

AASMA 2017 Relatório

Trânsito com Veículos Autónomos

Luís Espírito Santo
Instituto Superior Técnico
Lisboa, Portugal
luis.a.santo@tecnico.
ulisboa.pt

João Moreira
Instituto Superior Técnico
Lisboa, Portugal
joao.pedro.lavadinho.moreira
@tecnico.ulisboa.pt

Valentyn Hulevych
Instituto Superior Técnico
Lisboa, Portugal
valentyn.hulevych
@tecnico.ulisboa.pt

ABSTRACT

A motivação para a realização deste projecto foi a aplicação da matéria leccionada na cadeira de *Agentes Autónomos e Sistemas Multi-Agente* integrada no *Mestrado em Engenharia Informática e de Computadores* do Instituto Superior Técnico. Sendo a interação humana um sistema tão complexo e dinâmico, uma abordagem viável é a restrição e divisão do problema em problemas mais simples, focando o estudo em interações inseridas em contexto específicos mais controlados. De entre várias propostas, chamou-nos a atenção o contexto da circulação rodoviária uma vez que é algo actual, com o surgimento dos carros autónomos e o consequente desaparecimento de certos elementos reguladores da movimentação dos veículos, tais como semáforos. O nosso modelo pretende apresentar uma abordagem a estas condições de navegação automóvel em condições controladas, sendo que como objetivos principais temos a redução da sinistralidade, a fluência do sistema e a eficiência dos agentes que, em parte, condiciona os anteriores.

Keywords

Agentes Autónomos, Sistemas Multi-agentes, Controlo de Tráfego Automóvel

1. INTRODUÇÃO

No contexto da movimentação de carros existem elementos coordenadores que visam, acima de tudo, a segurança dos condutores e peões. Com o surgimento de paradigmas de condução automática, estes mediadores (semáforos, sinais verticais, sinais horizontais...) poderão ser dispensados, se outras formas melhores de coordenação forem encontradas e implementadas. Neste projecto pretendemos explorar algumas opções de modelação de um sistema multiagente de tráfego automóvel, cujos agentes apresentam uma variedade de sensores e atuadores que consideramos não limitadora para a simulação de um ambiente de trânsito.

O desenvolvimento será incremental tendo como ponto inicial um modelo em NetLogo ao qual se pretende acrescentar funcionalidades. Posteriormente será feita a análise comparativa entre vários modelos de forma a perceber o impacto das melhorias propostas.

Appears in: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017)*, S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.

Copyright © 2017, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

2. DESCRIÇÃO DO PROBLEMA

2.1 Modelo Inicial

O ponto de partida consiste num modelo de trânsito em movimento numa grelha de cidade. Este permite controlar os semáforos e variáveis globais, tais como a velocidade limite e o número de carros, e explorar dinâmicas de trânsito. Os carros iniciais estão implementados como simples agentes reativos que em cada instante de tempo tentam mover-se à sua velocidade atual e que se esta for menor que a velocidade limite e não existir um carro em frente, então aceleram. Se existe um carro mais lento à sua frente, eles assumem a mesma velocidade. Caso exista luz vermelha no próximo semáforo, os carros param.

2.2 Aspectos a incluir no modelo

O trânsito passará a ser com carros autónomos, ou seja, cada carro irá calcular a maneira de chegar autonomamente a um objetivo estipulado. Pretende-se retirar os semáforos. Esta opção obriga à implementação de mecanismos de negociação para se realizarem as passagens nos cruzamentos. Os carros irão passar a utilizar planeamento para calcular o trajecto até ao estado objetivo. A ideia é tornar os agentes mais deliberativos.

3. FORMALIZAÇÃO E MODELAÇÃO

Nesta secção irão ser descritas as implementações e alterações efetuadas sobre o modelo original. Começando com a descrição de pequenas funcionalidades base e acabando nas mais complexas, irão ser abordados todos os aspectos da implementação que abrange a solução final.

3.1 Descrição dos Agentes

Os agentes iniciais sobre os quais construímos o nosso sistema apresentavam um comportamento puramente reactivo. Com base nestes criámos agentes híbridos, incluindo no processo de escolha das ações algum planeamento a longo, médio e curto prazo no seu comportamento. Os agentes finais apresentam um comportamento híbrido entre deliberativo e reactivo. Criámos diferentes denominações para agentes com diferentes níveis de compromisso para com o plano:

- **Blind Agent:** este agente planeia o seu caminho antes de partir e mantém-se fiel a este plano até ao final da sua viagem;
- **Struggler Agent:** neste caso o agente pondera replanejar a sua trajectória quando se vê num caso de stress (ex. muito trânsito);

- **Provident Agent:** o agente verifica em espaços de tempo regulares se o trajeto escolhido continua a ser a melhor opção;

Algo que deve ser apontado é que todos os tipos de agentes aqui explorados podem ser de alguma forma comparados com diferentes perfis reais de condutores humanos.

Os agentes também tem um nível de perseverança, **Perseverance**, que significa a quantidade de vezes que o agente pondera replanear mas não o faz.

3.2 Descrição do Ambiente

O ambiente do nosso modelo é acessível pois os agentes conseguem obter informação completa dos estados do ambiente. Ao fazer o planeamento, por exemplo, o agente sabe automaticamente o percurso que tem de fazer com informação obtida do estado atual. O nosso ambiente é determinístico, uma vez que todas as ações do agente têm apenas um único efeito, mudar de direção para a direita ou para baixo. O nosso ambiente também é considerado estático. Sempre que o agente delibera e/ou toma uma ação o estado não muda, ou seja os carros não se mexem. Esta característica não se verifica num contexto de tráfego real.

O nosso ambiente é discreto, dado que nesta simplificação existe um número finito de ações e percepções que cada agente pode executar e obter. Por fim o nosso ambiente pode ser classificado como episódico, o tempo de execução dos agente pode ser dividido em intervalos independentes com condições iniciais semelhantes.

3.3 Sensores e Atuadores

3.3.1 Atuador de Mudança de Direção

Numa primeira abordagem ao problema era evidente a necessidade de acrescentar algumas funcionalidades básicas ao modelo inicial. Os veículos circulavam sempre no mesmo sentido, tornando o modelo bastante primitivo. De forma a aproximar o modelo ao mundo real, foi acrescentada um atuador de mudança de direção, permitindo assim uma maior liberdade na escolha dos trajetos.

Ainda que fosse desejado ter um modelo que simulasse o movimento em ambos os sentidos e a deslocação em todas as direções, devido aos prazos de entrega e ao nível de complexidade que seria introduzido, optou-se por utilizar um modelo que, à semelhança do inicial, só permitiria movimentos da esquerda para a direita e de cima para baixo, com a diferença de que os veículos tinham a possibilidade de mudar a sua direção, escolhendo um dos dois sentidos possíveis em cada instante.

Esta alteração introduzida nas opções de atuação dos veículos permitirá que o agente tivesse então a possibilidade de explorar todo o ambiente, consoante o seu planeamento.

3.3.2 Sensor do Objetivo

Após introduzir as alterações necessárias no movimento do agente, a próxima fase seria procurar acrescentar objetivos de forma a que o agente tivesse uma funcionalidade enquanto elemento do cenário em simulação. Tal como no mundo real, ao circular nas estradas, cada veículo tem como objetivo alcançar uma certa localização. Desta forma, foi implementada uma função objetivo que atribui uma posição do mapa a um carro, diferente da atual, e que é considerado o destino desejado. Este destino é gerado aleatoriamente entre todas as estradas disponíveis e permite criar agentes cujo

objetivo é alcançar uma posição aleatória, levando assim a um comportamento que simula as deslocações humanas. Alcançado o objetivo, é gerado um novo, pois a ideia principal é poder estudar aprofundadamente o comportamento dos agentes consoante as várias implementações e com mais iterações é possível fazer uma melhor análise.

3.3.3 Sensor e Actuadores de Velocidade

Como resultado da negociação surgem a necessidade de ajustes de velocidade mais orientados à resolução de conflitos nos cruzamentos. Cada agente em cada instante tem acesso aos dados do veículo com que está em conflito direto e verifica se é necessário ceder a passagem ou se tem prioridade. O que faltaria então, era uma função que dadas as posições e as velocidades fosse capaz de responder se os dois veículos iriam colidir. E com o intuito de conseguir exatamente esse resultado foi implementada essa mesma função, que em caso de colisão e necessidade de cedência de passagem, abrandava a velocidade do veículo para metade da velocidade atual. Os ajustes de velocidade já existentes, nomeadamente a reação de abrandar caso exista um veículo à frente e no mesmo sentido, foram mantidos similares ao modelo original, com a diferença que no novo modelo são tidas em conta as duas posições mais à frente. Deste modo o novo modelo introduz o conceito de “distância de segurança”, dando maior capacidade de manobra aos agentes em caso de conflito direto e diminuindo o perigo de sinistro.

3.4 Negociação

Uma funcionalidade crítica para todo o trabalho acaba por ser a negociação. Tendo em mente conseguir criar agentes autónomos e que sejam capazes de atingir os seus objetivos sem qualquer intervenção humana ou ajuda de estruturas externas como semáforos, surge a obrigação de pensar numa solução onde a negociação é indispensável. Estes agentes devem ter a capacidade de comunicar uns com os outros e tomar decisões em conjunto. Tendo todos estes aspectos em conta foi implementada uma função que, em cada instante e para cada agente permitia saber quantos veículos existiam a caminho do mesmo cruzamento. Não existindo ainda a distinção entre agentes em direções diferentes, a função serve essencialmente para saber se é necessário abrandar devido a um veículo mais à frente ou para reportar a posição em que o agente se encontra no cruzamento.

Agentes que circulam em direções perpendiculares apenas tem em conta o veículo em conflito direto, e toda a negociação será feita entre estes dois veículos. Deste modo foi utilizada uma segunda função que fazia uma análise às ordens dos carros em cada direção e atribuía um veículo, caso existisse, que estivesse em conflito direto. Decididos os pares em cada cruzamento, seria então necessário decidir quem iria passar primeiro. Para isso teve-se em conta as distâncias quando existiam velocidades nulas, e as distâncias a dividir pelas velocidades nos restantes casos. Veículos com maior distância e menor velocidade serão sujeitos a ceder a passagem. Caso exista um caso de empate são utilizados os identificadores únicos de cada carro.

3.5 Planeamento

3.5.1 Algoritmo

Como algoritmo de planeamento usámos o algoritmo de procura A*. Este recebe um estado inicial, um estado obje-

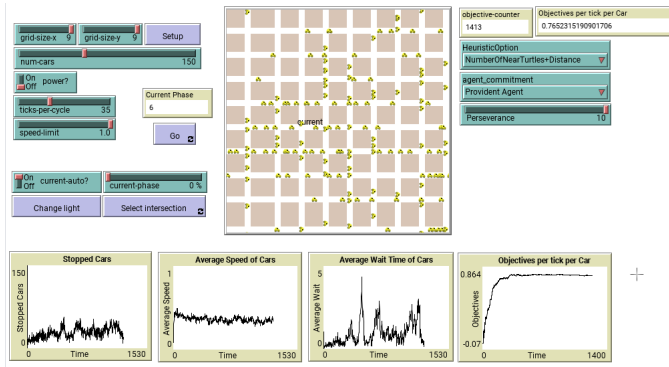


Figure 1: Exemplo da interface

tivo e um conjunto de células do mapa válidas para o carro navegar. Este algoritmo de planeamento é usado em três situações diferentes no nosso modelo. A primeira situação é na inicialização do modelo. Depois de todos os carros terem sido gerados, o A* é calculado para cada um deles para que assim o modelo comece já inicializado. Uma segunda situação onde o A* é usado é quando um carro chega a um objetivo. Assim que o agente atinge um objetivo, ele recebe um novo e usa novamente o A* para planejar o novo percurso, permitindo que a execução do modelo nunca pare. A terceira e última situação tem a ver com o facto de que o A* tem em conta no seu precedimento um ambiente estático. Contudo, depois da execução do plano o estado do ambiente altera-se e para colmatar este problema, o A* pode ser calculado novamente, de tempos a tempos, de forma a verificar se o caminho planeado continua a ser o trajecto ótimo. No modelo inicial os carros só podiam navegar numa direcção, logo foi necessário implementar a forma como o carro escolhe a próxima célula para onde ir. Como os carros só andam em frente, e o único local onde se pode virar é nos cruzamentos, a decisão é só feita neles. Vira para baixo ou para a direita de acordo com as coordenadas da célula seguinte.

3.5.2 Função de Custo

Uma única função de custo foi tida em conta que apenas tem em conta a distância percorrida. A fase de planeamento não possui mecanismos para prever a duração da viagem tendo em conta o trânsito e por isso não foram exploradas diferentes funções de custo.

3.5.3 Heurística

Foram implementadas duas heurísticas diferentes mas incrementais, ou seja, uma delas utiliza a outra. No início foi utilizada uma heurística simples definida apenas pela distância até ao objetivo, a qual denominámos **OnlyDistance**. Na segunda abordagem, para além da distância, também foi tido em conta o número de carros perto de determinado ponto. Com isto pretendia-se melhorar a distribuição de agentes pelo mapa, diminuindo o tempo de espera, as paragens e aumentando a velocidade média. Esta segunda heurística foi nomeada **NumberOfNearTurtles+Distance**.

4. INTERFACE

4.1 Operações

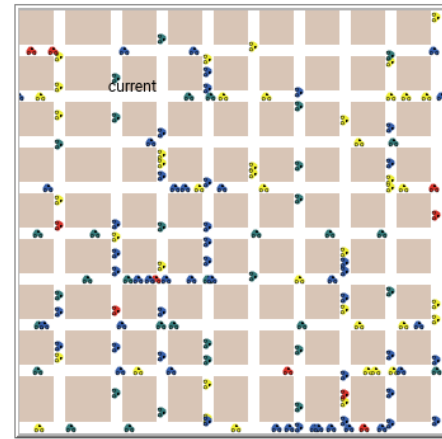


Figure 2: Exemplo ilustrativo do código de cores

Do lado esquerdo da janela da apresentação gráfica do nosso ambiente é apresentada a interface de base, onde se permite fazer o *Setup* e começar a correr, *Go*, escolher as dimensões do sistema, *grid-size-x* e *grid-size-y*, e o número de carros, *num-cars*. Também é possível ligar e desligar os semáforos, com o switch *power?*, alterar opções dos mesmos e configura-los individualmente. Toda a interface é apresentada na **Figure 1**. A interface para as novas funcionalidades é apresentada do lado direito, onde podemos escolher a heurística que queremos que os agentes utilizem no seu planeamento, onde também podemos escolher o tipo de agentes (ver 3.1) que queremos a circular e o nível de perseverança dos agentes.

4.2 Códigos de Cores

A nossa implementação utiliza um código de cores para identificar agentes em diferentes condições. Os carros que já atingiram pelo menos um objetivo são apresentados em amarelo, enquanto que os restantes são apresentados a azul se apresentam velocidade baixa, ciano para velocidades mais altas e a vermelho se precisaram de replanear o seu trajeto. As cores utilizadas podem ser observadas na **Figure 2**.

4.3 Gráficos e Contadores

Na janela de controlo também são apresentados alguns gráficos que permitem uma melhor análise do sistema. Por ordem da esquerda para a direita são apresentados: um gráfico que representa o número de carros parados ao longo do tempo, uma representação da velocidade média ao longo do tempo, a curva do tempo de espera médio e, por fim, a variação do número de objectivos conseguidos por carro por tick. Os três contadores que se encontram na mesma janela, na mesma ordem, apresentam respectivamente: a fase actual dos semáforos, o número de objectivos conseguidos e o número de objectivos conseguidos por carro por tick.

5. AVALIAÇÃO DE RESULTADOS

Nesta secção serão apresentados e discutidos alguns resultados obtidos. Todos os gráficos apresentados nesta secção serão conseguidos da execução de cerca de 1200 ticks do programa com 150 carros numa grelha de 9 por 9 com velocidade limite de 1. Estes números foram escolhidos experimentalmente de forma a melhorar os resultados represen-

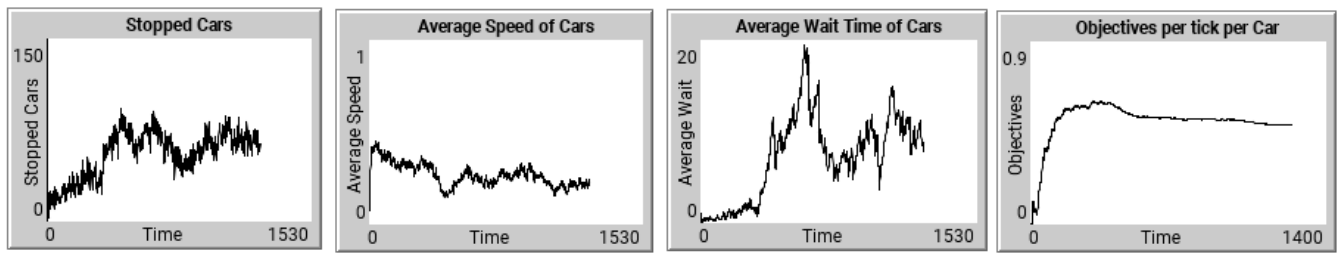


Figure 3: Gráficos de um agente Struggler utilizando a heurística OnlyDistance

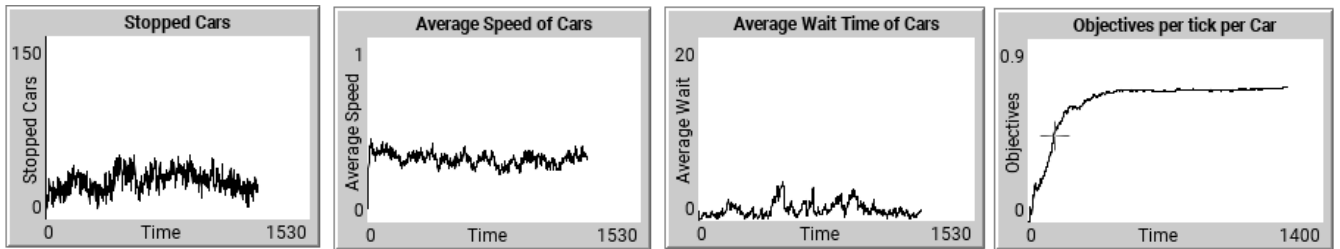


Figure 4: Gráficos de um agente Struggler utilizando a heurística NumberOfNearTurtles+Distance

tativos expostos no resto deste documento. O número de perseverança mantém-se no 5 ao longo de todos os testes apresentados excepto quando informado algo em contrário.

5.1 Semelhanças com a Realidade

Após a implementação das novas funcionalidades, consideramos o modelo final bastante mais próximo de um caso real de circulação rodoviária do que o modelo inicial. Ao permitir a mudança de direção e ao fornecer um sensor da localização de um objetivo tornamos o nosso modelo mais semelhante ao caso de um humano que pretende ir a algum sítio podendo escolher vários caminhos para o fazer.

5.2 Comparação de Heurísticas

Depois de implementadas as duas heurísticas verificámos que a utilização da heurística **OnlyDistance** apresentava resultados semelhantes para qualquer um dos tipos de agentes, o que vai de encontro ao esperado pois o valor da heurística não varia ao longo do tempo, depende apenas da localização espacial e desta forma o replaneamento não permite melhorar a performance do sistema. Esta semelhança de resultados, torna a heurística inútil para avaliação do compromisso do agente para com o plano uma vez que ela transporta com ela sempre as mesmas desvantagens independentemente do tipo de agente. Algumas dessas desvantagens são problemas de fluidez e estabilidade do sistema, o sistema pode ter uma boa velocidade média mas esse valor pode descer muito repentinamente.

A heurística **NumberOfNearTurtles+Distance**, na maioria dos casos, apresenta um menor número de carros parados, uma maior velocidade média, um maior número de objetivos atingidos e uma maior estabilidade.

Podemos comparar e verificar a instabilidade resultante da utilização da heurística **OnlyDistance** nos gráficos das Figuras 3 e Figura 4.

5.3 Comparação de Políticas de Compromisso

Tal como descrito na subsecção 3.1, foram implementa-

dos três agentes que diferem na sua política de compromisso com o plano: um **Blind Agent**, um **Struggler Agent** e por fim um **Provident Agent**, cujos os gráficos de uma simulação estão representados, respetivamente nas Figura 5, Figura 6 e Figura 7. Para avaliar a performance e eficiência dos agentes foi apenas utilizada heurística **NumberOfNearTurtles+Distance** e comparando os três tipos de agentes verificamos que quanto mais regular o planeamento também maior a performance, fluidez e regularidade do sistema. No entanto a eficiência da simulação decresce com agentes que planeiam mais frequentemente devido ao tempo do algoritmo de planeamento.

Sobre a **Perseverance**, um nível de perseverança muito elevado faz com que os agentes não replaneiem tão frequentemente como poderia ser necessário para manter o sistema sem problemas. O mesmo acontece com níveis de perseverança muito baixos, pois para esses valores o agente pode replanear a cada passo levando o agente a "ficar indeciso" e voltar para trás demasiadas vezes. Este efeito é o que leva aos gráficos das Figura 8 e Figura 9 demonstrem valores de menor performance que a Figura 7.

5.4 Semáforos e Negociação

Após a implementação da negociação, a utilização dos semáforos parece algo apenas perturbador da fluência do sistema. As medidas de sinistralidade não são muito influenciadas.

6. CONCLUSÕES

A modelação deste sistema propõe uma abordagem á circulação automóvel sem dispositivos de regulação de tráfego.

O desenvolvimento deste projecto forneceu-nos uma base prática para o desenvolvimento de sistemas multi-agente, assim como ferramentas para a modelação de sistemas complexos. Algo a explorar em trabalhos futuros no seguimento deste projecto, seria a coexistência dos diferentes tipos de agentes no mesmo ambiente e novos protocolos de negociação para veículos prioritários.

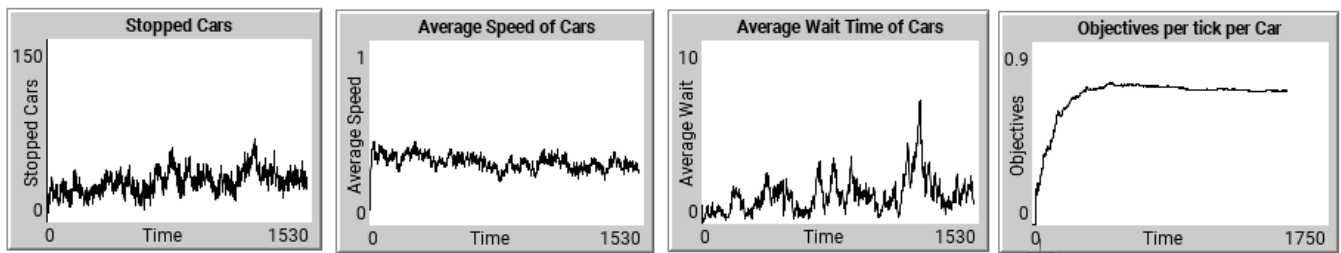


Figure 5: Gráficos de um agente Struggler utilizando a heurística `NumberOfNearTurtles+Distance`

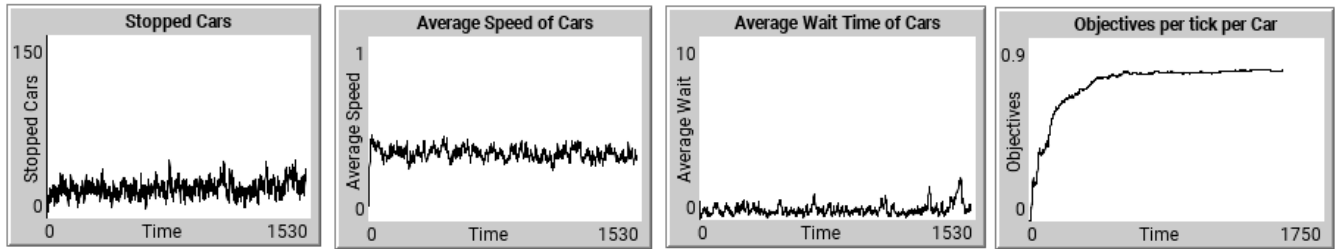


Figure 6: Gráficos de um agente Struggler utilizando a heurística `NumberOfNearTurtles+Distance` e com valor 5 de perseverança

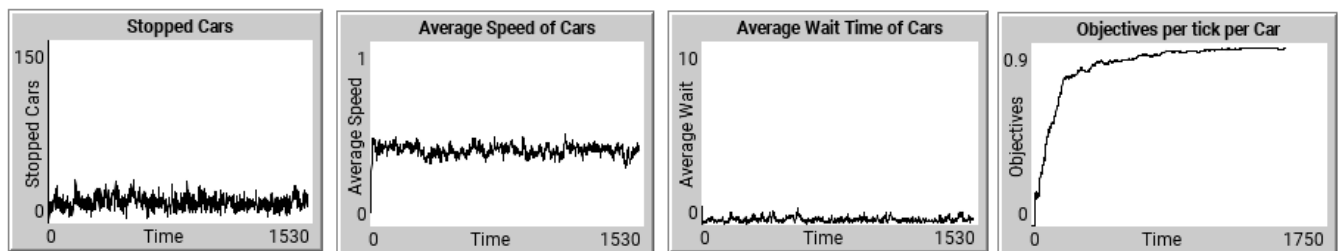


Figure 7: Gráficos de um agente Provident utilizando a heurística `NumberOfNearTurtles+Distance` e com valor 5 de perseverança

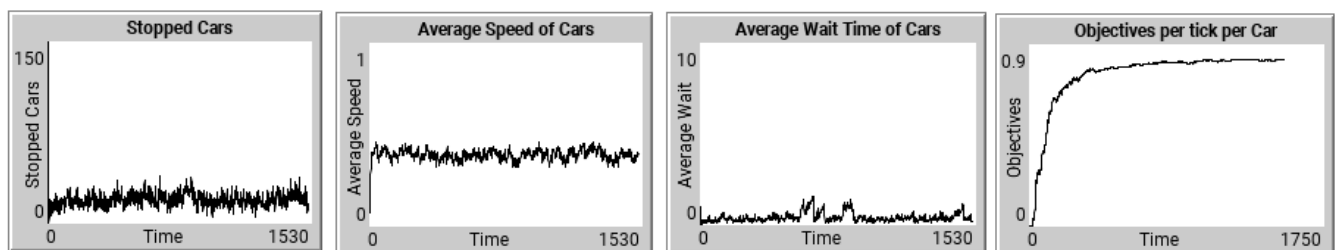


Figure 8: Gráficos de um agente Provident utilizando a heurística `NumberOfNearTurtles+Distance` e com valor 10 de perseverança

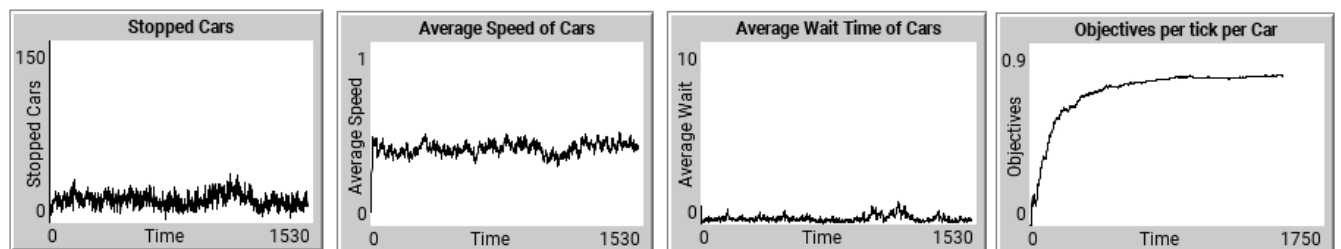


Figure 9: Gráficos de um agente Provident utilizando a heurística `NumberOfNearTurtles+Distance` e com valor 2 de perseverança