Modeling And Simulation Of Certain And Real Conditions Through Limits To Growth And Tragedy Of The Commons Archetypes

Dejan Krstev
Goce Delcev University
Faculty of Mechanical Engineering
Stip, North Macedonia
dejan.krstev@ugd.edu.mk

Sara Srebrenkoska
Goce Delcev University
Faculty of Mechanical Engineering
Stip, North Macedonia

Abstract— The archetype "Limits to Growth" serves as a foundational framework for modelling and understanding the dynamic interplay between population growth, resource availability, and environmental sustainability. In this context, MATLAB, a powerful computational tool, plays a crucial role in implementing and analysing various models that elucidate the potential consequences of unchecked growth. This abstract aims to explore the synthesis of the Limits to Growth and Tragedy of the Commons archetypes with MATLAB as a computational medium. We discuss the integration of mathematical models, including the World3 model, within MATLAB's environment to simulate scenarios and assess the limits imposed by finite resources on global development. The archetype embodies the recognition that exponential population and economic growth can ultimately lead to resource depletion, environmental degradation, and societal collapse. Through MATLAB, researchers and policymakers can explore these complex interactions by adjusting parameters, testing policies, and visualizing outcomes. This synergy offers insights into sustainable strategies, policy interventions, and potential pathways to mitigate the risks associated with unbridled growth. By leveraging MATLAB's computational capabilities, the Limits to Growth and   Tragedy of the Commons archetypes provides a dynamic platform for addressing the pressing challenges of resource management and environmental stewardship in an ever-changing world.

Keywords— Dynamic system, Archetype, Modelling and Simulation.

I. Introduction

In dynamic systems, archetypes refer to recurring patterns or structures that help us understand complex behaviours and interactions over time. These archetypes are often used in fields like systems thinking and system dynamics to analyse and model various phenomena. Here are some common archetypes in dynamic systems: Limits to Growth, Shifting the Burden, Eroding Goals, Escalation, Tragedy of the Commons, Success to the Successful, Reinforcing Feedback Loop. [1]

These archetypes help analysts and decision-makers identify and understand the underlying dynamics within complex systems. By recognizing these patterns, it becomes possible to anticipate potential issues, design more effective interventions, and make informed decisions to steer a system toward desired outcomes.

II. Explain archetypes known in dynamic systems

Archetypes in dynamic systems are recurring patterns or structures that help us understand complex behaviours and interactions over time. Here are some examples of these archetypes along with explanations:[2]

Limits to Growth: This archetype shows how a system can grow until it hits constraints, like limited resources or market saturation, causing growth to slow down.

Example: A company experiences rapid expansion and increases production, but eventually, it faces resource shortages, leading to reduced growth rates.
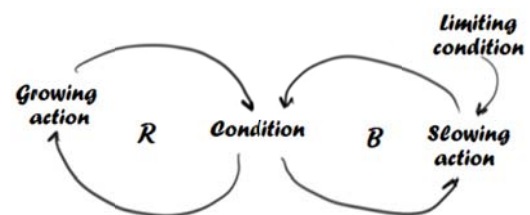

Fig. 1. *Limits to growth*

Shifting the Burden: This archetype demonstrates how reliance on short-term solutions can undermine efforts to address the root causes of problems.

Example: A person frequently takes pain relievers (quick fixes) for a recurring headache instead of addressing the underlying cause, which is poor sleep.
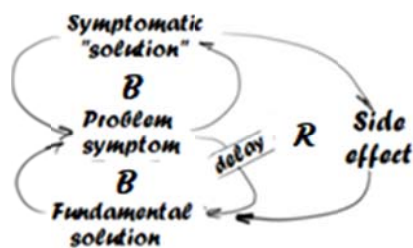

Fig. 2. *Shifting the burden/addiction*

Eroding Goals: This archetype illustrates how over time, lowering standards or goals can lead to a decrease in performance or quality.

Example: A sports team gradually lowers its training standards over the years, resulting in declining performance.
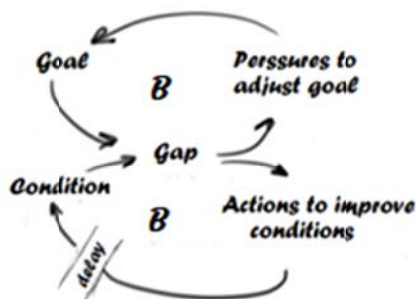

Fig. 3. *Eroding goals*

Escalation: This archetype shows how actions and reactions can escalate conflicts or competitions in a system.

Example: Two competing companies keep lowering their prices in response to each other's price cuts, leading to a price war.
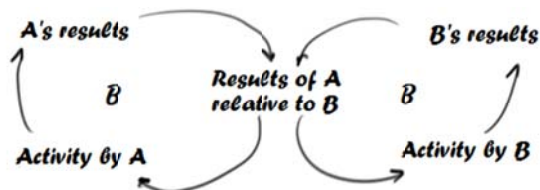

Fig. 4. *Escalation*

Tragedy of the Commons: This archetype demonstrates how the self-interest of individuals or groups can lead to the depletion of shared resources.

Example: Overfishing in a shared lake by multiple fishing communities leads to depletion of fish stocks and a crisis.
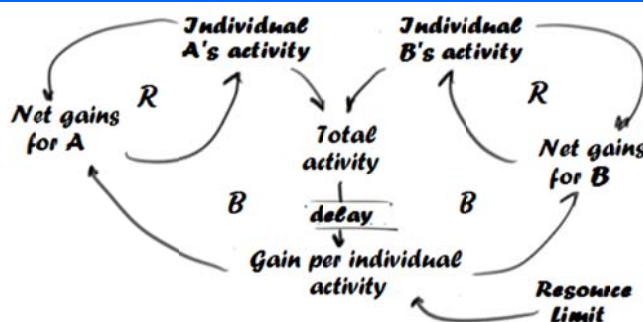

Figure 5. *Tragedy of the commons*

Success to the Successful: This archetype highlights how those who are already successful can accrue even more advantages, leading to growing disparities.

Example: In a competitive market, a leading company gains more customers and resources, making it even more dominant over time.
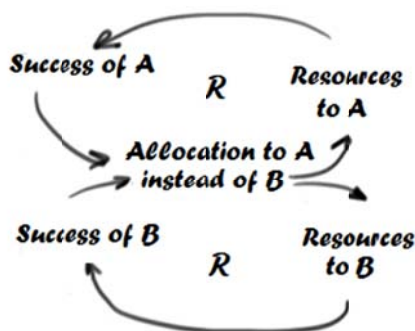

Figure 6. *Success to the successful*

Reinforcing Feedback Loop: This archetype shows how initial changes in one direction can lead to further changes in the same direction, often resulting in exponential growth or decline.

Example: A positive customer review boosts a product's sales, leading to increased visibility and even more positive reviews.
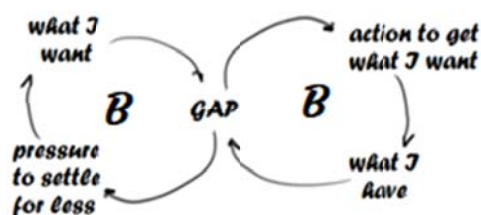

Figure 7. *Reinforcing Feedback Loop*

These archetypes provide valuable insights into the dynamics of various systems, helping us anticipate potential challenges, make informed decisions, and develop strategies to achieve desired outcomes. They are fundamental tools in the field of systems thinking and system dynamics.

III. MATLAB EXAMPLE FOR THE LIMITS TO GROWTH AND TRAGEDY OF THE COMMONS

The "Limits to Growth" archetype is a concept from system dynamics, a field that studies how the structure of systems influences their behaviour over time. In this archetype, the key idea is that when a system grows too quickly, it may eventually encounter limits or constraints that cause it to decline or collapse. [3]

Let's create a simple MATLAB example to illustrate the "Limits to Growth" archetype using a population growth model. In this model, we'll assume that a population grows at a certain rate but is limited by available resources. As the population approaches the resource limit, its growth rate decreases.

Algorithm 1 Matlab example for the Limits to Growth

```
x = 100;   % Initial population

growth_rate = 0.1; % Population growth rate

res_cap = 500;    % Resource carrying capacity

time_span = 100; % Simulation time span

% Arrays to store simulation results

population = zeros(1, time_span);

% Initialize the population

population(1) = x;

% Simulation loop

for t = 2:time_span

    % Calculate the population growth in the absence of constraints

    population_growth = population(t-1) * growth_rate;

    % Calculate the population growth limited by resource capacity

    limited_growth = min(population_growth, res_cap - population(t-1));

    % Update the population

    population(t) = population(t-1) + limited_growth;

end

% Plot the results

figure;

plot(1:time_span, population, 'b', 'LineWidth', 2);

xlabel('Time');

ylabel('Population');

title('Limits to Growth Archetype');

grid on;
```

In this MATLAB example: We define several parameters, including the initial population, growth rate, resource carrying capacity, and simulation time span. We create an array population to store the population at each time step and initialize it with the initial population. We simulate the population growth over time by iterating through a loop. In each iteration, we calculate the population growth based on the growth rate. However, we also consider the constraint of resource capacity by limiting the growth to the difference between the resource capacity and the current population. We update the population at each time step based on the limited growth.

Finally, we plot the population over time to visualize the "Limits to Growth" archetype. As the population approaches the resource capacity, its growth rate decreases, eventually reaching a steady state.
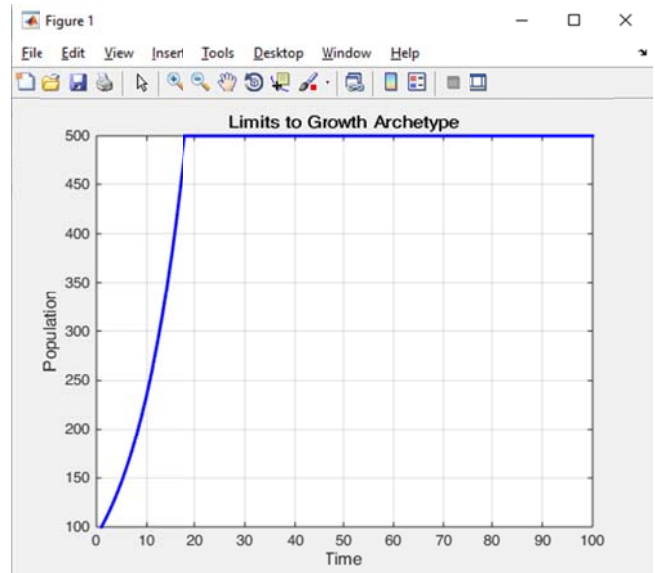

Fig. 8. *Output result in Matlab*

This simple example illustrates how a system can experience a limit to its growth when constrained by external factors. You can further refine and expand this model to explore more complex dynamics and scenarios in system dynamics modelling.

The Tragedy of the Commons is a concept in economics and environmental science that describes a situation in which individuals, acting in their self-interest, deplete shared resources, leading to negative consequences for the entire group. In the context of population growth, this might refer to the overuse of a common resource such as land, water, or food, resulting in environmental degradation or depletion.

Here's a simple MATLAB example illustrating the Tragedy of the Commons with a population growth model. This example assumes a common resource (e.g., land) that can support a certain maximum population. Each individual in the population decides how many offspring to have based on their perceived benefit, potentially leading to overuse of the resource.

Algorithm 2 Tragedy of the Commons Population Growth Model

```
% Parameters

max_population = 100; % Maximum sustainable population

initial_population = 10; % Initial population
```

```
growth_rate = 0.1; % Population growth rate

resource_capacity = 100; % Maximum resource capacity

% Simulation time

num_generations = 50;

% Initialize population

population = zeros(num_generations, 1);

population(1) = initial_population;

% Initialize resource usage

resource_usage = zeros(num_generations, 1);

% Simulation loop

for t = 2:num_generations

    % Each individual decides how many offspring to have

    offspring = round(population(t-1) * growth_rate);

    % Total population after reproduction

    new_population = population(t-1) + offspring;

    % Calculate resource usage (assuming each individual uses the resource equally)

    resource_per_individual = resource_capacity / new_population;

    % Update population and resource usage

    population(t) = new_population;

    resource_usage(t) = resource_per_individual;

end

% Plot results

figure;

subplot(2, 1, 1);

plot(1:num_generations, population, 'b-o');

title('Population Growth Over Time');

xlabel('Generation');

ylabel('Population');

subplot(2, 1, 2);

plot(1:num_generations, resource_usage, 'r-o');

hold on;

plot([1, num_generations], [1, 1] * (resource_capacity / max_population), 'k--');

hold off;

title('Resource Usage Over Time');

xlabel('Generation');

ylabel('Resource Usage per Individual');

legend('Resource Usage', 'Sustainable Resource Level');

% Display a warning if resource usage exceeds the sustainable level
```

```
    if any(resource_usage > (resource_capacity / max_population))

        disp('Warning: Resource usage exceeds sustainable level.');

end
```

In this example, individuals reproduce based on a growth rate, and the total population is limited by the maximum sustainable population. The resource usage per individual is calculated, and a warning is displayed if the resource usage exceeds the sustainable level. The second subplot shows the resource usage over time, and the dashed line represents the sustainable resource level per individual. (Figure 9.)
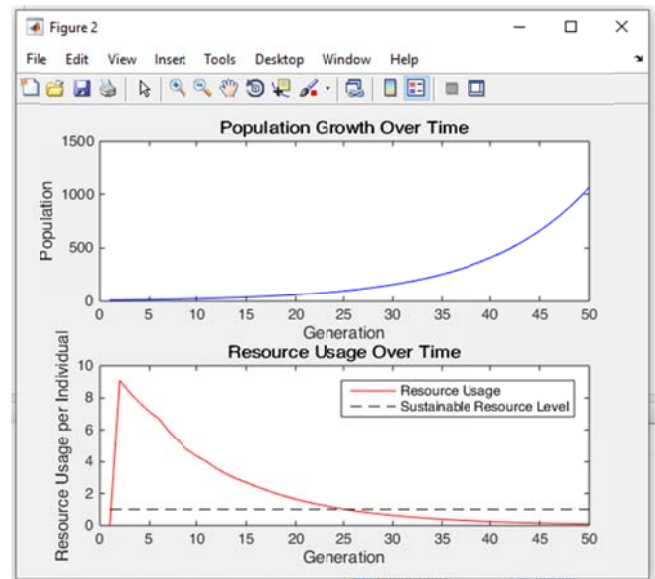

Fig. 9. *Output result in Matlab for Tragedy of the Commons Population Growth Model*

IV. RESULTS AND DISCUSSION

In conclusion, the archetype "Limits to Growth" when implemented and analysed using MATLAB offers a powerful platform for examining the critical theme of sustainable development and resource management. This combination provides valuable insights into the complex interactions between population growth, resource availability, and environmental sustainability.

By harnessing MATLAB's computational capabilities, researchers, policymakers, and scholars can simulate various scenarios, test policy interventions, and visualize potential outcomes. This enables a deeper understanding of the consequences of unchecked growth and the importance of sustainable practices.

The Limits to Growth archetype, rooted in the recognition of finite resources and environmental constraints, serves as a reminder of the need for responsible and informed decision-making to ensure a sustainable future. Through MATLAB, we have the tools to explore, analyse, and address the challenges posed by the limits to growth, ultimately guiding us

toward more sustainable and resilient societies. This theme remains as relevant as ever, and MATLAB continues to be an essential tool in our pursuit of a balanced and sustainable world.

System archetypes are models of behaviour of a system, understood as generic structures or as an overview of typical systems. There are recognized structures that show repetition in many different situations. Archetypes are depicted as appearances of common combinations through amplifying and balancing feedback loops. They are constantly used to facilitate a quick understanding of the system and their knowledge and already learned features, their insight and insight. The collection of systems archetypes has been published by Senge (1990), Braun (2002) and several other authors. As analytical features, they help people change their thinking for a much larger systemic perspective to understand a phenomenon or dynamic, and in some situations when real corrective action is not taken.[4]

This straightforward illustration demonstrates how a system might encounter a development limit when restrained by outside variables. This model may be improved and expanded upon to investigate more intricate dynamics and situations in system dynamics modelling.

This is a simplified example, and real-world systems may involve more complex interactions and additional variables. Depending on your specific research or modelling goals, you may need to extend the model and add more complexity. MATLAB provides a powerful environment for developing and simulating such dynamic models.

REFERENCES

[1] [1.] Braun, Wiliam, 2002, The System Archetypes Copyright © 2002

[2] [2.] Carlile, P.R., 2000. A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development. Paper Manuscript. MIT Sloan School of Management. Cambridge, MA.

[3] [3.] Carlile, P.R., E.S. Rebentisch., 2001. Into the Black Box: The Knowledge Transformation Cycle.. Paper Manuscript. MIT Sloan School of Management. Cambridge, MA.

[4.] Coyle, G. 1998. The practice of System Dynamics: Milestones, Lessons and Ideas from 30 Years Experience. System Dynamics Review, Vol. 14, No. 4, Winte