

**MINISTÉRIO DA DEFESA
EXÉRCITO BRASILEIRO
DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA
INSTITUTO MILITAR DE ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO E
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE DEFESA**

EQUIPE IME PROJETO S2C2

DEMONSTRAÇÃO DA PROVA DE CONCEITO

**RIO DE JANEIRO
NOVEMBRO/2025**

SUMÁRIO

1	INTRODUÇÃO	2
2	DEMONSTRAÇÃO	3
2.1	DESCRIÇÃO DO CENÁRIO	3
2.2	EXPERIMENTOS E ANÁLISES	5
2.3	APLICAÇÃO DA ONTOLOGIA NO S2C2 EMUSIM - <i>COMMAND AND CONTROL SIMULATION CONFIGURATION AND ORCHESTRATION</i>	10
3	CONCLUSÃO E TRABALHOS FUTUROS	12
3.1	POSSÍVEIS EXTENSÕES DO MISCMANAGER	12
3.2	UTILIZAÇÃO DA MISCON NOS APLICATIVOS DE C2	13
3.3	EXTENSÕES NA ONTOLOGIA MISCON	14
	REFERÊNCIAS	15

1 INTRODUÇÃO

O presente relatório complementa os relatórios anteriores de *Especificação de Procedimentos Sistemáticos para apoiar a Interoperabilidade para SoS de C2 (níveis conceitual e lógico)*, e o *Relatório de Construção da Prova de Conceito*, e corresponde a um indicador físico de execução da atividade de **Validação da Prova de Conceito em Ambiente Laboratorial** (atividade 6), prevista no Plano de Trabalho do projeto S2C2, para atingir a Meta física 5: *Interop - Conceber linhas de ação para a interoperabilidade de dados com relação à modelagem nos níveis conceitual e lógico*.

Como citado no *Relatório de Construção da Prova de Conceito* foram desenvolvidas ao todo quatro (4) ontologias de referência, a saber CROMO, HINT, MISCON e ATOP. As ontologias HINT e a MISCON foram operacionalizadas em linguagem OWL, enquanto a CROMO e a ATOP foram operacionalizadas e aplicadas diretamente em estudos de caso. Como a ontologia MISCON reusa elementos da CROMO e da HINT, a sua operacionalização e demonstração foram escolhidas como foco deste relatório, cujo conteúdo foi extraído de (1). As demais ontologias e seus estudos de caso estão descritos em (2)(3)(4).

Além da introdução, o presente relatório é estruturado como se segue. No Capítulo 2 são apresentados os estudos de caso que demonstram a aplicação das ontologias operacionais desenvolvidas. O Capítulo 3 apresenta uma conclusão que aponta para trabalhos futuros e extensões, evidenciando a aplicabilidade da ontologia MiScOn em outros sistemas como os da família de aplicativos de C2 da força terrestre (FAC2FTer).

2 DEMONSTRAÇÃO

A demonstração dos experimentos usando o software MiScManager e a Ontologia MiScOn está organizada da seguinte forma. Na seção 2.1 é apresentado o cenário em que se baseou a implementação da ontologia operacional. Na seção 2.2 são descritos os experimentos executados e suas análises. Por fim, a seção 2.3 descreve alguns pontos da utilização da ontologia operacional em outro software que foi desenvolvido no contexto do projeto S2C2, pela equipe da UFRGS.

2.1 Descrição do Cenário

Como forma de validação da proposta de se utilizar uma ontologia operacional para formalizar um cenário próximo da realidade, foi desenvolvido o cenário representado na Figura 1 na ontologia operacional MiScOn em OWL¹

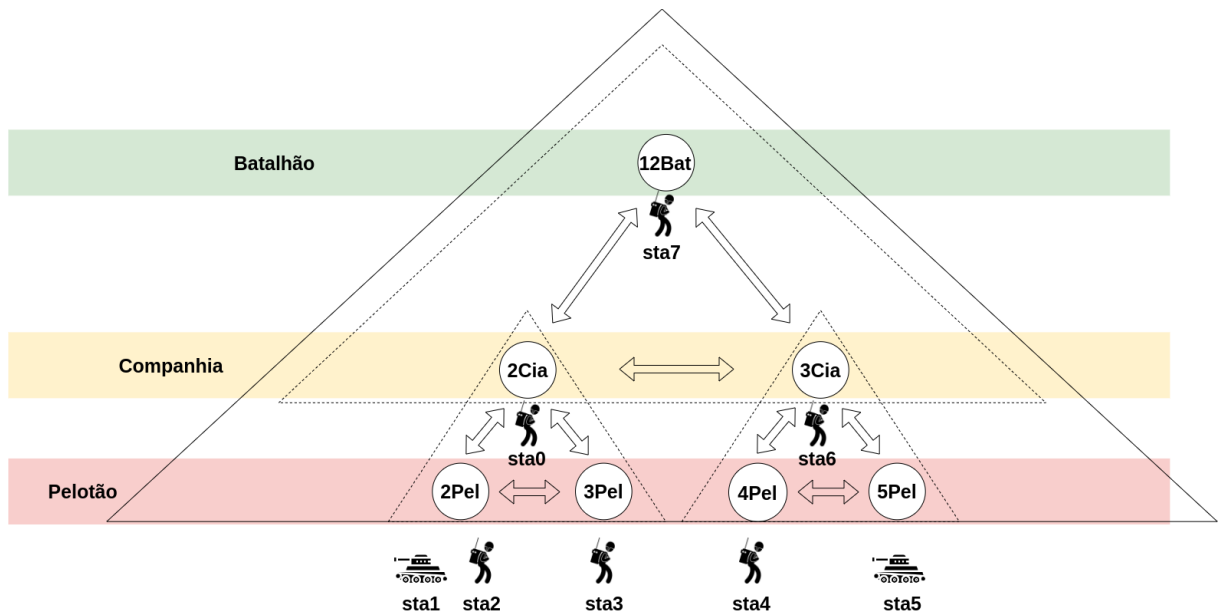


Figura 1 – Representação do cenário do experimento representando a hierarquia entre OMs, os militares, os veículos e as restrições na comunicação.

No cenário, estão representados os principais elementos para a realização de inferências com a ontologia MiScOn. A comunicação entre os nós da rede está sendo executada no emulador Mininet-WiFi através do MiScManager, onde a ontologia foi lida e reutilizada para criar os elementos no banco de dados. É possível observar que está sendo representada a hierarquia entre as Organizações Militares (OM's), os nós que estão com os militares à pé ou em veículos, e as restrições de comunicação através das setas. Nesse cenário, através

¹ <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/onto/miscon.ttl> em: 25 nov. 2025.

das regras SWRL e posteriormente através do resultado do raciocínio semântico, está sendo permitida a comunicação entre comandantes, subordinados e pares. Esse último se refere a quando os dois nós que estão se comunicando estão dentro de uma mesma OM. Ainda, está sendo permitida a comunicação entre equivalentes, isso é, quando as organizações militares possuem o mesmo comandante, estando no mesmo nível hierárquico. Assim, como é possível criar relações de comunicação entre comandantes subordinados, pares e equivalentes, os cenários possíveis podem ter uma OM ou várias OM's, tornando o planejamento flexível.

O cenário apresentado na Figura 1 é instanciado na ontologia operacional, como mostra a Figura 2, é composto de uma rede única com 3 níveis de hierarquia, sendo Batalhão, Companhia e Pelotão. Cada organização militar representada também possui um dispositivo, por exemplo sta4 que pertence ao 4Pel. Ainda, cada dispositivo desse possui duas interfaces, por exemplo, no caso de sta4, as interfaces são sta4-wlan0 e sta4-wlan1.

O **12Bat** possui um militar a pé responsável pela comunicação com os níveis inferiores. No nível de Companhia existem duas OMs: **2Cia** e **3Cia**, cada uma com um militar à pé. Por sua vez, no nível de Pelotão existem quatro OMs: **2Pel** com dois militares responsáveis pela comunicação, sendo um à pé e outro como passageiro em um veículo do tipo Guarani; **3Pel** com um militar à pé responsável pela comunicação; **4Pel** com um militar à pé responsável pela comunicação, e **5Pel** com um militar responsável pela comunicação como passageiro em outro veículo do tipo Guarani. O fato de um nó na rede estar com um militar à pé ou em um veículo implica na velocidade com que esse nó se move com o modelo de mobilidade no Mininet-WiFi. Cada nó na rede possui duas interfaces, uma para se comunicar com superiores (**UeUp**) e outra para se comunicar com subordinados (**UeDown**), simulando o que ocorre em ambiente real, onde existem diferentes rádios para se comunicar com diferentes níveis na hierarquia. Ainda, foi adotado como padrão a interface **UeUp** para se comunicar entre pares e equivalentes. Por fim, cada nó na rede possui um quadrante de movimento estabelecido entre (x_min, x_max) e (y_min, y_max) . A Figura 2 mostra o resultado da inferência realizada pelo raciocinador para a interface sta2-wlan1.

Apesar de a ontologia MiScOn ser reusada na criação de cenários de emulações de redes, o fragmento da ontologia que trata da parte de comunicação não representa as configurações do emulador, mas sim de sistemas reais de comunicação, e por isso alguns atributos configuráveis no Mininet-WiFi como modelo de mobilidade, modelo de propagação, ruídos e atenuação ficaram de fora da modelagem, recebendo valores padrão no momento da configuração dos cenários.

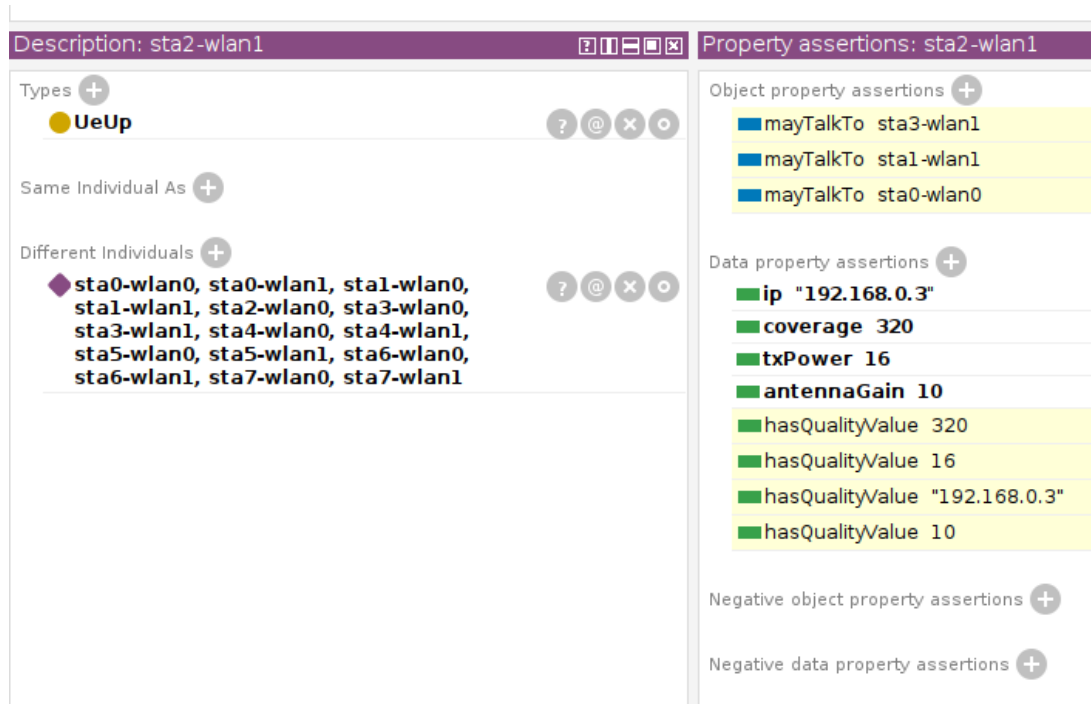


Figura 2 – *Printscreen* da tela do Protegé mostrando o resultado da inferência para o relacionamento *mayTalkTo* na interface *sta2-wlan1*.

2.2 Experimentos e Análises

Uma das primeiras vantagens demonstradas ao se utilizar o MiScManager para criação do cenário foi a automação na configuração, trazendo mais facilidade e ganho de tempo. Realizando a leitura da ontologia operacional ou criando pela interface do MiScManager, a configuração fica mais intuitiva em comparação com a forma como é feita no Mininet-WiFi, onde a topologia de rede deve ser desenvolvida através de linhas de código em linguagem Python. Adicionalmente, com o MiScManager foi possível criar *scripts* que fazem a leitura da ontologia operacional e transcrevem instantaneamente a estrutura conceitual em uma topologia de rede operacional, evitando uma configuração manual mais propensa a erros.

Adicionalmente, possíveis alterações nas regras, nos relacionamentos ou nas restrições de cardinalidade também tornam a manutenção mais flexível, fácil, ágil e dinâmica do que se fosse realizado tudo diretamente no código da aplicação (*hard coded*). Além desse ganho em relação à configuração, o fato de o experimento ser baseado em uma ontologia que fornece uma descrição bem fundamentada dos elementos do cenário militar e da rede, faz com que os experimentos sejam reproduzíveis, podendo replicar os resultados em outros ambientes mais facilmente. Por fim, ressalte-se que o uso da ontologia operacional ajuda na consistência da configuração, pois a detecção de erros não se limita a apenas erros lógicos no código, mas também às inconsistências ontológicas como definição de tipos incompatíveis, definição de hierarquia inadequada, definição de relacionamentos incorretos,

dentre outros.

Como o Mininet-WiFi é um emulador de redes que permite a criação de diversos tipos de topologias, seja em redes infraestruturadas ou *ad hoc*, e possibilita configurar os dispositivos de rede de diferentes maneiras através da passagem de parâmetros, o MiScManager faz com que a configuração de rede criada esteja dentro das restrições impostas pela ontologia. Ainda, ao executar o experimento, faz com que o comportamento na transmissão seja baseado na ontologia, que leva em consideração a segregação dos nós da rede por escalões.

É importante destacar que o propósito dos experimentos foi de avaliar a estratégia de se utilizar a ontologia para construir o cenário a ser reproduzido, e não de realizar experimentos de desempenho de redes. Como já descrito no Relatório de Construção da Prova de Conceito, e também em (1), foi aplicado o reúso de outras ontologias no desenvolvimento da MiScOn. Com isso, foram analisados quais elementos deveriam ser adicionados para se adaptar ao ambiente de comunicação militar e criar o comportamento desejado de forma a ajudar na tomada de decisões em um cenário militar, principalmente no que se refere ao comportamento da comunicação, realizando assim uma análise ontológica sobre o processo de tomada de decisão na comunicação. Por exemplo, a ontologia HINT apresentada por Tesolin (5) não leva em consideração a comunicação entre militares em diferentes níveis hierárquicos. Por isso foi importante especializar Ue em UeUp e UeDown, para imitar o que ocorre em ambiente real, onde são utilizados diferentes rádios para se comunicar, um com o nível de cima e outro com o nível de baixo na hierarquia. No emulador de redes, UeUp e UeDown são duas interfaces na mesma estação.

A Figura 3 mostra as fases para elaboração do experimento mostrando que a ontologia cria uma formalização e restringe as configurações das emulações de rede que acontecem no Mininet-WiFi, criando um cenário militar em um contexto sintético, fazendo com que o comportamento e a configuração da emulação esteja dentro do contexto esperado. Não obstante, apesar de o objetivo nos experimentos ser de fazer uma leitura da ontologia para construção do cenário e da topologia de rede, existem alguns pontos na ontologia HINT que divergem dos conceitos por trás do desenvolvimento do Mininet-WiFi. Essas divergências foram observadas durante o trabalho de desenvolvimento da ontologia que foi realizado sempre observando como os elementos da ontologia seriam passados para o Mininet-WiFi no momento da construção da topologia de rede. Um dos exemplos dessa divergência está na forma como o Mininet-WiFi implementa a criação de interfaces e realiza a atribuição de alguns parâmetros da rede ou do sinal. Mesmo assim foi possível reutilizar a ontologia, realizando sua leitura através do owlready2 e a linguagem Python, e criar a topologia de rede com base nas informações passadas.

Pode-se observar na Figura 4 uma tela mostrando a execução dessa topologia no MiScManager, onde foram coletados os valores referentes à taxa de entrega de pacotes na

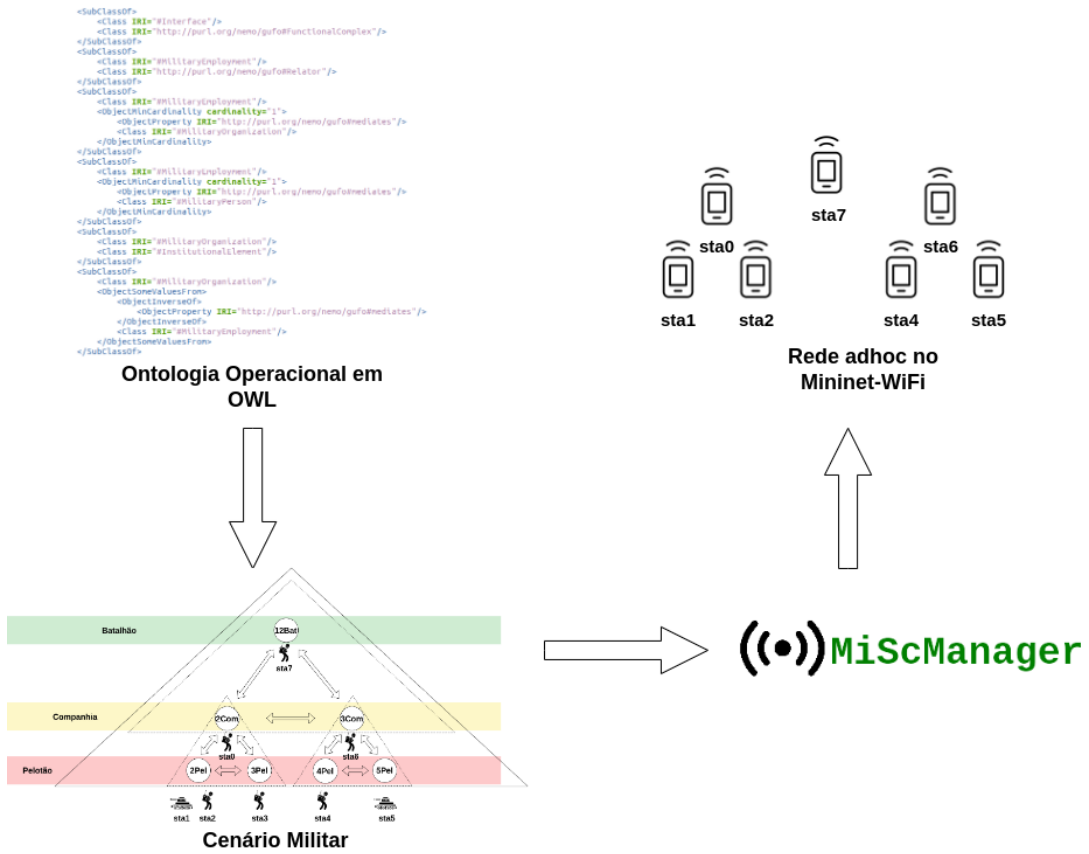


Figura 3 – Ciclo mostrando as fases do experimento desde o desenvolvimento da ontologia de referência até a execução da rede no Mininet-WiFi.

tabela da direita, e alguns atributos de cada nó em cada instante de tempo marcado. Através do raciocínio semântico gerado pela ontologia, o MiScManager testa a comunicação apenas nas interfaces que podem se falar baseado nas instâncias e relacionamentos criados. Ou seja, nesse caso, foi aplicado o resultado do raciocínio semântico que gerou os relacionamentos *mayTalkTo* através de instanciações, relacionamentos e regras SWRL baseadas na doutrina militar para tomar decisões na emulação, e definir o comportamento dos nós na rede. Apesar de estar sendo mostrado um exemplo enxuto, esse tipo de abordagem pode ser aplicada em cenários complexos onde cada nó da rede pode ter requisitos e restrições distintas.

A Figura 5 mostra as conversas capturadas pelo software WireShark durante a execução dos experimentos. Em **a)** o resultado de captura é mais fidedigno ao que seria em um ambiente real, pois foram capturadas as conversas realizadas na rede com a restrição imposta pelo *mayTalkTo* e em **b)** é mostrado um experimento sem nenhuma restrição. Dessa forma, é possível observar que a aplicação de políticas de restrição na comunicação resulta em uma diminuição no fluxo de conversas o que pode ser importante para aplicações que precisem receber essa informação para impor restrições em comunicações, reduzindo a chance de vazamento de informações, melhorando o desempenho da rede, e reduzindo a

Em andamento...

time	name	rss	txpower_if0	txpower_if1	ip_if0	ip_if1	position	ass
35	sta7	adhoc	16	16	192.168.0.16	192.168.0.8	22.53,35.61,0	adr
35	sta4	adhoc	16	16	192.168.0.13	192.168.0.5	42.03,11.86,0	adr
35	sta3	adhoc	16	16	192.168.0.12	192.168.0.4	27.03,10.8,0	adr
35	sta6	adhoc	16	16	192.168.0.15	192.168.0.7	68.24,9.23,0	adr
35	sta0	adhoc	16	16	192.168.0.9	192.168.0.1	8.86,3.36,0	adr
35	sta2	adhoc	16	16	192.168.0.11	192.168.0.3	2.59,2.05,0	adr
35	sta5	adhoc	16	16	192.168.0.14	192.168.0.6	73.17,10.64,0	adr
35	sta1	adhoc	16	16	192.168.0.10	192.168.0.2	18.74,3.03,0	adr
34	sta7	adhoc	16	16	192.168.0.16	192.168.0.8	22.65,35.48,0	adr
34	sta4	adhoc	16	16	192.168.0.13	192.168.0.5	41.89,11.74,0	adr
34	sta3	adhoc	16	16	192.168.0.12	192.168.0.4	27.07,10.91,0	adr
34	sta6	adhoc	16	16	192.168.0.15	192.168.0.7	68.25,9.02,0	adr
34	sta0	adhoc	16	16	192.168.0.9	192.168.0.1	8.89,3.29,0	adr
34	sta2	adhoc	16	16	192.168.0.11	192.168.0.3	2.65,2.15,0	adr
34	sta5	adhoc	16	16	192.168.0.14	192.168.0.6	73.24,10.93,0	adr

time	source	destination	data	value
36	sta0-wlan1-2Company	sta6-wlan1-3Company	ping	1 pacotes transmitidos, 1 recebidos, 0% perda de pacotes tempo 0ms rtt min/méd/máx = 26.166/26.166/26.166/0.000
35	sta6-wlan1-3Company	sta7-wlan0-12Battalion	ping	1 pacotes transmitidos, 1 recebidos, 0% perda de pacotes tempo 0ms rtt min/méd/máx = 5.060/5.060/5.060/0.000
34	sta1-wlan1-2Platoon	sta0-wlan0-2Company	ping	1 pacotes transmitidos, 1 recebidos, 0% perda de pacotes tempo 0ms rtt min/méd/máx = 17.453/17.453/17.453/0.000
33	sta6-wlan0-3Company	sta5-wlan1-5Platoon	ping	1 pacotes transmitidos, 1 recebidos, 0% perda de pacotes tempo 0ms rtt min/méd/máx = 1.268/1.268/1.268/0.000
32	sta1-wlan1-2Platoon	sta3-wlan1-3Platoon	ping	1 pacotes transmitidos, 1 recebidos, 0% perda de pacotes tempo 0ms rtt min/méd/máx = 1.257/1.257/1.257/0.000
31	sta4-wlan1-4Platoon	sta6-wlan0-3Company	ping	1 pacotes transmitidos, 1 recebidos, 0% perda de pacotes tempo 0ms rtt min/méd/máx = 2.271/2.271/2.271/0.000

Figura 4 – *Printscreen* da execução de um experimento mostrando que apenas os nós que possuem o relacionamento `mayTalkTo` se comunicam na emulação da rede.

possibilidade de ataque na rede.

a)							b)						
Ethernet	IPv4 · 12	IPv6 · 48	TCP	UDP			Ethernet	IPv4 · 28	IPv6 · 47	TCP	UDP		
Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B		Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	
192.168.0.1	192.168.0.16	22	3.080	11	1.540		192.168.0.9	192.168.0.11	12	1.680	6	840	
192.168.0.1	192.168.0.7	20	2.800	10	1.400		192.168.0.9	192.168.0.13	4	560	2	280	
192.168.0.2	192.168.0.9	12	1.680	6	840		192.168.0.9	192.168.0.16	4	560	2	280	
192.168.0.2	192.168.0.4	20	2.800	10	1.400		192.168.0.9	192.168.0.10	4	560	2	280	
192.168.0.2	192.168.0.3	24	3.360	12	1.680		192.168.0.9	192.168.0.14	4	560	2	280	
192.168.0.3	192.168.0.9	18	2.520	9	1.260		192.168.0.9	192.168.0.12	4	560	2	280	
192.168.0.3	192.168.0.4	16	2.240	8	1.120		192.168.0.9	192.168.0.15	12	1.680	6	840	
192.168.0.4	192.168.0.9	20	2.800	10	1.400		192.168.0.10	192.168.0.12	6	840	3	420	
192.168.0.5	192.168.0.6	12	1.680	6	840		192.168.0.10	192.168.0.15	10	1.400	5	700	
192.168.0.5	192.168.0.15	10	1.400	5	700		192.168.0.10	192.168.0.11	12	1.680	6	840	
192.168.0.6	192.168.0.15	18	2.520	9	1.260		192.168.0.10	192.168.0.14	6	840	3	420	
192.168.0.7	192.168.0.16	18	2.520	9	1.260		192.168.0.10	192.168.0.13	4	560	2	280	
							192.168.0.10	192.168.0.16	4	560	2	280	
							192.168.0.11	192.168.0.13	12	1.680	6	840	
							192.168.0.11	192.168.0.16	6	840	3	420	
							192.168.0.11	192.168.0.15	12	1.680	6	840	
							192.168.0.11	192.168.0.12	8	1.120	4	560	
							192.168.0.11	192.168.0.14	4	560	2	280	
							192.168.0.12	192.168.0.16	14	1.960	7	980	
							192.168.0.12	192.168.0.14	4	560	2	280	
							192.168.0.12	192.168.0.15	8	1.120	4	560	
							192.168.0.12	192.168.0.13	10	1.400	5	700	
							192.168.0.13	192.168.0.14	6	840	3	420	
							192.168.0.13	192.168.0.15	8	1.120	4	560	
							192.168.0.13	192.168.0.16	4	560	2	280	
							192.168.0.14	192.168.0.15	12	1.680	6	840	
							192.168.0.14	192.168.0.16	4	560	2	280	
							192.168.0.15	192.168.0.16	10	1.400	5	700	

Figura 5 – Registro das conversas capturadas pelo WireShark. Em a) é aplicado o `mayTalkTo`; Em b) todas as interfaces podem se comunicar.

Da mesma forma, outros tipos de configurações são trazidas da ontologia, como por exemplo, a velocidade com que os nós se movem na rede, que está relacionada ao fato de um militar estar à pé ou em um veículo, também levando em consideração qual tipo de

veículo é esse. Assim, são verificadas as relações das instâncias das classes **MilitaryPerson**, que pertencem à classe **MilitaryPersonAsPassenger** ou **MilitaryPersonAsDismounted**. Dessa forma, o sistema identifica as características de cada nó, uma vez que são definidos os atributos de cada *carrier* de acordo com o tipo de veículo, como a velocidade mínima e máxima em metros por segundo, que definirão a forma de locomoção do nó na emulação da rede, da mesma forma que os veículos e suas velocidades já estão definidos na ontologia. Da mesma forma, também são definidas as organizações militares a que cada militar pertence. Dessa forma, uma inconsistência ontológica pode ser detectada caso um militar seja passageiro em um veículo que pertence a uma OM diferente da dele. A Figura 6 mostra um *printscreen* da tela do MiScManager com uma representação gráfica dos nós se movendo dentro dos quadrantes estabelecidos, que é resultado dessa combinação das informações da ontologia com a emulação da rede.

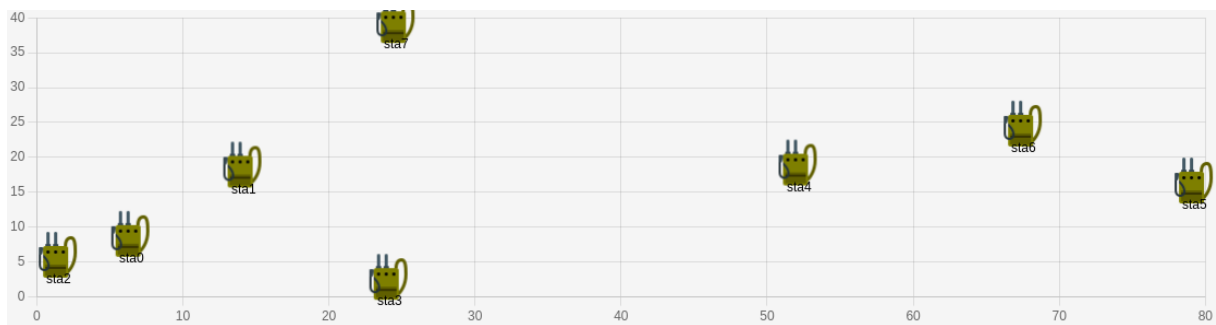


Figura 6 – *Printscreen* da movimentação dos nós dentro dos quadrantes definidos.

Ainda, como citado no Relatório de Construção da Prova de Conceito e também em (1), a ontologia MiScOn realiza algumas extensões da ontologia CROMO, principalmente especializando o *Kind MilitaryOrganization* em vários *Subkinds* e criando uma relação de subordinação e comando entre os tipos de OM e entre as instâncias de OM. Portanto, na Figura 7 é mostrada uma tomada de decisão diferente na emulação da rede, mas que também é baseada nas informações passadas pela ontologia. Nesse caso, é escolhida como comportamento de comunicação, a opção onde todos podem se comunicar, mas com a necessidade de encaminhar os pacotes para os superiores imediatos até chegar ao destino final. Sendo assim, cada escalão se comunica diretamente apenas com seus subordinados e superiores imediatos, utilizando interfaces específicas para cada caso, ou seja, não há possibilidade de comunicação direta com um nível acima ou abaixo na cadeia de comando de um determinado escalão. Em um ambiente real, ao realizar esse encaminhamento por uma cadeia de comando estabelecida, as mensagens têm menos chance de serem mal interpretadas, desviadas ou perdidas. Ainda, poderiam ser adotadas formas de tratamento ou medidas de segurança específicas para cada nível hierárquico ou para cada mensagem enviada. Esse exemplo de comunicação é inspirado nos manuais de doutrina (6, 7, 8). Por conseguinte, essa decisão de encaminhamento é feita verificando as relações de comando e subordinação que foram criadas na ontologia. Assim, um nó sabe para quem encaminhar a

mensagem de modo a fazê-la chegar no destino final. A Figura 7 mostra no quadrante 1, os valores coletados de cada dispositivo e suas interfaces sobre potência, posição, ip, dentre outros atributos no instante de tempo 22. No quadrante 2 é mostrada que no instante de tempo 22 que é quando se iniciou a comunicação, para a estação sta1 se comunicar com a estação sta5, a mensagem teve que ser enviada quatro vezes, pois sta1 do 2º Pelotão envia para sta0 da 22ª Companhia. Esse por sua vez envia para sta7 do 12º Batalhão. Então, sta7 envia para sta6 da 3ª Companhia. Por fim, sta6 envia para o destino final que é sta5 do 4º Pelotão. Dessa forma, esse experimento mostra uma outra forma de tomar decisões com base nas informações acerca da hierarquia militar que vieram da ontologia operacional.

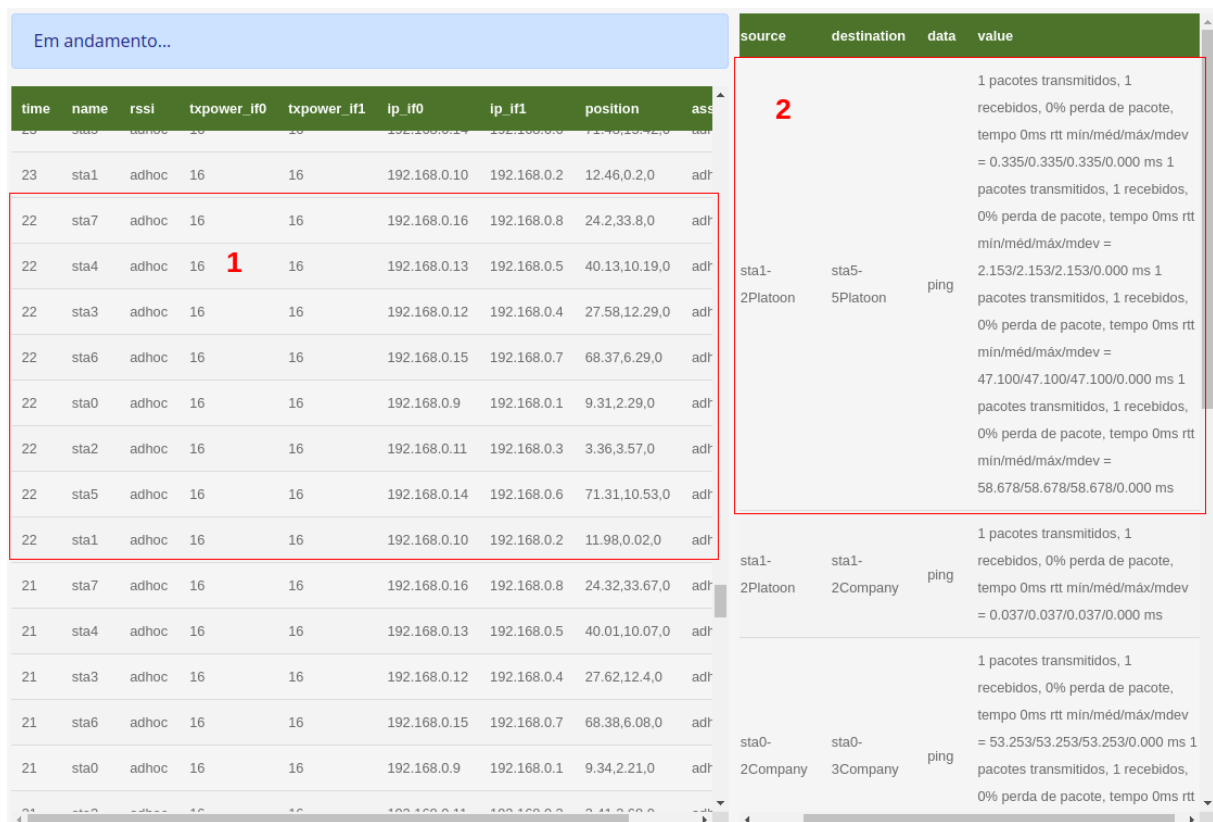


Figura 7 – Printscreen da tela de experimento do MiScManager com envio de pacote para os superiores até chegar ao destino final.

2.3 Aplicação da Ontologia no S2C2 EmuSim - *Command and Control Simulation Configuration and Orchestration*

O artigo intitulado *Integrating a Multi-Agent System Simulator and a Network Emulator to Realistically Exercise Military Network Scenarios*, apresentado pelos autores em (9), também contou com a contribuição da pesquisa apresentada nesta dissertação, onde o foco foi apresentar um simulador, ainda em desenvolvimento, denominado *S2C2 EmuSim*

- *Command and Control Simulation Configuration and Orchestration*. Esse simulador possui uma arquitetura composta por componentes que fazem a leitura de um cenário militar baseado em OWL, permitindo a execução de um sistema de simulação multiagente, que assim como o MiScManager, faz uso do Mininet-WiFi para a emulação de redes. Além disso, o EmuSim também faz uso do NetLogo, um ambiente de modelagem programável multiagente onde foram desenvolvidos os modelos de simulação de operações militares. Essa integração de simulação e emulação envolve desafios de sincronização e compartilhamento de dados entre diferentes softwares em execução e visa cobrir aspectos de tomada de decisão e de redes de comunicações em um cenário de batalha militar.

Para a construção dos cenários, o EmuSim trabalha com uma abordagem baseada na Web Semântica utilizando linguagens como a OWL, que representa os cenários de forma a permitir uma rica representação de entidades, indivíduos, categorias, inferências, atributos e relacionamentos no campo de batalha, igualmente feito no MiScManager.

Este relatório não fará uma descrição detalhada de todos os componentes, focando apenas no **S2C2 OWL Loader** que é usado para converter dados OWL, ou para OWL, e que faz uso da ontologia proposta. Mais detalhes sobre os componentes podem ser encontrados no artigo sobre o Emusim (9).

O Emusim, igualmente ao MiScManager, utiliza a biblioteca owlready2 para realizar a leitura da ontologia. Não obstante, diferentemente do MiScManager que já realiza o *upload* da ontologia operacional com o resultado do raciocínio semântico, o Emusim realiza o raciocínio dentro do próprio simulador utilizando o próprio owlready2 e o raciocinador Pellet (10).

3 CONCLUSÃO E TRABALHOS FUTUROS

Este relatório apresentou um estudo de caso onde o software MiScManager foi utilizado para demonstração do uso da ontologia MiScOn. Tal software é parte dos resultados deste projeto, e está disponível para uso e aberto para possíveis extensões¹.

Além de demonstrar o desenvolvimento de um banco de dados baseado na modelagem conceitual bem fundamentada, o MiScManager faz uso de um emulador de redes para demonstrar como pode ser aplicado o processo de tomada de decisões em redes de comunicações, de forma a fazer com que a rede emulada sofra interferências em seu comportamento causadas pelo cenário operacional militar.

Além disso, a ontologia operacional MiScOn foi utilizada pela equipe do projeto S2C2, mais especificamente, pelo software de simulações de operações militares S2C2 Emusim, desenvolvido pela equipe da UFRGS. A prova de conceito desta utilização foi descrita em (9).

Vale destacar ainda que a ontologia operacional HINT² foi utilizada em outro estudo de caso descrito no Capítulo 6 em (2).

3.1 Possíveis Extensões do MiScManager

Em relação ao MiScManager, versões futuras podem explorar padrões de mobilidade mais realísticos baseados em exercícios militares reais. Parte disso já vem sendo implementado na versão atual³ com padrões de mobilidade que imitam diferentes formações de tropas, como formação em linha, formação circular e formação em cunha.

Uma outra possibilidade de pesquisa está relacionada aos rádios cognitivos. Nesse sentido, poderiam ser exploradas as formas como as tomadas de decisões no rádio cognitivo podem usufruir do resultado do raciocínio semântico advindo da ontologia. Dessa forma, poderiam ser tomadas decisões não apenas relacionadas ao espectro eletromagnético, mas também relacionadas a questões táticas militares (11). Como mostrado na ontologia MiScOn, existe a possibilidade de criar restrições de comunicações ou decisões baseadas na hierarquia baseadas na doutrina. Dessa forma, as restrições impostas na ontologia poderiam ser usadas como um mecanismo de *enforcement* para o rádio cognitivo tomar decisões.

¹ <https://github.com/comp-ime-eb-br/S2C2-IME/tree/main/deliverables/MiScManager> - Acesso em: 25 nov. 2025

² <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/onto/hint.ttl>

³ disponível em <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/onto/miscon.ttl> ou em <https://gitlab.com/andredemori/miscon-repository> - Acesso em: 25 nov. 2025

3.2 Utilização da MiScOn nos Aplicativos de C2

Este trabalho contribui para a solução de problemas de interoperabilidade entre os sistemas de comando e controle da Família de Aplicativos de Comando e Controle da Força Terrestre (FAC2FTer), por criar uma ontologia que representa vários dos elementos presentes nesses sistemas com uma semântica rica.

A FAC2FTer possui sistemas implementados e em uso em diferentes escalões, mas que ainda carecem de um modelo de domínio comum para interoperar. Os sistemas C2 em Combate (C2Cmb6) e o Gerenciador de Campo de Batalha (GCB) são exemplos de sistemas que precisam melhorar sua interoperabilidade. Para isso, atualmente buscam-se algumas soluções para criar modelos de dados únicos que possam ser usados, criar um consenso acerca do cenário operacional, e também questões relacionadas à arquitetura de componentes e protocolos de comunicação usados nesses diferentes sistemas, que por vezes, possuem representações muito genéricas que dificultam a interoperabilidade.

A modelagem conceitual da ontologia MiScOn apresenta alguns dos elementos presentes nas representações desses sistemas, como por exemplo, militar, organização militar e os tipos de materiais do C2Cmb6, como os veículos, porém com uma definição semântica mais rica. Uma modelagem bem fundamentada poderia ajudar a encontrar melhores caminhos para a interoperabilidade entre os sistemas através da análise ontológica e do enriquecimento semântico. Na ontologia MiScOn, são representadas as classes **MilitaryOrganization** que por sua vez possuem seus respectivos tipos (*powertypes*). Ainda, todas as plataformas pertencem a uma organização específica. São também definidas as relações de subordinação e comando dos *powertypes* de OM's e as OM's em si. Toda essa representação na MiScOn possui interseção com os diagramas dos aplicativos de C2 da família FAC2FTer conforme mostram os relatórios do projeto S2C2.

Adicionalmente, pode-se extrair da ontologia MiScOn o que se refere à comunicação entre comandantes de diferentes escalões, que está descrito no manual do GCB como Redes de Comando Segregadas por Escalão, que se assemelha muito com a proposta colocada na ontologia. É possível observar uma semelhança muito grande com o experimento usando o MiScManager descrito na seção 2.2, onde foram configurados diferentes endereços IPs, um para se comunicar para baixo e outro para se comunicar para cima usando as interfaces **UeUp** e **UeDown**. Isso mostra que o uso da ontologia MiScOn possui interseções com esse tipo de configuração de rede, que leva em consideração a hierarquia para definição de redes e o comportamento da comunicação em ambiente real. Assim, a ontologia pode ser usada para ajudar a criar e representar tais redes no GCB.

3.3 Extensões na Ontologia MiScOn

Outra possível extensão em trabalhos futuros seria abordar com mais profundidade o fragmento da modelagem relacionada às organizações militares. No caso representado pela modelagem conceitual, as organizações militares (**MilitaryOrganization**) fazem parte da categoria de elementos institucionais (**InstitutionalElement**) que possuem um relacionamento de agregação com o cenário (**MilitaryScenario**). Sendo assim, por possuir uma relação de agregação, as organizações militares não possuem uma dependência existencial do cenário militar, podendo existir mesmo que o cenário não exista. Porém, as possibilidades de cenários não se limitam a apenas essa forma de representação. Em alguns casos, como em cenários que representam forças-tarefa, algumas organizações militares podem ser formadas especialmente para o cumprimento de uma missão específica, sendo desmanteladas após o seu encerramento. Sendo assim, **MilitaryOrganization** iria ter uma relação de composição e não de agregação com um **MilitaryScenario**, onde o objeto principal **MilitaryScenario** seria responsável pela criação e destruição dos objetos partes (**MilitaryOrganization**).

Ainda, outra possibilidade de extensão seria para representar OM's que são compostas por outras OM's. Por exemplo, em um cenário real, um Batalhão deve ser composto por duas ou mais Companhias. Da mesma forma, uma Companhia deve ser composta de dois ou mais pelotões, e assim por diante entre os diferentes níveis de hierarquia nas organizações militares. Porém na modelagem apresentada neste trabalho não está sendo representada a estrutura das organizações militares nesse nível de representação, importando-se mais com a relação entre comandante e subordinado, abstendo-se de representar a relação de composição.

Outra possibilidade de extensão seria aprofundar a modelagem no que se refere às interfaces e seus tipos, podendo ser utilizada mais uma vez a teoria Multi-Nível. Assim, os atributos **frequencyRange** e **txPowerRange** seriam atributos do tipo da interface e não da interface em si, definindo os limites (*ranges*) para cada interface de determinado tipo. Como a ontologia HINT, que foi reusada pra representar o sistema de comunicação, não representa os *powertypes* de interface, foi optado por não realizar esse tipo de extensão na ontologia MiScOn.

Uma vertente a ser explorada está relacionada ao uso da ontologia em simuladores, como o S2C2 Emusim, mas como uma forma de auxiliar durante a captura de registros, ajudando a capturar os valores de diferentes eventos e situações ao decorrer do tempo. Assim, ao final da execução de um experimento, ou ao final de um evento na simulação, documentos OWL podem ser usados para registrar os valores referentes a uma tomada de decisão ou a um evento que ocorreu. Dessa forma, os eventos e situações que ocorrem na simulação precisam ser modelados para que seja possível realizar esses registros.

REFERÊNCIAS

- 1 DEMORI, A. M. UMA ABORDAGEM BASEADA EM ONTOLOGIA PARA REPRODUÇÃO DE CENÁRIOS DE OPERAÇÕES MILITARES. Dissertação (Mestrado) — Instituto Militar de Engenharia (IME), 12 2023. Disponível em: <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/publi/Disserta%C3%A7%C3%A3oAndreMunizDemori_IME_versao_final.pdf?raw=true>.
- 2 TESOLIN, J. C. C. Towards A Mobile Wireless Network Ontology For Radio Access Points Selection Supported By Semantic Reasoning. Tese (Doutorado) — Instituto Militar de Engenharia, 07 2024. Disponível em: <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/publi/20240704_Tese_JTesolin.pdf?raw=true>.
- 3 SILVA, M. A. A. da. Combinando Técnica e Doutrina por Meio de Conceitos Ontológicos para Representar Cenários Operacionais Militares em Sistemas de Comunicações Cognitivos. Tese (Doutorado) — Instituto Militar de Engenharia, 02 2023. Disponível em: <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/publi/Tese_de_Doutorado_Marcus__IME_Final_Ass.pdf?raw=true>.
- 4 JESUS, V. da Silva de. AUTONOMIA DE MEIOS OPERATIVOS PARA APOIO À DECISÃO EM SISTEMAS DE COMANDO E CONTROLE: UMA ABORDAGEM PREDITIVA E CONCEITUAL UTILIZANDO APRENDIZADO DE MÁQUINA. Dissertação (Mestrado) — UFRJ, 05 2025. Disponível em: <https://github.com/comp-ime-eb-br/S2C2-IME/blob/main/publi/Dissertacao_Mestrado_PPGI_UFRJ_Valquire_Final_comAssinaturas.pdf?raw=true>.
- 5 TESOLIN, J. C. C.; DEMORI, A. M.; MOURA, D. F. C.; CAVALCANTI, M. C. Enhancing heterogeneous mobile network management based on a well-founded reference ontology. Future Generation Computer Systems, 2023. ISSN 0167-739X.
- 6 Brasil. Exército. Comando de Operações Terrestres. Manual de Campanha - Comando e Controle. Brasil, 2023.
- 7 TERRESTRES, B. E. C. de O. Manual de Campanha - As comunicações na Força Terrestre. Brasil, 2018.
- 8 TERRESTRES, B. E. C. de O. Manual de Campanha - As comunicações nas operações. Brasil, 2020.
- 9 BARONE., D.; WICKBOLDT., J.; C. Cavalcanti., M.; MOURA., D.; TESOLIN., J.; DEMORI., A.; ANJOS., J.; Silva de Carvalho., L.; GOMES., J.; Pignaton de Freitas., E. Integrating a multi-agent system simulator and a network emulator to realistically exercise military network scenarios. In: INSTICC. Proceedings of the 13th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - SIMULTECH. Roma, Itália: SciTePress, 2023. p. 194–201. ISBN 978-989-758-668-2. ISSN 2184-2841.
- 10 SIRIN, E.; PARSIA, B.; GRAU, B. C.; KALYANPUR, A.; KATZ, Y. Pellet: A practical owl-dl reasoner. Journal of Web Semantics, Elsevier, v. 5, n. 2, p. 51–53, 2007.

- 11 DEMORI, A. M.; CAVALCANTI, M. C.; TESOLIN, J. C. C.; SILVA, M. A. A. da. Implementing military hierarchical restrictions in communication applications. In: Proceedings of The Latin American Workshop on Information Fusion. Rio de Janeiro, RJ, Brasil: No prelo, 2023.