

LICENCIATURA EM ENGENHARIA INFORMÁTICA

Inteligência Artificial

2025/2026

# Relatório do **Trabalho Prático 1**

Docentes:

JOSÉ PAULO BARROSO DE MOURA OLIVEIRA  
EDUARDO JOSÉ SOLTEIRO PIRES

**Grupo de Trabalho (PL4):**

- Liane Raquel Madureira Duarte – AL79012  
- Pedro Miguel Costa Braz – AL81311

# Índice

1.	Introdução .....	3
2.	Fase de Implementação 1 (Eco1).....	3
2.1.	Ambiente do Eco1 .....	4
2.2.	Interface e controlos.....	5
2.3.	Comportamentos e dinâmica do modelo .....	5
2.4.	Resultados e observações .....	7
3.	Fase de Implementação 2 (Eco2).....	9
3.1.	Alterações ao ambiente .....	9
3.2.	Peixes e Ovos .....	10
3.3.	Poluição e Falha Geológica .....	13
3.4.	Barco e Rede de Pesca.....	15
3.5.	Ciclo de Dia e Noite .....	18
3.6.	Resultados e Observações.....	20
4.	Conclusão .....	22
5.	Referências Bibliográficas.....	24
6.	Anexo 1.....	24
7.	Anexo 2.....	25

# 1. Introdução

Este trabalho tem como principal finalidade promover a compreensão e aplicação dos conceitos de sistemas multiagente através da modelação e simulação de um ecossistema aquático na plataforma NetLogo. A simulação envolve a interação entre diferentes tipos de agentes, como peixes, plantas e poluidores, num ambiente dinâmico, onde é possível observar fenómenos como a propagação da poluição, a reação dos agentes ao meio e a evolução do equilíbrio ecológico.

O modelo criado permite observar fenómenos como a dispersão da poluição, o comportamento adaptativo dos agentes e a evolução do equilíbrio ecológico ao longo do tempo.

O desenvolvimento do projeto foi dividido em duas fases:

- Na **Fase 1 (Eco1)**, foi criado um modelo inicial que representa o ambiente aquático e o comportamento básico dos agentes, com foco na movimentação e na introdução da poluição.
- Na **Fase 2 (Eco2)**, foram introduzidas melhorias e inovações, ajustando o modelo para alcançar um maior equilíbrio no ecossistema e explorar a interação entre os diferentes tipos de agentes.

Desta forma, o presente relatório procura documentar o processo de conceção, implementação e análise dos modelos desenvolvidos, apresentando uma descrição detalhada das principais funcionalidades, decisões de design e resultados obtidos. Além disso, pretende-se evidenciar como a utilização de sistemas multiagente pode ser uma ferramenta poderosa para compreender fenómenos complexos e simular dinâmicas ecológicas reais em ambientes virtuais.

## 2. Fase de Implementação 1 (Eco1)

A primeira fase do trabalho consistiu na criação de um modelo base (Eco1) que representa um ecossistema aquático simples, onde coexistem peixes, plantas e fontes de poluição (canos de esgoto). O objetivo foi estruturar o ambiente, definir o comportamento elementar dos agentes e introduzir as primeiras dinâmicas de contaminação e limpeza.

## 2.1. Ambiente do Eco1

O ambiente do modelo é constituído por um conjunto de células (patches) que representam o espaço físico do ecossistema.

Estas células foram divididas em três zonas distintas:

- **Céu** (topo): pcolor = 87.
- **Água** (zona central): pcolor = 95, região de movimento dos peixes e onde ocorre a poluição.
- **Areia** (base): pcolor = 44.

No início da simulação, todas as células de água encontram-se limpas e partilham a mesma cor, criando assim um ambiente aquático puro e equilibrado, como ilustrado na Figura 1.

No setup são limpos os dados, pintado o ambiente, criados os agentes e reposto o tempo (**ticks**). As plantas são posicionadas na faixa de chão (areia) junto à costa, aleatoriamente, e é criado 1 cano colocado na linha de costa (dentro da água, adjacente a patches de areia), que funciona como **fonte local de poluição**.

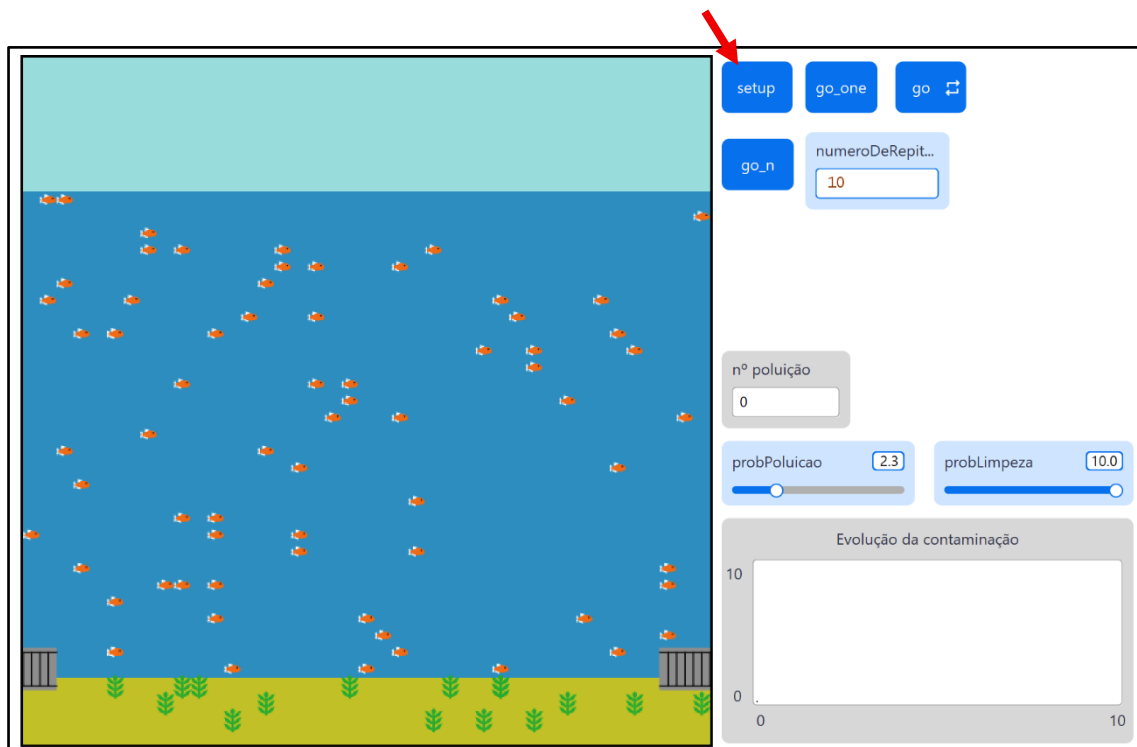


Figura 1 - Vista inicial do mundo após setup

## 2.2. Interface e controlos

Foram implementados os controlos pedidos, como representado na Figura 2:

- **Botões:** setup, go\_one (1 tick), go\_n (N ticks), go (contínuo).
- **Deslizadores:** nPeixes, nPlantas (tamanhos iniciais), probPoluicao (probabilidade de contaminação), probLimpeza (probabilidade de recuperação/limpeza), numeroDeRepiticoes (para go\_n).
- **Monitores:** contador “nº poluição” (n.º de patches contaminados).
- **Gráfico:** “Evolução da contaminação”, que regista a variação do número de patches poluídos ao longo dos ticks.

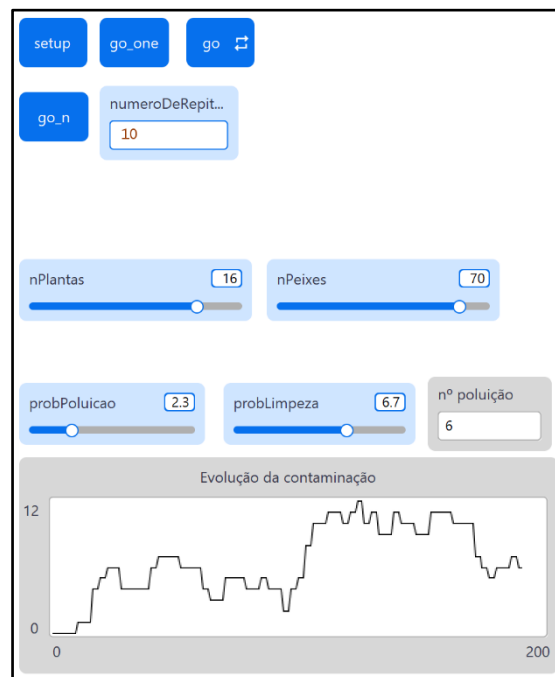


Figura 2 - Interface com botões, deslizadores, monitor e gráfico.

## 2.3. Comportamentos e dinâmica do modelo

### Movimento dos peixes

Em cada tick um peixe roda um ângulo aleatório e avança 1 unidade. Se o patch destino não for água ( $pcolor \neq 95$ ), o passo é anulado e o peixe inverte direção, garantindo que nunca sai da água, como mostra o código apresentado no Anexo 1.1.

## **Geração, propagação e limpeza da poluição**

A cada tick, o cano de esgoto atua como fonte emissora de poluição, contaminando patches de água num raio de 2 unidades.

Cada patch tem uma probabilidade igual a  $\text{probPoluicao} / 100$  de passar de azul (95) para castanho-escuro (34), conforme ilustrado no Anexo 1.2, representando a descarga de esgoto doméstico e resíduos industriais diretamente no mar.

Esta contaminação inicial ocorre junto à costa e simboliza a libertação de substâncias poluentes, que se vão diluindo gradualmente devido ao grande volume de água existente no ecossistema marinho.

A difusão da poluição é modelada através dos patches já contaminados (34), que podem propagar a contaminação aos patches vizinhos (`neighbors4`) ainda limpos (95) com a mesma probabilidade.

Assim, a mancha poluída expande-se de forma progressiva, imitando a forma como o esgoto e os poluentes se dispersam na água devido às correntes marítimas, como ilustrado na Figura 3.

O modelo inclui também um mecanismo de limpeza natural que ocorre a cada 3 ticks ( $\text{ticks} \bmod 3 = 0$ ), simulando o processo de diluição e regeneração da água. Neste processo:

- Os patches 34 (castanho-escuro) podem evoluir para 35 (castanho-claro) com probabilidade  $\text{probLimpeza} / 100$ , representando uma fase de transição, em que a poluição começa a misturar-se com a água.
- De seguida, os patches 35 podem regressar ao estado limpo (95, azul) com a mesma probabilidade, completando a recuperação gradual do ambiente aquático.

Este mecanismo traduz uma limpeza progressiva e mais realista, permitindo observar o equilíbrio dinâmico entre a contaminação causada pelo cano de esgoto e o processo natural de purificação da água ao longo do tempo.

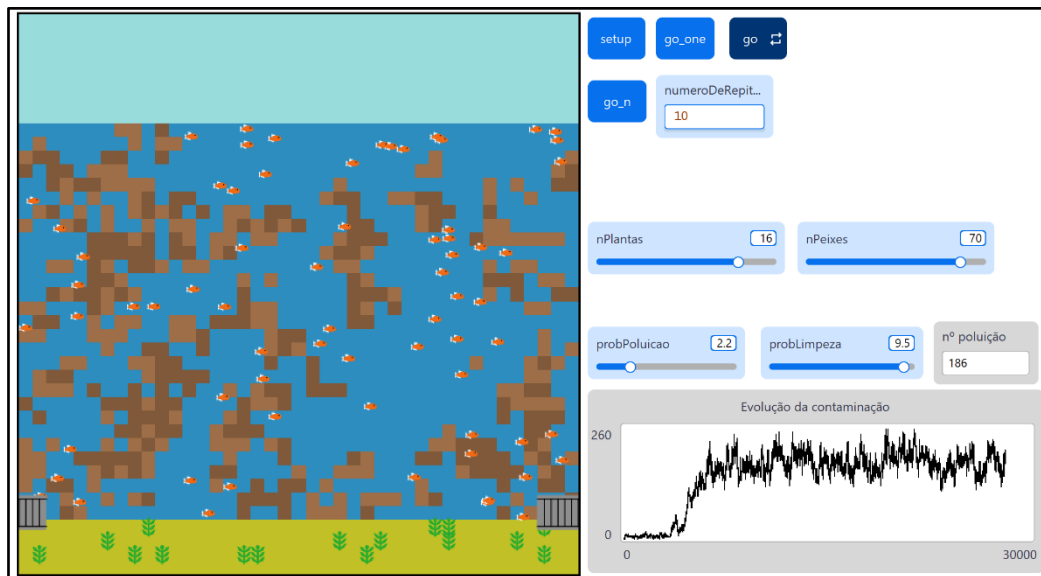


Figura 3 - Propagação da poluição e evolução temporal da contaminação.

## 2.4. Resultados e observações

Durante a execução do modelo Eco1, observou-se que a poluição se propaga progressivamente a partir do cano de esgoto, afetando uma área cada vez maior do ecossistema. O gráfico de evolução da contaminação mostra um crescimento inicial rápido, seguido de oscilações causadas pelo equilíbrio entre os processos de poluição e limpeza, como representado na Figura 4.

Os peixes continuam a movimentar-se indiferentemente em zonas limpas e contaminadas, uma vez que não possuem comportamento adaptativo. Já as plantas, permanecem estáticas, funcionando apenas como elementos visuais, e o processo de limpeza mostra que, mesmo com baixa probabilidade, o mar tende a recuperar lentamente a sua cor original.

Esta primeira versão do modelo permitiu compreender a dinâmica básica entre os agentes e os processos de contaminação, servindo de base para as melhorias e inovações da Fase 2 (Eco2).

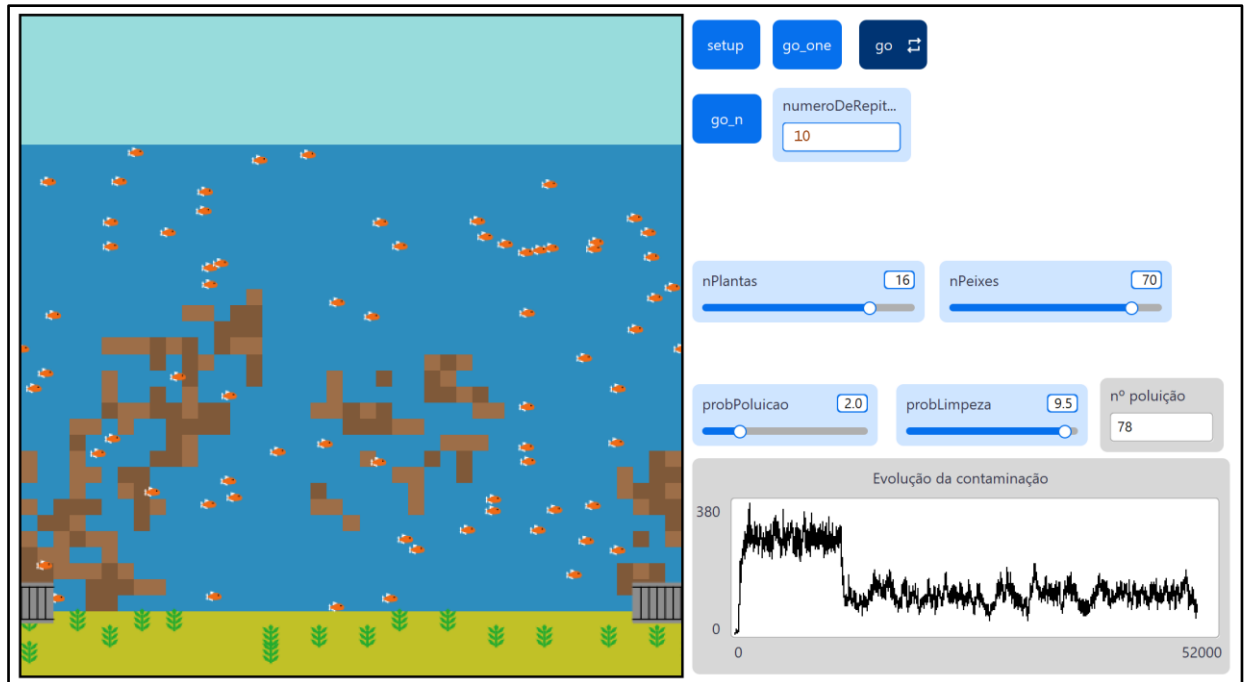


Figura 4 - Eco1 após 50057 ticks (com uma alteração na probPoluicao, nos 9000 ticks, como apresentado no gráfico).



### 3. Fase de Implementação 2 (Eco2)

Na segunda fase, o modelo foi ampliado com novos agentes, eventos e mecanismos de equilíbrio, aproximando a simulação de um ecossistema mais realista, como ilustrado na Figura 5. Introduziram-se peixes predadores, reprodução por ovos, ciclo dia/noite, poluição como “turtles” (esgoto) e uma falha geológica que liberta petróleo. Adicionalmente, foi integrado um barco com rede de pesca e lógica de avaria/afundamento, o que acrescenta pressões adicionais sobre a população de peixes.

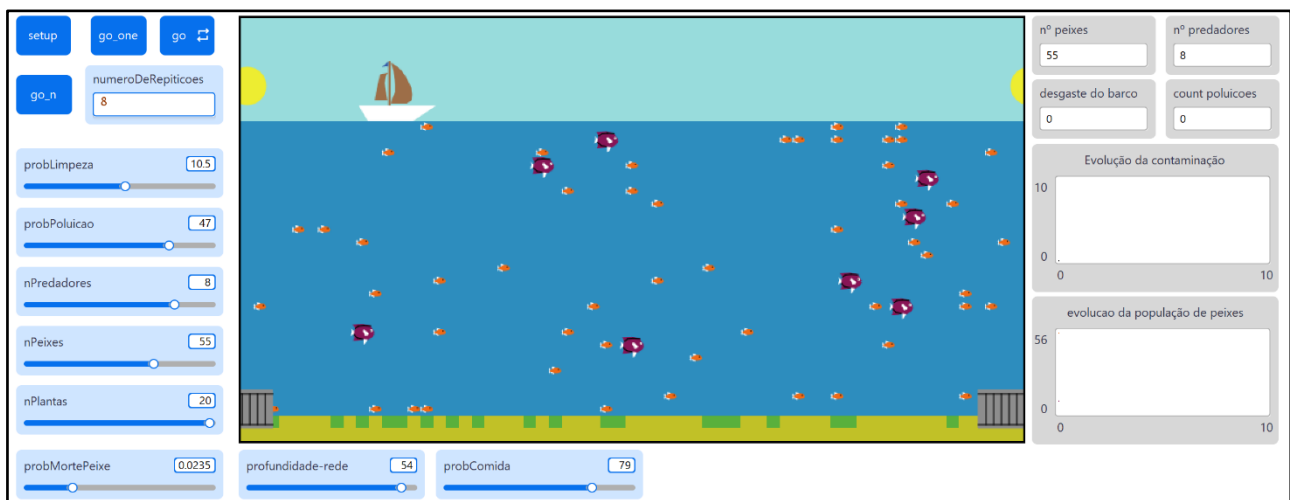


Figura 5 – Eco2 na fase inicial

#### 3.1. Alterações ao ambiente

Nesta fase, o ambiente do modelo foi profundamente reformulado para suportar os novos elementos introduzidos no ecossistema. O objetivo foi criar um cenário mais realista e dinâmico, capaz de representar as variações ambientais e de integrar novos agentes e comportamentos.

Uma das principais alterações foi a substituição das plantas como agentes (turtles) por patches estáticos, representados por uma cor específica (cor 55), localizados na zona costeira, retratado na Figura 6.

Esta decisão teve como finalidade reduzir a complexidade visual do ambiente, uma vez que o número de elementos aumentou consideravelmente com a introdução de novos agentes.

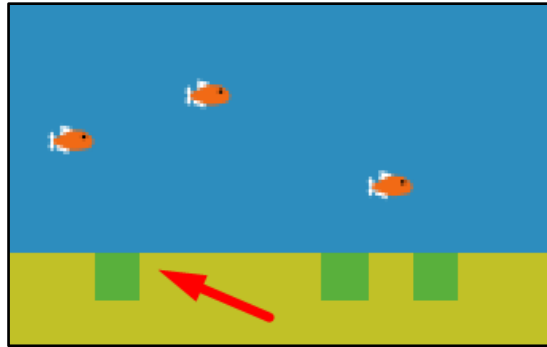


Figura 6 – Representação das plantas no Eco2.

Outra alteração significativa foi a forma como a poluição do esgoto passou a ser representada.

No modelo Eco1, a poluição era simulada através da mudança de cor dos patches, a água contaminada era pintada de castanho, refletindo a propagação e posterior limpeza por meio de alteração do valor de cor.

Já no modelo Eco2, a poluição de esgoto é representada por turtles móveis (do breed *poluicoes*), que se deslocam pelo ambiente de forma independente, apresentado na Figura 7.

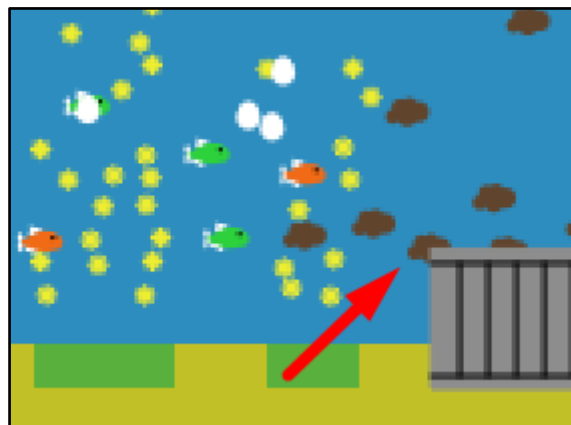


Figura 7 – Representação da poluição (cano de esgoto) no Eco2.

### 3.2. Peixes e Ovos

Nesta fase do modelo, foram introduzidos dois tipos de peixes, representados por *turtles* distintas: os peixes herbívoros, também designados por peixes normais, e os peixes predadores.

Esta distinção permitiu representar dois níveis tróficos dentro do ecossistema, criando relações de dependência e competição que contribuem para o equilíbrio do sistema.

Cada peixe possui três propriedades fundamentais: vida, idade e tipo.

A vida representa a energia e o estado de saúde do agente, iniciando com um valor médio de 50 e podendo variar entre 0 e 100. A idade é incrementada a cada *tick*, permitindo controlar o tempo de vida dos peixes e simular a mortalidade natural associada ao envelhecimento. Já a variável tipo define o comportamento e as interações de cada peixe no ambiente, cuja implementação pode ser consultada no Anexo 2.1.

Para além destes parâmetros, foi implementado um sistema visual de estados, ilustrado na Figura 8, que permite identificar graficamente a condição dos peixes:

- Quando a vida é superior a 90, o peixe assume uma cor viva e intensa (verde nos herbívoros e vermelha nos predadores), indicando vitalidade e boa saúde.
- Quando a vida desce abaixo de 40, a cor do peixe torna-se acinzentada, simbolizando fraqueza ou doença.
- Caso a vida chegue a 0, o peixe morre, sendo removido do ecossistema.

Este sistema não só facilita a análise visual do estado geral da população, como também reforça o realismo do modelo, ao associar cor a condição fisiológica.

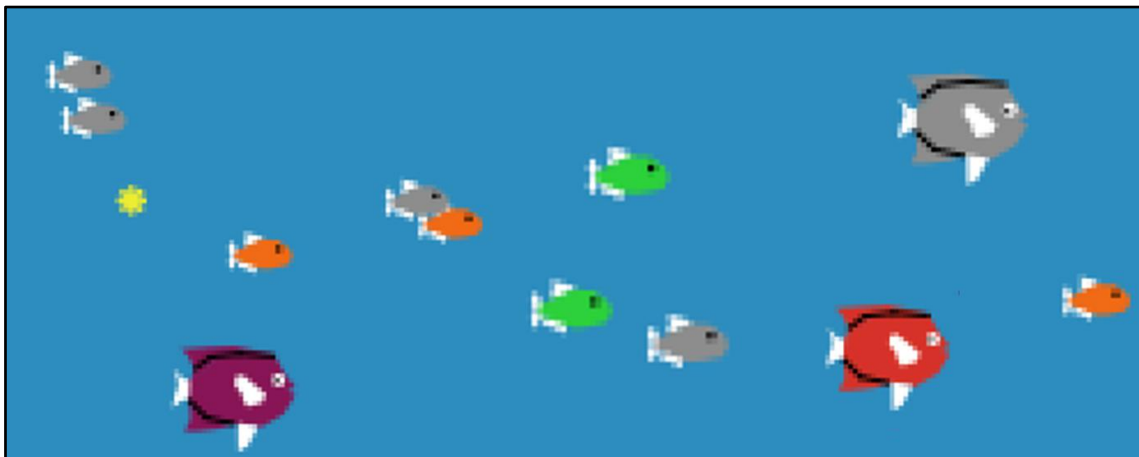


Figura 7 – Representação dos peixes no Eco2.

Quando um peixe atinge um nível de vida superior a 90, passa a ter uma determinada probabilidade de pôr ovos (*probPôrOvo*), de acordo com o código apresentado no Anexo 2.2.

Nos peixes herbívoros, esta probabilidade é mais elevada, enquanto nos predadores é mais reduzida, refletindo taxas reprodutivas distintas.

Cada ovo permanece no ambiente durante um número definido de *ticks* (*tempoOvo*) e, após esse período, eclode, originando um novo peixe do mesmo tipo e com valores iniciais de vida e idade, demonstrado na Figura 8.

Além disso, os predadores podem alimentar-se dos ovos dos peixes herbívoros, e, também, obtêm energia ao consumir peixes herbívoros, um comportamento que ajuda a controlar o crescimento populacional e mantém o sistema próximo de um equilíbrio estável.

Já os peixes herbívoros alimentam-se das partículas de alimento libertadas pelas plantas, representadas no modelo por turtles do tipo *comidas*. Sempre que um peixe herbívoro se encontra sobre uma destas partículas, o alimento é consumido e removido do ambiente, aumentando o valor da variável *vida* do agente. Este processo, ilustrado no Anexo 2.3, permite simular a transferência de energia entre produtores e consumidores primários, reforçando o realismo ecológico do sistema.

Por fim, foi implementado um mecanismo de resgate populacional: caso todos os peixes de uma espécie (herbívoros ou predadores) desapareçam, o último indivíduo antes de morrer deixa um ovo, garantindo a continuidade da espécie, tal como evidenciado no Anexo 2.4.

Esta funcionalidade foi inspirada em comportamentos reais observados em algumas espécies aquáticas, reforçando a sustentabilidade ecológica do modelo.

O conjunto destas dinâmicas, alimentação, predação, envelhecimento, reprodução e morte, confere ao ecossistema um comportamento emergente e autorregulado, onde as variações populacionais resultam naturalmente das interações entre agentes e do estado do ambiente.

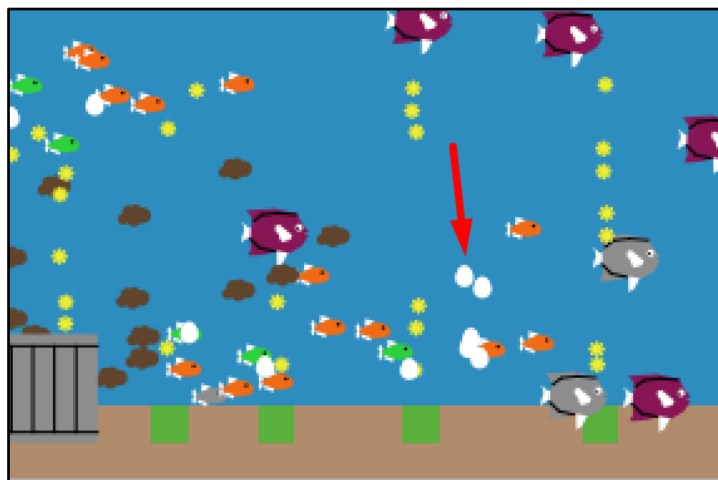


Figura 8 – Representação dos ovos no Eco2.

### 3.3. Poluição e Falha Geológica

Com o objetivo de tornar o ecossistema mais realista e complexo, foi introduzido no modelo um sistema de poluição multifatorial, composto por canos de esgoto e por uma falha geológica que libertam petróleo. Estas duas fontes de contaminação afetam diretamente a qualidade da água e a sobrevivência dos agentes vivos, refletindo as consequências ambientais da ação humana e de fenómenos naturais.

#### Poluição por esgoto

Os canos de esgoto, como já mencionado anteriormente, funcionam como fontes constantes de poluição, libertando *turtles* do tipo *poluicoes* que representam partículas de resíduos, retratado na Figura 9.

Cada partícula move-se aleatoriamente pelo mar, simulando a dispersão natural dos poluentes devido às correntes marítimas.

Estas partículas podem desaparecer com o tempo, de acordo com a probabilidade definida pelo parâmetro *probLimpeza*, que representa a capacidade de regeneração do oceano.

Os peixes herbívoros e predadores interagem com estas partículas de poluição. Quando ingerem uma delas, confundindo-a com comida, a sua variável *vida* diminui significativamente, o que reproduz o efeito tóxico do esgoto e dos metais pesados sobre a fauna marinha.

Em alguns casos, uma exposição prolongada pode levar à morte dos peixes, evidenciando a gravidade da poluição sobre o equilíbrio ecológico.

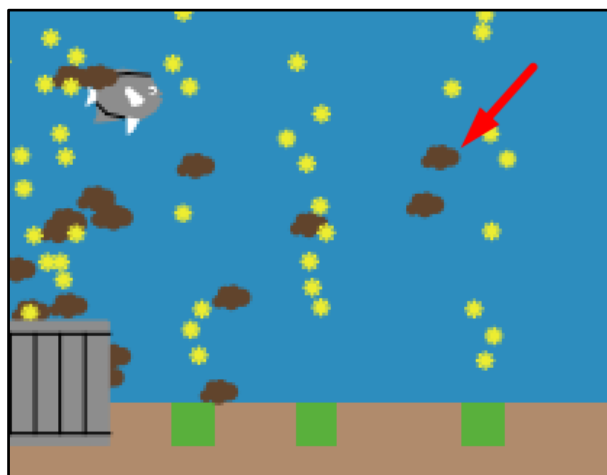


Figura 9 – Representação das partículas de poluição provenientes dos canos de esgoto no Eco2.

## Falha geológica e libertação de petróleo

Além da poluição provocada por atividades humanas, o modelo contempla também fontes naturais de contaminação.

Foi implementado um evento estocástico que simula a formação de uma falha geológica no fundo do mar.

Esta falha pode surgir a qualquer momento da simulação, desde que não exista uma já ativa, com uma probabilidade definida pelo parâmetro *probFalha*, e marca a sua posição através de uma *turtle* do tipo falha, que simboliza o ponto de rutura da crosta oceânica, conforme ilustrado no Anexo 2.5.

Quando ocorre, o terreno sofre um abalo sísmico, representado pela mudança temporária da cor do chão para um tom castanho mais escuro, indicando a movimentação da crosta oceânica e a libertação de sedimentos.

Durante este período, a falha liberta partículas de petróleo que sobem lentamente até à superfície, reproduzindo o comportamento físico do petróleo no meio aquático.

Cada partícula, representada por uma *turtle* do tipo petroleo, desloca-se verticalmente e espalha-se horizontalmente quando atinge a camada superior da água, formando uma mancha negra visível na superfície, representado na Figura 10.

Os efeitos do petróleo sobre os peixes são ainda mais severos do que os do esgoto, como se demonstra no Anexo 2.6.

Quando um peixe entra em contacto com estas partículas, perde uma grande quantidade de vida e, se o contacto for prolongado, morre por intoxicação. Esta abordagem permite observar o impacto acumulado da poluição e a forma como eventos naturais, como falhas tectónicas, agravam o desequilíbrio ecológico.

Com o tempo, parte das partículas de petróleo desaparece devido ao processo de limpeza natural (*probLimpeza*), simulando a diluição lenta e parcial do petróleo no oceano.

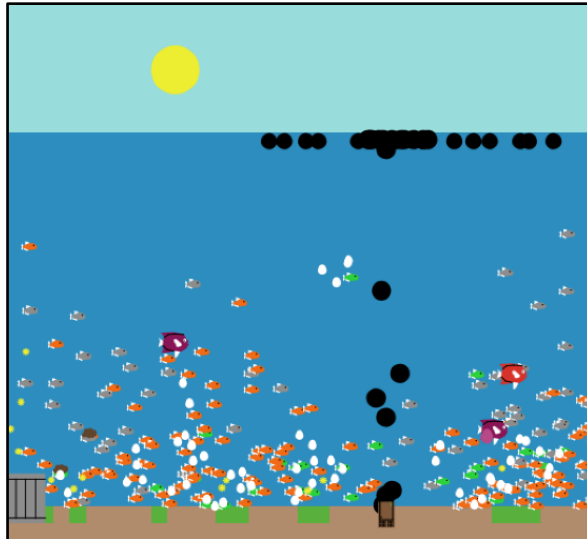


Figura 10 – Representação da falha geológica, alteração do solo e libertação de petróleo no Eco2.

### 3.4. Barco e Rede de Pesca

Para representar a intervenção humana no ecossistema, foi adicionado ao modelo um barco que navega lentamente à superfície do mar. Este agente simboliza as atividades humanas de pesca e exploração marítima, permitindo observar o impacto direto dessas ações sobre as populações de peixes e a qualidade ambiental do oceano.

O barco é um agente móvel que se desloca horizontalmente de um lado ao outro, percorrendo a superfície da água.

O seu movimento é controlado por uma variável interna de direção (*dir*), que alterna entre 1 e -1, definindo o sentido de deslocação.

Quando o barco atinge um dos limites do mundo, inverte a direção, criando um ciclo contínuo de navegação.

Durante a simulação, o barco pode entrar em contacto com partículas de petróleo ou zonas poluídas, o que aumenta a variável *contaminacao*.

À medida que o valor de *contaminacao* cresce, a cor do barco escurece gradualmente, simulando o acúmulo de resíduos na embarcação e o dano estrutural causado por poluentes, ilustrado na Figura 11.

Quando a *contaminacao* atinge o valor máximo de 100, o barco avaria e deixa de se mover, representando uma embarcação danificada, cuja implementação pode ser consultada no Anexo 2.7.

A partir desse momento, inicia-se um processo de afundamento gradual, onde o barco desce lentamente até ao fundo do mar e muda de cor para um tom acinzentado, indicando que já se encontra submerso e inativo.

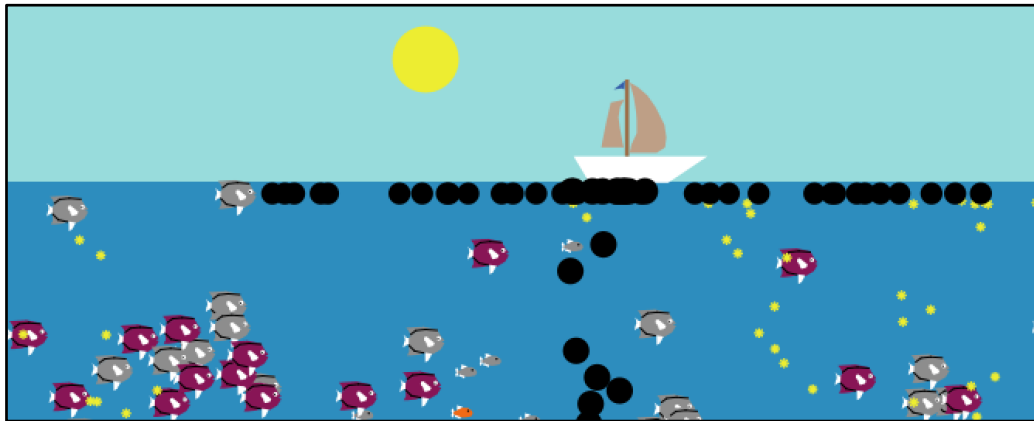


Figura 11 – Representação do barco a navegar à superfície e acumular contaminação durante a simulação.

Após o afundamento, o barco permanece no fundo durante um período de tempo, enquanto um contador interno (*tempo-para-contaminar*) é incrementado a cada tick, tal como se observa no Anexo 2.8.

Quando este contador atinge o limite de 100 ticks, o barco torna-se pronto para contaminar (*pronto-para-contaminar? = true*).

A partir desse momento, as células de água próximas do barco, *patches* num raio de 2 unidades, mudam de cor para violeta, representando a libertação de substâncias tóxicas e combustíveis provenientes do naufrágio, representado na Figura 12.

Esta contaminação local provoca a morte de peixes nas redondezas e aumenta temporariamente o nível global de poluição, simulando os impactos ambientais de acidentes marítimos reais.



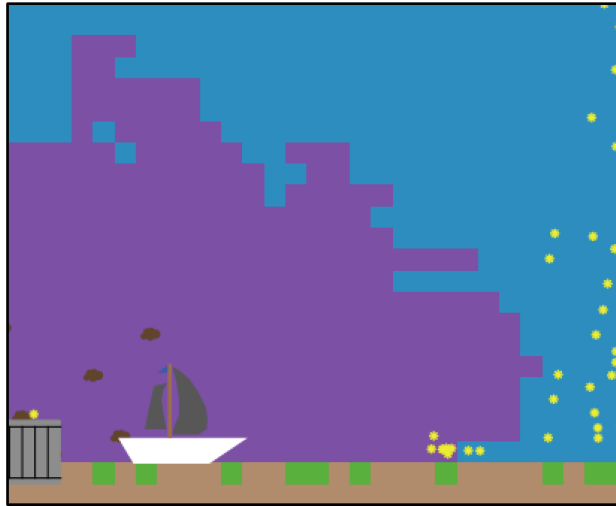


Figura 12 – Barco afundado e contaminação local da água após a libertação de toxinas.

O barco está também equipado com uma rede de pesca, que é lançada periodicamente para capturar peixes, demonstrado na Figura 13.

A ativação da rede é controlada por um temporizador interno (*tempo-rede*), que impede que a rede seja lançada continuamente.

Após o lançamento, é necessário aguardar 20 ticks antes que a rede possa ser utilizada novamente, simulando o tempo necessário para içar e preparar o equipamento.

Cada rede tem uma duração limitada (*tempo* = 5 ticks) e desaparece automaticamente quando o tempo chega a zero, conforme ilustrado no Anexo 2.9. Durante esse intervalo, a rede atua numa área de captura definida em torno da sua posição.

Os peixes que se encontram dentro dessa zona são removidos (morrem), representando o processo de pesca e redução da população aquática.

A profundidade a que a rede é lançada pode ser ajustada através do deslizador *profundidade-rede*, permitindo analisar o impacto da pesca em diferentes níveis da coluna de água.

Redes mais profundas capturam mais peixes, mas também aumentam o risco de desequilíbrio ecológico, afetando as espécies que normalmente se mantêm longe da superfície.

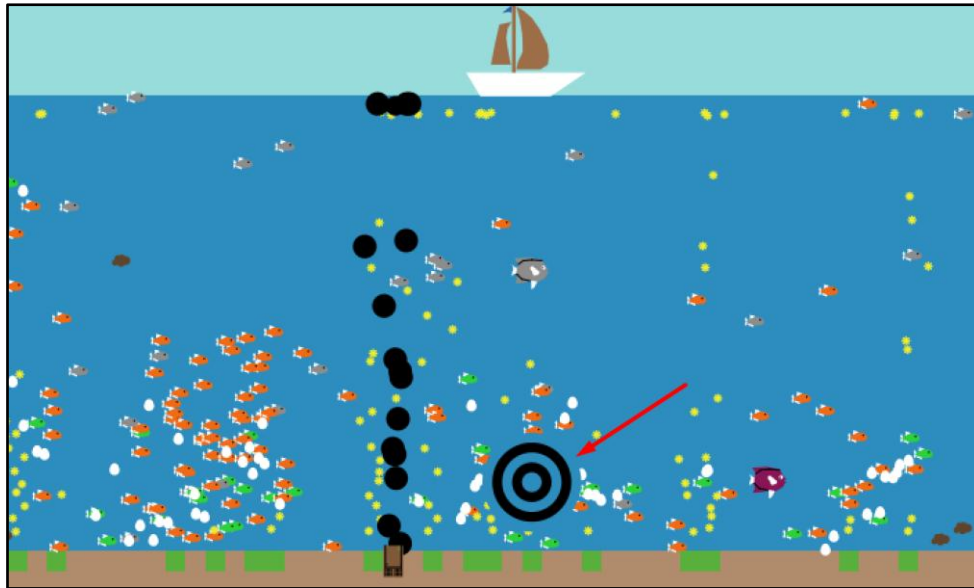


Figura 13 – Barco a lançar a rede de pesca, capturando peixes dentro da área definida pela profundidade configurada.

### 3.5. Ciclo de Dia e Noite

Para tornar o ecossistema mais dinâmico e aproximá-lo de um ambiente real, foi implementado um ciclo de dia e noite, controlado por um agente designado por astro.

Este agente representa o Sol e a Lua, alternando entre ambos à medida que percorre o céu.

A implementação deste ciclo permitiu introduzir variações visuais e comportamentais nos agentes, conferindo ritmo e realismo ao ecossistema.

O astro move-se horizontalmente ao longo do céu, de uma extremidade à outra do mundo, com um movimento suave e contínuo.

A sua trajetória é ligeiramente curvada, graças à utilização de uma função senoidal, que cria o efeito de subida e descida característico do movimento aparente do Sol e da Lua no horizonte, de acordo com o código apresentado no Anexo 2.10.

Quando o astro atinge o limite direito do ecrã, ocorre a transição entre o dia e a noite.

Neste momento, o astro troca de cor, amarelo durante o dia e cinzento à noite, e regressa à posição inicial, reiniciando o ciclo.

Simultaneamente, o céu altera a sua cor:

- Durante o **dia**, os *patches* superiores do mundo assumem um azul-claro (pcolor 87), representando na Figura 14.

- Durante **a noite**, esses mesmos *patches* mudam para um azul-escuro (pcolor blue - 2), ilustrado na Figura 15.

Esta variação cromática cria um efeito visual muito marcante, tornando o ambiente mais envolvente e facilitando a distinção entre os diferentes períodos do ciclo.

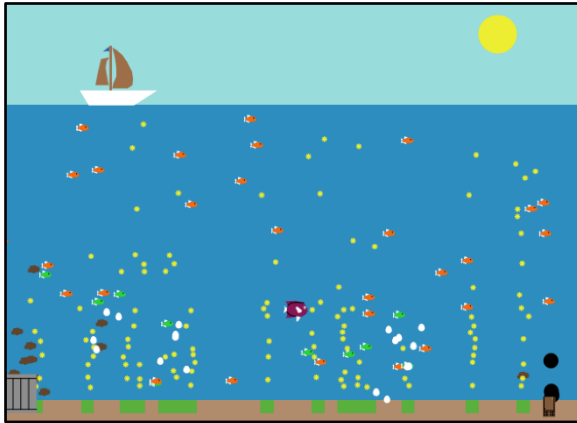


Figura 14 – Representação do ciclo de dia e noite no modelo Eco2, com o astro a percorrer o céu de dia.



Figura 15 – Representação do ciclo de dia e noite no modelo Eco2, com o astro a percorrer o céu de noite.

O ciclo de dia e noite tem um impacto direto no comportamento dos peixes, refletindo os padrões naturais de atividade das espécies marinhas. Durante o dia, os peixes herbívoros são mais ativos e deslocam-se com maior velocidade, explorando o ambiente em busca de alimento libertado pelas plantas. Nesta fase, os peixes predadores tornam-se mais lentos e menos agressivos, representando o período de menor caça. À noite, ocorre o inverso, os predadores tornam-se mais ativos e aumentam a sua velocidade de deslocação, enquanto os herbívoros reduzem o movimento, permanecendo próximos do fundo, cuja implementação pode ser consultada no Anexo 2.11, o que simula o comportamento de procura de refúgio durante períodos de menor luminosidade.

Esta alternância de comportamentos gera ciclos naturais de alimentação e repouso, tornando o ecossistema mais equilibrado e menos previsível. Além disso, o ciclo de luz influencia indiretamente a taxa de mortalidade: durante a noite, o aumento da atividade dos predadores resulta numa maior predação de peixes normais, o que por sua vez regula o crescimento populacional.

A introdução do ciclo de dia e noite permitiu criar um ecossistema com ritmos próprios, aproximando o modelo de condições reais observadas na natureza. Visualmente, o movimento do astro e a alteração da cor do céu enriquecem a

simulação, enquanto o impacto comportamental reforça a ideia de que os agentes reagem ao ambiente e adaptam-se às condições externas.

Esta funcionalidade contribui também para a noção de tempo contínuo dentro do modelo, permitindo observar efeitos cumulativos ao longo de vários ciclos, como o envelhecimento dos peixes, a variação na taxa de reprodução e a influência indireta nas dinâmicas populacionais.

### **3.6. Resultados e Observações**

Após a implementação do modelo Eco2, foi possível observar um conjunto de comportamentos emergentes que refletem o equilíbrio e a complexidade de um ecossistema aquático real.

O aumento do número de agentes e das interações entre eles permitiu que o sistema evoluísse de forma dinâmica e menos previsível, revelando padrões ecológicos e ciclos de adaptação entre as diferentes espécies.

Durante as simulações, verificou-se que as populações de peixes tendem a oscilar de forma natural ao longo do tempo.

Os peixes herbívoros reproduzem-se mais rapidamente, enquanto os predadores ajudam a controlar esse crescimento, evitando a sobrepopulação.

O mecanismo de ovos e o sistema de vida e energia mostraram-se eficazes para representar a sobrevivência e o envelhecimento das espécies, resultando num equilíbrio entre nascimento, alimentação e morte.

Mesmo quando uma espécie chega a níveis muito baixos, o ovo final de resgate permite a continuidade da população, simbolizando a resiliência do ecossistema.

A poluição, proveniente dos canos de esgoto e das falhas geológicas, exerce um impacto significativo na vida marinha.

As partículas de esgoto espalham-se rapidamente, enquanto o petróleo libertado pelas falhas e pelos barcos afundados causa uma mortalidade mais lenta, mas persistente.

Quando as taxas de limpeza (*probLimpeza*) são elevadas, observa-se uma recuperação gradual da qualidade da água, o que reduz a mortalidade e favorece a regeneração ambiental.

Por outro lado, quando a poluição é intensa e a regeneração é baixa, ocorre uma redução acentuada da população de peixes, evidenciando o efeito cumulativo das pressões ambientais.

O ciclo de dia e noite acrescenta uma componente rítmica e realista. Durante o dia, os peixes herbívoros tornam-se mais ativos e aumentam o consumo de alimento, enquanto os predadores reduzem a caça. À noite, a situação inverte-se, equilibrando a cadeia alimentar.

O barco e a rede de pesca introduzem uma forma de intervenção humana que altera este equilíbrio, reduzindo as populações locais e aumentando temporariamente a poluição quando ocorre um afundamento.

De forma geral, o modelo demonstrou ser autossustentável e ecologicamente coerente, exibindo um equilíbrio dinâmico entre os fatores de poluição, regeneração, reprodução e predação, representado na Figura 15.

Os resultados obtidos evidenciam que, mesmo num ambiente sujeito a perturbações constantes, o ecossistema tende a reorganizar-se e a recuperar, refletindo a capacidade adaptativa e resiliente dos sistemas naturais, ilustrado na Figura 16.

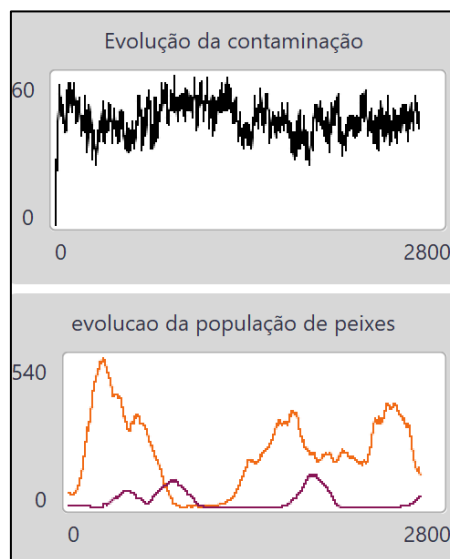


Figura 15 – Demonstração do equilíbrio ecológico no modelo Eco2.

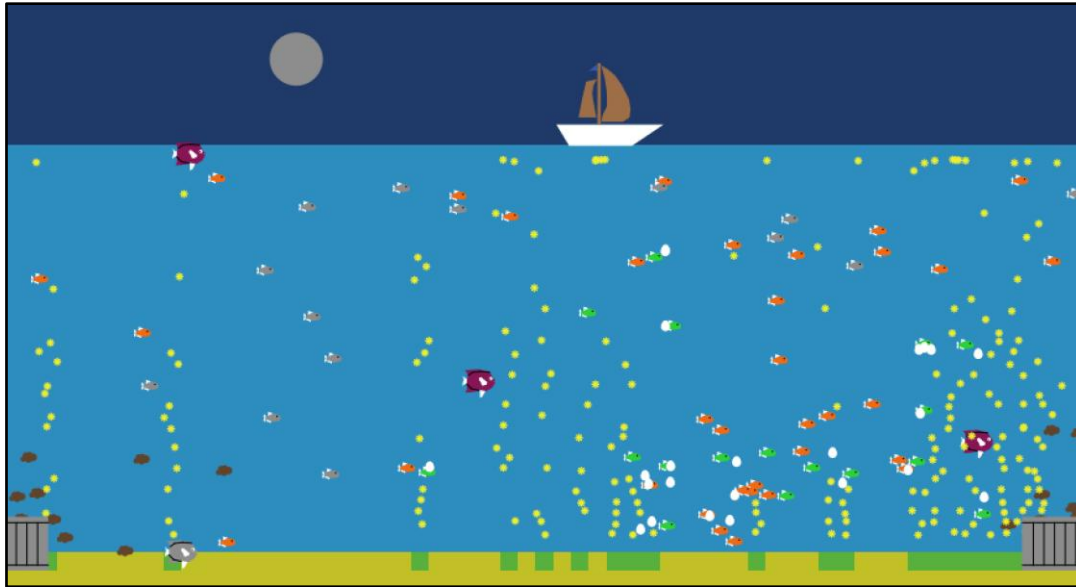


Figura 16 – Exemplo de evolução temporal da simulação Eco2: variação da população de peixes e do nível de poluição ao longo do tempo.

## 4. Conclusão

O desenvolvimento deste trabalho permitiu aplicar, de forma prática, os conceitos de sistemas multiagente na modelação de um ecossistema aquático.

Através da criação e evolução dos modelos Eco1 e Eco2, foi possível compreender como diferentes tipos de agentes, interagindo entre si e com o ambiente, originam comportamentos complexos e realistas, emergindo a partir de regras simples de atuação individual.

A transição do Eco1 para o Eco2 representou uma evolução significativa na complexidade e no realismo da simulação.

Enquanto o primeiro modelo se focava apenas na representação básica da poluição e no movimento dos peixes, o segundo integrou novos elementos ecológicos e comportamentais, como peixes predadores, alimentação e reprodução por ovos, poluição dinâmica através de turtles, falhas geológicas, barco com rede de pesca e um ciclo de dia e noite.

Estas adições permitiram observar um sistema mais equilibrado, onde a vida, a morte, a poluição e a regeneração coexistem de forma dinâmica.

Durante as simulações, o modelo demonstrou a capacidade de auto-organização e resiliência do ecossistema.

Mesmo perante perturbações, como o aumento da poluição, os derrames de petróleo ou o afundamento do barco, o sistema revelou uma tendência natural para o equilíbrio ecológico, graças aos mecanismos de reprodução, limpeza e

interação entre espécies.

A introdução do ciclo de dia e noite e do comportamento predador reforçou o caráter realista do ambiente, mostrando que a adaptação e a sobrevivência dependem tanto das condições externas como das estratégias individuais de cada agente.

Em suma, o trabalho atingiu os seus objetivos principais, modelar um ecossistema com múltiplos agentes, analisar a influência da poluição e da ação humana, e demonstrar como a utilização do NetLogo é eficaz na representação de fenómenos naturais complexos.

O modelo desenvolvido não só cumpre os requisitos propostos, como também oferece uma base sólida para futuras melhorias, como a introdução de novas espécies, fenómenos climáticos ou mecanismos de cooperação entre agentes.

## 5. Referências Bibliográficas

Wilensky, U. (1997). NetLogo. <http://ccl.northwestern.edu/netlogo/> . Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL

## 6. Anexo 1

### Anexo 1.1

```
to criacao_peixes
  ask peixes [
    right random 360
    forward 1
    if [pcolor] of patch-here != 95 [
      back 1
      right 180
    ]
  ]
end
```

### Anexo 1.2

```
to criacao_poluicao
  ;;sujar água perto do cano
  ask canos [
    ask patches in-radius 2 with [pcolor = 95] [
      if random-float 1 < (probPoluicao / 100) [
        set pcolor 34
      ]
    ]
  ]

  ;;patches castanhos contaminam vizinhos azuis
  ask patches with [pcolor = 34] [
    ask neighbors4 with [pcolor = 95] [
      if random-float 1 < (probPoluicao / 100) [
        set pcolor 34
      ]
    ]
  ]

  if ticks mod 3 = 0 [
    ;;Limpeza: cada patch castanho pode voltar a azul
    ask patches with [pcolor = 34] [
      if random-float 1 < (probLimpeza / 100) [
        set pcolor 35
      ]
    ]
    ask patches with [pcolor = 35] [
      if random-float 1 < (probLimpeza / 100) [
        set pcolor 95
      ]
    ]
  ]
end
```



```
]
end
```

## 7. Anexo 2

### Anexo 2.1

```
peixes-own [
  vida
  tipo
  idade ;; idade em ticks
]
to gastar-energia
  ask peixes [
    ;; aumenta idade a cada tick
    set idade idade + 1

    ;; decremento de energia
    if tipo = "predador" [
      set vida vida - 0.2
    ]
    if tipo = "normal" [
      set vida vida - 0.05
    ]

    ;; morte por idade
    if (tipo = "normal" and idade > 150) or
      (tipo = "predador" and idade > 200) or
      vida <= 0 [
      die
    ]
  ]
end
```

### Anexo 2.2

```
to por-ovos
  ;; peixes normais
  ask peixes with [tipo = "normal"] [
    if vida >= 90 [
      if random-float 100 < probPôrOvo [
        hatch-ovos 1 [
          set color white
          set shape "egg"
          set size 1
          set ticks-para-eclodir tempoOvo
          set tipo-ovo "normal"
        ]
      ]
    ]
  ]

  ;; predadores
  ask peixes with [tipo = "predador"] [
```

```

if vida >= 90 [
  ;; menos chance de por ovos
  if random-float 100 < probPôrOvo * 0.9 [
    hatch-ovos 1 [
      set color (magenta + 1)
      set shape "egg"
      set size 1.5
      set ticks-para-eclodir tempoOvo
      set tipo-ovo "predador"
    ]
  ]
]
end
to atualizar-ovos
ask ovos [
  set ticks-para-eclodir ticks-para-eclodir - 1
  if ticks-para-eclodir <= 0 [
    if tipo-ovo = "normal" [
      hatch-peixes 1 [
        set vida 50
        set color orange
        set shape "fish"
        set size 1
        set tipo "normal"
        setxy [xcor] of myself [ycor] of myself
      ]
    ]
    if tipo-ovo = "predador" [
      hatch-peixes 1 [
        set vida 50
        set color (magenta - 1)
        set shape "fish 2"
        set size 2
        set tipo "predador"
        setxy [xcor] of myself [ycor] of myself
      ]
    ]
  ]
  die
]
end

```

## Anexo 2.3

```

to comer-comida
ask peixes with [tipo = "normal"] [
  let comida-no-patch one-of comidas-here
  if comida-no-patch != nobody [
    ask comida-no-patch [ die ]
    set vida vida + 10
    if vida > 100 [ set vida 100 ] ;; vida máxima
  ]
  set vida vida - 0.2 ;; decremento
  if vida <= 0 [ die ]
]
end
to comer-predador
ask peixes with [tipo = "predador"] [

```

```
;; comer ovos (menos ganho)
let ovo-no-patch one-of ovos-here with [tipo-ovo != "predador"]
if ovo-no-patch != nobody [
  ask ovo-no-patch [ die ]
  set vida vida + 5 ;; menos ganho que antes
  if vida > 100 [ set vida 100 ]
]

;; comer peixes normais (menos ganho)
let peixe-no-patch one-of peixes-here with [tipo = "normal"]
if peixe-no-patch != nobody [
  ask peixe-no-patch [ die ]
  set vida vida + 10 ;; menos ganho que antes
  if vida > 100 [ set vida 100 ]
]
]
end
```

## Anexo 2.4

```
to resgatar-peixes
;; Resgatar peixes normais
if count peixes with [tipo = "normal"] = 0 [
  let n-ovos 1 + random 2 ;; 1 ou 2 ovos
  repeat n-ovos [
    let p one-of patches with [pcolor = 95]
    if p != nobody [
      ask p [
        ;; em vez de criar ovo, cria peixe diretamente
        sprout-peixes 1 [
          set vida 50
          set color orange
          set shape "fish"
          set size 1
          set tipo "normal"
          set idade 0
        ]
      ]
    ]
  ]
]

;; Resgatar predadores
if count peixes with [tipo = "predador"] = 0 [
  let n-ovos 1 + random 2
  repeat n-ovos [
    let p one-of patches with [pcolor = 95]
    if p != nobody [
      ask p [
        sprout-peixes 1 [
          set vida 50
          set color (magenta - 1)
          set shape "fish 2"
          set size 2
          set tipo "predador"
          set idade 0
        ]
      ]
    ]
  ]
]
```

```
]
]
end
```

## Anexo 2.5

```
to verificar-falha
;; só cria uma falha se ainda não existir
if not falha-ativa? [
  if random-float 100 < probFalha [
    criar-falha
  ]
]
end

to criar-falha
;; Limites da zona central em x
let x_min (min-pxcor * 0.3)
let x_max (max-pxcor * 0.7)

;; Patches de fundo dentro da zona central
let fundo-central patches with [
  pcolor = 44 and
  any? neighbors with [pcolor = 95] and
  pxcor >= x_min and pxcor <= x_max
]

if any? fundo-central [
  ask one-of fundo-central [
    sprout-falhas 1 [
      set shape "computer server"
      set color (brown - 1)
      set size 2
    ]
  ]
  set falha-ativa? true
  set tremor-contador tempo-tremor
  mostrar-tremor
]
end

to libertar-petroleo
ask falhas [
  ;; petróleo surge com menor probabilidade
  if random-float 100 < 20 [ ;; 5% de chance por tick
    ask patch-here [
      sprout-petroleos 1 [
        set shape "circle"
        set color black
        set size 1.2
        set heading random 360 ;; direção inicial aleatória
      ]
    ]
  ]
]
end

to act-petroleo
ask petroleos [
  ;; patch acima (pode ser nobody se no topo)
```

```
let acima patch-at 0 1

;; se ainda estiver debaixo da superfície (ycor < superfície)
if ycor < max-pycor * 0.5 [
  ;; sobe lentamente em cone
  set ycor ycor + 0.3
  set xcor xcor + (random-float 0.4 - 0.2)
]
;; se chegou à superfície
if ycor >= max-pycor * 0.5 [
  ;; movimentação lenta horizontal
  set xcor xcor + (random-float 0.2 - 0.1)

  ;; espalhar petróleo ao redor lentamente
  if random-float 100 < 5 [
    let vizinho one-of patches in-radius 1 with [pcolor = 95 and not any?
petroleos-here]
    if vizinho != nobody [
      ask vizinho [
        sprout-petroleos 1 [
          set shape "circle"
          set color black
          set size 1
        ]
      ]
    ]
  ]
]

;; limpeza natural
if random-float 100 < probLimpeza * 0.05[
  die
]
end
```

## Anexo 2.6

```
to peixe-petroleo
ask peixes [
  let petroleo-aqui one-of petroleos-here
  if petroleo-aqui != nobody [
    ;; peixe entra em contato com petróleo → perde 30 de vida
    set vida vida - 30

    ;; atualiza cor conforme a vida
    if vida <= 0 [
      die
    ]
  ]
]
end
```

## Anexo 2.7

```

to -petroleo
ask barcos [
;; coleta petróleo se ainda não estiver cheio
if contaminacao < 100 [
let petroleo-perto one-of petroleos in-radius 3
if petroleo-perto != nobody [
if random-float 100 < 8 [
set contaminacao contaminacao + 1
if contaminacao > 100 [ set contaminacao 100 ]
ask petroleo-perto [ die ]
]
set color scale-color brown (100 - contaminacao) 0 100
]
]

;; se contaminação chega a 100 → pára e começa a afundar
if contaminacao >= 100 [
set velocidade 0
afundar-barco
]
end

```

## Anexo 2.8

```

to afundar-barco
ask barcos with [velocidade = 0] [
;; procura todos os patches abaixo do barco que representam areia
let patches-debaixo patches with [pycor < [ycor] of myself and (pcolor =
44 or pcolor = (brown + 1))]

if contaminacao >= 100 and not afundado? [
set velocidade 0
set afundado? true ;; marca que já afundou
afundar-barco
]

if any? patches-debaixo [
let topo-areia max [pycor] of patches-debaixo ;; topo da areia
let alvo topo-areia + 2 ;; para 5 acima da areia

if ycor > alvo [
set ycor ycor - 0.2 ;; desce gradualmente
]
if ycor <= alvo [
set ycor alvo
set color gray - 2
]
]
]
end
to contar-ticks-apos-afundar
ask barcos with [afundado? and not pronto-para-contaminar?] [
set tempo-para-contaminar tempo-para-contaminar + 1
if tempo-para-contaminar >= 100 [
set pronto-para-contaminar? true
]
]
]

```

```

end
to contaminar-agua
  ask barcos with [pronto-para-contaminar?] [
    ;; seleciona patches de água próximos do barco
    let agua-proxima patches in-radius 2 with [pcolor = 95]

    ;; muda a cor deles para indicar poluição
    ask agua-proxima [
      ;; podemos usar violet como exemplo, ou outro tom
      set pcolor violet
    ]
  ]
end
to espalhar-poluicao
  ;; pega todos os patches que já estão contaminados
  let poluidos patches with [pcolor = violet]

  ask poluidos [
    ;; pega vizinhos limpos (água)
    let vizinhos-limos patches in-radius 1 with [pcolor = 95]
    ask vizinhos-limos [
      ;; cada patch vizinho tem uma chance de ser contaminado
      if random-float 100 < 2 [ ;; 2% de chance, ajusta como quiser
        set pcolor violet
      ]
    ]
  ]
end

```

## Anexo 2.9

```

to lancar-rede
  if tempo-rede = 0 [ ;; só lança se o timer chegou a 0
    ask barcos with [contaminacao < 90] [ ;; só barcos com poluição menor
que 90
      hatch-redes 1 [
        set tempo 5 ;; duração da rede
        set color 95
        set shape "target"
        set size 5
        setxy [xcor] of myself [ycor] of myself - profundidade-rede
      ]
    ]
    set tempo-rede 20 ;; espera 20 ticks até lançar de novo
  ]
end

to atualizar-redes
  ask redes [
    set tempo tempo - 1 ;; decreuenta o tempo
    if tempo <= 0 [ die ] ;; desaparece quando o tempo acabar
  ]
end

to rede-captura-peixes
  ask redes [
    ;; só redes de barcos com velocidade > 0 pescam
    let barco-mestre one-of barcos with [velocidade > 0 and distance myself <
1]
  ]
end

```

```

if barco-mestre != nobody [
  let largura size * 0.5
  let profundidade 2
  let peixes-a-capturar peixes with [
    xcor >= [xcor] of myself - largura and
    xcor <= [xcor] of myself + largura and
    ycor <= [ycor] of myself and
    ycor >= [ycor] of myself - profundidade
  ]
  ask peixes-a-capturar [ die ]
]
end

```

## Anexo 2.10

```

to mover-astro
ask astros [
  ;; movimento no X
  set xcor xcor + 1

  ;; largura do caminho
  let largura (max-pxcor - min-pxcor)

  ;; converte a posição em fase de 0 a 180 graus
  let fase ((xcor - min-pxcor) / largura) * 180

  ;; agora limitamos o Y para ficar só no céu
  let base max-pycor * 0.65 ;; altura média no céu
  let amplitude max-pycor * 0.25 ;; o quanto sobe acima da base
  set ycor base + (sin fase * amplitude)

  ;; quando chega à margem direita → troca entre sol e lua
  if xcor >= max-pxcor [
    set astro-dia? not astro-dia?
    ifelse astro-dia? [
      set color yellow
      mudar-ceu-dia
    ] [
      set color gray
      mudar-ceu-noite
    ]
    set xcor min-pxcor
  ]
]
end

```

## Anexo 2.11

```

to move-peixes
ask peixes [
  rt random 360

  if tipo = "normal" [
    ;; movimento mais lento à noite
    ifelse astro-dia? [
      fd 1
    ]
  ]
]

```



```
    ] [
      fd 0.4
    ]
  ]

  if tipo = "predador" [
    ;; predadores mais ativos à noite
    ifelse astro-dia? [
      fd 0.7
    ] [
      fd 1
    ]
  ]

  ;; impedir que saiam da água
  if not member? [pcolor] of patch-here [95 violet] [
    bk 1
    rt 180
  ]

  ;; morte natural
  if random-float 100 < probMortePeixe [
    die
  ]
]
end
```