PROCEEDINGS, 50th Workshop on Geothermal Reservoir Engineering
Stanford University, Stanford, California, February 10-12, 2025
SGP-TR-229

GeoCLUSTER v2.0: A Closed-Loop, Techno-Economic Simulator Supporting New Case Studies

Anastasia Bernat[1], Alexander Buchko[1], Aaron Moreno[1] and Koenraad Beckers[2]

[1]Pacific Northwest National Laboratory, 902 Battelle Blvd, Richland, WA 99354

[2]National Renewable Energy Laboratory, 15013 Denver West Parkway Golden, CO 80401

anastasia.bernat@pnnl.gov, alexander.buchko@pnnl.gov, aaron.moreno@pnnl.gov, Koenraad.Beckers@nrel.gov

Keywords: closed-loop, techno-economic, simulator, Slender-Body Theory (SBT)

ABSTRACT

Closed-loop geothermal systems (CLGSs) utilize a closed-loop heat exchanger such as U-loop or co-axial systems for subsurface heat extraction. These systems have recently received significant attention and investment, with several companies developing and commercializing this technology. Additionally, access to consolidated, independent, high-quality simulations for early scoping and/or project management purposes has become increasingly useful. However, open-source software for rapid data exploration and decision-making for this emerging technology is limited. Existing tools may extensively consider geothermal system design but operate on legacy software (STOMP-GT, GEOPHIRES), require user training to run the tool (GES-CAL), and/or need to be purchased (Eavor-SuiteTM). Likewise, tools often focus on including closed-loop configurations and simulating the thermal performance of ground source heat pump systems but fail to fully assess the economic viability of various closed-loop configurations. To this end, we present GeoCLUSTER v2.0: a cloud-native, techno-economic web simulator that enables start-up developers and venture capitalists to explore the economic viability of closed-loop geothermal systems, such as capital and levelized costs. Users can explore scenarios through several methods: 1) toggling between the heat-exchanger designs, working fluids, and end-use, 2) optimizing power output and economic competitiveness by clicking on the scenario buttons and moving easy-to-use sliders, and 3) visualizing simultaneous graphics and downloading its data. In GeoCLUSTER's v2.0 release, we integrate a Slender-Body Theory (SBT) model that allows users to simulate any type of U-loop and co-axial system, significantly reduce the application's memory footprint, and address user feedback. In large part, GeoCLUSTER is also an emergent software for quickly adopting to emergent geothermal feasibility research. Over the past few years, multiple studies have been undertaken by our closed-loop geothermal working group, including a 1) general feasibility study by White et al. (2023), 2) a database of pre-calculated reservoir simulations of multiple closed-loop designs (Beckers et al., 2023), 3) a heat transfer performance study of closed-loop geothermal systems with thermally conductive enhancements (Beckers et al., 2024), and 4) an impact assessment of convection on performance of closed-loop systems (Hakes et al., 2024). In turn, we present continued systematic evaluation of the technical performance and cost-competitiveness of closed-loop geothermal systems for heat production and electricity generation.

1. INTRODUCTION

Risk averse compared to enhanced geothermal systems (EGSs) [1] but capital-intensive [2,3], closed-loop geothermal systems (CLGSs) hold the promise of clean electricity on the upwards of 300+ GW by 2050 across the United States [4,5] but also face the challenges of high drilling costs [3,6] and efficient heat production over extended operational periods. These systems are a next-generation geothermal technology whose heat-exchanger design circulates a fluid through closed wellbores drilled in the subsurface, and their commercial viability has largely depended on drilling wells at depths far deeper than those of current geothermal wells in order to reach commercial levels of power production [6]. Their next-gen counterpart, the EGS, conducts far greater heat extraction than CLGSs by requiring hydraulic fracturing to create pathways for the fluid to directly penetrate the reservoir, but closed loop projects are afforded the benefit of avoiding risks from hydraulic fracturing, such as induced seismicity, reservoir sustainability, soil contamination, and water drawback [3,7]. To then bridge the "heat gap", technical efforts have been underway to further enhance the heat extraction and reduce the costs of CLGSs by, for example, repurposing abandoned oil wells and adding insulation like polyurethane foam [2]. In turn, identifying how to close these gaps for optimized heat extraction and economic feasibility of CLGSs can add further value for both current and future closed loop projects who aim to commercialize CLGSs.

Efforts to increase investments towards commercializing CLGSs can then significantly benefit from decision support tools that easily enable early scoping on the performance of different system designs. This is becoming more relevant as geothermal tools continue to integrate sensitivity studies and model increasingly complex and wider parameter spaces that compute vast simulations in support of high-quality analyses (Table 1). Examples include software like STOMP-GT that extensively considers geothermal system design to simulate heat and mass transport [3]. Independent tools, such as GES-CAL also exist to evaluate the design of shallow geothermal energy systems with 24 economic and environmental input parameters that users can define [8]. Additionally, there are cloud-native tools such as Eavor-Suite™ that offer advanced builder capabilities for users to simulate their case studies but need to be purchased. Simulators alike have become increasingly useful for evaluating the techno-economic feasibility of next next-generation geothermal technologies, in particular easy-to-access online tools with high quality data and a low memory footprint enables further interoperability and scoping. In this paper, we then introduce GeoCLUSTER v2.0 (Figure 1-3): a cloud-native, techno-economic web simulator that enables start-up developers and venture capitalists to explore the economic viability of closed-loop geothermal systems. In its latest release, GeoCLUSTER has been updated to support a parameter space of 28 user editable parameters, has significantly dropped its memory footprint, and offers fast new on-the-fly computed case studies that can reach in the quadrillions.

Table 1: Sensitivity parameters of closed-loop geothermal systems (CLGSs) by publication. Table was recreated from publications with more than five sensitivity parameters in Table 6 of the literature view conducted by Budiono et al. 2022. Modifications to the table include adding GeoCLUSTER v1.0 and v2.0 as rows and adding new heat transfer and economic parameters added as columns. There are a total of 23 sensitivity parameters considered and compared.

	Heat Transfer Performance Parameters and Sensitivity Variables																	**Economic Parameters**					
Reference	**T**	**Qr**	**P**	**Pth**	**H**	**q**	**t**	**Tgrd**	**Kins**	**Pin**	**Tin**	**d**	**F**	**Lhor**	**Kres**	**Ws**	**HE**	**EU**	**D**	**DR**	**L**	**PCt**	**PCe**
Song et al.	x			x	x	x	x				x												
Sun et al.	x	x	x			x				x	x												
Zhang et al.	x			x		x	x		x				x										
Sun et al.	x		x			x	x		x					x									
Yu et al.	x	x	x		x	x				x													
Sun et al.	x	x	x		x	x				x													
Yuan et al.	x			x			x				x				x	x							
Zhang et al.	x	x				x	x							x	x								
Wang et al.	x	x				x	x					x	x	x									
Sun et al.	x	x	x		x	x		x	x	x			x	x									
GeoCLUSTER v1.0					x	x	x	x			x	x	x	x	x		x	x	x	x	x	x	x
GeoCLUSTER v2.0					x	x	x	x			x	x	x	x	x		x	x	x	x	x	x	x

Note: T—Temperature distribution; Qr—Heat extraction rate; P—Pressure/Pressure drop; Pth—Thermal power; H—Wellbore depth; q—Flowrate; t—Production time; Tgrd—Reservoir gradient temperature; Kins—Conductivity coefficient of the insulation; Pin—Inlet pressure; Tin—Inlet temperature; d—Wellbore diameter; F—Fluid type; Lhor—Horizontal length; Kres—Conductivity coefficient of the reservoir; Ws—Wellbore spacing (multi-lateral type CLGSs only); HE—Heat-exchanger; EU—End-use; DC—Drilling cost; DR—Discount rate; L—Lifetime; PCt—Plant CAPEX ($/kWt); PCe—Plant CAPEX ($/kWe).

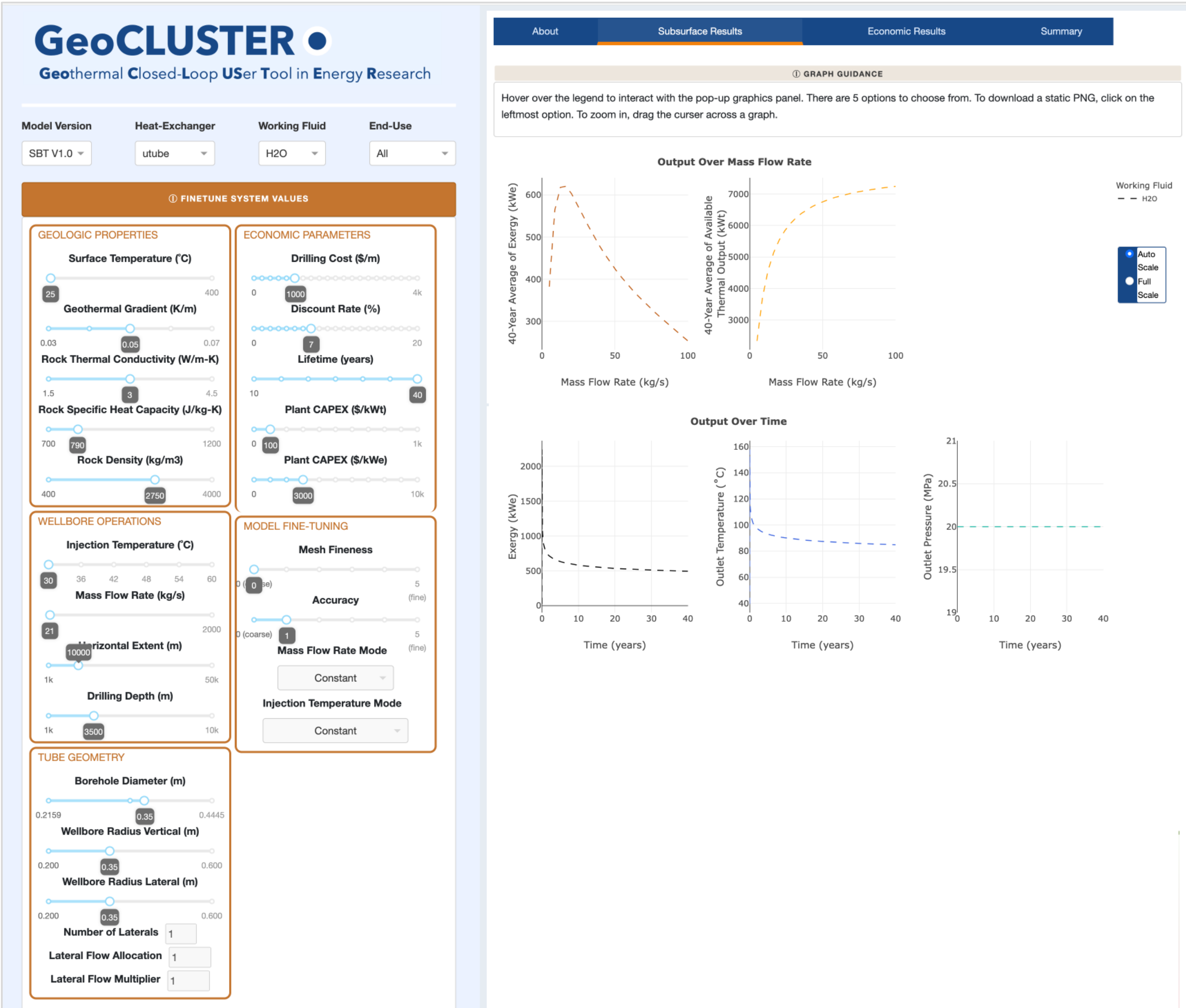

Figure 1: GeoCLUSTER v2.0 displaying SBT parameter space and its subsurface results.

Figure 2: GeoCLUSTER displaying HDF5 subsurface contours.

Figure 2: GeoCLUSTER displaying HDF5 techno-economic results.

2. UPGRADES

GeoCLUSTER is a user-friendly, closed-loop, techno-economic simulator hosted on Amazon Web Services (AWS) and publicly accessible via the web (url: https://geocluster.labworks.org/) or on GitHub (codebase: https://github.com/pnnl/GeoCLUSTER). The tool is intended to enable users to rapidly explore numerous techno-economic simulations for closed-loop geothermal systems. GeoCLUSTER v1.0 was originally developed by a team of multiple national laboratories (i.e., Closed-Loop Geothermal Working Group) with their contributions listed in [citation 1][citation 2][citation etc.]; likewise, GeoCLUSTER has now been enhanced in part by the Pacific Northwest National Laboratory to deploy a more interoperable, light-weight, cost effective, and energy efficient 2.0 version. GeoCLUSTER v2.0 also serves a Slender-Body Theory (SBT) model developed by the National Renewable Energy Laboratory (NREL) and Sandia National Laboratory (Sandia). Between GeoCLUSTER v1.0 and v2.0, major differences and similarities are documented in Table 2 and further described below.

Table 2: Parameter metrics and their ranges between versions of GeoCLUSTER.

Parameter	**GeoCLUSTER v1.0**	**GeoCLUSTER v2.0 (SBT Integrated)**
Heat-Exchanger Design	U-Loop, Co-axial	U-Loop, Co-axial
Working Fluid	H_2O, sCO_2	H_2O
End-Use	Heating, Electricity	Heating, Electricity
Mass Flow Rate	5 kg/s to 100 kg/s	1 kg/s to 2000 kg/s
Horizontal Extent	1,000 m to 20,000 m	1,000 m to 50,000 m
Drilling Depth	1,000 m to 5,000 m	1,000 m to 10,000 m
Geothermal Gradient	0.03 K/m to 0.07 K/m	0.01 K/m to 0.09 K/m
Borehole Diameter	0.2159 m to 0.4445 m	0. 2000 m to 0.6000 m
Injection Temperature	30°C to 60°C	25°C to 400°C
Injection Pressure	200 bar (**fixed parameter**)	200 bar (**fixed parameter**)
Rock Thermal Conductivity	1.5 W/m-K to 4.5 W/m-K	0.1 W/m-K to 7 W/m-K
Rock Specific Heat Capacity	790 J/kg-K (**fixed parameter**)	790 J/kg-K to 1,200 J/kg-K
Rock Density	2,750 kg/m^3 (**fixed parameter**)	400 kg/m^3 to 4,000 kg/m^3
System lifetime	10 years to 40 years	10 years to 40 years
Drilling Cost	0 $/m to 4,000 $/m	0 $/m to 4,000 $/m
O&M Cost Plant as Percentage of Capital Cost	1.5% (**fixed parameter**)	1.5% (**fixed parameter**)
Discount Rate	0 % to 20 %	0 % to 20 %
Direct Use Heat Plant CAPEX	0 $/kWt to 1,000 $/kWt	0 $/kWt to 1,000 $/kWt
Power Plant CAPEX (for electricity generation)	0 $/kWe to 10,000 $/kWe	0 $/kWe to 10,000 $/kWe
Pre-cooling	0 °C to 40°C	0 °C to 40°C
Turbine Outlet Pressure	75 bar to 200 bar	75 bar to 200 bar
Dead-State Temperature	20°C (**fixed parameter**)	20°C (**fixed parameter**)
Dead-State Pressure	1 bar (**fixed parameter**)	1 bar (**fixed parameter**)
Turbine Isentropic Efficiency (for sCO_2 electricity)	90% (**fixed parameter**)	90% (**fixed parameter**)
Generator Efficiency (for sCO_2 electricity)	98% (**fixed parameter**)	98% (**fixed parameter**)
Compressor Isentropic Efficiency (for sCO_2 electricity)	90% (**fixed parameter**)	90% (**fixed parameter**)
Turbine Outlet Pressure (for sCO_2 electricity)	79 bar (**fixed parameter**)	79 bar (**fixed parameter**)
Pre-cooling Temperature Decline (for sCO_2 electricity)	5°C (**fixed parameter**)	5°C (**fixed parameter**)

Table 3: Setup, performance, and parameter metrics between versions of GeoCLUSTER.

	GeoCLUSTER	
Setup	**Version 1.0 (Cite Paper)**	**Version 2.0**
Web Framework	Dash-Plotly	Dash-Plotly
Python Version	3.8	3.11
	GeoCLUSTER	
Performance	**Version 1.0 (Cite Paper)**	**Version 2.0**
Memory (Data File)	6500 MB (HDF5 ~90 MB compressed)	200 MB

EC2 Instance (Memory; Cost)	r6i.large (16 GB; $90; Memory Optimized; 2 vCPUs)	t3.small (2 GB; $15; Burstable Performance; 4 CPUs)
Number of Simulations	631,800 simulations per each of the four combinations of heat-exchanger / fluid type, totaling to over 2.5 million simulation runs (if all results were precomputed, it would require approximately 500 petabytes).	Approximately an additional 19 quadrillion simulations can be generated on the fly.

2.1 Back End Optimizations

2.1.1 Cloud Architecture

Between GeoCLUSTER v1.0 and v2.0, there was no change in cloud architecture. Figure 1 shows a minimal setup on Amazon Web Services where the Apache webserver, Dash-Plotly web framework, and data files are all on the same Elastic Container Compute (EC2) instance, with traffic distributed by an Elastic Load Balancer (ELB). In this architecture, the webserver does not just serve GeoCLUSTER, but it is also involved in processing requests, making it responsible for both the view (the display of the data) and the controller (the processing behind requests). The HDF5 file serves as the model (storage of data)

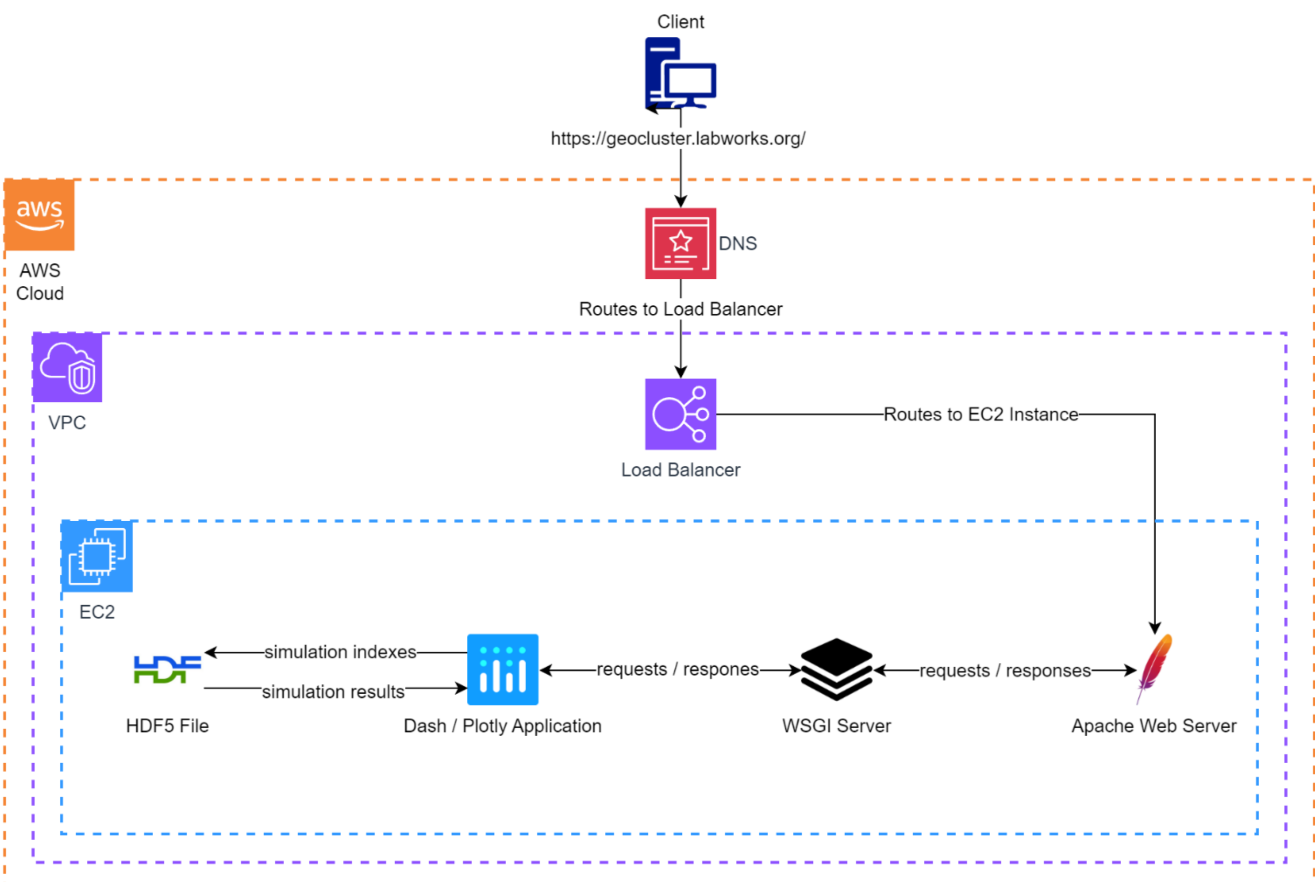

Figure 3: Simple AWS architecture diagram of GeoCLUSTER deployed on AWS Cloud.

2.1.2 Memory Optimizations

The previous version of GeoCLUSTER is light on CPU-load but required 6.5 GB of RAM. As such, backend optimizations to reduce the memory used were required to make GeoCLUSTER more energy and cost efficient. Due to project constraints, the application was limited to being hosted on EC2 instance (described above) and optionally a MySQL database. So, optimization efforts were focused on reducing the number of computational resources the hosting instance needed for the application instead of towards moving towards a traditional split model-view-controller paradigm.

Some explanation of the working of GeoCLUSTER is a prerequisite to explaining the memory optimizations. GeoCLUSTER contains a subsurface model (i.e., clgs.py) which calculates the output of a CLGS, and an economic model (i.e., clg_tea_module.py), which uses the subsurface results to calculate the economic performance of that CLGS. The subsurface model leverages Beckers et al. 2023 dataset, which contains the output heat, and pressure over time of a CLGS at 631,800 different CLGS for each of the 4 combinations of working fluid working fluid (H2O or sCO2) and tube shape (coaxial or utube). This dataset was generated by simulating a CLGS over the cartesian production of 7 different parameters describing a CLGS. At any combination of these parameters, the dataset contains the output pressure and temperature of the CLGS at that configuration over 161 points in time, representing 40 years of output for a CLGS. This data is stored as an 8D matrix where the first 7 dimensions are the CLGS configuration parameters, and the 8th dimension is the output over time for either the heat or the pressure. The dataset contains 8 of these 8D matrices: one for the output pressure and one for the output temperature for each of the 4 possible combinations of working fluids and tube shapes.

While these 8D matrices take only 90MB of disk space when compressed into an HDF5 thanks to Becker's single value decomposition compression, they take much more when decompressed and read into memory. The 8D matrices were stored in float 64s at runtime, so each 8D matrix required $(631{,}800 * 161)\text{datapoints} * 8\,\frac{bytes}{datapoint} \approx 0.81\text{GB}$. Since the application uses 8 of these matrices, GeoCLUSTER the application requires 8 times this amount, about 6.5GB.

The 8D matrices are used in 2 ways. First is to look up the output over time for a given set of input parameters, as shown in Figure 4. This is equivalent to slicing the matrix by holding the first 7 dimensions constant at the input parameters and fetching the output over time.

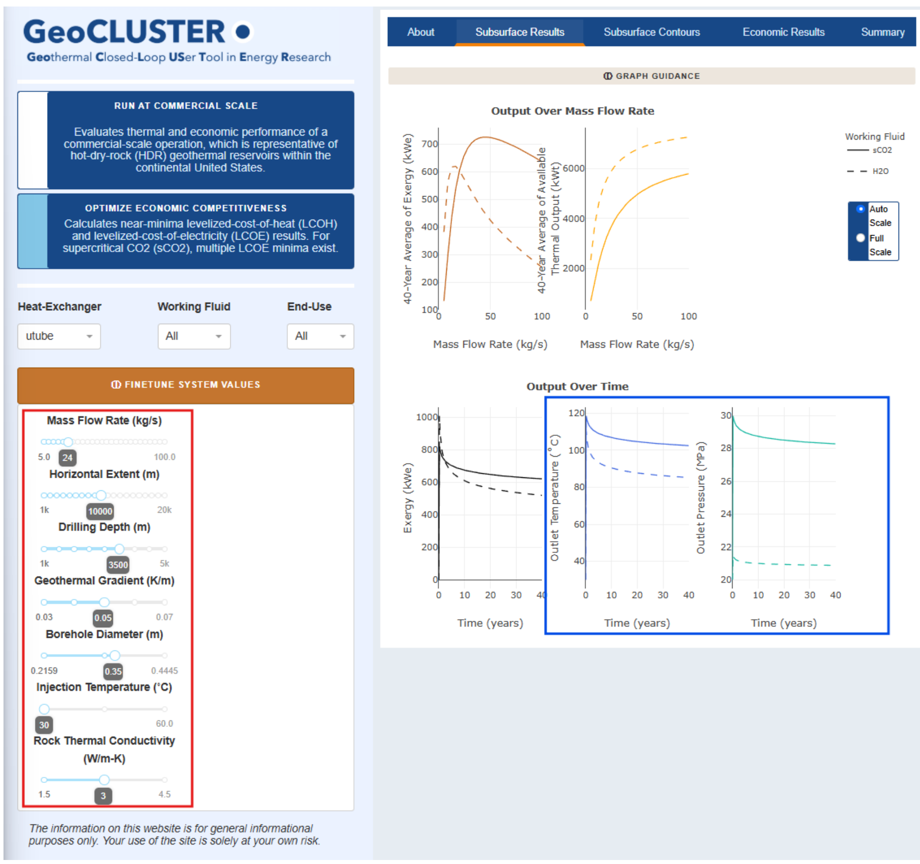

Figure 4. Input parameters in red and output temperature and pressure over time in blue

Secondly, we generate the subsurface contours, which shows how the outlet state changes with to the cartesian product of mass flow rate and a second parameter (any one of the 6 other input parameters), at a CLGS configuration defined by the remaining 5 parameters. This is essentially slicing the matrix by mass flow rate and the second parameter, holding the 5 other input parameters and time constant. In both cases, only a small subset of the data of the entire matrix is needed at a given moment.

Figure 5. Subsurface contours with parameter second parameter in green, variation of outlet state with respect to mass flow rate and second parameter in red, and parameters held constant in blue.

However, the access patterns are complicated by the use of interpolation to approximate the outlet states of the CLGS when the input parameters are not equal to their precomputed values. For example, in figure 4, the borehole diameter is 0.35m, which is not one of the precomputed values for it. In this case, GeoCLUSTER cannot simply slice into the matrix where the bore hold diameter is equal to 0.35m, as that data point does not exist. Instead, the application uses multidimensional linear interpolation to approximate the output value at that point. GeoCLUSTER treats the 8D matrix like an 8D grid, where the location of the point is the parameters of the CLGS configuration, and the value at that point is the outlet state for that configuration. When a given CLGS configuration isn't precomputed, it falls "between" the points on the grid, and the application uses linear interpolation and the precomputed outputs at points near the given point to approximate the output at the given point.

To remove this large memory burden, a way was needed to offload the storage of these matrices from directly into memory to some database. One considered option was to simply avoid the challenge of interpolation entirely and pre-interpolate the outlet states at a every possible value in GeoCLUSTER, save the output to S3 (Amazon's Simple Storage Service), and fetch the simulation output at the given parameters. However, due to the curse of dimensionality, this would take an astronomical amount of space. In order to have the same degree of granularity in a pre-interpolated matrix as the granularity offered with interpolation on GeoCLUSTER, the shape would need to change from (26, 20, 9, 5, 3, 3, 3, 161) to (190, 190, 80, 79, 114, 150, 150, 161). This would require $190 * 190 * 80 * 79 * 114 * 150 * 150 * 161 \approx 9.42 * 10^{16}$ datapoints, which would require approximately 377 petabytes if they were stored as float32s. Before even constructing an access pattern for this data, the raw amount of storage needed alone disqualified this approach.

Instead, focus was turned to offloading the data from memory and then only fetching the points the application needed to approximate the output of a given CLGS configuration. In the same way only the 2 sample points around a given point are needed to linearly approximate in 1D, and only the corners of the square around a point are needed to bilineraly approximate in 2D, only the corners of the surrounding 8-cube in the 8D matrix are needed to interpolate a point on an 8D grid. Code was written to compute the indices of the corners of the given points to interpolate the outlet state for. Then, the outlet states at these points could be fetched from a database and fed into scipy's internpn function, which would perform the linear interpolation.

To ease integration with the pre-existing codebase, the database chosen was an 8D hdf5 dataset in the same format as the 8D numpy matrix in memory, but the key difference is that this new dataset is chunked. Instead of the data being stored in one contiguous file on disk that's then all read into memory, it's stored as many different individual subsets of the data, chunks, which can be read into memory independent of each other. When the application slices into this chunked matrix, it loads only the chunks containing the slice into memory and reads the points requested in the slice from those chunks. Given our 2 access patterns, a chunk size of (26, 1, 1, 1, 1, 1, 1, 161) was chosen as a balance between reducing the amount of in unnecessary data per chunk, and while still minimizing the number of chunks needed per lookup. Another difference between the 2 versions is that the new dataset is not stored using singular value decomposition, but instead the 8D matrix is stored as-is to disk, increasing the disk size needed to 3.2GB. This was done to minimize latency in the application, as GeoCLUSTER would need to decompress the data every time it read a chunk from disk.

This chunked HDF5 dataset on the disk of the EC2 Instance acts as the database for GeoCLUSTER and meets all the demands of the application. It supports concurrent reads (meaning that two threads can read from the dataset as the same time when the website is serving multiple users simultaneously) and Partial I/O, which allows us to only read a portion of the dataset into memory at a time, instead of the entire dataset. Since the application only reads the dataset and never writes to it, a traditional database is not needed. A prototype version of the application using MySQL as a database instead was tested, but the memory overhead and performance were comparable between the two versions, and the MySQL version unnecessarily increased the complexity the application with no added benefit.

In all, the output of a CLGS configuration at a point in time is treated as point on an 8D grid. When that configuration is not precomputed, the point is essentially "between" the grid lines. The application finds the corners of the 8-cube that contains the point, fetches the output at those corners from a database, and the linearly interpolates on them to approximate the output at the given point. With this strategy of offloading the 8D matrixes from memory onto disk, the RAM needed for the application was reduced by roughly 97%, from 6.5GB to 200MB.

Table 4: Available output datasets, [hx] = "utube" or coaxial", [fluid] = "H2O" or "sCO2". Note the left, right singular vectors correspond to a rank k approximation. This table has been expanded from Table 3 of Beckers et al. 2023.

HDF5 v1.0 dataset path	Description	HDF5 v2.0 dataset path	Description
/[hx][fluid]/output/Tout/U	Left singular vectors for outlet temperature state	/[hx][fluid]/output/chunked_tout	Chunked temperature outlet state 8D matrix
/[hx][fluid]/output/Tout/sigma	Singular values for outlet temperature state	/[hx][fluid]/output/chunked_tout	
/[hx][fluid]/output/Tout/Vt	Right singular vectors for outlet temperature state	/[hx][fluid]/output/chunked_tout	
/[hx][fluid]/output/Pout/U	Left singular vectors for outlet pressure state	/[hx][fluid]/output/chunked_pout	Chunked pressure outlet state 8D matrix
/[hx][fluid]/output/Pout/sigma	Singular values for outlet pressure state	/[hx][fluid]/output/chunked_pout	
/[hx][fluid]/output/Pout/Vt	Right singular vectors for outlet pressure state	/[hx][fluid]/output/chunked_pout	
/[hx][fluid]/output/We	Available work over forty years, units [GWhr]	/[hx][fluid]/output/We	Available work over forty years, units [GWhr]
/[hx][fluid]/output/Wt	Heat output over forty years, units [GWhr]	/[hx][fluid]/output/Wt	Heat output over forty years, units [GWhr]

2.1 Web/Front End Optimizations

A detailed memory and object data analysis of GeoCLUSTER was made using Python's memory_profiler and guppy3 utility libraries. The analysis focused on two primary areas: Numpy objects used for model calculations, and Plotly graph objects associated with the Dash components rendered on the front-end web page. The analysis revealed that the data class gets instantiated 4 times on app start up accumulating 10MB of memory in the heap each time. Each time new parameters would be passed, and outputs required interpolation, it involved manipulating Numpy array and matrix objects causing additional memory spikes ranging from 1-12MB. This was then outputted through the generation and rendering of new, or existing graphs. Consequently, Dash components were revealed to cause significant memory spikes ranging from 1-15MB when rendering these graphs.

Further heap analysis using the guppy3 utility revealed that objects of the numpy.ndarray type, while few in number (159 instances), accounted for the largest portion of memory. 40MB of the total heap storage while GeoCLUSTER was running was from storing Numpy objects in memory. This was primarily caused by GeoCLUSTER's data class which is responsible for parsing the data from the HDF5 file (Table 4). String objects and tuples accounted for 26MB and 23MB in the heap memory respectively.

In exploring solutions to optimize memory usage, Pyodide presented interesting possibilities. Pyodide enables Python code to be run the web browser by compiling CPython to WebAssembly, effectively allowing computational tasks to be offloaded from the GeoCLUSTER server to the end user's internet browser. This decentralization was thought to bring significant reductions to the server's memory load by utilizing the user's local resources for storing data and running memory intensive computations.

2.2.1 Slender-Body Theory (SBT) Model

We integrated the Slender-Body Theory (SBT) model into GeoCluster to allow simulating closed-loop geothermal scenarios and configurations that were not originally included in the pre-calculated database [9]. For example, with the SBT model, designs can be simulated for depths deeper than 5 km, geothermal gradients larger than 70°C/km and with number of laterals larger than 1, which were originally upper limits considered for the respective parameters when generating the database. The SBT model combines Green's functions with a one-dimensional discretization along a closed-loop heat exchanger for computationally-fast simulations of heat transfer with closed-loop systems in conduction-only reservoirs. The model was originally developed by Beckers et al. (2015) [10] for simulating shallow geothermal heat exchangers, and was updated by Beckers et al. (2022) [11] for simulating deep co-axial and U-loop closed loop systems. The model requires constant rock properties but can dynamically update the heat transfer fluid properties over time and along the heat exchanger as a function of temperature and pressure. Thermal interference between laterals is captured and variable flow rate and injection temperature can be simulated. Recently, Aljubran et al. (2024) simulated dispatchable geothermal power production with closed-loop systems using the SBT model by varying injection temperature and flow rate on an hourly time step over a 20-year lifetime. For integrating into GeoCLUSTER, we converted the SBT model code from MATLAB to Python.

4. ACKNOWLEDGEMENTS

This work was authored in part by the Pacific Northwest National Laboratory, operated by Battelle Memorial Institute, for the U.S. Department of Energy (DOE) under contract GT0100000-05450-1005612 (2022) and the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, under contract DE-AC36-08GO28308. Funding was provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Geothermal Technologies Office. The views expressed herein do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

REFERENCES

[1] Budiono, A., Suyitno, S., Rosyadi, I., Faishal, A., and Ilyas, A.X.: A Systematic Review of the Design and Heat Transfer Performance of Enhanced Closed-Loop Geothermal Systems, Energies, 15, (2022), 742. https://doi.org/10.3390/en15030742.

[2] Liu, S., Taleghani, A., and Ji, K.: An Advanced Closed-Loop Geothermal System to Substantially Enhance Heat Production, Energy Conversion and Management, 322, (2024), 119168. https://doi.org/10.1016/j.enconman.2024.119168.

[3] White, M., et al.: Numerical Investigation of Closed-Loop Geothermal Systems in Deep Geothermal Reservoirs, Geothermics, 116, (2023), 102852. https://doi.org/10.1016/j.geothermics.2023.102852.

[4 Blankenship, D., Gertler, C., Kamaludeen, M., O'Conner, M., and Porse, S.: Pathways to Commercial Liftoff: Next-Generation Geothermal Power, Department of Energy, (2023). https://liftoff.energy.gov/next-generation-geothermal-power/.

[5] Idaho National Laboratory: Impact of Enhanced Geothermal Systems (EGS) on the United States in the 21st Century, MIT Energy Initiative, (2006). https://energy.mit.edu/research/future-geothermal-energy/.

[6] Fervo Energy: Fervo Energy Breaks Ground on the World's Largest Next-Gen Geothermal Project, Fervo Energy, (2023), 27 Nov. https://fervoenergy.com/fervo-energy-breaks-ground-on-the-worlds-largest-next-gen-geothermal-project/.

[7] Alimonti, C., Soldo, E., Bocchetti, D., and Berardi, D.: The Wellbore Heat Exchangers: A Technical Review, Renewable Energy, 123, (2018), 353. https://doi.org/10.1016/j.renene.2018.02.055.

[8] Bláquez,C. S., et al.: GES-CAL: A new computer program for the design of closed-loop geothermal energy systems, Geothermics, 87 (2020), 101852. https://doi.org/10.1016/j.geothermics.2020.101852.

[9] Beckers, K., Vasyliv, Y., Bran-Anleu, G.A., Martinez, M., Augustine, C. and White, M., 2023. Tabulated Database of Closed-Loop Geothermal Systems Performance for Cloud-Based Technical and Economic Modeling of Heat Production and Electricity Generation Proceedings, 48th Stanford Geothermal Workshop, Palo Alto, CA (2023).

[10] Beckers, K. F., Koch, D. L., and Tester, J. W., 2015. "Slender-body theory for transient heat conduction: theoretical basis, numerical implementation and case studies." Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 471(2184), p. 20150494.

[11] Beckers, K. F., Rangel-Jurado, N., Chandrasekar, H., Hawkins, A. J., Fulton, P. M., and Tester, J. W., 2022. "Techno-economic performance of closed-loop geothermal systems for heat production and electricity generation." Geothermics 100, p. 102318.