground watEr Retention (TIGER) map, and dividing it into effective irrigation management zones that accurately reflect water retention properties. This method is based on a novel computerized method of analysis and integration, which analyzes topographical terrain attributes, and integrates them with an analysis of EC mapping data. The Topography Integrated Ground watEr Retention (TIGER) zone map of the present invention for the first time, allows automation of the differential irrigation planning process, as illustrated in FIG. 1.

In accordance with a preferred embodiment of the present invention, a differential irrigator 100, which preferably is embodied in an automated irrigation decision support software module running on a general purpose computer, or on a mobile computing and or communication device in conjunction with an internet-based computing server, is used to enable efficient irrigation of a field 105, by differentially irrigating different parts of the field 105. It is typically the case that the soil composition and the topography of agricultural fields are not homogeneous, and hence different parts of the field often require different amounts of irrigation.

In accordance with a preferred embodiment of the present invention, the differential irrigator 100 preferably initially performs a one-time initial assessment 110 of the field 105, based at least in part on Electro-Conductivity Mapping Data, designated EC data 112 and topographical Digital Elevation Mapping Data, designated DEM data 114, both of the field 105. EC data is preferably obtained from EM mapping. EM mapping measures the apparent electrical conductivity of soil through the use of electromagnetic sensors that are towed on the surface soil of a field, typically by a quad bike, which is fitted with RTK GPS. The EM sensor uses a transmitting coil that induces a magnetic field that varies in strength according to soil depth.

A receiving coil reads primary and secondary induced currents in the soil. It is the relationship between these primary and secondary currents that measures soil conductivity. EM mapping may be performed using commercially available EM mapping hardware, such as Geomatrix' EM31 and EM38, data is processed into an EC map using publicly available software. It may also be obtained from service providers that provide both EM sensing service in the field, as well as processing the obtained data into an EC map. A recent report summarizes the current practices, and illustrates examples of suitable equipment, and service providers ('Standards for Electromagnetic Induction mapping in the grains industry', GRDC Precision Agriculture Manual, Australia 2006).

DEM data 114 may also be obtained from EM mapping output, since DEM data is typically collected as part of the EM survey, since EM survey is typically performed using a RTK GPS, which logs DEM data 115. It is important to note that DEM data 114 is unrelated to EC data, and is typically discarded in the prior art. Alternatively, DEM data 114 may be obtained from other sources of DEM data 114, including databases of DEM data 114, instruments that record DEM data 114 and services of DEM data 114 mapping. EC data 112 and DEM data 114 and the modes for obtaining them are further described herein below with reference to FIG. 2.

The initial assessment 110 generates a Topography Integrated Ground water Retention (TIGER) zone map 115, which preferably provides for each location in the field 105, a soil wetness potential score, reflecting relative 'potential for retaining water' of this location in the field 105, relative to all other locations therein. This soil wetness potential score is based on an analysis of EC data 112 and DEM data 114, and reflects a calculation of an integrated effect of physical soil properties, reflected in the EC data 112, and of topographical terrain attributes, which are calculated based an analysis of the DEM data 114), both of the field 105.

The Topography Integrated Ground water Retention (TIGER) zone map 115 preferably also divides the field 105 into several irrigation zones according to their soil wetness potential score. In a preferred embodiment of the present invention, the several irrigation zones, typically three irrigation zones, zone-1 120, zone-2 125 and zone-3 130. Each one of these irrigation zones preferably has soil-physics properties and topographical terrain attributes that indicate that it would retain water differently and hence require different amount and timings of irrigation from each one of the other irrigation zones.

The Topography Integrated Ground water Retention (TIGER) zone map 115 is preferably also used to define one or more suitable locations for placing one or more soil sensors within each of zone-1 120, zone-2 125 and zone-3 130. In a preferred embodiment of the present invention, sensor-1 140 is a sensor node, located within zone-1 120, sensor-2 145 is a sensor node located within zone-2 125, and sensor-3 150 is a sensor node located within zone-3 130.

In a preferred embodiment of the present invention, a location determined by the Topography Integrated Ground water Retention (TIGER) zone map 115 for sensor-1 140 is such that based at least in part on measurements of sensor-1 140, the differential irrigator 100 can effectively predict an irrigation condition of the entire zone-1 120. The same is true for sensor-2 145 and sensor-3 150 and their corresponding zone-2 125 and zone-3 130. Each of sensor-1 140, sensor-2 145 and sensor-3 150 - is a sensor node that preferably comprises one or more sensors. In a preferred embodiment of the present invention, each sensor node may comprise two soil moisture sensors, installed at two different soil depths, depending on crop type. In a preferred embodiment of the present invention, each node also comprises a temperature sensor. The initial assessment 110 and the Topography Integrated Ground water Retention (TIGER) zone map 115 are further described herein below with reference to FIG. 2.

Sensor-1 140, sensor-2 145 and sensor-3 150 are preferably connected, preferably wirelessly, preferably via a gateway 155 to the differential irrigator 100. In a preferred embodiment of the present invention, other sensors, including but not limited to sensors operative to detect rainfall, climatic conditions, and plant parameters, may also be utilized and similarly connected to the differential irrigator 100; these are not required for operation of the present invention, but may be useful in improving its performance.

Once the installation described hereinabove is complete, the differential irrigator 100 preferably enables effective irrigation of the field 105, through the following iterative process.

A step designated SENSE 165, receives measurements from each of sensor-1 140, sensor-2 145 and sensor-3 150. These measurements preferably represent a soil moisture and an irrigation condition of zone-1 120, zone-2 125 and zone-3 130 respectively.

Next, a step designated ASSESS 170 assesses the measurements received from each of the sensor-1 140, sensor-2 145 and sensor-3 150. Based at least in part on these measurements, assess 170 determines an amount of irrigation appropriate for each of zone-1 120, zone-2 125 and zone-3 130, which amounts of irrigation may preferably be different from one another. Preferred operation of ASSESS 170 is further described hereinbelow with reference to FIG. 4.

Finally, a step designated IRRIGATE 175, preferably communicates a daily irrigation map 180 to an irrigator controller 185, which controls an irrigator 190. The irrigator 190 may preferably be a mechanized irrigation device, such as a pivot irrigator, a lateral move irrigator, or other. The irrigator 190 then irrigates the field 105 accordingly. Preferred operation of IRRIGATE 175 is further described hereinbelow with reference to FIG. 4.

In a preferred embodiment of the present invention, this iterative process of SENSE 165, ASSESS 170 and IRRIGATE 175, may be performed at scheduled intervals, such as daily. In other preferred embodiments of the present invention, it may take place following each irrigation event, or prior to each planned irrigation event, or upon demand of a user of the system.

Reference is now made to FIG. 2, which is a simplified schematic diagram illustrating the rationale and operation of the initial assessment 110 of FIG. 1, a process which is central to the present invention.

Reference numeral 200 designates a schematic image depicting a field to be irrigated which is non-flat topologically. Judging by its external appearance, it appears quite 'normal'. Its vegetation appears quite uniform. It does not seem to be different from other fields, which have a similar external appearance. Current irrigation systems would irrigate a field like this uniformly, or at best - would base irrigation exclusively on EC data 112. The present invention takes a different approach, through an appreciation that EC data 112 is not the only factor affecting the wetness of the ground and takes into account topographic terrain attributes, which significantly influence soil water retention and hence irrigation. Harnessing an analysis of these various features produces the Topography Integrated Ground watEr Retention (TIGER) zone map 115, which enables automation of differential irrigation planning. These topographic terrain attributes and the method by which they are analyzed and integrated with the EC data are further described herein below.

Reference numeral 205 designates a schematic image depicting an EC map of the field of schematic image 200, showing EC-based irrigation management zones. While the field of 200 seems 'normal', underlying it is the EC data, which indicates different soil zones.

Reference numeral 210 designates a schematic image depicting catchment area mapping of the field of image 200. A catchment area is an area that is topographically lower than its surroundings, the soil of which tends to be more 'soggy'.

Reference numeral 215 designates a schematic image depicting 'aspect mapping' of the field of image 200: Aspect mapping indicates the extent of exposure to the sun and utilizes the fact that areas that are facing the sun, receive more solar radiation and hence dry up more rapidly than those that don't.

Reference numeral 220 designates a schematic image depicting 'slope mapping' of the field of image 200 and utilizes the fact that areas that have a steeper slope retain water differently than ones of moderate slopes. It is appreciated from schematic images 205-220 that there are multiple factors affecting the water-retention properties of the field of 200.

Reference numeral 225 designates a schematic image depicting the superimposition of the four above mentioned datasets: EC mapping 205, catchment mapping 210, aspect mapping 215 and slope mapping 220. In accordance with a preferred embodiment of the present invention at least one and preferably all of the aforesaid mappings are integrated into a single coherent map, the Topography Integrated Ground water Retention (TIGER) map.

As noted above, reference numeral 205 depicts an Electro Conductivity (EC) map of the same field, divided into three irrigation zones, based on the EC data. EC data may be derived from Electro-Magnetic (EM) mapping. EM mapping is acquired using EM sensors, such as Geonics EM38Mk2 and EM31 sensors, which are preferably combined with RTK- DGPS and dataloggers mounted on an all-terrain vehicle to acquire high resolution EM38 and EM31 vertical mode datasets in two separate surveys. A Trimble Ag170 field computer may be used for simultaneous acquisition of high resolution positional and ECa data.

The sensors preferably measure a weighted mean average value for apparent electrical conductivity (EC) to 1.5 m depth (EM38) and 5.0 m depth (EM31). Survey data points are preferably collected at 1-s intervals, at an average speed of 15 kph, with a measurement recorded approximately every 4 m along transects 10 m apart. Filtered data comprising latitude, longitude, height above mean sea level and ECa (mSm^{-1}) may preferably be imported into ArcGIS (Environmental Systems Research Institute, (ESRI© 1999). Points are preferably kriged in Geostatistical Analyst (ESRI© 1999) using a spherical semivariogram and ordinary kriging to produce a soil ECa prediction surface map. Three management zones may preferably be defined on this map (using Jenks natural breaks) for further soil sampling. EM surveys quantify soil variability largely on a basis of soil texture and moisture in non-saline conditions.

A process designated compute and map catchment area 230 computes a catchment layer 210, which is a spatial representation of the Catchment Area value of every point in the field 105. A catchment area is defined as the $\ln(a/\tan\beta)$ where a is the local upslope area draining through a certain point per unit contour length and $\tan\beta$ is the local slope. A location has a high catchment area value when it is topographically depressed relative to its surrounding area. Accordingly, a soil in a location which has a high catchment area value tends to retain more water and be 'more soggy'. As an example, water would more likely accumulate at the bottom of a valley than at the top of a hill. There are various methods to compute catchment.

In a preferred embodiment of the present invention the surface and subsurface runoff is parameterized by catchment area estimations. The catchment area (CA), defined as the discharge contributing upslope area of each grid cell and the specific catchment area, defined as the corresponding drainage area per unit contour width are computed using the multiple flow direction method of FREEMAN (1991).

In another preferred embodiment the SAGA Wetness Index is used in conjunction with the Topographic Wetness Index (TWI). SWI is similar to TWI but it is based on a modified catchment area calculation (out.mod.carea), which does not treat the flow as a thin film as done in the calculation of catchment areas in conventional algorithms. As a result, the SWI tends to assign a more realistic, higher potential soil wetness than the TWI to grid cells situated in valley floors with a small vertical distance to a channel. A computer code is then preferably used to integrate the different predictors, remove sinks, and correct for overlapping results. The computer code performing the calculation of catchment area, in a way that has been found effective in predicting irrigation management zones and is enclosed as computer code listing.

A process designated compute and map aspect 235 computes the aspect layer 215, which is a spatial representation of a set of 'aspect' values of every point in the field 105. By aspect, is meant in which direction the land is facing. As an example, land facing the sun, will dry faster and hence require more water than land facing away from the sun. A process designated compute and map slope 240 computes the aspect layer 220, which is a spatial representation of the slope in value in degrees of every point in the field 105. As an example, steeper sloped land will require a different amount of water than flatter land. Computer code performing the calculation of slope and of aspect, in a way that has been found effective in predicting irrigation management zones and is enclosed as computer code listing.

Having calculated the above mentioned four datasets, conductivity score map 205, catchment score map 210, aspect score map 215 and slope score map 220, the next step is create the Topography Integrated Ground water Retention (TIGER) map. It is appreciated that each one of these maps on its own is not useful for guiding irrigation. It is further appreciated, as images 250 and 255 illustrate, that simply overlying these maps one on top of the other, is similarly not useful. The following algorithm and methodology is preferably used in order to carefully analyze each data point in each of these datasets, integrating them to generate an integrated wetness potential map 115.

It is appreciated that each of the above datasets 205-220 is a map of the field 105, wherein each location in this map of the field 105 is associated with a value. As an example, the catchment score map 210 comprises a catchment score for each point in the map. Same is true for the EC value map, aspect score value map and slope value map. To integrate these scores, a large set of vectors is created, corresponding to all locations in the field 105 which are investigated, for example all locations for which EC data 112 and DEM data 114 has been obtained. This set of vectors is designated vector pool. Each vector preferably comprises eight attributes: a location property (its location within the field 105, preferably an x location and a y location, and a set of six measured or calculated attributes, relating to the above mentioned four data sets: superficial EC score, deep EC score, catchment score, aspect score, slope score, and elevation (as per DEM data 114 for that location). Importantly, elevation is not associated with soil wetness, but has been found to be an important attribute, useful in creating the integrated wetness potential map 115, as described herein below.

A number of vectors are randomly selected. Each of these serves as a nuclei of an integrated wetness potential score zone. In a preferred embodiment of the present invention, the number of initial tentative nuclei is preferably 100, providing a detailed map of the integrated wetness potential scores in the field 105. In another preferred embodiment, the number of initial nuclei is preferably a much smaller number: a desired number of irrigation zones, typically 3 or 4. In yet another preferred embodiment, the number may be double the number of the desired irrigation zones, so as to have within each irrigation zone an 'inner zone', in which the sensors are

to be placed, so that sensors are placed in a location which best represents the irrigation zone they are in.

Each vector in the vector pool is assessed for its distance to the each of the nuclei, and added to the closest nuclei. By distance is meant an integrated distance, that is a distance which takes into account the distance of each attribute of the vector to that attribute in each of the nuclei. In a preferred embodiment of the present invention, this distance may preferably be calculated as a squared error function.

When all vectors in the pool have been thus assigned to nuclei, the barycenter of each nucleus is calculated, and the process of assessing each vector in the vector pool to a nucleus and assigning it to the nearest nucleus is repeated. With each iteration, the centre of the nuclei of each further optimized. This process is repeated until the location of the centre of the nuclei does not move between iteration. In a preferred embodiment of the present invention, the process is preferably repeated 1000 iterations.

In a preferred embodiment of the present invention, a function describing the calculation performed in evaluating the integrated affect of each location in each of the conductivity score map 205, catchment score map 210, aspect score map 215 and slope score map 220 - on each corresponding location the integrated wetness potential map 115 - may be described calculated as follows:

$$J = \sum_{j=1}^K \sum_{i=1}^N \|x_i^{(j)} - c_j\|^2$$

where K is the number of zones, N is number of vectors (i.e. locations evaluated in the field 105), X is an attribute, and i is the type of attribute.

It is appreciated that topographical terrain attributes other than the ones listed above may be used to calculate the integrated wetness potential map 115, and that the above mentioned ones are provided as an example only and are not meant to be limiting. It is further appreciated that the above description of methodology of integrating topographical terrain attributes and EC data may be performed using other methodologies, and that the above methodology is provided as an example only and is not meant to be limiting.

The Topography Integrated Ground water Retention (TIGER) zone map 115, and the irrigations zones therein, may preferably be represented in suitable formats, including but not limited to polygons and shape-files. Conversion into such formats is well known in the art, for example using a 'Raster-to-Polygons' and 'Polygon-to-Shapefile' in 'R' Programming language (www.r-project.org). Such formats are useful for comparing the irrigation zones to other data and for communicating with irrigation system controllers and other agricultural systems.

According to a preferred embodiment of the present invention, if more than one crop is grown in the field 105 under the same irrigator 190, than the irrigation zones may preferably divided into soil-crop zones, such that there is only one crop per irrigation zone. As an example, if there are two crops, wheat and corn, grown within single soil-topography irrigation zone 'A', then this zone 'A' would preferably be divided into zone 'A-Wheat' and zone 'A-Corn'. This, since the water uptake and hence irrigation balance of these two crops may be different, and hence would require separate sensors monitoring them, and separate irrigation planning logic.

Lastly, for each of the irrigation zones determined in the Topography Integrated Ground water Retention (TIGER) zone map 115, a soil type is determined, by a process designated

an automated soil type ascertaining process 270, which is further described herein below, with reference to FIG. 3.

Accuracy of the the initial assessment 110 and Topography Integrated Ground water Retention (TIGER) zone map 115 both of FIG. 1 was validated in the field as follows. Three replicate soil samples (at three depth intervals) were randomly collected from each of the three classes identified from the Topography Integrated Ground water Retention (TIGER) zone map 115, avoiding spray truck and irrigator tracks. The soil samples were intact soil cores (100 mm diameter and 80 mm in height) taken from the middle of three sample depths (0-200mm, 200-400mm, 400-600mm) for laboratory characterisation of bulk density and soil moisture release characteristics (at 10 kPa); and smaller cores (50 mm diameter and 20 mm in height) were taken for soil moisture release at 100 kPa. A bag of loose soil was also collected (0-200 mm, 200-400 mm, 400-600 mm soil depth) for laboratory estimation of permanent wilting point (1500 kPa) (Burt, 2004) and particle size distribution. Total available water holding capacity (AWC) was estimated as the difference between volumetric soil moisture content (mcv) at 10kPa and 1500kPa, where 10kPa is taken as field capacity and 1500kPa is wilting point. Readily available water holding capacity (RAWC) was estimated as the difference between mcv at 10kPa and at 100kPa. Percent sand, silt and clay was determined on these soil samples by organic matter removal, clay dispersion and wet sieving the >2-mm soil fraction and then by a standard pipette method for the <2-mm soil fraction (Claydon, 1989).

Table 1 summarizes some significant measured differences between the soil hydraulic characteristics of the three classes identified from the Topography Integrated Ground water Retention (TIGER) zone map 115 of FIG. 1. These measured differences reflect differences in pore size distribution and justify the efficacy of the Topography Integrated Ground water Retention (TIGER) zone map 115, as the basis for management of irrigation. An increasing Available Water Capacity (AWC) with class number reflects an increasing proportion of pores in the range where plant-available water is stored, in particular readily available water which is stored between 10kPa and 100kPa (pore size diameters 0.03 – 0.003 mm).

Table 1

Soil texture and hydraulic characteristics ($\pm$ standard deviation) of soils in the three management classes

Class	Soil moisture release at					Sand	Clay
	10 kPa	100 kPa	1500 kPa	RAWC*	AWC*		
	m^3m^{-3}	m^3m^{-3}	m^3m^{-3}	m^3m^{-3}	m^3m^{-3}	%	%
1	0.11 $\pm$ 0.02	0.06 $\pm$ 0.01	0.03 $\pm$ 0.00	0.05 $\pm$ 0.02	0.08 $\pm$ 0.02	96	2
2	0.14 $\pm$ 0.04	0.09 $\pm$ 0.01	0.03 $\pm$ 0.01	0.05 $\pm$ 0.04	0.11 $\pm$ 0.04	95	2
3	0.24 $\pm$ 0.02	0.13 $\pm$ 0.02	0.04 $\pm$ 0.00	0.11 $\pm$ 0.03	0.20 $\pm$ 0.02	90	4

*RAWC=readily available water-holding capacity; AWC=available water-holding capacity.

The soil moisture sensors used also tracked large differences in soil moisture between soil classes within this study area (Fig. 2), reflecting their contrasting soil moisture release characteristics, and the varying influence of a high water table, especially noticeable in Class 3 soils. Prior to commencement of irrigation in late spring 2010, the soil moisture sensors simultaneously monitored $0.11 \pm 0.06 \text{ m}^3 \text{ m}^{-3}$ in the dry classes (lowest EC values) compared with $0.17 \pm 0.26 \text{ m}^3 \text{ m}^{-3}$ (intermediate EC classes) and $0.27 \pm 0.64 \text{ m}^3 \text{ m}^{-3}$ in the wettest classes (highest EC values). The dry classes (Class 1 in Fig. 1) hold less available water and require irrigation sooner than Class 3.

For the period: February–March 2011, the depth to water table varied at any one time by about 70 cm (Fig. 2). A 66 mm rainfall event between 4th and 6th March caused the water table to rise by about 50 cm in Class 1 and 70 cm in Class 3 (Fig. 2). This difference is due to different storage capacities of the soils and landscape position. Class 3 soils occupy low-lying areas where water tends to accumulate by overland runoff and lateral flow, and the water table is closest to the surface. These soils, typically being wetter, require less rainfall to bring them to saturation; and once saturated the water table rises to the surface, at a faster rate than in soils starting at a drier soil moisture content.

Continuous soil moisture sensor recordings, at 15 minute intervals during an entire irrigation season, from a network of 9 sensors, placed in the different irrigation zones defined by the Topography Integrated Ground waterEr Retention (TIGER) zone map 115, provided an unprecedented high resolution temporal dataset, confirming the efficacy of the Topography Integrated Ground waterEr Retention (TIGER) zone map 115 and providing important input for its fine-tuning.

In another preferred embodiment of the present invention, predictive modelling of an underground water table may be useful, preferably using a random forest regression trees data mining algorithm (RF, Breiman, 2001). This approach and experiments validating its are useful is described as follows. The use of EM38, EM31, digital elevation and rainfall data were investigated for incorporating into the predictive models. Rainfall data was obtained from the closest weather station (six kilometres away), and rainfall was assumed constant over the study area at any one time. TWI and SWI were extracted from the digital elevation map (see 2.3). The data was fused by projecting it onto a common grid, and modelling the co-variables in space. Two predictive modelling approaches were developed and compared to explain observed patterns of water table depth and soil moisture status, i.e. a simple approach using multiple linear regression (MLM), and a data-mining approach using random forest regression trees (RF, Breiman, 2001).

Three predictors have been selected to dynamically model soil moisture content and water table depth: EM38, SWI and rainfall. EM38 and SWI data have been log-transformed to overcome skewness, as modelling approaches assume normal distribution. The rainfall data have been integrated over three days to account for the time required for the rain event to fully affect water table depth. These variables were selected as the best predictors, and other attributes, including elevation, EM31 and TWI, although tested, did not improve model predictions, and were therefore not included, with our objective being to develop the best parsimonious prediction model.

Reference is now made to FIG. 3, which is a simplified schematic diagram illustrating operation of automated soil type ascertaining process 270.

As is known in the art, different types of soils have different water release properties. For example, clay retains water well, whereas sand does not. These soil water release properties are typically studied in the lab, for example by taking intact soil core samples, drying them under lab conditions, and recording the release of water from the soil over time, also known as a soil-drying curve. Such curves are useful in guiding irrigation. Of special importance are three points that are on the curve and are derived from it. Field Capacity is the maximal amount of water which

the soil can retain without runoff. At Wilting Point plants will wilt. And Refill Point, which is calculated based on these two, represents the level of water in the soil, below which irrigation is needed.

Refill Point and Field Capacity are useful in controlling irrigation; since a goal of efficient irrigation is preferably to maintain a soil moisture level that is in the range between these two. A severe limitation of existing irrigation solutions is that these values can currently only be obtained through a manual scientific laboratory process, which is therefore expensive. Importantly, it also prevents automation of the irrigation planning process.

The automated soil type ascertaining process 270 is a novel automated process to determine the soil type of irrigation zones in the field 105, without requiring a manual laboratory process. This process is preferably an automated process which trains a classifier 300, using a set of known field soil-drying curves 305 and preferably a set of known lab soil-drying curves 305. Once trained, the classifier 300 is operative to analyze an unknown Field soil-drying curve and determine its soil-class properties 320, or its site specific soil properties 325, as further explained herein below.

The classifier 300 is preferably embodied in machine learning computer software. In a preferred embodiment of the present invention the classifier 300 may preferably be a Decision Tree algorithm. It is appreciated however that there are many powerful, easily applicable machine learning methodologies, algorithms and tools known in the art, and the following embodiment described is provided as an example only and is not meant to be limiting.

Each one of the known Field soil-drying curves 305, is a set of soil-moisture measurements along a time axis, made in the field, by a soil-moisture sensor, in a soil type. These measurements may be plotted as a soil drying curve. The set of known Field soil-drying curves 305 comprises of a plurality of such soil drying curves, from each of a plurality of locations and soil types.

Similarly, each one of the known lab soil-drying curves 305, is a set of soil-moisture measurements along a time axis, but ones which were made in the laboratory, where the water content in the soil is accurately measured by weighing the soil sample as it is being dried in an oven. The set of known lab soil-drying curves 305 comprises of a plurality of such sets of moisture measurements, or soil drying curves, taken from each of a plurality of locations and soil types. Preferably, at least part of the known Field soil-drying curves 305 and the known lab soil-drying curves 310 are taken from an identical location and soil type.

In a preferred embodiment of the present invention, a linear modeling process 330 fits the known Field soil-drying curves 305 and the known lab soil-drying curves 310 to corresponding plurality of line graphs 335. For each of the line graphs 335, an extract LINEAR parameters 340 process is performed, which derives parameters 345, preferably an Intercept and a Slope of each of the line graphs 335. The parameters 345 are a convenient abstraction of each of the known Field soil-drying curves 305 and the known lab soil-drying curves 310. It is appreciated that the classifier 300 may be trained on curves directly using various methodologies well known in the art, and may also be trained on abstractions or models other than the linear modeling process 330, which is provided as an example only.

In a preferred embodiment of the present invention, a divide into training sets 350 process, divides the parameters 345 derived from the known Field soil-drying curves 305 into two datasets: a soil-drying calibration set 355 and a soil-drying validation set 360. In another preferred

embodiment of the present invention, the parameters 345 derived from the known lab soil-drying curves 310 are similarly divided into these two datasets.

The train classifier 365 process uses the soil drying calibration set 355 and the soil drying validation set 360, to train the classifier 300. The classifier 300 is trained to identify patterns which appear in the soil-drying calibration set 355, and then tests its success in identifying these patterns, on the soil drying validation set 360. In a preferred embodiment, the soil drying calibration set 355 and the soil drying validation set 360 may preferably be grouped by their soil type, and or by other criteria, and the classifier 300 may be trained to identify a drying curve, or its abstraction, which typifies this drying curve in the soil type.

Various methodologies are known in the art to train machine learning classifiers and other comparable software algorithms. These include, but are not limited to: an iterative process of training and validation, processes in which the training and validation sets are dynamically changed and overlap, and other methodologies. It is appreciated therefore that the description herein of the training of the classifier 300 are simplified and provided as an example only and are not meant to be limiting.

Once trained, the classifier 300 is operative to analyze an unknown Field soil-drying curve 315 and based on this analysis to determine a soil type 370 to which the unknown Field soil-drying curve 315 corresponds. By soil-class, is meant soil type of a 'class' of soils, such as 'clay', 'sand', 'sandy-loam' etc. It is understood, that as an example, soil in two different farms may be classified as 'sandy loam' in both, although there may be a difference between the 'sandy loam' of one, compared to the other.

In various preferred embodiments of the present invention a list of 8-12 of following soil types, is preferably used, and their Field Capacity and Wilting Point values may preferably be used (v%):

Texture	Capacity	Wilting
Sand	10	5
Loamv sand	12	5
Sandv loam	18	8
Sandv clay loam	27	17
Loam	28	14
Sandv clay	36	25
Silt loam	31	11
Silt	30	6
Clay loam	36	22
Siltv clay loam	38	22
Siltv clay	41	27

Clav	42	30
------	----	----

In another preferred embodiment of the present invention, the classifier 300 determines SITE-SPECIFIC soil properties 325 of the unknown Field soil-drying curve 315. As mentioned above, grouping soils into 'classes' such as 'Clay loam' etc., is a generalization, whereas in fact the soil in each site has its own specific water retention properties. These are referred to here as SITE-SPECIFIC soil properties 325.

As is known in the art, the accuracy, sensitivity and specificity of a machine learning classifier depends on the size and quality of the training and validation sets and on the quality of the unknown sample to be analysed. The accuracy of the classifier 300 increases over time, as it continues to be trained by the train classifier 365. Its increasing accuracy over time is further facilitated by two factors. First, the known Field soil-drying curves 305 is constantly growing, as more users use the system. This, since the system continuously streams all readings from all sensors of all users to its central data repository, and thus accumulates a growing number of soil-drying curves, obtained from various soil types. Second, over time, the readings from a specific irrigation zone in a specific farm also accumulate. Over time, therefore, the unknown Field soil-drying curve 315, rather than being a single curve, may preferably be a plurality of soil-drying curves obtained from the same location. Providing as input such a plurality of 'natural variants' of the sample to be identified greatly increases the accuracy of a classifier, as is well known in the art.

According to another preferred embodiment of the present invention, the soil type 370 may be obtained by the farmer-user manually selecting a type of soil, as designated by manually select 375. The differential irrigator 100 may preferably be implemented as a computer-web application or more preferably as mobile-web application, wherein clear guidelines describe the differences between preferably 8-12 types of soil. Preferably, short videos and photographs guide the farmer in selecting the correct type of soil-class.

Reference is now made to FIG. 4, which is a simplified schematic diagram illustrating operation of ASSESS 170 and IRRIGATE 175, both of FIG. 1.

A compute irrigation process 400 preferably receives as input, sensor data 405, soil properties 410 and irrigation goal 415. The sensor data 405 comprises readings received from soil moisture and other sensors, such as sensor-1 140, sensor-2 145 and sensor-3 150 all of FIG. 1. The soil properties 410, comprises soil-class properties 320 and site-specific soil properties 325 both of FIG. 3, including field capacity and refill point properties. The irrigation goals 415 preferably comprises user defined guidelines, indicating up to which soil moisture level the user would like to irrigate, preferably relative to the field capacity and refill point values of the soil of the zone in which the sensor is located. In a preferred embodiment of the present invention, the user may provide as one of the irrigation goals 415, a percentage number, relating to the range between refill point and field capacity. Irrigation goals 415 may comprise global irrigation goals and crop specific irrigation goals.

The compute irrigation 400 compares each sensor reading received, with the soil properties of the soil of the irrigation zone, and the irrigation goal defined by the user, and calculates accordingly the recommended irrigation for that zone.

Next step, present to user via app 420, preferably presents a tentative irrigation map, for each of the zones of the field 105 of FIG. 1, preferably via an app on a mobile device, or a computer, or a web browsing device.

A step designated user modifies and confirms 425 allows the user to review the irrigation recommendation, and very simply modify it. In a preferred embodiment of the present invention, this modification may be performed via the mobile app, preferably using under 4 or less clicks and or gestures, in most cases. FIG. 9 presents several screen layouts of an app constructed and operative in accordance with a preferred embodiment of the present invention, illustrating the total automation, and simplicity and ease of use, with which steps present to user via app 420 and user modifies and confirms 425, are preformed.

Format and send to irrigator 430 illustrates operation of IRRIGATE 175 of FIG. 1. This process formats the irrigation map approved by the user in the previous step, in to a formatted irrigation plan 435, such that it is suitable for the irrigator controller 185 and the irrigator to the irrigator 190. It is appreciated that there are different types, brands and providers of mechanical irrigators, such as pivot irrigators and lateral move irrigators. As an example, the format and send irrigator 430 may format formatted irrigation map 435 as a 'full-VRI' map (that is, where every point in the field may receive a different amount of irrigation), or to pivot speed or section control irrigator (that is, where different sectors of a circular field, receive different amounts of irrigation), for section or speed control of lateral move irrigator (that is, where different cross-sections of a rectangular field receive different amounts of irrigation). In another preferred embodiment of the present invention, the format and sent to irrigator 430 may provide an amount to irrigate, to be applied uniformly onto a field, such that the irrigation is optimized based on the assessment of the irrigation needs of each part of the field, and preferably one or more user preferences. This step also formats the irrigation map to the technical format, suitable for a specific vendor of an irrigator 190 or irrigator controller 185.

Reference is now made to FIG. 5, which is a simplified schematic diagram illustrating embodiment of the invention that guides a drip irrigation system.

In accordance with another preferred embodiment of the present invention, the differential irrigator 100 of FIG. 1 may automatically control differential irrigation of the field 105, through use of a drip irrigation system.

In this embodiment, the Topography Integrated Ground water Retention (TIGER) zone map 115 preferably also defines a pattern for laying drip irrigation pipes, such that a separate drip irrigation pipe is placed in each of the irrigation zones, zone-1 120, zone-2 125 and zone-3 130. This pattern for laying drip irrigation pipes allows a farmer to LAY DRIP PIPES 118 accordingly: a pipe designated zone-1-PIPE 131 in zone-1 120, a pipe designated zone-2-PIPE 132 in zone-2 125, and a pipe designated zone-3-PIPE 133 in zone-3 130.

Each of the three pipes preferably connect to a corresponding tap: zone-1-PIPE 131 connects to TAP-1 134, zone-2-PIPE 132 connects to TAP-2 135, zone-3-PIPE 133 connects to TAP-3 136.

In a preferred embodiment of the present invention, TAP-1 134, TAP-2 135 and TAP-3 136 are remotely operated taps, preferably controlled by the irrigator controller 185.

Similar to the process described hereinabove with reference to FIG. 1, the differential irrigator 100 operates in an automated iterative manner: sense 165 receives measurements from each of sensor-1 140, sensor-2 145 and sensor-3 150. assess 170 assesses these measurements and determines an amount of irrigation appropriate for each of zone-1 120, zone-2 125 and zone-3 130, which amounts of irrigation may preferably be different from one another. Lastly, irrigate 175, preferably communicates the daily irrigation map 180 of FIG. 1 to the irrigator controller 185, which in turn controls TAP-1 134, TAP-2 135 and TAP-3 136, thereby delivering suitable irrigation amounts to each of zone-1 120, zone-2 125 and zone-3 130.

As mentioned hereinabove with reference to FIG. 1, in a preferred embodiment of the present invention, this iterative process of sense 165, assess 170 and irrigate 175, may be performed on scheduled intervals, such as daily. In other preferred embodiments of the present invention, it may take place following each irrigation event, or prior to each planned irrigation event, or upon demand of a user of the system.

Reference is now made to FIG. 6, which illustrates ascertaining an Irrigation Water Utilization Metric (IWUM) in accordance with a preferred embodiment of the present invention, which is useful in optimizing water pricing and allocation by a water provider.

Uniform irrigation, which is the current norm, is often wasteful, since different parts of a field often have different irrigation needs. The damages from this are waste of water, reduced crop due to overwatering, and damage to ground water reservoirs through chemical leaching and waste overflow. Water owners and governments bear much of this consequence, since water provided to agriculture is often heavily subsidized or discounted. Governments and state agencies further suffer from this, by means of damage to the state's natural resources.

It would be advantageous for water owners, governments and state agencies, to have tools which allow monitoring of the efficiency with which water is used for irrigation. An important aspect of this would be a tool which monitors and grades the differential irrigation efficiency, that is to what extent irrigation of a field is optimized for the different needs of different parts of a field. Currently such tool does not exist. The present invention provides such a tool, which is described herein below.

The present invention provides a Irrigation Water Utilization Metric (IWUM) 600, which empowers a water owner 605 to affect a water pricing and allocation 610 of water 615 that the water owner 605 provides to each of a plurality of farms 620.

Each of the plurality of farms 620 may comprise a plurality of Topographic Integrated Ground water Retention zones, designated TIGER zones 625, which are derived from the Topographic Integrated Ground water Retention zone map designated Topography Integrated Ground water Retention (TIGER) zone map 115 of FIG. 1. The differential irrigator 100 of FIG. 1 is operative to analyze and determine an amount of irrigation each of the TIGER zones 625, needs at any time, if suitable sensors are installed in each of these zones.

According to a preferred embodiment of the present invention, one or more sensor 630 is preferably installed in each of the TIGER zones 625. The sensor is preferably a soil moisture sensor node, similar to sensor-1 140, sensor-2 145 and sensor-3 150 of FIG. 1, and preferably comprises two soil moisture sensors installed at two soil depths.

Using mechanisms described hereinabove with reference to FIGS. 1-4, a calculate responsive differential irrigation amount 635, may calculate a responsive irrigation amount 640 based on input from one or more sensor 630, from each of the plurality of sensor-zones 625, for any one of the farms 620. By comparing the responsive irrigation amount 640 (that is: calculating how much water would have been irrigated, if this farm would have irrigated differentially and effectively) to an actual irrigation amount 645 (that is the amount of water that this farm actually

used) - the Irrigation Water Utilization Metric (IWUM) 600 is calculated. As an example, the Irrigation Water Utilization Metric (IWUM) 600 may be a ratio between the responsive irrigation amount 640 and the actual irrigation amount 645.

The Irrigation Water Utilization Metric (IWUM) 600 may then be used by a water owner 605, to affect the water allocation and pricing 610 of the water 615 provided to this one of the farms 620. It is appreciated that the Irrigation Water Utilization Metric (IWUM) 600 may be used by the water owner 605 as well as by other interested parties, in various ways, and in combination with various other elements, to govern the use of water, encourage water savings, and for other purposes, and that the above description is meant as an example only and is not meant to be limiting.

COMPUTER PROGRAM LISTING

The following sections of a computer code used in a preferred embodiment of the present invention, may be useful for understanding of the invention. It is appreciated the following computer code sections are provided as an example only and are not meant to be limiting.

ANALYZE TERRAIN ATTRIBUTES

```
library(RSAGA)
# Gaussian filtering of both EM and DEM maps
rsaga.geoprocessor(lib = "grid_filter", module = 1, param = list(INPUT = "data/em38.sgrd",
  RESULT = "data/em38_filtered.sgrd", RADIUS = 5), show.output.on.console = FALSE)
rsaga.geoprocessor(lib = "grid_filter", module = 1, param = list(INPUT = "data/dem.sgrd",
  RESULT = "data/dem_filtered.sgrd", RADIUS = 5), show.output.on.console = FALSE)
# SAGA Wetness Index
rsaga.wetness.index(in.dem = "data/dem_filtered.sgrd", out.wetness.index = "data/swi.sgrd",
  show.output.on.console = FALSE)
# Slope
rsaga.slope(in.dem = "data/dem_filtered.sgrd", out.slope = "data/slope.sgrd",
  show.output.on.console = FALSE)
# Aspect
rsaga.aspect(in.dem = "data/dem_filtered.sgrd", out.aspect = "data/aspect.sgrd",
  show.output.on.console = FALSE)
```

INTEGRATION

```
# Load libraries
library(raster)

# Path to raster files
dem <- raster("data/dem_filtered.sdat")
em38 <- raster("data/em38_filtered.sdat")
swi <- raster("data/swi.sdat")
slope <- raster("data/slope.sdat")
aspect <- raster("data/aspect.sdat")

# Stack rasters together
st <- stack(dem, em38, swi, slope, aspect)
# Sort layer names out
names(st) <- c("dem", "em38", "swi", "slope", "aspect")
# Make sure your mask is right
msk <- rasterize(bnd, dem)

## Found 1 region(s) and 1 polygon(s)
st <- mask(st, mask = msk)
plot(st)

# Convert RasterStack to data.frame
spdf <- as(st, "SpatialPixelsDataFrame")
```

Classification on attributes

```
# In this case we put slope and aspect out
attributes <- c("dem", "em38", "swi")
n.clust <- 3
```

```
# Here we use k-means
```

```
clust.res <- kmeans(x = subset(spdf@data, select = attributes), centers = n.clust,
  iter.max = 1000)
```

```
# Setting the names of the clusters using simple lettering
```

```
spdf$cluster <- clust.res$cluster
spdf$mgt <- factor(spdf$cluster)
levels(spdf$mgt) <- LETTERS[1:n.clust]
```

```
# Convert back to RasterStack
```

```
st <- stack(spdf)
plot(raster(st, "mgt"), col = topo.colors(3))
```

```
# Write to Geotiff
```

```
writeRaster(raster(st, "mgt"), "mgt_zones.tif", overwrite = TRUE)
## class      : RasterLayer
## dimensions : 339, 251, 85089 (nrow, ncol, ncell)
## resolution : 5, 5 (x, y)
## extent      : 1502175, 1503430, 5127390, 5129085 (xmin, xmax, ymin, ymax)
## coord. ref. : NA
## data source : /home/pierre/Dropbox/tmp/varigate/river-block/mgt_zones.tif
## names       : mgt_zones
## values      : 1, 3 (min, max)
# Convert raster data to Polygons
mgt <- rasterToPolygons(raster(st, "mgt"), dissolve = TRUE)
spplot(mgt)
```

```
# Save the management zone polygons
```

```
writeOGR(mgt, dsn = "mgt_zones.shp", layer = "mgt_zones", driver = "ESRI Shapefile",
  overwrite_layer = TRUE)
```

MAPS

```
# Read WSN data
```

```
wsn <- read.table(file = "data/wsn_bh.csv", header = TRUE, as.is = TRUE, sep = ",")
```

```
# Get zone IDs
```

```
zone_ids <- unique(wsn$zone)
```

```
# Affect IDs to spatial data
```

```
mgt$zone <- zone_ids
```

```
# remove the existing fields as they are useless now
```

```
mgt$value <- NULL
```

```
# Data manipulation
```

```
library(stringr)
```

```
library(reshape2)
```

```

library(lubridate)

wsn_df <- melt(wsn, c("zone", "variable", "units", "depthcm"))
head(wsn_df)
##   zone variable units depthcm variable value
## 1  z1    mcv percent    20 X31.10.2011    12
## 2  z2    mcv percent    20 X31.10.2011    13
## 3  z3    mcv percent    20 X31.10.2011    37
## 4  z1     fc percent    20 X31.10.2011     0
## 5  z2     fc percent    20 X31.10.2011     0
## 6  z3     fc percent    20 X31.10.2011     0

# There are two columns with the same name so let's change the second one
# to 'date'
names(wsn_df)[5] <- "date"
# Removing the 'X' in front of the dates
wsn_df$date <- str_replace(wsn_df$date, "X", "")
# Convert strings to time objects
wsn_df$date <- dmy(wsn_df$date, tz = "NZ")

# The dynamic variables are 'mcv' and 'smd', the rest is fixed for each
# zone and obtained from the soil physics lab
idx <- which(wsn_df$variable %in% c("fc", "rp", "wp"))
soil_physics <- wsn_df[idx, ]
wsn_realtime <- wsn_df[-idx, ]

# We can plot the realtime WSN data
library(ggplot2)

# Produce a plot
p_wsn <- ggplot(wsn_realtime) + geom_line(aes(x = date, y = value, colour = zone)) +
  facet_grid(depthcm ~ variable)
print(p_wsn)

```

Irrigation logic

Let's first load the libraries we need:

```

library(raster)
library(rgdal)
library(plyr)
library(lubridate)
library(ggplot2)
library(RColorBrewer)
library(gridExtra)

```

Soil characterisation

The characteristics of the various soil types can be read from a stand-alone look-up table,

soil_lut.csv:

```

# Read soil look-up table
soil_lut <- read.csv("data/soil_lut.csv", stringsAsFactors = FALSE)
print(soil_lut)
##           soil fc pwp  rp

```

```
## 1      sand 10  5 7.5
## 2    loamy sand 12  5 8.5
## 3    sandy loam 18  8 13.0
## 4 sandy clay loam 27 17 22.0
## 5      loam 28 14 21.0
## 6    sandy clay 36 25 30.5
## 7    silt loam 31 11 21.0
## 8      silt 30  6 18.0
## 9    clay loam 36 22 29.0
## 10 silty clay loam 38 22 30.0
## 11    silty clay 41 27 34.0
## 12      clay 42 30 36.0
```

This look-up table will give us the hydraulic properties of soil for 12 classes of soil. For example's sake, we will have the following classification:

This can be read from a dedicated file, data/soil_setup.csv, which is generated at the beginning of the season:

```
# Read the paddock specific file
```

```
soil_setup <- read.csv("data/soil_setup.csv", stringsAsFactors = FALSE)
```

```
# Add soil characteristics
```

```
soil_setup <- join(soil_setup, soil_lut, by = "soil")
```

There is a maximum soil moisture deficit for each soil, and at each depth. This is given by the available water holding capacity. This can be defined as the difference between field capacity and permanent wilting point. We can add this information:

```
idx_top <- which(soil_setup$depth == 20)
```

```
idx_bottom <- which(soil_setup$depth == 60)
```

```
# Update sensor values
```

```
soil_setup$smd_max <- NA
```

```
soil_setup$smd_max[idx_top] <- 2 * (soil_setup$fc[idx_top] - soil_setup$pwp[idx_top])
```

```
soil_setup$smd_max[idx_bottom] <- 2 * (soil_setup$fc[idx_top] - soil_setup$pwp[idx_top]) +  
  4 * (soil_setup$fc[idx_bottom] - soil_setup$pwp[idx_bottom])
```

```
soil_setup$smd_max <- -1 * soil_setup$smd_max
```

```
print(soil_setup)
```

```
## id   zone depth   soil fc pwp  rp smd_max
## 1 1    dry  20   sandy loam 18 8 13.0  -20
## 2 1    dry  60   loamy sand 12 5 8.5   -48
## 3 2 intermediate 20   sandy loam 18 8 13.0  -20
## 4 2 intermediate 60   loamy sand 12 5 8.5   -48
## 5 3    wet  20   silty clay loam 38 22 30.0  -32
## 6 3    wet  60   sandy loam 18 8 13.0  -72
```

We will then read the management zones polygons produced earlier:

```
# Read management zones file
```

```
mgt <- readOGR(dsn = "data/mgt_zones.shp", layer = "mgt_zones")
```

```
## OGR data source with driver: ESRI Shapefile
```

```
## Source: "data/mgt_zones.shp", layer: "mgt_zones"
```

```
## with 3 features and 1 fields
```

```
## Feature type: wkbMultiPolygon with 2 dimensions
```

```
# Re-level zones IDs to use characteristics
```

```
mgt$zone <- factor(mgt$zone, levels = 1:3, labels = c("wet", "intermediate",
"dry"))
```

```
summary(mgt)
## Object of class SpatialPolygonsDataFrame
## Coordinates:
##   min   max
## x 1793739 1795019
## y 5552504 5553359
## Is projected: TRUE
## proj4string :
## [+proj=tmerc +lat_0=0 +lon_0=173 +k=0.9996 +x_0=1600000
## +y_0=10000000 +ellps=GRS80 +units=m +no_defs]
## Data attributes:
##      wet intermediate      dry
##      1         1         1
```

Soil moisture data

The WSN data is supposed to be a table with four columns:

timestamp

zone

depth

mcv

2011-09-29 09:19:24.000

1

20

41.605

2011-09-29 09:19:24.000

1

60

53.212

2011-09-29 09:23:39.000

2

20

12.913

2011-09-29 09:23:39.000

2

60

32.795

...

...

...

...

In this example, we will read such WSN data from a file, `wsn_bh.csv`. We are using the `lubridate` library to explicitly store the date and/or time information as a `POSIXct` object. We are of course using the NZ timezone here.

Read WSN data

```
wsn <- read.csv("data/wsn_bh.csv", stringsAsFactors = FALSE)
```

Re-level zones IDs to proper characteristics

```
wsn$zone <- factor(wsn$zone, levels = 1:3, labels = c("dry", "intermediate",
"wet"))
```

```
# Transform timestamps from characters to time objects
wsn$timestamp <- dmy(wsn$timestamp, tz = "NZ")
```

```
# Add zones field capacity information
wsn <- join(wsn, soil_setup, by = c("zone", "depth"))
```

```
# Here's what the data looks like
head(wsn)
```

```
## timestamp zone depth mcv id soil fc pwp rp smd_max
## 1 2011-10-31 dry 20 12 1 sandy loam 18 8 13 -20
## 2 2011-11-01 dry 20 11 1 sandy loam 18 8 13 -20
## 3 2011-11-02 dry 20 11 1 sandy loam 18 8 13 -20
## 4 2011-11-03 dry 20 11 1 sandy loam 18 8 13 -20
## 5 2011-11-04 dry 20 11 1 sandy loam 18 8 13 -20
## 6 2011-11-05 dry 20 13 1 sandy loam 18 8 13 -20
```

Processing

To facilitate processing, we are writing two processing functions. The first one does two things. First, it is extracting the raw data from the WSN data for a given timestamp, and then, it is transforming that raw data into the “real” soil moisture status using the soil hydraulic characteristics at any one zone.

```
# Associate soil water status to zones
```

```
get_soil_moisture_status <- function(timestamp, zones) {
```

```
  # Get the WSN data for the current timestamp
```

```
  cur_wsn_df <- wsn[which(wsn$timestamp %within% new_interval(timestamp, timestamp)),
  ]
```

```
  # Join soil information to zones
```

```
  zones@data <- join(zones@data, cur_wsn_df, by = "zone")
```

```
  # Update MCV values using the soil data
```

```
  zones$mcv_mm <- zones$fc_depth <- zones$smd <- NA
```

```
  # First the top sensor
```

```
  idx_top <- which(zones$depth == 20)
```

```
  idx_bottom <- which(zones$depth == 60)
```

```
  # Update sensor values to millimeters from volumetric values
```

```
  zones$mcv_mm[idx_top] <- zones$mcv[idx_top] * 2
```

```
  zones$mcv_mm[idx_bottom] <- 2 * zones$mcv[idx_top] + 4 * zones$mcv[idx_bottom]
```

```
  # Compute FC values between 0-20cm and 0-60cm
```

```
  zones$fc_depth[idx_top] <- 2 * zones$fc[idx_top]
```

```
  zones$fc_depth[idx_bottom] <- 2 * zones$fc[idx_top] + 4 * zones$fc[idx_bottom]
```

```
  # Compute soil moisture deficit
```

```

zones$smd <- zones$mcv_mm - zones$fc_depth

# Compute water left in soil
zones$water_left <- zones$rp - zones$smd

# Return a SpatialPolygonDataFrame object
zones
}

# Test
res <- get_soil_moisture_status(timestamp = wsn$timestamp[1], zones = mgt)
summary(res)
## Object of class SpatialPolygonsDataFrame
## Coordinates:
##   min   max
## x 1793739 1795019
## y 5552504 5553359
## Is projected: TRUE
## proj4string :
## [+proj=tmerc +lat_0=0 +lon_0=173 +k=0.9996 +x_0=1600000
## +y_0=10000000 +ellps=GRS80 +units=m +no_defs]
## Data attributes:
##   zone   timestamp      depth    mcv
## wet      :2 Min. :2011-10-31 Min. :20 Min. :12.0
## intermediate:2 1st Qu.:2011-10-31 1st Qu.:20 1st Qu.:13.0
## dry       :2 Median :2011-10-31 Median :40 Median :14.0
##           Mean :2011-10-31 Mean :40 Mean :22.2
##           3rd Qu.:2011-10-31 3rd Qu.:60 3rd Qu.:31.5
##           Max. :2011-10-31 Max. :60 Max. :43.0
##   id    soil      fc    pwp
## Min. :1.00 Length:6    Min. :12.0 Min. :5.00
## 1st Qu.:1.25 Class :character 1st Qu.:13.5 1st Qu.:5.75
## Median :2.00 Mode  :character Median :18.0 Median :8.00
## Mean :2.00          Mean :19.3 Mean :9.33
## 3rd Qu.:2.75          3rd Qu.:18.0 3rd Qu.:8.00
## Max. :3.00          Max. :38.0 Max. :22.00
##   rp    smd_max    smd    fc_depth
## Min. :8.50 Min. :-72 Min. :-12.0 Min. :36.0
## 1st Qu.:9.62 1st Qu.: -48 1st Qu.: -9.5 1st Qu.:46.0
## Median :13.00 Median : -40 Median : -5.0 Median :80.0
## Mean :14.33 Mean : -40 Mean :11.3 Mean :77.3
## 3rd Qu.:13.00 3rd Qu.: -23 3rd Qu.: 1.0 3rd Qu.:84.0
## Max. :30.00 Max. : -20 Max. :98.0 Max. :148.0
##   mcv_mm    water_left
## Min. :24.0 Min. : -85.0
## 1st Qu.:38.0 1st Qu.: 9.0
## Median :75.0 Median :19.8
## Mean :88.7 Mean :3.0
## 3rd Qu.:83.5 3rd Qu.:24.5
## Max. :246.0 Max. :32.0

```

The second function is the irrigation logic algorithm. It takes the soil moisture status at 20cm and at

50cm, and spits out a recommendation.

Irrigation logic function

```
irrigation_logic <- function(smd_top, smd_bottom, smd_max_top, smd_max_bottom) {

  if (smd_top >= 0 & smd_bottom >= 0)
    res <- 0
  if (smd_top >= 0 & smd_bottom < 0)
    res <- 0
  if (smd_top < 0 & smd_bottom >= 0)
    res <- ifelse(-1 * smd_top >= smd_max_top, -1 * smd_top, smd_max_top)
  if (smd_top < 0 & smd_bottom < 0) {
    smd_top <- ifelse(smd_top >= smd_max_top, smd_top, smd_max_top)
    smd_bottom <- ifelse(smd_bottom >= smd_max_bottom, smd_bottom, smd_max_bottom)
    idx <- which.min(c(smd_top, smd_bottom))
    res <- -1 * c(smd_top, smd_bottom)[idx]
  }

  res
}
```

Test

```
irrigation_logic(smd_top = -7, smd_bottom = 0, smd_max_top = -10, smd_max_bottom = -12)
## [1] 7
irrigation_logic(smd_top = -17, smd_bottom = -9, smd_max_top = -10, smd_max_bottom = -12)
## [1] 10
```

Finally everything can be processed inside a single list. The result of the code below is a list of SpatialPolygonsDataFrame containing the recommendation. There's one for each timestamp available in the WSN data.

```
bh_irrigation <- llply(unique(wsn$timestamp), function(t) {
  # Get current moisture data
  cur_mgt <- get_soil_moisture_status(timestamp = t, zones = mgt)

  # Apply the irrigation decision
  irrigation_decision <- ddply(cur_mgt@data, .(zone), function(x) {
    smd_top <- x[which(x$depth == 20), "smd"]
    smd_bottom <- x[which(x$depth == 60), "smd"]
    smd_max_top <- x[which(x$depth == 20), "smd_max"]
    smd_max_bottom <- x[which(x$depth == 60), "smd_max"]
    smd_df <- data.frame(smd_top = smd_top, smd_bottom = smd_bottom, smd_max_top =
smd_max_top,
      smd_max_bottom = smd_max_bottom)

    decision <- irrigation_logic(smd_top, smd_bottom, smd_max_top, smd_max_bottom)

    data.frame(zone = unique(x$zone), timestamp = unique(x$timestamp), decision = decision)
  })
  # Merge decision back to the zones
  res <- mgt
  res@data <- join(res@data[, "zone", drop = FALSE], irrigation_decision,
    by = "zone")
})
```

```
# Return zones object
res
})
names(bh_irrigation) <- unique(wsn$timestamp)
```

Plotting

```
msk <- raster("data/swi.sdat")
msk[!is.na(msk)] <- 1
writeRaster(msk, "data/msk.tif")
Anyway, back to my maps:
# Plotting function
#
plot_irrigation <- function(t, msk, range_data = c(0,10)){
  # If a char is passed to the function
  if(!is.POSIXt(t)) t <- dmy(t, tz = "NZ")

  idx <- which(names(bh_irrigation) == as.character(t))
  cur_spdf <- bh_irrigation[[idx]]

  # Switching to raster for more efficient visualisation
  cur_raster <- rasterize(cur_spdf, msk)
  cur_raster <- na.exclude(as.data.frame(cur_raster, xy = T))

  # Create plot object
  p <- ggplot(cur_raster, aes(x=x, y=y)) +
    geom_raster(aes(fill = decision)) +
    scale_fill_gradientn(
      "Recommended\nIrrigation (mm)",
      colours=brewer.pal(n=5, name="YlGnBu"),
      limits = range_data
    ) +
    labs(x="E (m)", y = "N (m)", title = t) +
    coord_equal()

  p
}
```

```
# Find the min and max recommendations for the whole dataset
# so we use a fixed colour scale
min_range <- min(lapply(bh_irrigation, function(x) min(x$decision)))
max_range <- max(lapply(bh_irrigation, function(x) max(x$decision)))
```

```
# Get the raster mask of the paddock
# (I'm using raster rather than vector data for
# plotting purposes, it's faster)
msk <- raster('data/mask.tif')
```

```
# Generate plots
plots <- lapply(
  # Here I only take a subset of the available date
  # to save on processing time
```

```

.data = unique(wsn$timestamp)[50:55],
.fun = plot_irrigation,
msk = msk, range_data = c(min_range, max_range)
)

# You can either print maps one by one....

# Here the first map
print(plots[[1]])

# Here the third one
print(plots[[3]])

# ...or print on the same page
do.call(grid.arrange, plots)

# We can also check the recommendations for each zone
reco_zones <- ldply(bh_irrigation, function(x) data.frame(zone = x@data$zone,
  decision = x@data$decision))

# Probability density functions for each zone (Note that I log-transform the
# X axis)
ggplot(reco_zones) + geom_density(aes(x = decision, colour = zone)) + scale_x_log10()

# We can check what's happening on a monthly basis
reco_zones$month <- laply(reco_zones$.id, function(x) as.character(month(ymd(x),
  label = TRUE, abbr = FALSE)))

# Probability density functions for each zone (Note that I log-transform the
# X axis)
ggplot(reco_zones) + geom_density(aes(x = decision, colour = zone)) + facet_wrap(~month)

SOIL TYPE RECOGNITION
library(plyr)
library(stringr)
library(reshape2)
library(ggplot2)
library(caret)
## Loading required package: cluster Loading required package: foreach
## Loading required package: lattice

setwd("/home/pierre/Dropbox/tmp/varigate/curves/code")
# Load NSD
nsd <- read.csv("../data/nsd.csv")

# Select attributes
nsd <- subset(nsd, select = c("Type.qualifier", "X0.025.bar", "X0.05.bar", "X0.1.bar",
  "X0.2.bar", "X0.4.bar", "X1.bar", "X15.bar"))

# Remove NAs nsd$texture <- str_replace(nsd$texture, "", as.character(NA))
nsd <- na.exclude(nsd)

```

```

# Add some kind of ID
nsd$id <- 1:nrow(nsd)

# Re-arrange data
nsd <- melt(nsd, c("id", "Type.qualifier"))
# Better colnames
names(nsd) <- c("id", "texture", "pressure", "moisture")
nsd$texture <- factor(nsd$texture)
# Better pressure values
nsd$pressure <- as.numeric(as.character(str_replace(str_replace(nsd$pressure,
  "X", ""), ".bar", "")))
# Group similar groups
nsd$texture <- str_replace(nsd$texture, "CLAY LOAM, PALE TOPSOIL PHASE", "CLAY LOAM")
nsd$texture <- str_replace(nsd$texture, "MOTTLED SILT LOAM", "SILT LOAM")
nsd$texture <- str_replace(nsd$texture, "PEAT DRAINED", "PEAT")
nsd$texture <- str_replace(nsd$texture, "PEAT UNDRAINED", "PEAT")
nsd$texture <- factor(nsd$texture)
ggplot(nsd) + geom_line(aes(x = pressure, y = moisture, group = id, colour = texture),
  alpha = 0.2) + geom_point(aes(x = pressure, y = moisture, group = id, colour = texture),
  alpha = 0.2) + # geom_smooth(aes(x=pressure, y=moisture, colour=texture), method = lm, se =
# TRUE, lwd=2) +
scale_colour_discrete("Texture") + scale_x_log10() + labs(x = "Pressure (bar)",
  y = "Moisture content (%)") + theme_bw()

pmax <- 0.5
nsd <- subset(nsd, pressure <= pmax)
ggplot(nsd) + geom_line(aes(x = pressure, y = moisture, group = id, colour = texture),
  alpha = 0.2) + geom_point(aes(x = pressure, y = moisture, group = id, colour = texture),
  alpha = 0.2) + geom_smooth(aes(x = pressure, y = moisture, colour = texture),
  method = lm, se = TRUE, lwd = 2) + scale_colour_discrete("Texture") + scale_x_log10() +
  labs(x = "Pressure (bar)", y = "Moisture content (%)") + theme_bw()

fits <- ddply(nsd, .(id), function(x) {
  fit <- lm(moisture ~ pressure, data = x)
  data.frame(texture = as.character(x$texture), intercept = fit$coefficients[1],
    slope = fit$coefficients[2])
})
m_fits <- melt(fits, c("id", "texture"))

ggplot(m_fits) + geom_boxplot(aes(x = texture, y = value)) + facet_wrap(~variable,
  scales = "free_y")

pct_calib <- 0.5
set.seed(20130920)
idx_calib <- sample(1:nrow(fits), size = floor(pct_calib * nrow(fits)), replace = FALSE)
calib <- fits[idx_calib, ]
valid <- fits[-idx_calib, ]

```

```

ctrl <- trainControl(method = "repeatedcv", repeats = 5)

fit <- train(texture ~ intercept + slope, data = fits, method = "C5.0", tuneLength = 10,
  trControl = ctrl)
## Loading required package: class

summary(fit)
##
## Call:
## C5.0.default(x = "scrubbed", y = "scrubbed", trials = 1, rules =
## "CF", "minCases", "fuzzyThreshold", "sample", "earlyStopping",
## "label", "seed"))
##
##
## C5.0 [Release 2.07 GPL Edition]   Mon Sep 23 09:02:19 2013
## -----
##
## Class specified by attribute `outcome'
##
## Read 1220 cases (3 attributes) from undefined.data
##
## No attributes winnowed
##
## Decision tree:
##
## intercept > 59.85417:
## :...slope > -24.79032: CLAY LOAM (75)
## : slope <= -24.79032:
## : :...intercept > 66.29583: PEAT (150)
## : : intercept <= 66.29583:
## : : :...slope <= -42.33333: SILT LOAM (20)
## : : : slope > -42.33333:
## : : : :...intercept <= 61.45417: PEAT (10)
## : : : : intercept > 61.45417: SILT LOAM (5)
## intercept <= 59.85417:
## :...intercept <= 40.39167:
## : :...intercept > 33.6125: SILT LOAM (255)
## : : intercept <= 33.6125:
## : : :...intercept <= 31.48333: SILT LOAM (25)
## : : : intercept > 31.48333: LOAMY SAND (5)
## intercept > 40.39167:
## :...slope <= -30.80645: SILT LOAM (180)
## : slope > -30.80645:
## : :...intercept <= 51.30416:
## : : :...slope <= -27.02688:
## : : : :...intercept <= 47.1125: CLAY LOAM (15)
## : : : : : intercept > 47.1125: SILT LOAM (15)
## : : : : slope > -27.02688:
## : : : : :...intercept <= 40.65:
## : : : : : :...intercept > 40.4: CLAY LOAM (5)
## : : : : : : : intercept <= 40.4:

```

```

##      :      : ...slope <= -18.71505: CLAY LOAM (5)
##      :      :      slope > -18.71505: SILT LOAM (5)
##      :      :      intercept > 40.65:
##      :      :      ...slope <= -8.827957: SILT LOAM (295)
##      :      :      slope > -8.827957:
##      :      :      ...intercept > 45.825: SILT LOAM (40)
##      :      :      intercept <= 45.825:
##      :      :      ...intercept <= 43.39583: SILT LOAM (10)
##      :      :      intercept > 43.39583: CLAY LOAM (10)
##      :      :      intercept > 51.30416:
##      :      :      ...slope > -5.564516: SILT LOAM (15)
##      :      :      slope <= -5.564516:
##      :      :      ...intercept <= 52.3625: CLAY LOAM (10)
##      :      :      intercept > 52.3625:
##      :      :      ...intercept <= 53.8375: SILT LOAM (20)
##      :      :      intercept > 53.8375:
##      :      :      ...intercept <= 54.42083: CLAY LOAM (5)
##      :      :      intercept > 54.42083:
##      :      :      ...slope <= -16.45161: SILT LOAM (15)
##      :      :      slope > -16.45161:
##      :      :      ...slope <= -14.43011: CLAY LOAM (5)
##      :      :      slope > -14.43011:
##      :      :      ...slope > -7.564516: CLAY LOAM (5)
##      :      :      slope <= -7.564516:
##      :      :      ...intercept > 55.72917: SILT LOAM (10)
##      :      :      intercept <= 55.72917:
##      :      :      ...intercept <= 55.40833: SILT LOAM (5)
##      :      :      intercept > 55.40833: CLAY LOAM (5)
##
##

```

Evaluation on training data (1220 cases):

```

##
##  Decision Tree
##  -----
##  Size   Errors
##
##  28  0( 0.0%) <<
##
##
##  (a) (b) (c) (d) <-classified as
##  ----
##  5          (a): class LOAMY SAND
##  915        (b): class SILT LOAM
##  140        (c): class CLAY LOAM
##  160        (d): class PEAT
##
##

```

Attribute usage:

```

##
## 100.00% intercept
## 76.64% slope

```

##

##

Time: 0.0 secs

postResample(obs = valid\$texture, pred = predict(fit, newdata = valid))

Accuracy Kappa

INDUSTRIAL APPLICABILITY

The invention will be useful in the areas of irrigation of any type of pasture, crop or other agricultural environment where irrigation of land is required.

The invention provides and exemplifies a system and a method for reducing the amount of water required to irrigate an area of land, by applying different amounts of water to different parts of the field, based at least in part on an analysis of spatial soil properties of the field including topological features, and extrapolation of data from soil sensors placed in different parts of a field.

The invention thus provides a useful system and method for irrigating land in an environmentally friendly manner.

CLAIMS

1. A computerized differential irrigation system comprising:
 - a computerized Topography Integrated Ground water Retention (TIGER) map generator receiving at least the following inputs:
 - a topographical input describing topographical features of an area to be irrigated; and
 - an electromagnetic input describing conductive features of the area to be irrigated,
 - and in which the computerized Topography Integrated Ground water Retention (TIGER) map generator includes:
 - a computerized topographic feature processing functionality providing information relating to at least one of slope, aspect and catchment area features of said area to be irrigated; and
 - a computerized topographic feature utilization functionality employing at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated; and
 - a computerized computing functionality employing the Topography Integrated Ground water Retention (TIGER) map together with at least current outputs of wetness sensors located at the plurality of different regions within the area to be irrigated to generate a current irrigation plan; and
 - a computerized irrigation control subsystem automatically utilizing the current irrigation map to control irrigation within the area to be irrigated based on the current irrigation instructions and to cause different amounts of water to be provided to the different regions within the area to be irrigated.
2. A computerized differential irrigation system according to claim 1 and in which the computerized Topography Integrated Ground water Retention (TIGER) map generator employs automatically generated soil type data.
3. A computerized differential irrigation system according to claim 1 and in which the computerized Topography Integrated Ground water Retention (TIGER) map generator includes a computerized automatic soil type analysis functionality which obviates the need for laboratory testing of soil in the area to be irrigated.
4. A method of using a computerized system according to claim 1 by
 - ascertaining an amount of water required to irrigate said area based on said current irrigation plan;
 - ascertaining an amount of water required to irrigate said area if differential irrigation is not employed; and
 - calculating an irrigation efficiency metric representing a water savings produced by employing the current irrigation plan.

5. A method according to claim 4 and also comprising employing the irrigation efficiency metric for at least one of controlling supply and pricing of water and mandating irrigation policy.
6. A computerized irrigation planning system comprising:
 - a computerized Topography Integrated Ground water Retention (TIGER) map generator receiving at least the following inputs:
 - a topographical input describing topographical features of an area to be irrigated; and
 - an electromagnetic input describing conductive features of the area to be irrigated,
 - and in which the computerized Topography Integrated Ground water Retention (TIGER) map generator includes:
 - a computerized topographic feature processing functionality providing information relating to at least one of slope, aspect and catchment area features of the area to be irrigated; and
 - a computerized topographic feature utilization functionality employing the at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated; and
 - a computerized computing functionality employing the Topography Integrated Ground water Retention (TIGER) map together with at least current outputs of wetness sensors located at the plurality of different regions within the area to be irrigated to generate a current irrigation plan.
7. A computerized system according to claim 6 and wherein the computerized Topography Integrated Ground water Retention (TIGER) map generator employs automatically generated soil type data.
8. A computerized system according to claim 6 and wherein the computerized Topography Integrated Ground water Retention (TIGER) map generator includes computerized automatic soil type analysis functionality which obviates the need for laboratory testing of soil in the area to be irrigated.
9. A method of using a computerized system according to any one of claims 6-8, to generate an irrigation plan obviating the need for laboratory testing of soil in the area to be irrigated.
10. An automated Topography Integrated Ground water Retention (TIGER) map generating system comprising:
 - a data input interface receiving at least the following inputs:
 - a topographical input describing topographical features of an area to be irrigated; and
 - an electromagnetic input describing conductive features of the area to be irrigated,

computerized topographic feature processing functionality automatically deriving from the inputs, information relating to at least one of slope, aspect and catchment area features of the area to be irrigated; and

computerized topographic feature utilization functionality employing the at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated.

11. A computerized system according to claim 10 and wherein the computerized Topography Integrated Ground water Retention (TIGER) map generating system also employs automatically generated soil type data which is input at the data input interface.

12. A computerized system according to claim 10 and wherein the computerized Topography Integrated Ground water Retention (TIGER) map generating system includes computerized automatic soil type analysis functionality which obviates the need for laboratory testing of soil in the area to be irrigated.

13. A method of using a computerized system according to claim 10 by:
ascertaining an amount of water required to irrigate the area based on the current irrigation plan;
ascertaining an amount of water required to irrigate the area if differential irrigation is not employed; and
calculating an irrigation efficiency metric representing a water savings produced by employing the current irrigation plan.

14. A method according to claim 13 and wherein the irrigation efficiency metric is employed for at least one of controlling supply and pricing of water and mandating irrigation policy.

15. An automated soil type classification system comprising:
an input interface receiving:
offline pre-existing laboratory generated soil drying curves, which indicate at least the following parameters for a plurality of different types of soils: field capacity, wilting point and refill point; and
empirical field drying curves for a field for which irrigation is to be planned;

and a computer operated automatic correlator employing the offline pre-existing laboratory generated soil drying curves and the empirical field drying curves for a field for which irrigation is to be planned to automatically provide a soil type map for the field for which irrigation is to be planned.

16. An automated soil type classification system according to claim 15 and wherein the computer operated automatic correlator employs automated learning techniques whereby correlator performance improves over time based on accumulated correlation data.
17. An automated soil type classification system according to claim 15 and wherein the computer operated automatic correlator employs automated learning techniques whereby correlator performance improves over time based on accumulated correlation data from multiple automated soil type classification systems in other fields, which data is shared via a computer network.
18. A computerized differential irrigation system comprising:
 - a computerized Topography Integrated Ground watEr Retention (TIGER) map generator receiving at least the following inputs:
 - a topographical input describing topographical features of an area to be irrigated; and
 - an electromagnetic input describing conductive features of the area to be irrigated,
 - and in which the computerized Topography Integrated Ground watEr Retention (TIGER) map generator includes:
 - a computerized automatic soil type analysis functionality which obviates the need for laboratory testing of soil in the area to be irrigated.
19. A computerized system according to claim 18 and wherein the computerized Topography Integrated Ground watEr Retention (TIGER) map generating system also employs automatically generated soil type data which is input at the data input interface.
20. A computerized system according to claim 18 and wherein the computerized Topography Integrated Ground watEr Retention (TIGER) map generating system includes computerized automatic soil type analysis functionality which obviates the need for laboratory testing of soil in the area to be irrigated.
21. A method of using a computerized system according to claim 18 and also:
 - ascertaining an amount of water required to irrigate the area based on the current irrigation plan;
 - ascertaining an amount of water required to irrigate the area if differential irrigation is not employed; and
 - calculating an irrigation efficiency metric representing a water savings produced by employing the current irrigation plan.
22. A method according to claim 21 and also comprising employing the irrigation efficiency metric for at least one of controlling supply and pricing of water and mandating irrigation policy.
23. A computerized irrigation efficiency metric generating system comprising:

a computerized Topography Integrated Ground watEr Retention (TIGER) map generator receiving at least the following inputs:

a topographical input describing topographical features of an area to be irrigated; and

an electromagnetic input describing conductive features of the area to be irrigated,

and in which the computerized Topography Integrated Ground watEr Retention (TIGER) map generator includes:

a computerized topographic feature processing functionality providing information relating to at least one of slope, aspect and catchment area features of the area to be irrigated; and

a computerized topographic feature utilization functionality employing the at least one of slope, aspect and catchment area features of the area to be irrigated for automatically ascertaining water retention at a plurality of different regions within the area to be irrigated; and

a computing functionality employing the Topography Integrated Ground watEr Retention (TIGER) map together with at least current outputs of wetness sensors located at the plurality of different regions within the area to be irrigated to generate a current irrigation plan; and

an irrigation efficiency analyzer operative to:

ascertain an amount of water required to irrigate the area based on the current irrigation plan;

ascertain an amount of water required to irrigate the area if differential irrigation is not employed; and

calculate an irrigation efficiency metric representing a water saving produced by employing the current irrigation plan.

24. A method of using a computerized irrigation efficiency metric generating system as claimed in claim 23 to calculate an irrigation efficiency metric representing a water saving by

ascertaining an amount of water required to irrigate an area based on the current irrigation plan;

ascertaining an amount of water required to irrigate the area if differential irrigation is not employed; and

calculating an irrigation efficiency metric representing a water savings produced by employing the current irrigation plan

25. A method of employing an irrigation efficiency metric for at least one of controlling supply and pricing of water and mandating irrigation policy using a computerized irrigation efficiency metric generating system according to claim 1.

26. A computerized system according to any of the preceding claims 1-3, 6-8, 10-12, 15-20 and 23 wherein the irrigation plan is a non-differential irrigation plan, based on the Topography Integrated Ground watEr Retention (TIGER) map.

27. A computerized system according to any of the preceding claims 1-3, 6-8, 10-12, 15-20 and 23 and also comprising an output interface providing irrigation instructions to at least one predetermined type of irrigators.

28. A method according to any one of the preceding claims 4, 5, 9, 13-14, 21-22 and 24-25 and also calculating a first irrigation amount suitable for a first area and a second irrigation amount suitable for a second area, and wherein the first irrigation amount is different from the second irrigation amount, the first area is different from the second area, and the first area and the second area are sectors of a circle and are irrigated by a pivot mechanical irrigator.

29. A method according to any one of the preceding claims 4, 5, 9, 13-14, 21-22 and 24-25 and also calculating a first irrigation amount suitable for a first area and a second irrigation amount suitable for a second area, said first irrigation amount is different from said second irrigation amount, said first area is different from said second area, and said first area and said second area are rectangular cross-sections of a rectangle and are irrigated by a lateral-move mechanical irrigator.