CSCI 191T
Project Presentations

Presented By:

Edgar Ernesto Rodriguez Pablo

Ryan Vu Hang

Saishnu Ramesh Kumar

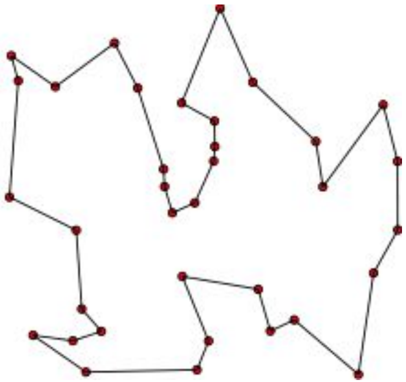
Overview

- **Project 1:** Ant Colony Optimization & Heuristic for the TSP Problem with 50 cities
- **Project 2:** Gradient Descent Minimization For Rastrigin Function & Simulated Annealing
- **Project 3:** Play And Its Role In The Mental Development Of The Child And The Zone Of Proximal Development (by L. S. Vygotsky)

Project 1:
Ant Colony Optimization & Heuristic For The TSP Problem

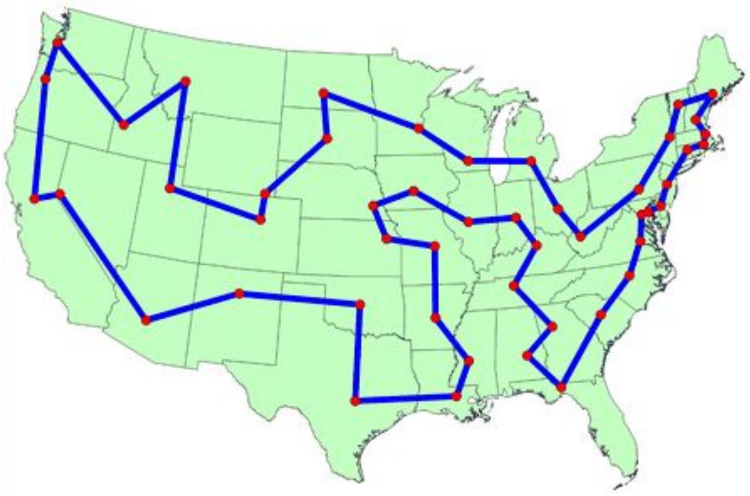
Introduction

- The Traveling Salesman Problem (TSP) is a dilemma that Computer Scientists have faced due to how simple it is to solve but how difficult it is to optimize.
- The problem can be stated as finding the shortest possible route with the constraint of only visiting a city (node) exactly once and returning back to the original city with a given list of cities and distances (edges) between each pair of cities.



Motivation

- Identify efficient solutions to the TSP which has significant real world applications. E.g: Transportation and Logistics.
- Explore the effectiveness of a metaheuristic algorithm.
- Compare other optimization techniques and evaluate its performance and computational efficiency.

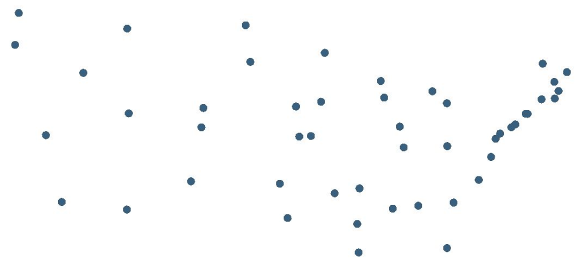


Problem Statement

- How can we implement the Ant Colony Optimization (ACO)/ Ant System (AS) with the Nearest Neighbor (NN) algorithm for solving the TSP with 50 cities.
- Among the two algorithms, which results in better performance, optimization, and efficiency.

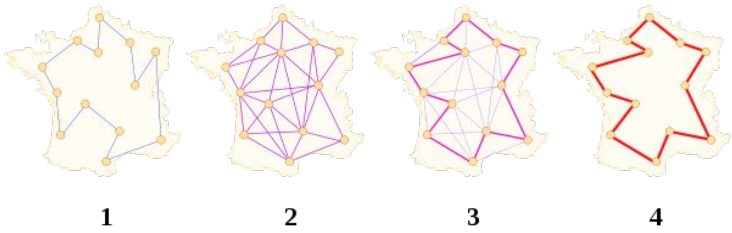
Nearest Neighbors

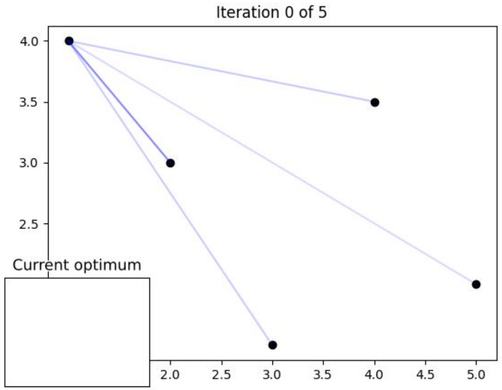
- Nearest Neighbor (NN) is a heuristic algorithm for solving the TSP by greedily searching for the closest city from the current node.
- The algorithm starts with an arbitrary city and iteratively selects the closest unvisited city as the next destination.
- This process continues until all cities have been visited, forming a tour.



Ant System

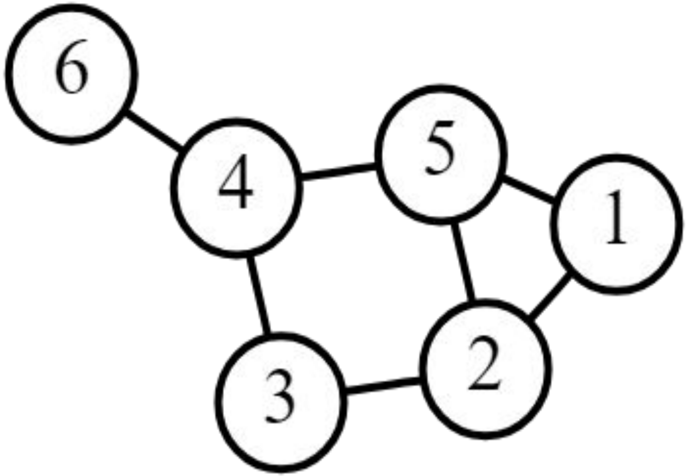
- In Ant System (AS), artificial ants construct solutions by iteratively building paths through cities, depositing pheromone along the edges they traverse.
- Pheromone evaporation and reinforcement mechanisms are used to update the pheromone levels on the edges.
- Ants make probabilistic decisions based on pheromone levels and heuristic information to guide their search for optimal solutions.
- AS iteratively improves solutions over multiple iterations, converging towards an optimal or near-optimal solution for the TSP.





Approach

- Compare AS with NN involves creating a simulation of 50 cities.
- Both algorithms are allowed to compute their shortest route within this simulation.
- The simulation and environment are created using a Graph Data structure.
- Cities within the simulation are represented as nodes, and their respective paths are represented as edges.
- The weight of each edge is based on the distance between the two cities.



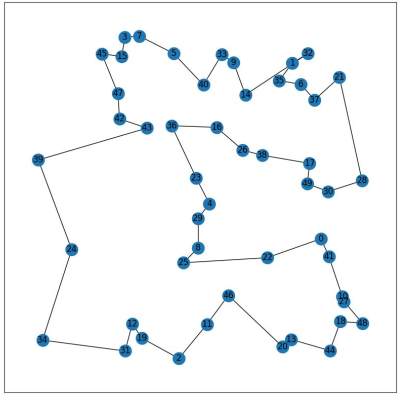
Experiment

- The configuration used for both algorithms involves generating the weights of the edges based on the distance of the cities' randomly generated coordinates.
- The coordinate range for the city is (0,500).
- For the Ant System algorithm:
    - The number of ants used is 50.
    - The iteration count is set to 500.
    - The pheromone evaporation rate used is 0.2.
    - The influence factors used are alpha = 1 and beta = 2.

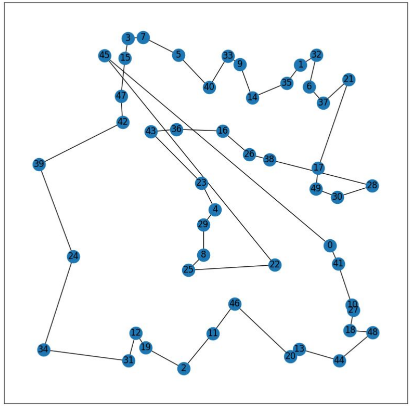
$$p_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum_{l \epsilon J_i^k}[\tau_{il}(t)]^{\alpha} \cdot [\eta_{il}]^{\beta}}$$

$$\tau_{xy} \leftarrow (1-\rho)\tau_{xy} + \sum_{k} \Delta\tau_{xy}^{k}$$

Results



ACO - Total Distance: 5908.12



NN - Total Distance: 7307.37

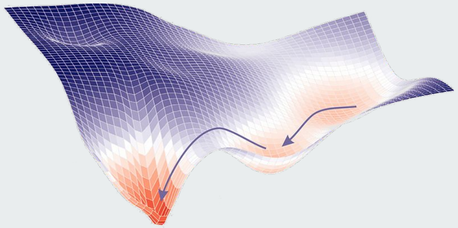
Results Cont.

- Ant System (AS) generally outperforms Nearest Neighbor (NN) in terms of total distance.
- AS Strengths:
  - Global perspective and adaptability lead to more effective exploration of diverse solutions compared to NN.
- NN Strengths:
  - Computationally less intensive due to its simple iterative nature.
- Limitations:
  - NN: Greedy nature can lead to missed cities and overlap in the path.
  - AS: Iterative nature may prolong computation of optimal solutions.

Conclusion

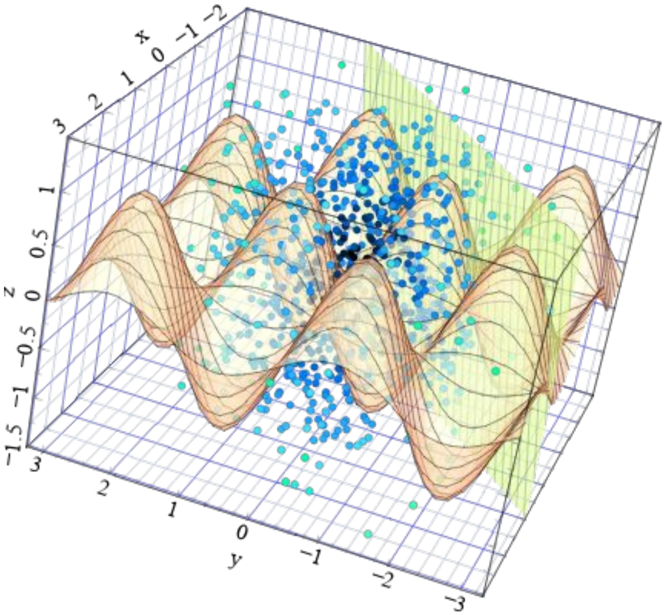
- This study contributes to the understanding of the strengths and weaknesses of NN and ACO algorithms in solving combinatorial optimization problems such as the TSP.
- NN is suitable for solving basic optimization problems.
- AS is more suited for complex combinatorial optimization problems.
- The findings provide valuable insights for researchers and practitioners in selecting suitable algorithms for similar optimization tasks based on specific requirements and constraints.

Project 2:
Gradient Descent Minimization For Rastrigin Function & Simulated Annealing



Motivation

- Implementing Gradient Descent and Simulated Annealing to comprehend and compare the two optimization techniques.
- Identify which algorithm works better in this scenario.
- Optimization is crucial for application in reworld application(used in AI, engineering, computation)

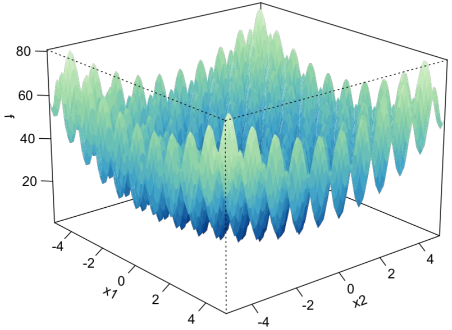


Problem Statement

- Which algorithm performs better and why does it outperform the other?
- Analyze and compare their results to determine which algorithm is better for minimizing the Rastrigin function.
    - We want to evaluate their performances in terms of speed and accuracy.

Background

- Gradient Descent used to optimize:
  - Neural networks and Deep learning: Speech recognition, and Natural language processing.
    - backpropagation
  - Logistics: Energy sector to predict energy consumption and load forecasting, and distribution.
  - Healthcare: Develop models that predict patient outcomes, disease progression, and treatment effectiveness.
- Simulated Annealing used to optimize:
  - The shortest route within different cities and a route to every city.
  - Parameters to minimize cost, weight, and distribution of products.
  - Circuit Design.
  - Telephone and Network Design.



Gradient Descent

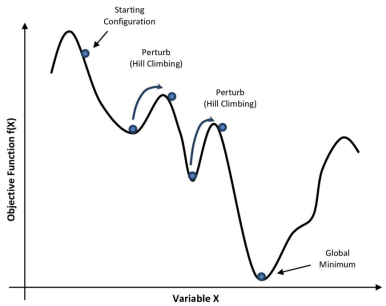
- Utilizes a first-order iterative approach to minimize function values by moving opposite to the gradient direction.
- Enhanced with Momentum and a cosine-adjusted learning rate for improved convergence and stability.
- Employs a dynamic step size adjustment across iterations and momentum for directional consistency.

Simulated Annealing - Adaptive Variant

- Inspired by the annealing process, it aims to minimize function values through probabilistic exploration.
- Employs a temperature-controlled exploration mechanism, allowing initial broad search space exploration and refined searching as temperature decreases.
- Adapted to include an adaptive exploration mechanism based on the Metropolis criterion/Boltzmann Probability and a logarithmic cooling schedule.

Approach

- Tested Gradient Descent and Simulated Annealing algorithms using the Rastrigin function, known for its complex landscape with numerous local optima.
- Utilized in a two-dimensional form for this experiment, maintaining its characteristic complexity.
- Development Environment: Implemented in Visual Studio Python, leveraging numpy libraries to emphasize algorithm mechanisms.
- Preliminary Testing: Initial tests conducted with simpler functions to validate algorithm implementations.



Approach - Key Functions

- gradient_descent(): For performing Gradient Descent with parameters such as the mathematical expression, initial parameters, learning rate, momentum coefficient, max iterations, and convergence threshold.
- general_gradient_compute(): Computes the gradient of any function given its SymPy expression and variables.
- cosine_learning_rate(): Calculates the learning rate at a given iteration time, aiding in dynamic step size adjustment.
- Simulated Annealing Specifics:
    - adaptive_simulated_annealing(): Implements the SA algorithm, requiring inputs like the mathematical expression, initial guess, max iterations, initial temperature, and kappa for adjustment.

Experiment - Gradient Descent

- Tested algorithms with a preliminary function to see if they work.
- set max_iter = 10,000
- For Gradient Descent gamma_coef = 0.8 (can be set between 0.8 and 0.9)
- Cosine learning rate initial rate = 0.01
- Random initial positions—some examples
  - [(2,-2), (1.5,-1), (8,3), (0,0),(0.1,0.1) (0.5, 0.7)]
- Convergence threshold: $1 * 10^{-6}$
- Test conducted on the 2D Rastrigin function where the abs min(0,0).

$$f(x, y) = 20 + x^2 + y^2 - 10(\cos(2\pi x) + \cos(2\pi y))$$

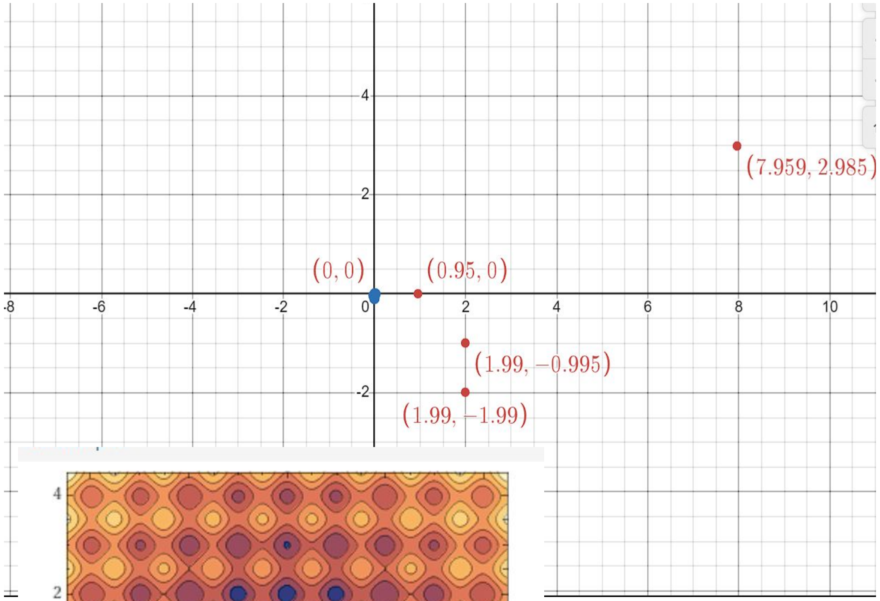
Experiment - Simulated Annealing

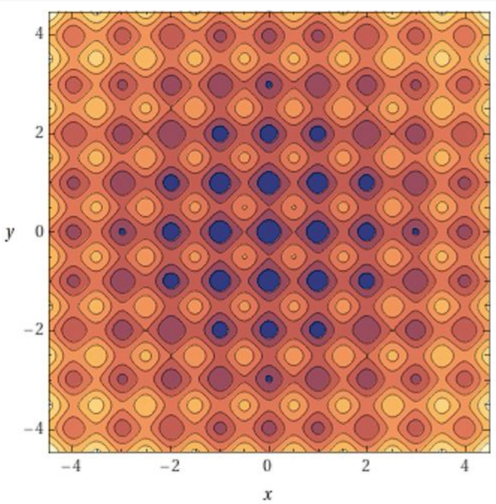
- Tested algorithms with a preliminary function to see if they work.
- set max_iter = 10,000
- Initial_temp = 100.0 (gave best results)
- Cooling_constant =10.0
- Random initial solutions—some examples
  - [(2,-2), (1.5,-1), (8,3), (0,0), (0.1,0.1), (0.5, 0.7)]
- Convergence threshold: $1 * 10^{-6}$
- Test conducted on the 2D Rastrigin function where the abs min(0,0).

$$f(x, y) = 20 + x^2 + y^2 - 10(\cos(2\pi x) + \cos(2\pi y))$$

Results

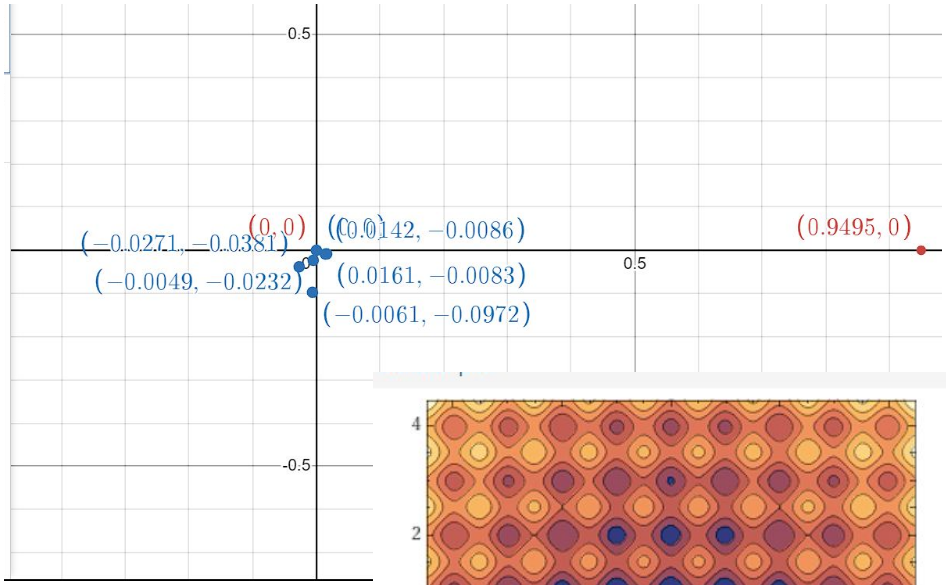
- Where did they tend to land: Gradient Descent
  - [(1.9899, -1.9899), (1.9899, -0.9949), (7.9592,2.9848), (0,0), (0.000000339,0.000000339), (0.9495,0.00000046)]
  - Note that I had to increase max_iter to get better results

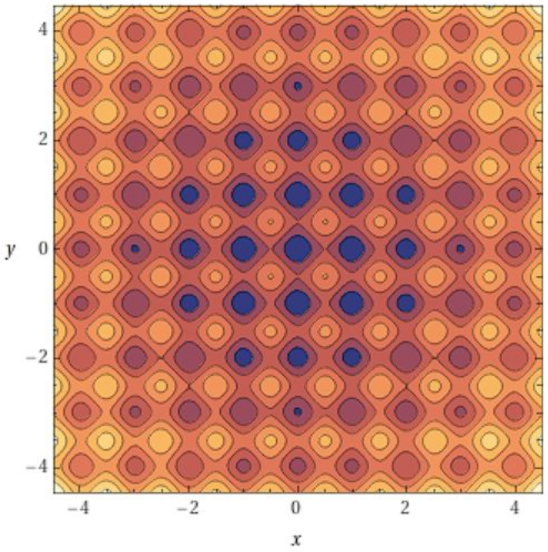




Results Cont.

- Where did they tend to land: SA
  - [(-0.0061305, -0.097170) ,(-0.004949, -0.02316837), (-0.027099, -0.03812), (0,0) , (0.01614, -0.00835), (0.014155, -0.008622),]
  - Note that it did get to the max iteration





Conclusion

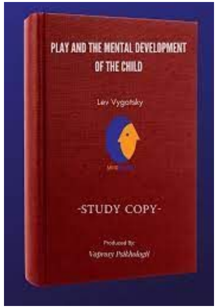
- Given the same initial points, SA constantly outperformed gradient descent.
- It requires less iterations than Gradient descent:
  - Gradient descents(~1000000)
  - SA(~19000)
- SA does not have the precision require to get zero exactly.
- For initial conditions close to absolute min, Gradient Descent is has more precision.
- For initial conditions far from absolute min, SA constantly gets close to absolute min.
- Demonstrates SA superiority over Gradient Descent but also demonstrates Gradients Descent superior precision when in close proximity.

Project 3:
Play And Its Role In The Mental Development Of The Child And The Zone Of Proximal Development

Introduction

- L. S. Vygotsky (1896 - 1934) was a former Soviet psychologist known for his work in children's development.
- Created a framework for cultural-historical activity theory which helped to understand the relationship between the human mind (relating to thinking and feeling) and activity (the act of doing).

Motivation



- Highlight the importance of play on the mental development of a child.
- Discuss the importance of social interactions that play an important role in play-based learning.
- Understand and identify the Zone of Proximal Development and scaffolding.

Problem Statement

- What is the developmental process with the emergence of play, its origin, and its genesis?
- What is play and how does it contribute to a child’s cognitive, social, and emotional development?
- How does the Zone of Proximal Development concept and its relationship with scaffolding support children’s learning and development?

Approach

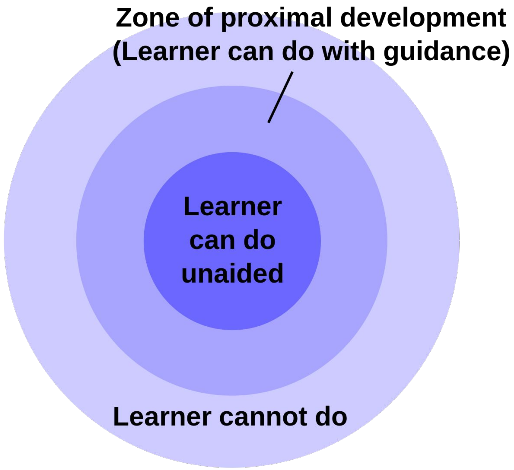
- Define play in this context from the reading published by Vygotsky.
- How does the role of play relate to cognitive development in this context.
- Examine the relationship between play, culture, and social interactions and how they play a critical aspect in play-based learning.
- Identify the Zone of Proximal Development and focus on the concept of scaffolding.

Main Points

- **Play & Imagination:** Fosters imaginative thinking and representation and allows children to engage in make-believe scenarios and engage in role-playing activities. Allows children to explore and this promotes creativity and abstract thinking.
- **Social Interaction & Collaborative Play:** Gives children the opportunity to interact with peers to collaborate and reach a common goal. This enhances their social skills and promotes language development.
- **Cultural & Contextual Influence:** Children often incorporate elements of their cultural environment into their activities.
- **Role of Adults:** Important in regard to scaffolding as they can provide guidance, support, and encouragement to during play. Help children to face challenges and expand their skills within the Zone of Proximal Development.

Zone of Proximal Development (ZPD)

- Concept that refers to the difference between what a learner can do without help and what they can do with help from someone more knowledgeable.
- Vygotsky believed that social interaction and collaborative learning should happen within the ZPD.
- **Scaffolding:** A technique to where temporary support is provided along with guidance and assistance as the learner works to acquire new skills and knowledge.
    - Primary goal is to promote independence and self-regulated learning.



Related Work and Background Material

1. **Piaget’s Theory of Cognitive Development:** Emphasizes the importance of play in the aspect of helping children to develop their cognitive abilities that include, problem-solving, language, and abstract thinking.
2. **Erikson’s Theory of Psychosocial Development:** Discusses the importance of play in aiding children to develop their social-emotional skills which include trust, autonomy, and initiative.
3. **Vygotsky’s Theory of Sociocultural Development:** Emphasizes the role of social interaction and play in helping children learn from more knowledgeable people and develop their cognitive and social skills altogether.

Conclusion

- Consider play to be an important part of a child's mental development as it allows them to engage in various activities beyond their current level of understanding and skill.
- Play is a form of social activity that includes collaboration, negotiation, and communication.
- Play can foster growth among children by establishing a Zone of Proximal Development and with scaffolding, allow them to gain more independence and initiate their self-regulated learning.
- Enable children to take on challenging and manageable tasks.
    - If required, seek advice from people of higher level of experience.

Thank you!