**Process simulation theory in practice**

**(Turtle Sim developer edition)**

# Content

# Intro

## Author introduction

Hello my name is Erik Palenčík, while writing this book I work as Senior Software developer and starting my PhD study on the faculty of the mechanical engineering – Industry Engineering field.

During study of IT (which I failed) I continued in this field by work, but I finished bachelor’s in management and master’s degree in Industry Engineering.

## Motivation to write the book

As student I went to multiple books – maybe great in simulations community, but for me as free time simulation worker, they were useless, after reading four hundred pages, I read no information. So, this was the reason why to create my own publication to help students and bring fun to the simulations world.

## Motivation to create Turtle Engine

During my work I started to experiment with Three JS and web and backend technologies, as my skillset grown and my life passed through I decided to create my own digitalization platform – for uploading 3D models, point clouds, 360 panoramas and another interactive content.

By the time I undergone Modeling and Simulations and learning about manual simulation in this course I decided to create my own simulation engine with strong 3D visualization and knowledge of web technologies.

Turtle Engine is part of my free time activities and my university research. Platform also contains modules to boost learning and visualization of students, not only to make practical examples.

# Simulation theory

## Monte Carlo method

Monte Carlo is the most popular simulation method

## Heuristics

## Genetics algorithm

## Probability distribution

## Statistics

# Data structures and algorithms

# Entities and definitions

## Basic entitiesƒ

The purpose of basic entities is to create simplified simulation with basic set of basic behavior.

**Spawn**

Spawn is behavior responsible for creation of entities. Spawn can work in the following modes:

1. Constant spawn time – spawn every 5 seconds
2. Normal distribution spawning – spawn in NormDist(10, 30)

**Process**

**Delay**

**Buffer**

**Queue**

**Merge**

**Split**

**Switch**

**Sink**

An opposite to spawn, sink destroys entities, sink is responsible for removing entity from the world.

## Smart entities

The purpose of smart entities is to add additional value to

# Platform architecture overview

Many computer languages support different architecture allowing different patterns and code architecture. Because Turtle Sim was developed with usage of Golang (as backend) – functional static typed language and JavaScript (frontend – React and Three JS) - duck typing OOP language, there are special kinds of architectures used in this cases.

## Entities and actors explanation

In Turtle we differentiate **Entities** and **Actors** – in IT theory they can be the same thing, but in this cases I decided to use this two names separately.

Entities are objects in world which has some role and behavior, e.g. Process, Spawn, Sink, …

Actors are objects traveling between entities being able to change it’s state, wear states.

## Simulation initialization

As every system also Turtle models have to somehow prepare load data, prepare them and then simulate.

A diagram of a process

Description automatically generated

Figure Simulation Initialization

The following loading stages are:

1. **World deserialization** – in this phase world is loaded from database.
2. **Init 1** – in this phase there is called function **Init1** in which you entity should prepare data

# Statistics and analysis

# Working with turtle

After opening simulation platform, we can find similar interface being seen in environments like Anylogic or Tecnomatix Plant simulation.

A screen shot of a computer

Description automatically generated