Nick Smith

Provo, UT | (385) 285-4099 | nicholas.smith0014@gmail.com
LinkedIn | GitHub | ORCID: 0009-0009-8621-6710

SUMMARY

Highly motivated Master's in Computer Science student and published researcher, specializing in Swarm Intelligence, Multi-Agent Systems, and Stochastic Modeling. Advanced Python developer with experience building a modular simulation framework and optimizing performance. Co-author on swarm intelligence research published in Philosophical Transactions of the Royal Society A, focused on performance prediction of hub-based swarms. Strong foundation in linear algebra, calculus, probability, and graph theory, with experience modeling complex systems. Interested in controls, robotics, autonomous systems, and simulation.

EDUCATION

Brigham Young University, Provo, UT | MS in Computer Science | *Expected Apr 2028*
Focus: Swarm Robotics & Multi-Agent Systems
Relevant Coursework: Robotic Localization & Mapping, SLAM, CUDA, Verification & Validation.

Worcester Polytechnic Institute, Worcester, MA | Graduate Certificate in Robotics Engineering (Online) | *Expected Apr 2028*
Status: Accepted; Resuming in Jan 2027

Brigham Young University, Provo, UT | BS in Computer Science: Data Science | *April 2025*
Minors: Math & Statistics
Relevant Coursework: Artificial Intelligence, Reinforcement Learning, Deep Learning, Natural Language Processing (NLP), Machine Learning, Network Science, Linear Programming, Algorithms, Data Structures, Computer Vision, Database Modeling.

TECHNICAL SKILLS

AI & ML: Deep Learning, Reinforcement Learning, Machine Learning, Graph Neural Networks (GNNs), Grammatical Evolution, Multi-Agent Systems, Computer Vision, State Estimation, Kalman Filtering, EKF-SLAM, Factor Graph SLAM (In Progress).
Mathematics: Multivariable Calculus, Linear Algebra, Probability & Statistics, Bayesian Statistics, Regression, Ordinary Differential Equations (ODEs), Linear Programming, Graph Theory.
Systems & Formal Methods: Formal Verification (SPIN/Promela), Model Checking, Systems Programming, Linux/Unix Systems, Debugging.
Programming Languages: Python (Advanced), C/C++, Java, R, SQL, Visual Basic for Applications (VBA).
Tools & Frameworks: CUDA, PyTorch, OpenCV, ROS2 (Some Exposure), OpenMP, NumPy, Pandas, Matplotlib, SciPy, Jupyter Notebooks, Git, GitHub, Microsoft Azure Functions, Microsoft Power Platform, Excel.

RESEARCH EXPERIENCE

BYU Computer Science, Provo, UT | Undergraduate Research Assistant | *May 2023 - April 2025*
Human-Centered Machine Intelligence Lab

Swarm Dynamics/Distributed Intelligence:

- **Multi-Agent Simulation:** Implemented several foundational models of swarm behavior, including Reynolds' Boids (vector-based steering) and Couzin's Three-Zone Model, to serve as benchmarks and to gain familiarity with Swarm Dynamics.
- **Stochastic Modeling:** Independently modeled a multi-agent Best-of-N problem as an **Absorbing Markov Chain**. Extracted transition probabilities (for the collective state of agents) using **Monte Carlo methods** to predict convergence to optimal or sub-optimal sites.
- **State-Space Analysis:** Leveraged **Strongly Connected Component (SCC) Analysis** to evaluate predictive limits of the state representation; found ~80% of states produced convergence probabilities in a narrow band around the global baseline (~20% optimal), indicating minimal predictive signal.

- **Theoretical Analysis & Embeddings:** Proved theoretical links between GCNs and Markov processes to validate them as tools for creating low-dimensional embeddings of states, while preserving underlying transition dynamics.
- ***Publication:*** Co-author of Puneet Jain et al. 'Performance Prediction of Hub-Based Swarms,' *Philosophical Transactions of the Royal Society A* (2025). [DOI: 10.1098/rsta.2024.0141]

Grammatical Evolution Framework:

- **Systems Engineering:** Engineered a modular **Discrete-Time Agent-Based Modeling (ABM)** simulation engine and framework for Grammatical Evolution in Python; utilized a **Self-Evaluating Abstract Syntax Tree (AST)** architecture and stack-based program execution to decouple experiment design from complex grammar parsing and interpretation logic; integrated a **transactional rollback system** to detect cycles and revert invalid state changes during crossover and node mutation, thus maintaining the structural integrity of resulting programs.
- **Performance Optimization:** Achieved a ~**10x speedup** in program generation by reducing redundant initialization overhead across millions of nodes; implemented efficient sparse-matrix encoding, state-change tracking, and on-the-fly reconstruction for creating memory-efficient animations.
- **Research Utility:** Eliminated boilerplate code and streamlined the end-to-end pipeline for world/agent definition, simulation setup, and visualization, reducing new experiment setup time by weeks.
- ***Link here:*** *https://github.com/n-smith-byu/GrammaticalEvolutionTools*

WORK EXPERIENCE

BYU Computer Science, Provo, UT | ***Teaching Assistant (CS 470 - Intro to Artificial Intelligence)*** | *Sep 2026 - Present*

- **Student Mentorship:** Conducted 1-on-1 sessions with students to clarify complex mathematical and algorithmic concepts and support learning and understanding.
- **Technical Feedback:** Graded assignments for 60+ students, evaluating understanding of course content and providing specific, written feedback.

BYU Computer Science, Provo, UT | ***Teaching Assistant (CS 575 - Intro to Graph Data Science)*** | *Oct 2021 - Dec 2023; Feb 2026 - Apr 2026*

- **Technical Feedback:** Graded assignments for 20+ students, evaluating understanding of network properties, centrality measures, modularity, community detection, knowledge graphs, and graph machine learning (GML) methods; provided written feedback.
- **Content Engineering:** Enhanced existing project specifications and instructions to improve clarity and reduce implementation ambiguity for complex graph-based programming tasks.
- ***Material Covered:***

• ***Network Topology:*** Random Graphs, Small-World (Watts-Strogatz), Scale-Free (Barabási-Albert).
• ***Centrality Measures:*** Eigenvector, Katz, Betweenness, PageRank, Degree Centrality.
• ***Community Detection:*** Louvain, Girvan-Newman, Newman Hill-Climbing, Laplacian/Spectral Graph Cut
• ***Graph Machine Learning:*** Node2Vec, Knowledge Graphs, Graph Convolutional Neural Networks
• ***Tools:*** NetworkX, Gephi

BYU Dining Technology, Provo, UT | **Systems Developer** | *Oct 2021 - Dec 2023*

- **Technical Initiative:** Independently mastered the tools to deploy a suite of in-house applications and automated workflows for Dining Administration and Retail Restaurants, with **zero prior experience** in most of the stack (Wrike, Microsoft Power Platform, Azure); Leveraged existing expertise in Python, Excel, SQL, and Databases to deliver production-ready solutions.
- **Systems Integration:** Built automation pipelines using Wrike REST APIs, Power Automate, Azure Functions, and Excel (VBA) to streamline reporting, task assignment, and support ticket workflows; reduced ticket response times and improved cross-team communication across all of Dining Services.
- **Performance Optimization:** Resolved a critical reporting bottleneck by migrating legacy code from Microsoft Power Automate to Azure Functions. Used Python and Pandas DataFrames to reduce total processing time from **120 minutes to 2 minutes (60x speedup)**.
- **Full-Stack Development (Agile):** Built an order fulfillment and delivery app using SQL Server, Bite Order API, and Microsoft Power Apps to bypass current ERP limitations; **delivered ~90% of core functionality within a two-week sprint**, anticipating stakeholder needs in advance to accelerate deployment and reduce iteration cycles.

COURSEWORK & PROJECTS

Computer Vision: Object Detection with Convolutional Neural Networks; Edge Detection; Image Segmentation; Image Stitching using homogeneous coordinate transforms; Hough Transform; SIFT Features, SURF Features, FLANN (Feature Matching).

AI and Reinforcement Learning: Discrete Bayes' Filter (Localization), Particle Filtering, Occupancy-Grid Mapping, Potential Fields (Motion Planning), Game Theory, Minimax, Alpha-Beta Pruning, Dynamic Programming, Value and Policy Iteration, Q-learning, Approximate Value Iteration, and Deep Q-learning.

Deep Learning and NLP: Built generative models using PyTorch, including image style transfer, machine translation, and text/image generation; Implemented Transformer, Convolutional Neural Network (CNN), Graph Neural Network (GNN), Recurrent Neural Network (RNN), Diffusion Models, and other Neural Network architectures from scratch.

Networks & Graph Data Science: Modeled and simulated simple/complex contagions using SEIR models; implemented graph algorithms and analyzed common network metrics (centrality, clustering coefficients, density, modularity, etc.); constructed knowledge graphs to represent structured information; generated node embeddings using Node2Vec and Graph Convolutional Neural Networks (GCNs).

Computational Theory & Formal Languages: Built an interpreter and execution engine for the Datalog language using finite-state automata for lexical analysis, a context-free grammar for parsing, and an evaluator for queries using relational algebra.

Systems Programming: Developed a Linux-environment mini shell for I/O redirection and piping; built a multithreaded HTTP proxy server using a thread pool and another implementation using I/O multiplexing to manage concurrent requests.

PERSONAL PROJECTS

Graph Method to Create Unbiased Word Embeddings | *Nov 2023 - Dec 2023*

Final Project for Natural Language Processing. Created a graph mapping words to the words in their dictionary definitions. Used a variant of Node2Vec to generate embeddings on the graph. Showed some promise in creating unbiased word embeddings, with respect to gender, when compared to Google's pre-trained Word2Vec embeddings.

Modeling Photosynthesis Board Game with Linear Programming | *Mar 2022 - Apr 2022*

Final Project for Linear Programming. Demonstrated how the rules of Photosynthesis (BlueOrange Games) can be represented as a set of Linear Constraints and an optimization problem set up to solve for the set of best moves given some heuristic function. Modeled the game as a resource flow graph, with edges representing actions, states representing game pieces or spaces on the board, and constraints on the resource flow across edges representing the cost of actions and the rules of the game.

PUBLICATIONS

Puneet Jain, Chaitanya Dwivedi, Nicholas Smith, Michael A. Goodrich; **Performance prediction of hub-based swarms**. Philos Trans A Math Phys Eng Sci 30 January 2025; 383 (2289): 20240141. https://doi.org/10.1098/rsta.2024.0141

AWARDS & HONORS

The Honor Society of Phi Kappa Phi, Membership | *March 2022*

OTHER SKILLS

Languages: Russian (A2), ASL (Elementary)