Action-Free Inverse Reinforcement Learning for Evaluating Satellite Similarity and Anomaly Detection

D. Witman, T. Olson, B. Williams, D. Kesler, B. Marchand
Slingshot Aerospace, 841 Apollo Street, Suite 350, El Segundo, CA 90245

ABSTRACT

Human operators and analysts will be increasingly challenged to detect anomalous satellites (or actors) within mega-constellations as more are deployed. Automated identification and flagging of events and anomalous actions are critical to ensure safe operations in congested and contested space environment. Given observed actor information (states and actions), inverse reinforcement learning (IRL) provides a framework for quantifying the reward function (how desirable a state/action pair is) of an actor in an environment. In many real-world environments, the actions that an actor takes are unknown. This paper will present a novel action-free IRL approach that is used within a larger suite of scalable machine-learning based capabilities for characterizing inter-satellite similarity amongst a large collection of resident space objects (RSOs). Inter-satellite similarity is then used to identify distinct or anomalous satellites that may be operating outside of typical mission boundaries. We first describe the implementation of action-free IRL and then share its ability to distinguish known anomalies within a large simulated low earth orbit constellation. The results of our method show similar performance to the widely used dynamic time warping method. Notably, we are able to achieve these results at a significant computational speedup (15-20x), which allows our approach to efficiently scale to very large constellations.

1. INTRODUCTION

As the number and size of constellations operating in low Earth orbit expands, there is a need for understanding behavioral characteristics of interacting satellites within larger constellations. Near-real time algorithms that quantify expected behaviors and detect anomalous departures from the norm will be required for owner operators and constellation orbital neighbors to ensure safe operations in a congested and contested environment. Existing and planned space domain awareness data enables new methods to analyze the behaviors of satellites.

In 2023 Slingshot's data science team developed a unique machine learning (ML) based time-series, or sequential, comparison pipeline called Agatha, a high level representation is illustrated in Fig. 1. The Agatha pipeline can incorporate many heterogeneous feature sets across a variety of domain applications and compare individualized entities. These comparisons can be used for downstream insights, including anomaly detection, grouping behavioral neighbors and many other space situational awareness applications. Additionally, Agatha uses a variety of similarity and detection algorithms as well as many feature decomposition and pre-processing techniques in an ensemble framework. This allows the ability to consider many unique cross correlations and methods for answering questions of interest.

One component of the Agatha pipeline is the use of action-free inverse reinforcement learning (AFIRL). AFIRL is a framework for representing agent actions and behaviors given only observed state information. This is different from many common inverse reinforcement learning algorithms for which observed state *and* action information is required. AFIRL can be used in cases where an agent's actions are challenging to determine or difficult to represent. The main goal of AFIRL is to generate a reward function that ideally represents an agent's behavioral intent[1]. This reward function is similar to an optimization objective where larger rewards are generated for behaviors that mimic the observed agent. With a reward function, one can accomplish many tasks such as training forward reinforcement learning agents to serve as surrogates in simulated environments. But for the purposes of this paper, we make use of AFIRL in order to compare satellites behavioral intent via the reward function. We refer to this specific application of the AFIRL technique as action-free inverse reinforcement learning for assessing satellite-similarity (AFIRLS).

This paper first presents a background on the space situational awareness application area for which the Agatha pipeline was designed, along with requisite overviews of reinforcement learning and time series analysis terminology. Then our

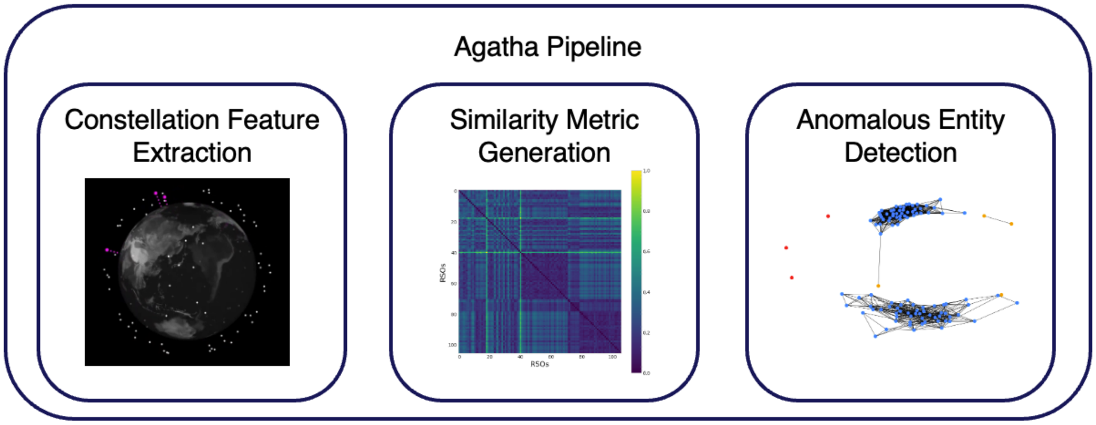

Agatha Pipeline
Constellation Feature Extraction
Similarity Metric Generation
Anomalous Entity Detection
0
20
40
60
80
100
RSOs
RSOs
1.0
0.8
0.6
0.4
0.2
0.0

Fig. 1: The Agatha pipeline was developed in 2023 to detect satellites that exhibited anomalous behaviors within a large constellation

methodology is discussed in the context of the detection of anomalous satellites within a constellation. To illustrate the application of this methodology we share details on simulated constellation benchmark examples. Finally, we demonstrate performance results on detecting anomalous satellites amongst the simulated constellations and compare against a popular existing technique. Conclusions and discussions address where and when this method is applicable.

2. BACKGROUND

The fields of space situational awareness (SSA), inverse reinforcement learning, and time-series analysis have a large existing corpus of material in their respective areas. AFIRLS brings together these three concepts to provide a novel similarity measure that can be used for contrasting satellite behavioral differences. This section will provide a high-level overview of these three fields and share some remarks on why AFIRLS fills a needed gap.

2.1 Space Situational Awareness: Constellation Profiling

Existing and planned satellite mega-constellations have become normal for space operators seeking to accomplish missions that include geo-spatial image collection, weather forecasting, and providing communications and internet. Substantial prior work has focused on how best to construct and manage a constellation, given specific mission parameters. Additionally, there has been some work on detecting deception in space [2] based on publicly available SSA data. Even more recently, there has been related work on detecting potential network intrusions and anomalies within low Earth orbit (LEO) constellations [3]. But detecting differences in satellite operations and characteristics within a constellation based on SSA data is a relatively new field.

2.2 Time Series Similarity

Generating time-series similarity metrics is of great interest to a number of application areas including finance, medicine, and human speech [4] [5]. Successfully relating multiple multi-dimensional time-series signals allows for better correlation of financial stocks, patient drug responses, and distinct bio-metric indicators. For our work, we were interested in comparing astrodynamic signals across time for multiple unique entities (satellites). Given n satellites, we intended to generate an $n \times n$ similarity matrix that could be used for downstream correlation and anomaly detection. The Agatha pipeline supports multiple algorithms that provide similarity matrices in this format.

Dynamic time warping (DTW) [6] [7] is perhaps one of the most popular time series comparison methods. DTW determines the minimum Euclidean distance between shifted signals. Under this construction, similarly scaled sine and cosine functions should be exactly similar according to DTW. But there are a few drawbacks, including the computational complexity which in most implementations is dependent on the length of each time series ($O(n^2)$) being compared. Additionally, DTW can be sensitive to noise in the time-series signal leading to biased similarity metrics.

2.3 Inverse Reinforcement Learning

The field of reinforcement learning (RL) was developed on the concept of Markov decision processes (MDP) wherein one considers an agent that must make sequential decisions in an environment in order to accomplish a defined objective. In any RL formulation, there are four main components (see Fig. 2) that must be considered: states (or sometimes referred to as observations[1]), actions, rewards, and policies. States (s) comprise information (encoded images, kinematic state etc.) about the world or environment that must be used to make decisions. Actions (a) are the numerical representations of the decisions (maneuvering, attitude changes etc.) that are made by the agent. Rewards ($R(s,a)$) provide a functional mapping that generates scalar feedback to the agent based on how well it is performing given its defined objective. And finally the policy ($\pi(s) \to a$) is the model that decides how an agent acts in its environment.

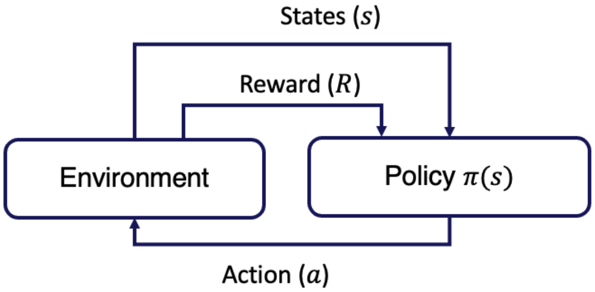


Fig. 2: In reinforcement learning, a policy (or agent) is tasked with generating actions that maximize the long-term expected reward given observed information

In the traditional, and generally more popular, forward RL framework [8], the goal is to determine an optimal π such that the overall long term reward is maximized. For example, deep RL [9] [10] [11] makes use of deep neural networks to form a policy based on a pre-defined reward function. Deep RL has emerged as the predominant method for solving forward RL problems. But there are many real world tasks for which the reward function is unknown or challenging to specify. Conversely, the inverse RL (IRL) framework (Fig 3) has access to perceived state and action data from a teacher or expert agent demonstrations. The goal of which is extracting the policy or the reward function, and in many cases both.

Within the field of IRL there are many related fields that attempt to solve a specific aspect of the overall IRL problem. One related field is behavior cloning, wherein a surrogate policy is created based on collected state and action pairs generated by an expert or teacher policy. More recent behavior cloning methods introduce state based techniques [12] in which policies can be developed to mimic large datasets of states without explicit actions. Behavior cloning can be quite effective when the observed teacher dataset is large and spans the space of states/actions. But behavior cloning can also introduce dangerous pitfalls if the teacher policy injects a sub-optimal bias in addition to over-fitting considerations that can lead to unintentional behaviors [13].

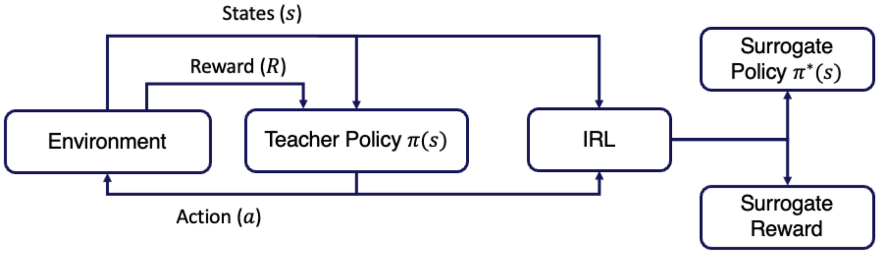


Fig. 3: In inverse reinforcement learning, observed state information and (optionally) action data are used to generate a surrogate reward representation. Some techniques also build a surrogate policy along with the surrogate reward.

[1] Generally, RL literature uses the terminology *states* as opposed to *observations*. In this context a *state* can refer to any data (or feature set) that can be used to define an instance in which an agent can make a decision

In contrast to behavior cloning methods in which the student policy is learned directly from teacher demonstrations, core IRL methods seek to develop a formulation of a surrogate reward function that approximates the preferences demonstrated by the underlying teacher/expert agent.[2] Recall, this reward often is a function based on the unique state and action pair. One technique that has been used to represent the reward function uses a linear combination of weighted feature vectors that map the states to a reward scalar [1] [14]. Maximum entropy IRL provides an alternate approach [15] [16] that builds a discrete representation of the reward based on the expected transition between states, given actions. For the purposes of this work, we are interested in comparing objectives and behaviors of individual satellites via the reward function. The action-free formulation of the IRL reward representation in AFIRLS is distinct from previous IRL approaches and provides a novel reward formulation that is especially applicable when action histories are not directly available in the historical data.

3. METHODOLOGY

There are multiple components of AFIRLS: this section outlines each of the components and how they are used to generate a similarity metric between satellites. We first define the action-free IRL methodology and explain how we make use of it to generate a reward representation. From there, we develop a pairwise similarity metric, given the reward representation, that can be used to generate a similarity matrix over all satellites. Finally, we apply this pre-computed similarity matrix to identify anomalous satellites. This section will discuss each of these three steps in detail.

3.1 Action Free Inverse Reinforcement Learning

Inverse reinforcement learning relies on a set of collected data from a teacher policy referred to as a set of trajectories τ. Generally, these trajectories comprise a sequence of state/action pairs: (s_i, a_i) for $i \in 1...N$ where N is the length of a single trajectory. Given these trajectories, we seek to extract a surrogate reward representation that captures the teacher policy's dynamics. A challenge in many applications is that the actions a_i are not available or overly difficult to obtain. For instance, a non-cooperative satellite operator is unlikely to share information about their constellation's station-keeping maneuvers. Though it may be possible to infer certain details from independently gathered data sources, representing the specific maneuvers that took place and correlating them with the states that led to those actions is itself a non-trivial problem.

3.1.1 Markov Decision Processes

Before introducing the details of our action-free IRL approach, we first define a few useful relations and provide a background to stochastic Markov decision processes. Recall, an MDP defines the sequential decision making process where a policy (π) is able to make actions (a) given some observed state (s) information. At its core, an MDP must obey:

$$\mathcal{P}(s_{i+1}|s_i) = \mathcal{P}(s_{i+1}|s_1, ..., s_i) \tag{1}$$

which is to say that the probability of transitioning to s_{i+1} can be fully represented by s_i. This allows a policy to act independently of all previous states ($s_1 ... s_{i-1}$) and only rely on s_i.

Given a state, s_i, and a subsequent state, s_j, we can write the probability of transitioning directly (in a single step) from state i to state j as:

$$p_{i,j} = p(s_i, s_j) = p(s_{i+1} = s_j | s_i)\,. \tag{2}$$

This relation implies a transition matrix of the form:

$$\mathbf{T} = \begin{bmatrix} p_{1,1} & \cdots & p_{1,d} \\ \vdots & \ddots & \vdots \\ p_{1,d} & \cdots & p_{d,d} \end{bmatrix} \tag{3}$$

[2] The IRL reward surrogate need not bear any formal relationship to the teacher's underlying behavioral motivations. In fact, an explicit reward function may not even exist for the teacher policy.

where the rows of the matrix sum to one and d represents the dimensionality of our state space. It is important to note the spaces in which we are operating are constrained to discrete representations; additional comments on how to operate with continuous spaces will be presented below.

3.1.2 Reward Representation

Given the state transition matrix relation in equation 3, we now seek to build a reward structure that is dependent *only* on states. AFIRLS considers two perspectives on the reward definition: global and local rewards. In this context, we refer to global rewards (R_G) as the most desirable states that a teacher policy is striving for. Local rewards (R_L) on the other hand represent the path that observed trajectories take to reach the global goal. For instance, global rewards would be able to capture the high level station keeping parameters of an earth observing satellite, whereas local rewards would provide feedback on the path a satellite takes to maintain that station. Providing both a global and a local representation of the reward allows us to capture not only distributional differences (global) but also transitional differences (local). This is one potential advantage over other time-series comparison methods where the bias tends to be more on the transitional shape of the time-series.

Global Rewards

Global rewards (R_G) are defined by a mapping from the overall state visitation frequencies to the reward:

$$p(s_i) \to R_G(s_i) \text{ for } s_i \in \mathbf{S}^d\,, \tag{4}$$

where $\mathbf{S}^d$ represents the space of possible states, and $p(s_i)$ is the frequency by which s_i was encountered in the expert trajectories. The translation from frequency to reward depends on the choice of normalization for the reward scale. Generally speaking, in practice most forward RL methods learn best when $R_G(s_i) \in [-1,1]$ [17].

Local Rewards

Local rewards (R_L) capture path-dependent properties of the expert trajectories, so we make use of the transition matrix $\mathbf{T}$ from Eq. 3. Similar to the global rewards obtained from the state visitation frequencies in equation 4, we can define a local reward mapping from the state *transition* frequencies

$$\mathbf{T}_{ij} \to R_L(s_i, s_j) \text{ for } s_i, s_j \in \mathbf{S}^d\,. \tag{5}$$

3.2 Mapping Continuous Spaces to Discrete

From the definition of both the global and local reward structures, it is clear that they are dependent on discrete state spaces. Yet, many practical applications involve continuous state spaces. To bridge this gap, we implemented a bin-based discretization technique that built a compact map between continuous and discrete feature spaces. Fig. 4 shows how we are able to decompose a continuous space into discrete regions of the feature space. It is important to note that we are able to consider a sparse representation of this space. This means that even though our discrete dimensionality grows exponentially as new features are added, it remains relatively compact with respect to the spread of the feature distributions.

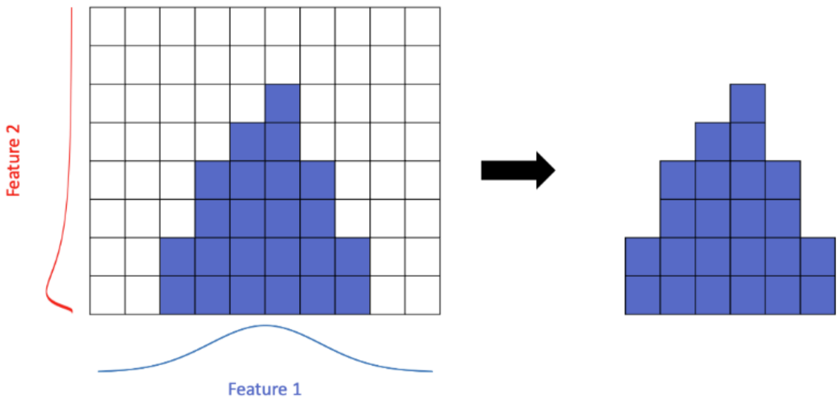


Fig. 4: AFIRLS bin-based discretization scheme allows continuous feature spaces to be represented in a discrete sense. Bins are defined and stored in a sparse format allowing for compression of the overall dimensionality, d.

3.3 Similarity Metrics

Once we have representations for both the global and local rewards, the next step is to define a comparison metric to assess similarity. We make the assumption that we have a set of m trajectories (τ_m) of discrete d dimensional feature data. Additionally, we assume that each of these τ_m trajectories are m unique entities (or in our context satellites) with aligned temporal states $s_i \in \mathbf{S}^d$. In practice, each entity can also be segmented into multiple time-history batches to form τ. Given these assumptions, we can write the similarity metric as the L^2 norm between either the global or the local rewards:

$$\begin{aligned} \mathcal{S}_G^{i,j} &= ||R_G^i - R_G^j|| \\ \mathcal{S}_L^{i,j} &= ||R_L^i - R_L^j|| \text{ for } i,j \in [1,m] \end{aligned} \tag{6}$$

where S represents a matrix of collected similarity metrics and i and j denote the anomalous satellites with their associated global (R_G) and local (R_L) reward representations. We have found in practice that combining reward structures prior to computing similarity is not as effective as combining the regularized similarity matrices. The left side of Fig. 5 shows an example similarity matrix for a set of satellites, using the benchmark problem that will be defined in the next section. It is useful in some contexts to consider this similarity matrix as a network diagram (right side of Fig. 5) where the nodes represent the individual entities and the edges represent strong similarity between entities.

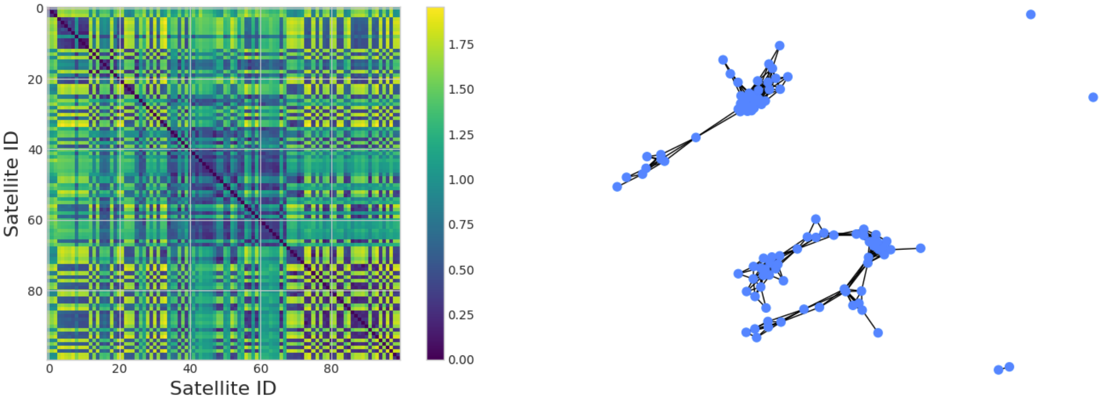


Fig. 5: An example similarity matrix and its graph representation. Edges were defined based on a threshold of the similarity between entities.

3.4 Detecting Anomalous Satellites

The final component required for detecting distinct or anomalous satellites is to develop scoring algorithms that are able to generate outlier scores given similarity matrices. Anomaly detection can be considered an unsupervised learning problem where data is unlabelled. Under this framework, common algorithms like HDBScan [18] and Local Outlier Factor (LOF) [19] will attempt to cluster and partition the feature dataset such that anomalous entities can be identified when they are unable to associate with a cluster. Under the hood, many of these outlier cluster detection techniques build a similarity matrix that is then used to identify which entities are the most distinct. In our situation, we already have the similarity matrix and are able to bypass the initial clustering and similarity matrix generation step.

All outlier scoring algorithms have their own unique capabilities, but for simplicity of presentation in this paper we will make use of the LOF for determining which satellites are the most anomalous. The LOF creates a scoring factor that is dependent on the local neighborhood of similar entities. It is important to note that this method creates a continuous value (referred to here as the interest factor, or I) of how different an outlier is, as opposed to a binary property. This enables entities to be ranked by the size of their interest factors, and the continuous scoring mechanism is also useful for cases where one might want to combine scores from multiple algorithms and/or feature sets into a weighted ensemble.

4. SIMULATION

To validate the effectiveness of AFIRLS for assessing satellite similarity, we created a validation dataset with known anomalous behaviors present in a constellation. Slingshot Aerospace's PHASE (Physically High-accuracy Astrodynamics Simulation Engine)[20] was employed to create said validation dataset. PHASE is a simulation tool that is able to generate large quantities of astrodynamic data at varying levels of fidelity. Additionally, PHASE can simulate sparse and imperfect information using realistic sensor observation processing as well as modelling downstream Orbit Determination (OD) processes. PHASE was critical in creating this benchmark dataset so that we could test known constructed characteristic/behavior differences in a realistic setting.

The benchmark problem chosen to validate this technique involved a constellation of satellites operating in low Earth Orbit, wherein a small subset of satellites (less than 5%) had varying masses that were different from the remainder of the constellation. This scenario was contrived but alludes to a feasible reality in which a satellite with an extra payload is attempting to hide in a larger constellation. To thoroughly test this benchmark, we created an experimental design with 10 different variations that included multiple parameter permutations. Table 1 shows a sample of some of the parameters that were varied to generate the data necessary for this experiment.

Table 1: Design of experiments parameters for scenarios to serve as a benchmark for assessing the performance of AFIRLS at detecting anomalous satellites within a constellation. All resultant data was simulated with underlying control parameters and realism assumptions using the PHASE simulation tool.

Experiment	Orbital Shells	Orbital Planes	Unique Bus types	Scaled Mass Difference
1 (Least Difficult)	1	10	1	5
2	1	10	1	5
3	1	10	2	4.5
4	1	10	2	4
5	2	10	2	3.5
6	2	12	2	3
7	2	12	3	2.5
8	2	10	3	2
9	2	12	3	1.5
10 (Most Difficult)	2	12	3	1.25

To illustrate the different mass difference configurations Fig. 6 shows the differences in mass between the anomalous satellites and the nominal satellites for the most/least difficult scenarios. The most challenging scenario presented numerous challenges that increased the complexity of detecting anomalous satellites.

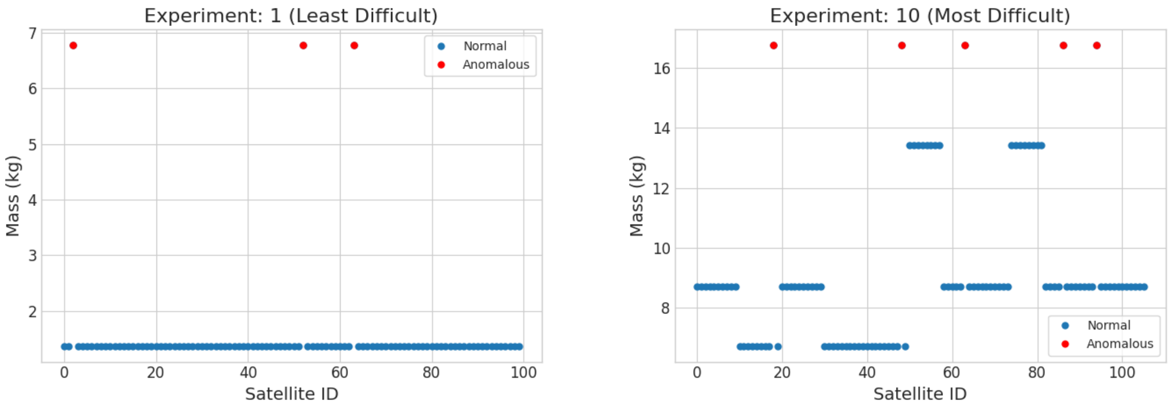


Fig. 6: The mass differences between the least/most difficult scenarios.

In addition to the parameters presented in Table 1, a few underlying characteristics of the simulation were controlled to increase realism, such as the number and geo-location of the ground sensors and the noise levels for associated state vectors. Satellite attitude and online station-keeping maneuver controllers were also included in the simulation to account for realistic maneuvers that would be needed to maintain a mission. More broadly, this simulated constellation was intended to mimic a LEO internet or communications provider over a two year time period.

4.1 ML features

Given the data generated using the PHASE simulation engine, it is worth describing some of the features that will be used for assessing the performance of the AFIRLS algorithm. One straightforward approach to detecting mass differences amongst satellites would be to detect how frequently maneuvers occur based on an external maneuver detection algorithm. As the mass of a satellite increases, the number of maneuvers required to maintain the object's station will decrease. All else being equal, this is due to the inertial force of said object relative to to its less-massive companions. It is important to note that the thrust required to maneuver the larger satellites will be greater but the overall frequency of maneuvers will decrease. Fig. 7 shows the differences in the total number of maneuvers that a satellite in the simulated constellation makes over the two year time period.

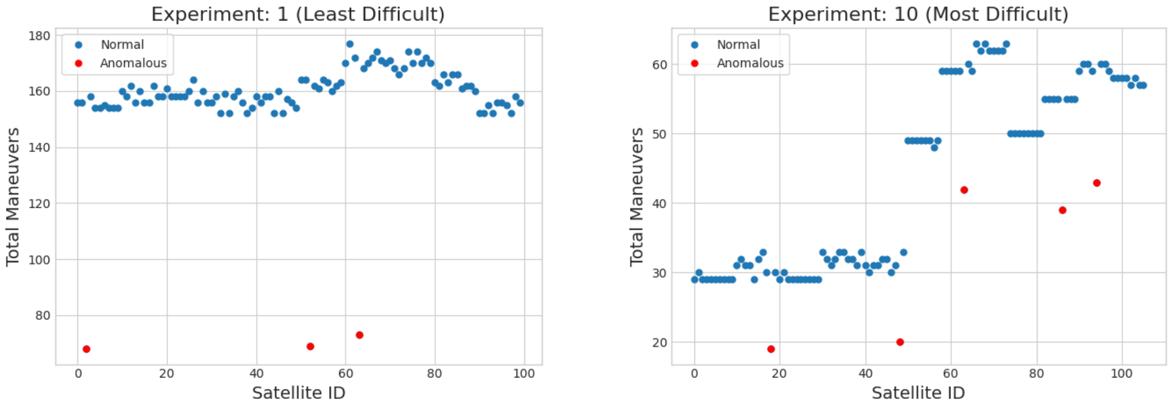


Fig. 7: The total number of maneuvers performed over the entire simulated two year period. The least and most difficult scenarios are presented with the anomalous satellites indicated in red.

From Fig. 7 we can see that the total number of maneuvers (and corresponding maneuver frequency) will not be sufficient in detecting all anomalous satellites. For example, the maneuver frequency of some anomalous satellites in the most difficult scenario fell within the distribution of maneuver frequencies of the nominal satellites. This is due to other underlying simulation parameters including the orbital shell/plane the satellite is operating in. Thus in order to capture all relevant anomalies, we include orbital features (such as semi-major axis, inclination, and eccentricity) into our time dependent feature set, too. The feature processing component of Agatha is able to ingest a large heterogeneous set of features including (but not limited to): astrodynamic, photometric, and contextual data.

4.2 Metrics

To evaluate the performance of AFIRLS to detect anomalous spacecraft we define a set of metrics that will be used to validate the accuracy and precision of our approach. Prior to defining the metrics, it is important to note that the AFIRLS outlier scoring algorithm is not a classification scheme. This means that instead of providing results in the form of classes (anomalous vs nominal), we provide a scored ranking, or *interest factor* (denoted I), based on how anomalous a given entity is. An entity's interest factor provides a numerical ranking for how different that entity is in relation to all other entities considered. The two performance metrics that will be employed are the top-5 and top-10 recall.

The top k recall can be defined as the number of successfully identified anomalies within the top k ranked entities divided by the total number of true anomalous entities. Or in equation form:

$$r_{recall} = \text{top-}k \text{ Recall } = \frac{\text{correct anomalies identified in top-}k\text{ ranked list}}{\text{total anomalous entities}} \tag{7}$$

This metric is useful in that it is operationally relevant. If an SSA operator were to be tasked with investigating potential anomalous satellites, there would be a finite number of satellites that they could conceivably profile in a reasonable time frame. Ensuring that the anomalous satellites are included in the list would be critical. For our purposes, we consider two values for k: 5 and 10. For all experiments, the number of anomalous satellites is constrained to be less than or equal to 5 which is why we limit the lower bound to $k = 5$.

5. RESULTS

Now that we have defined our methodology, described our simulated benchmark data, and proposed some metrics to evaluate the performance, we present some results. Recall, our objective is to show that using the reward function generated via an action-free IRL methodology, produces sufficient information to assess satellite similarity. To that end, we will compare our Action-Free Inverse Reinforcement Learning for Satellite-similarity (AFIRLS) performance results and relate them to a common time-series comparison technique: Dynamic Time Warping (DTW).

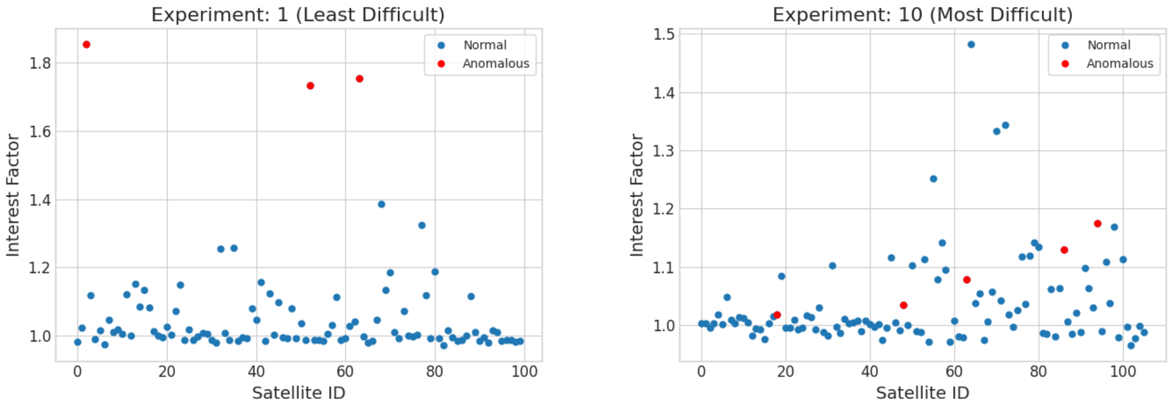


Fig. 8: The AFIRLS interest factor for most/least difficult experiments in the benchmark problem.

Prior to sharing the performance metrics, it is useful to understand how well our technique applies across the space of benchmark experiments. Fig. 8 illustrates the interest factors for each satellite ID, where the red markers indicate the true anomalous satellites. As we can see, the AFIRLS method for comparing satellites is very accurate in identifying the distinct satellites for the least difficult scenario. Performance expectedly degrades as the scenarios under consideration become more difficult. Additional interest factor charts for the remaining set of experiments as well as the interest factors when using dynamic time warping can be found in appendix 8.

Recall, DTW is perhaps one of the most prevalent methods used for time series similarity. For that reason, we compare our AFIRLS approach to an efficient implementation of DTW [7]. For both methods, we parallelize the individual satellite similarity computations in order to reduce the cost of computing the entire similarity matrix. Performance metric comparisons across the space of benchmark experiments are provided in Fig. 9. The top-10 and top-5 recall metrics are provided as a function of the scaled mass difference for each experiment described in table 1. From this chart it is clear that AFIRLS exhibits similar performance in identifying anomalous satellites compared to DTW.

Both AFIRLS and DTW are able to capture the large mass differences between anomalous and standard satellites. In a few cases (scaled mass differences between 2.5 and 3.5) a combination of simulation noise and a smaller number of anomalous satellites, contributed to a small performance dip for both algorithms. Understandably, the performance drops off as the differences become more subtle. This is arguably a good result that indicates that both models are not producing spurious predictions of anomalies.

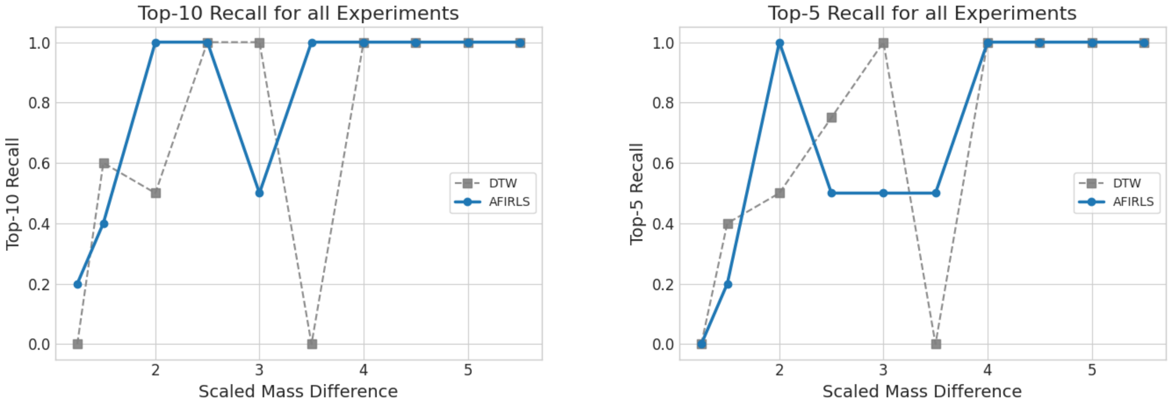


Fig. 9: The top-10 and top-5 recall metrics for AFIRL and DTW as a function of the mass differences between the anomalous satellites and the regular satellites.

It is important to note that although AFIRLS and DTW exhibit similar performance across the space of benchmark scenarios, there is a significant difference in the computational complexity for each method. Since DTW's complexity is quadratically dependent on the sequence length of a time-series, it can be somewhat costly for long sequences and many satellites. Both local and global forms of AFIRLS are dependent linearly on the size of the discrete feature space. This can be costly in cases where the feature space and associated discrete space are highly discretized, but in practice, we have found that coarser discretizations work quite well in capturing the most important components of the underlying feature space. The dimensionality for the AFIRLS results presented here use only 20 total bins allowing it to compare even very large trajectories very quickly.

Fig. 10 shows timing results indicating the speed difference between AFIRLS and DTW. The results are normalized with respect to the mean of the AFIRLS inference time. It is important to highlight this difference if we consider the need to assess many additional feature sets and even larger constellations.

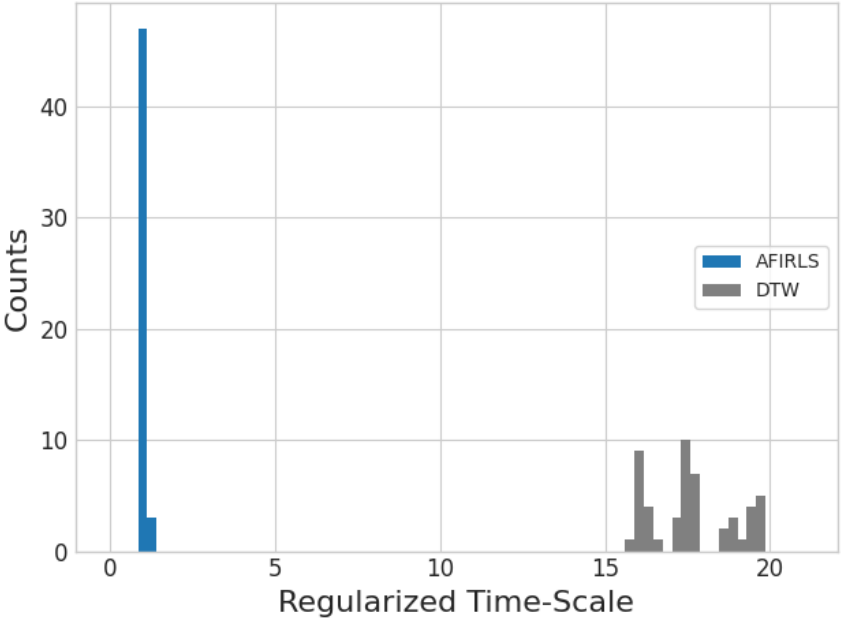


Fig. 10: Comparison of scaled time differences in compute cost between AFIRLS and DTW. Times are regularized with respect to the mean cost of AFIRLS. Importantly, computing a similarity matrix scales quadratically with respect to the total number of satellites. Therefore any minor cost savings in computing the individual similarity metrics can become significant as the size of a constellation grows.

6. CONCLUSION

This work presented Action Free Inverse Reinforcement Learning for Satellite-similarity (AFIRLS) as a new option for better understanding differences between satellite behaviors and characteristics. We also presented a benchmark problem where a small number of satellites had larger masses than the remainder of their constituent constellation. Finally results for both AFIRLS and Dynamic Time Warping (DTW) were presented on how well they are able to detect the anomalous satellites within the constellation. AFIRLS showed similar performance to DTW in detecting the anomalous satellites but exhibited superior computational complexity.

The objective of this paper is not to pose AFIRLS as a replacement of DTW, but instead to show it is worth considering as a complement to other time-series comparison methods. In fact, many different time-series comparison methods should be considered when approaching this problem operationally in a weighted ensemble. Ensemble approaches will be able to capture the entirety of the feature spaces and weigh the strengths/weaknesses of constituent algorithms. The larger Agatha pipeline, of which both AFIRLS and DTW are components, considers many different feature and algorithm representations to identify anomalies.

7. ACKNOWLEDGMENTS

This work, and its supporting research, would not have been possible without the tremendous organizational and technical leadership at Slingshot Aerospace. Additionally, we'd like to thank the entire Slingshot team for their dedication and commitment in preparing datasets, reviewing material, and ensuring accuracy. Finally, we are extremely grateful for our colleagues at DARPA and USSF as well as our project manager, Erin Pennington, for helping to refine and craft an application of this work.

[1] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*, p. 1, 2004.

[2] J. Pavur and I. Martinovic, "On detecting deception in space situational awareness," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, pp. 280–291, 2021.

[3] M. Indaco and D. Guzzetti, "Transformer-based anomaly detection in p-leo constellations: A dynamic graph approach," *Acta Astronautica*, vol. 218, pp. 177–194, 2024.

[4] J. Serra and J. L. Arcos, "An empirical evaluation of similarity measures for time series classification," *Knowledge-Based Systems*, vol. 67, pp. 305–314, 2014.

[5] M. D. Morse and J. M. Patel, "An efficient and accurate method for evaluating time series similarity," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pp. 569–580, 2007.

[6] P. Senin, "Dynamic time warping algorithm review," *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA*, vol. 855, no. 1-23, p. 40, 2008.

[7] T. Giorgino, "Computing and visualizing dynamic time warping alignments in r: the dtw package," *Journal of statistical Software*, vol. 31, pp. 1–24, 2009.

[8] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[9] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, *et al.*, "Mastering the game of go without human knowledge," *nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[10] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[11] D. Miller and R. Linares, "Low-thrust optimal control via reinforcement learning," in *29th AAS/AIAA Space Flight Mechanics Meeting*, vol. 168, pp. 1817–1834, American Astronautical Society Ka'anapali, Hawaii, 2019.

[12] F. Torabi, G. Warnell, and P. Stone, "Behavioral cloning from observation," *arXiv preprint arXiv:1805.01954*, 2018.

[13] F. Codevilla, E. Santana, A. M. López, and A. Gaidon, "Exploring the limitations of behavior cloning for autonomous driving," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9329–9338, 2019.

[14] R. Linares and R. Furfaro, "Space objects maneuvering detection and prediction via inverse reinforcement learning," *Proceedings of the Advanced Maui Optical and Space Surveillance, Maui, HI, USA*, pp. 19–22, 2017.

[15] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, *et al.*, "Maximum entropy inverse reinforcement learning.," in *Aaai*, vol. 8, pp. 1433–1438, Chicago, IL, USA, 2008.

[16] A. J. Snoswell, S. P. Singh, and N. Ye, "Revisiting maximum entropy inverse reinforcement learning: New perspectives and algorithms," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 241–249, IEEE, 2020.

[17] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. d. L. Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq, *et al.*, "Deepmind control suite," *arXiv preprint arXiv:1801.00690*, 2018.

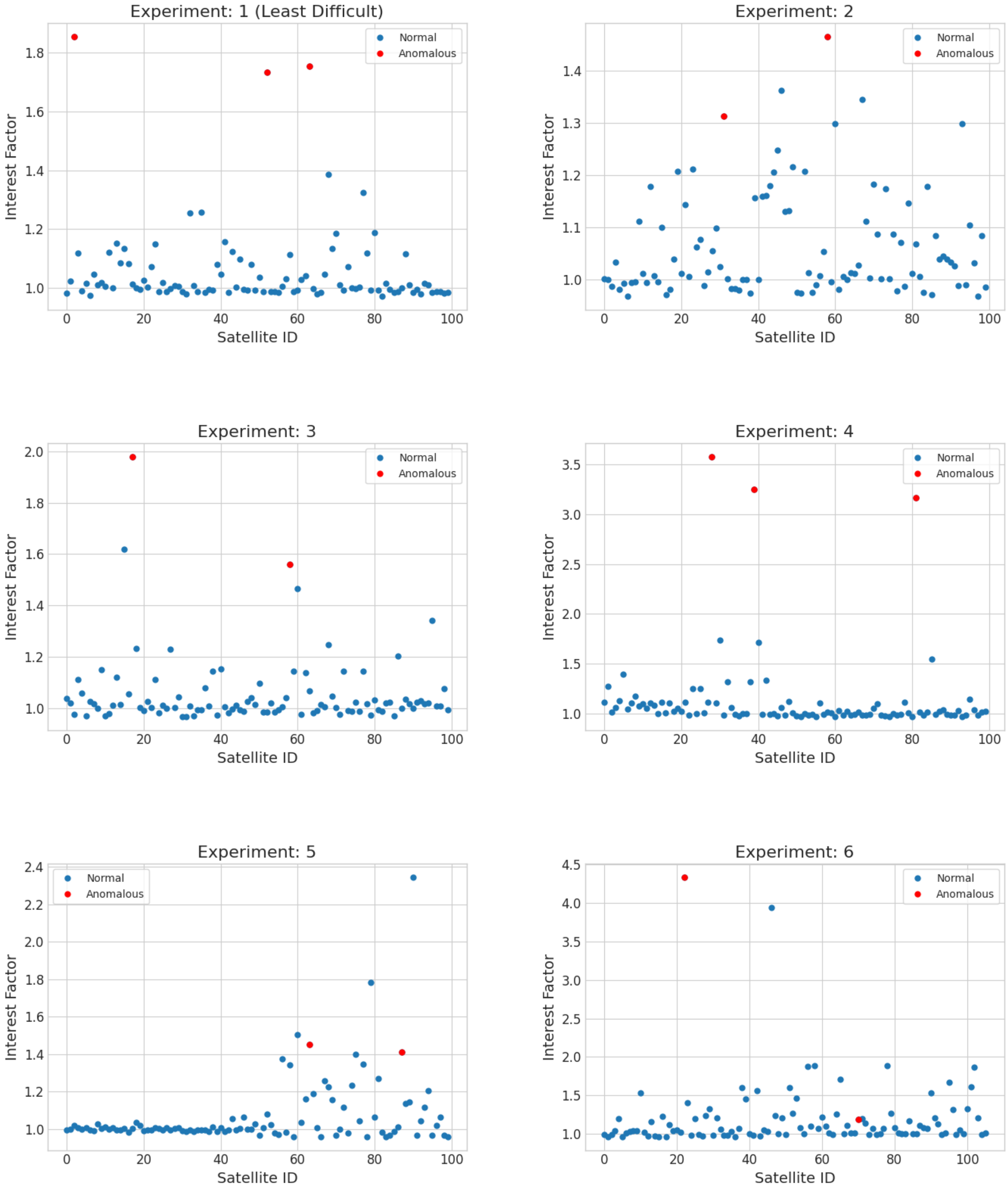
[18] R. J. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in *Pacific-Asia conference on knowledge discovery and data mining*, pp. 160–172, Springer, 2013.

[19] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.

[20] B. Williams, "Slingshot space modeling and simulation." https://www.slingshot.space/solutions/capabilitiesspace-modeling-simulation.

8. EXPERIMENT INTEREST FACTORS

8.1 AFIRL Interest Factors



Experiment: 1 (Least Difficult)

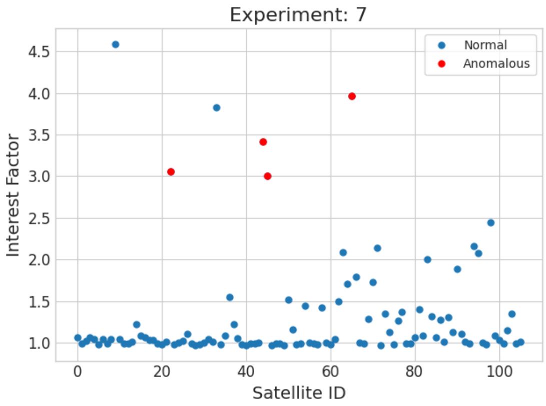
Experiment: 2

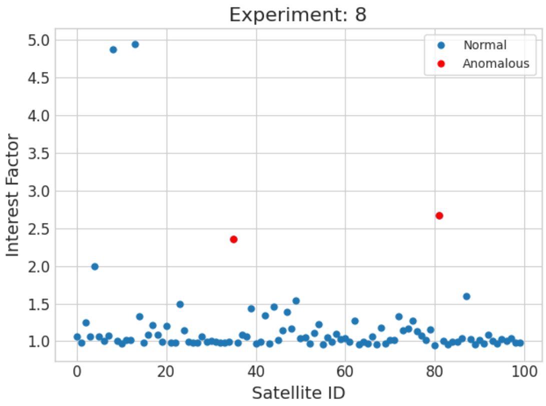
Experiment: 3

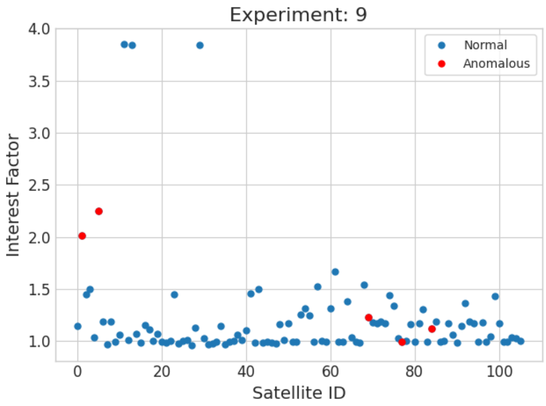
Experiment: 4

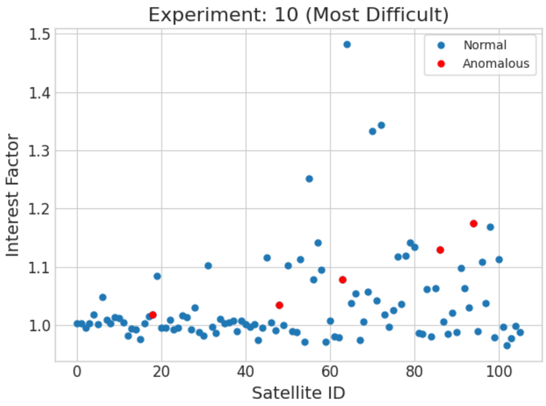
Experiment: 5

Experiment: 6

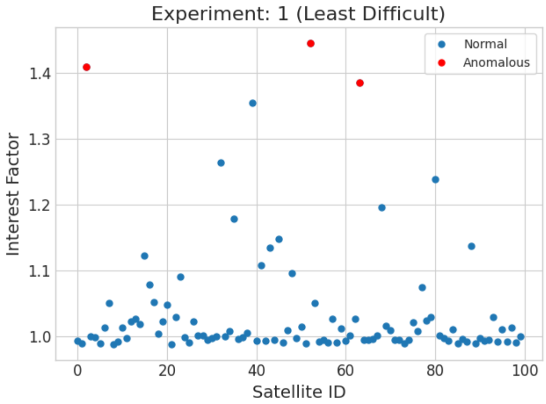


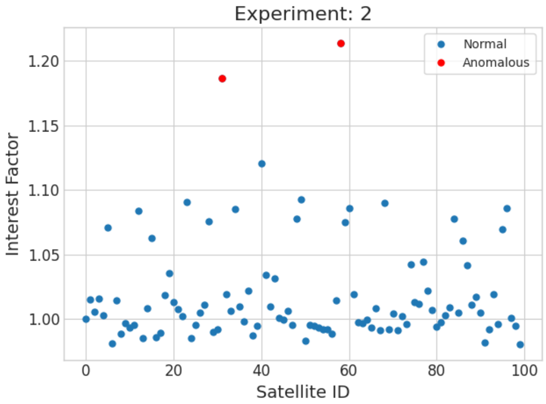


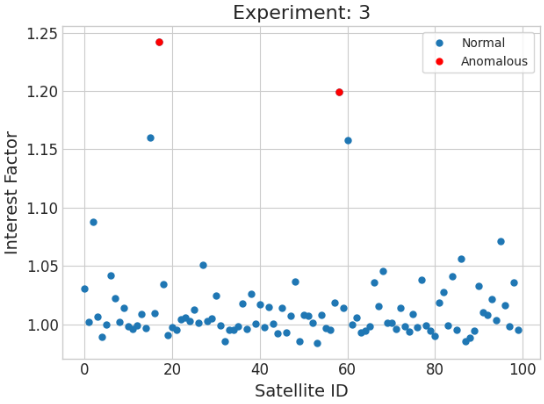


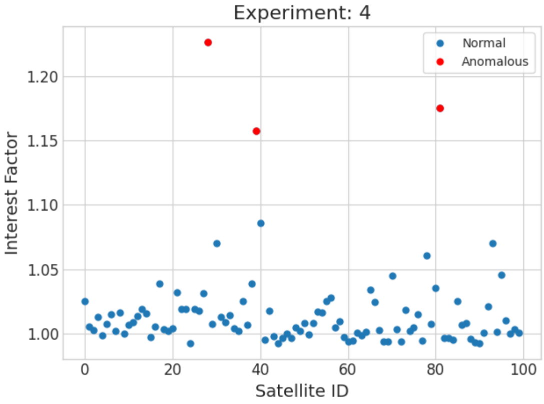


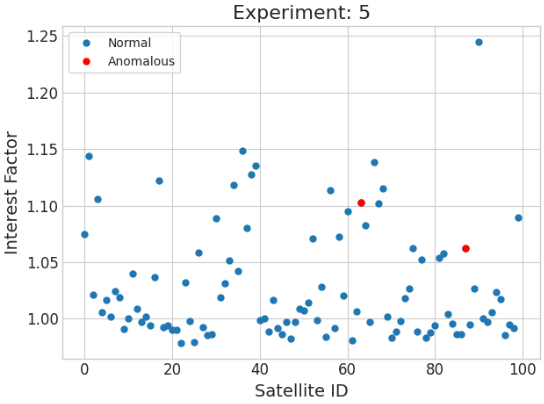
8.2 DTW Interest Factors

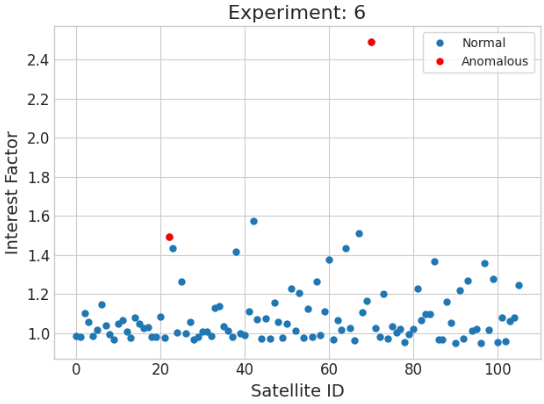


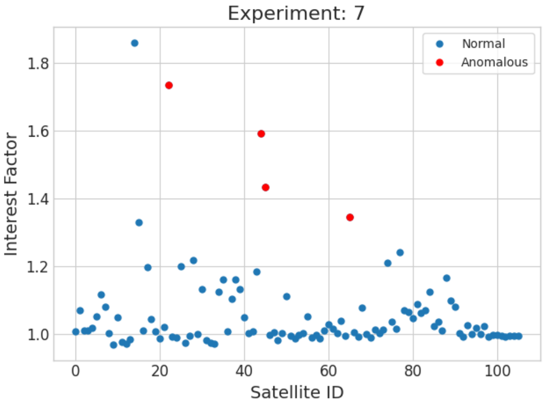


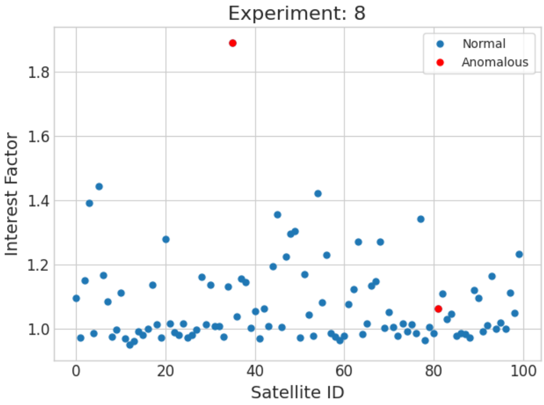

Experiment: 3

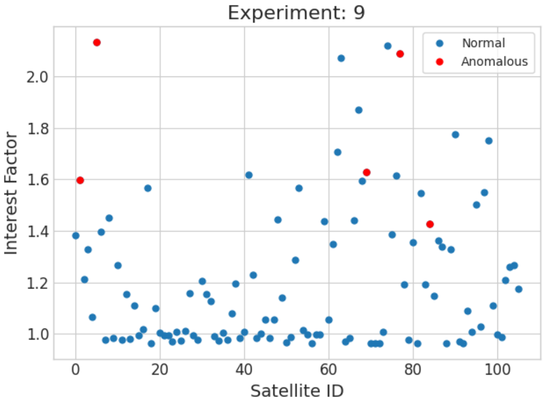

Experiment: 4

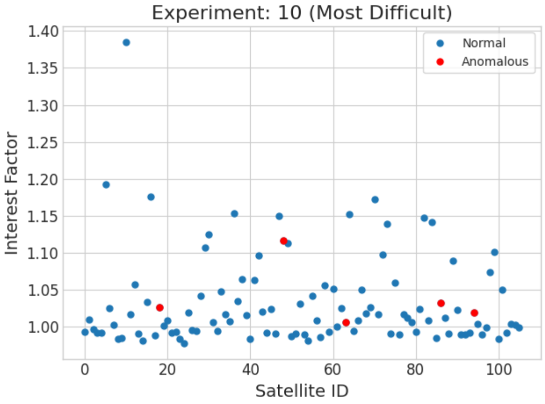

Experiment: 5


Experiment: 6


Experiment: 7


Experiment: 8


Experiment: 9
Interest Factor
Satellite ID
2.0
1.8
1.6
1.4
1.2
1.0
0
20
40
60
80
100
● Normal
● Anomalous


Experiment: 10 (Most Difficult)
Interest Factor
Satellite ID
1.40
1.35
1.30
1.25
1.20
1.15
1.10
1.05
1.00
0
20
40
60
80
100
● Normal
● Anomalous