Sustainable Groundwater Management Using Water Markets

Aaron McGarvey – MSc Applied Geosciences candidate

Advisors: Vince Uhl, PG, Uhl and Associates Inc; Carl Mastropaolo, Lecturer, Dept of Earth and Environmental Science

INTRODUCTION

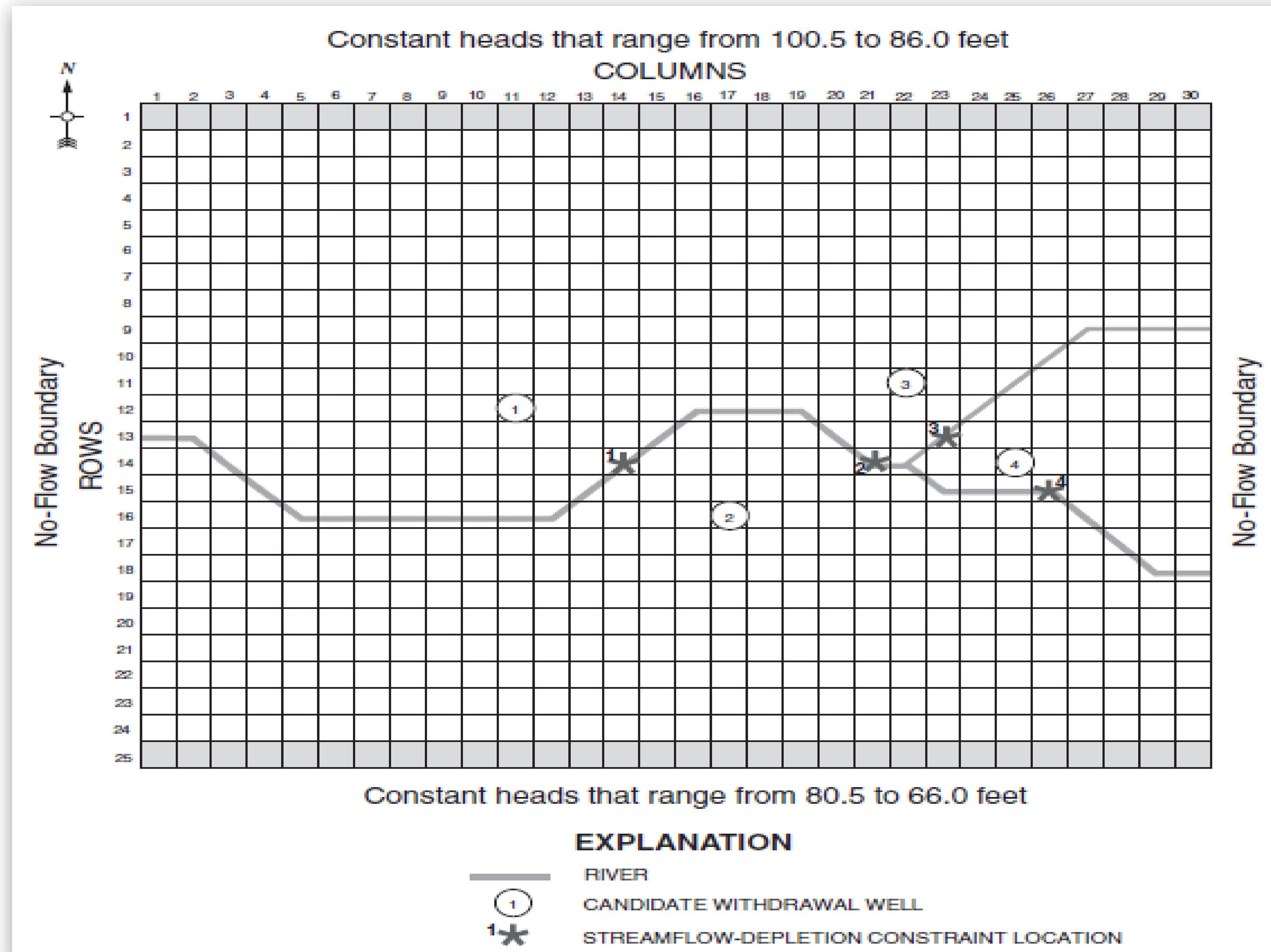
Increasing regulation of the world's groundwater resources is prompting many governments to consider water markets as a water management tool. Yet despite considerable research and a dearth of new technology, few digital tools exist for creating and operating water markets. This project proposes a digital framework for a groundwater market that would allow agricultural groundwater users to exchange temporary water leasing rights over a digital exchange. The ultimate realization of this idea would take the place of a digital web application built using Python.

BACKGROUND

For decades, California's groundwater sustainability problem has been characterized by declining water levels, increasing demand, and extreme drought. In 2014, the State passed the Sustainable Groundwater Management Act requiring 127 groundwater basins to halt overdraft and return to balanced levels of pumping. The SGMA act allows for the use of groundwater markets to manage this problem. In a groundwater market, stakeholders are given an allocation they can either use or trade to a neighboring stakeholder in the same basin.

METHODS AND MATERIALS

As part of a literature review, Raffensperger's model was identified as the most comprehensive model for groundwater market creation. The model uses a response matrix to account for spatial variabilities in groundwater pumping. To provide a framework for a digital application, the author examined three main computational frameworks: MODFLOW, MODFLOW-GWM, and PULP, a Python library for linear programming optimization. A sample problem from MODFLOW-GWM literature was used to simulate a real world scenario in which four farmers trade for groundwater pumping rights subject to environmental constraints.



$$Maximize\ Q1 + Q2 + Q3 + Q4 \text{ (Objective Function)}$$

Subject To:

$$Flow\ at\ constraint\ 1 \geq 1000cfs$$
$$Flow\ at\ constraint\ 2 \geq 900cfs$$
$$Flow\ at\ constraint\ 3 \geq 800cfs$$
$$Flow\ at\ constraint\ 4 \geq 700cfs$$

Figure 1- Model grid and linear programming formulation for a MODFLOW-GWM sample problem with four candidate wells (circles) and four constraint locations (asterisks). The objective is to maximize the total water withdrawn subject to the environmental constraints.

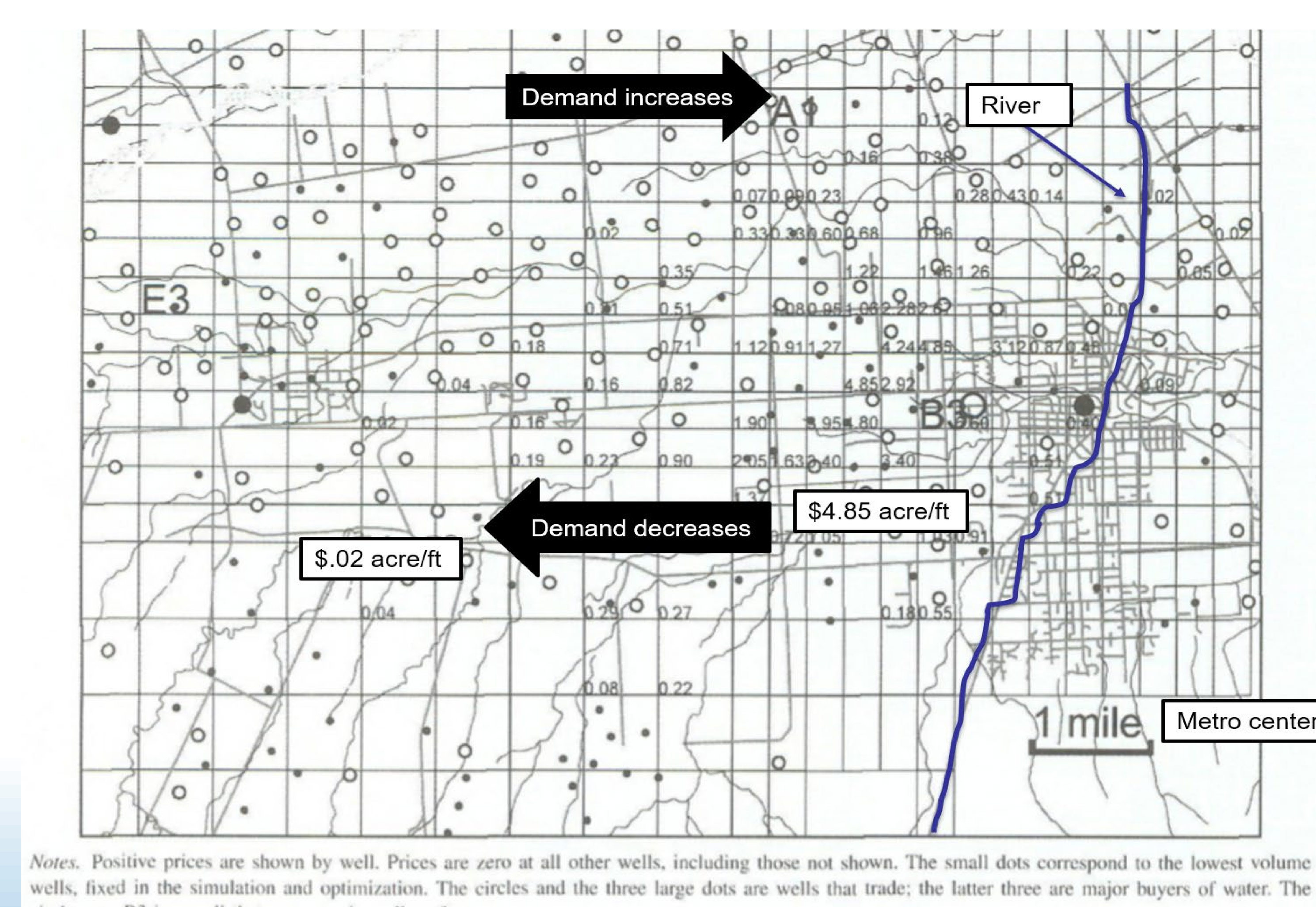


Figure 2- Raffensperger's evaluation of his model in a basin in Marlborough, New Zealand. User's bids for water are entered as coefficients of the objective function in the optimization problem. The MODFLOW-GWM response matrix values are entered as coefficients in the constraint summations.

CONCLUSIONS

1. **Raffensperger's model is the most comprehensive model for groundwater market creation and represents the best chance for successfully implementing a groundwater market under SGMA.**
2. **The digital application that will allow water users to sell temporary groundwater leasing rights will need to be based on three computational systems: MODFLOW, MODFLOW-GMW, and the PULP Python library.**
3. **The creation of a groundwater market assumes proper monitoring of groundwater abstraction is possible for enforcement purposes.**

In [4]:

```
prob =LpProblem('Sample Water Market Problem', LpMaximize)

# The 4 flow rate variables (Qs) are created with a lower limit of zero
x1=LpVariable("Q1",0,None)
x2=LpVariable("Q2",0,None)
x3=LpVariable("Q3",0,None)
x4=LpVariable("Q4",0,None)

# The objective function is added to 'prob' first, bids entered as coefficients in objective function
prob += 4*x1 + 3*x2 + x3 + 8*x4, "Total Economic Value"

# The constraints are entered
# Initial head in the MODFLOW model is 80ft at all active cells
# Regulatory head levels must be at a minimum of 50ft as a regulatory requirement
# Therefore drawdown at all constraint locations must be >= 30ft
prob += 0.010195*x1 + 0.007789*x2 + 0.007970*x3 + 0.007505*x4 <= 30, "Head Constraint 1"
prob += 0.010305*x1 + 0.010342*x2 + 0.009296*x3 + 0.007611*x4 <= 30, "Head Constraint 2"


# The problem data is written to an .lp file
prob.writeLP("WaterMarketSample1.lp")

# The problem is solved using PuLP's choice of Solver
prob.solve()

# The status of the solution is printed to the screen
print("Status:", LpStatus[prob.status])

# Each of the variables is printed with it's resolved optimum value
for v in prob.variables():
    print(v.name, "=", v.varValue)

# The optimised objective function value is printed to the screen
print("Total value of the objective function = ", value(prob.objective))
```

```
Status: Optimal
Q1 = 0.0
Q2 = 0.0
Q3 = 0.0
Q4 = 3941.6634
Total value of the objective function =  31533.3072
```

Figure 3- A Python code sample of how to utilize the PuLP library to solve an optimization problem with four wells and two constraints.

REFERENCES

Raffensperger, J. F., & Milke, M. W. (2017). Smart Markets for Water Resources: A Manual for Implementation. Springer International Publishing. Retrieved from https://books.google.com/books?id=_1K5DgAAQBAJ

Raffensperger JF, Milke M. (2005) A Design for a Fresh Water Spot Market. Water Science and Technology: Water Supply.;5(6):217-224.

Raffensperger JF, Milke MW, Read EG. A Deterministic Smart Market Model for Groundwater. Oper Res. 2009;57(6):1333-1346.