Automating Collision Avoidance Maneuvers

On the use of available space monitoring data for improving the risk of collision avoidance

Prof. Adam F. Abdin, Assistant Professor

Laboratory of Industrial Engineering, CentraleSupélec, Université Paris Saclay

adam.abdin@centralesupelec.fr

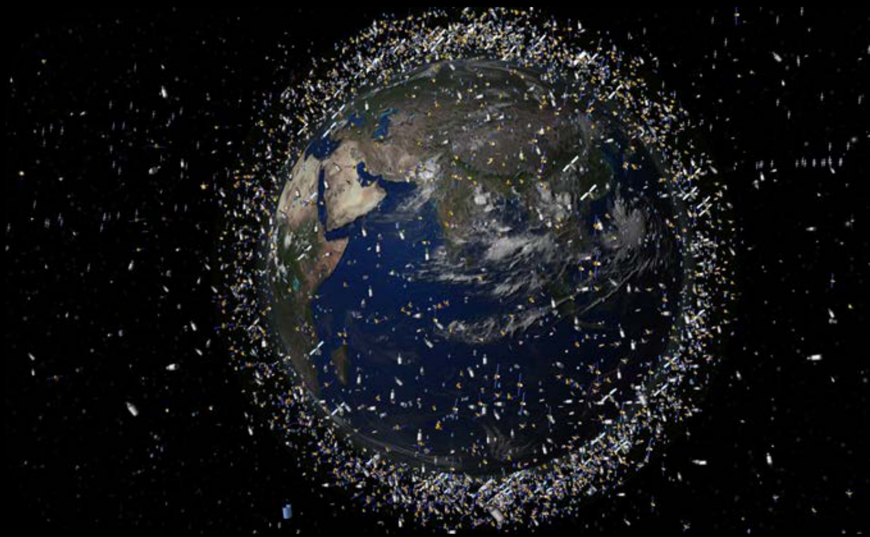
Students : Nicolas BOURRIEZ; Adrien LOIZEAU; Susmitha Patnala

Kessler Syndrome

2009

Crash between the American satellite *Iridium-33* and Russian *Kosmos-2251*

330 pieces of debris from the Iridium satellite were successfully catalogued | **1,400** pieces of debris for the Kosmos-2251 satellite



~5,300
Active satellites

~2,500
Inactive satellites

2019

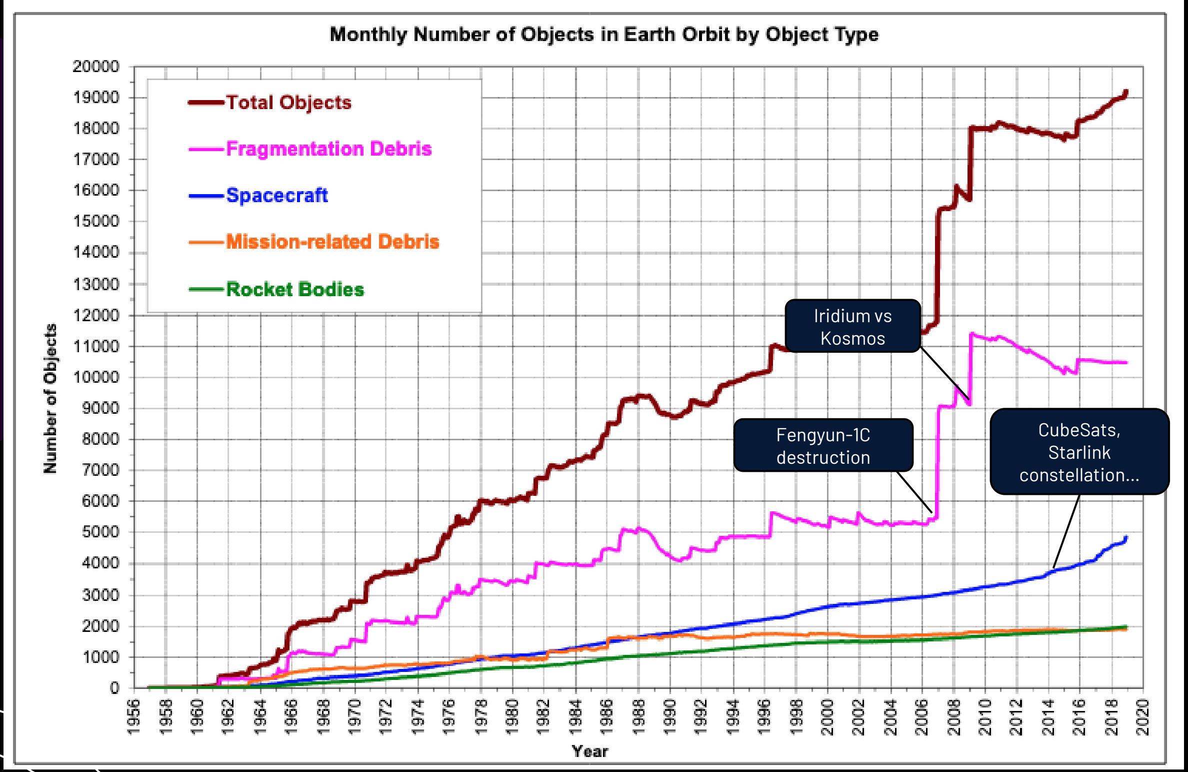
~35,000
Objects (> 10cm) orbiting Earth
>
~22,300 (64%)
Tracked by SSN
>
~13000 (37%)
Debris from collisions

2030

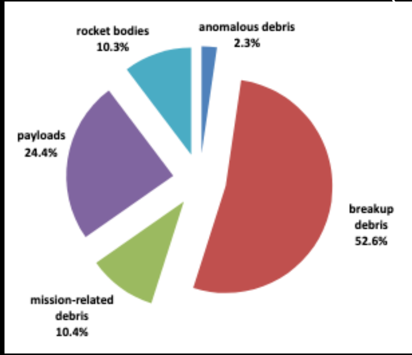
~75,000
Active satellites
(projection)

"Trending analysis of historical conjunction data messages", 2019
"Space Navigator: a tool for the optimization of collision avoidance maneuvers", 2019

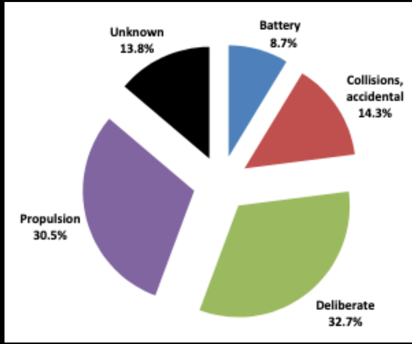
Orbit Debris Analysis



Summary of all objects in Earth orbit officially cataloged by the U.S. Space Surveillance Network since 1956



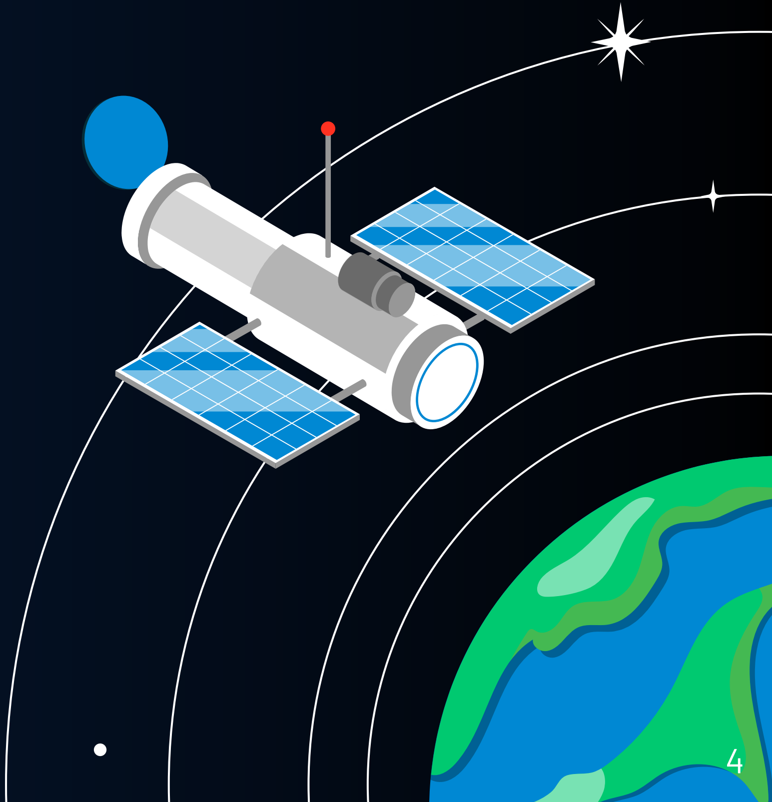
Types of debris



Causes analysed for spatial debris to be created in LEO

NASA Orbital Debris Quarterly News (https://orbitaldebris.jsc.nasa.gov/quarterly-news/pdfs/odqnv23i1.pdf)

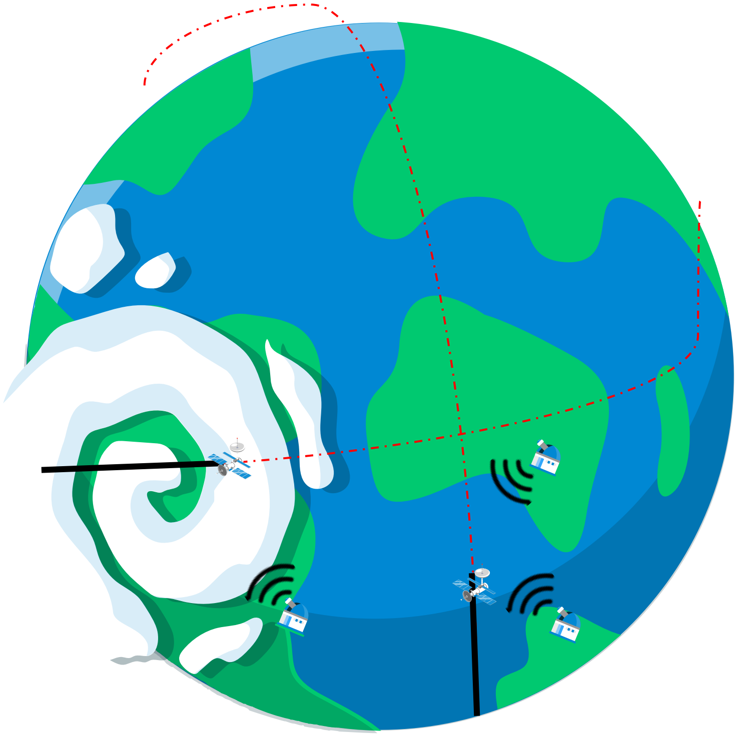
Data usage for Conjunction Assessment



Cunjunction Event Prediction

- 22040 Catelogued items in space and many more uncatalogued
- Collision models –MASTER(ESA), ORDEM(NASA)
- Lethality of debris depends on size, but no consensus reached
  –> Aluminum sphere of 1cm of radius at 14km/s = 1MJ = 1500kg car at 130km/h.
  –> Aluminum sphere of 1mm of radius at 14km/s = 1KJ = bowling pin at 100 Km/h.[1]
- Mean time between two successive collisions is inversely proportional to the size
- How to protect our spacecraft?? CAM –> for debris $> 10cm$ else severe damage(CAM)

Cunjunction Event Prediction



Global Objective: *Determine within 7 days into the future which space objects/**debris** will come within proximity/**cunjunction** of a primary/protected asset (i.e. a satellite)*

Objects type: Near-earth orbiting objects (< 225min revolution)

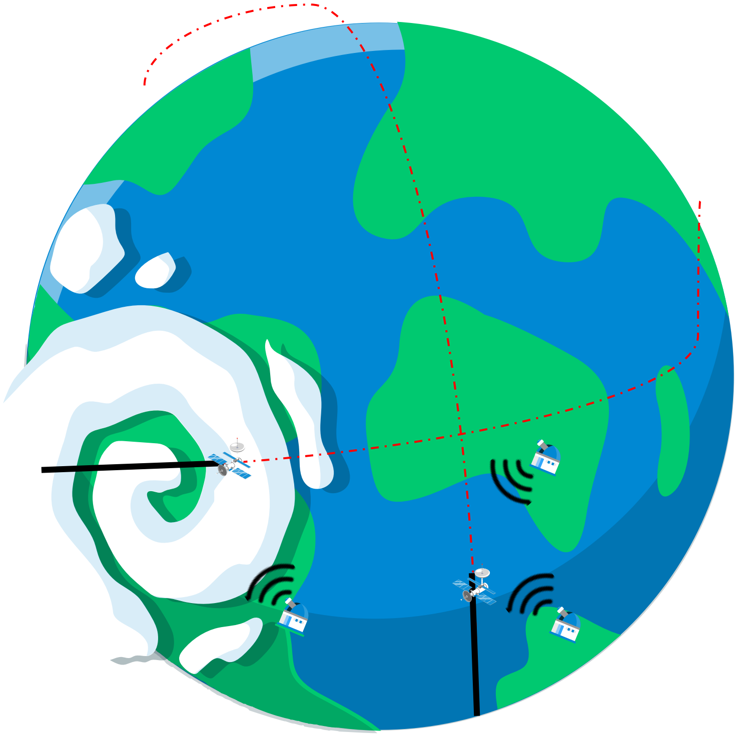
How: Assess the future <u>predicted</u> positions of both objects

Raw TLE (Two-Line Elements) received from Sensors & Radars Tracking (TDRSS, C-band, GPS)

State vector & **Covariance 7-day projection** (/min) (*Simplified General Pertubations-4* propagator)

https://www.nasa.gov/cara/conjunction-event-prediction

Cunjunction Event Prediction



What is the uncertainty in the propagated orbits ?

- *Normally distributed uncertainty [1]*
- *SGP4 model has an error ~1 km at epoch and grows at ~1–3 km per day [2]*

What is behind the SGP4 Propagator algorithm ? [3]

1. **Long-term propagation** from ***Keplerian elements*** *(Line 2 of TLE)*
1. **Short-term perturbations from *external data*** *(Line 1 of TLE)*

[1] https://stacks.stanford.edu/file/druid:dg552pb6632/Foster-estes-parametric_analysis_of_orbital_debris_collision_probability

[2] https://celestrak.org/publications/AIAA/2006-6753/AIAA-2006-6753.pdf

[3] https://celestrak.org/NORAD/documentation/spacetrk.pdf

Cunjunction Event Prediction

- European space surveillance network - EUSST database and catalogue –> 7 EU states collab. Source
- US Space surveillance network - Open source data –> TLE data - Source
- CDM US Joint Space Operations Center JSpOC
- SSN network

Cunjunction Event Prediction

Initial Orbit Determination

- Principle (Kepler's method –> Taylor's expansion and dynamical model)
- Gauss method (3 pairs of angles) –> time-sequential and co-planar
- Gibbs method (3 position vectors) –> orbit from 3 position vectors

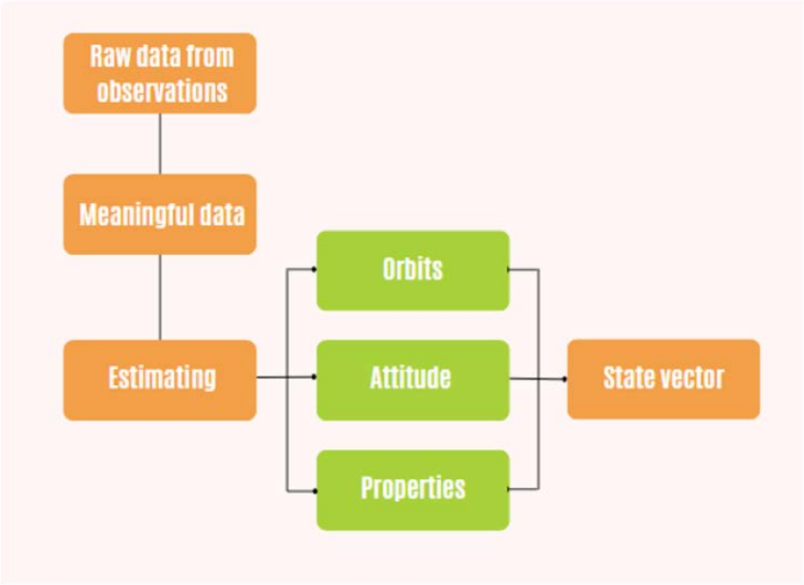
Orbit Determination

- Bayesian filtering –> prediction and correction
- Kalman filter (KF)
- Extended Kalman filter (EKF) –> For non-linear, we linearize using Taylor's expansion

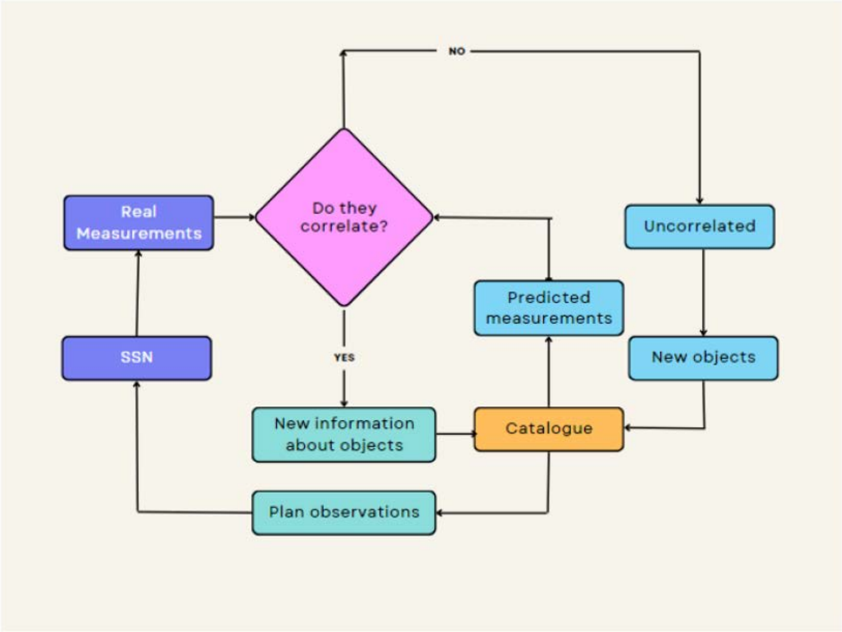
Collision Risk Assessment

- Close approach detection
- Collision Data Message (CDM)
- Probability of collision

Cunjunction Event Prediction - Cataloguing by Space Surveillance Network (SST)

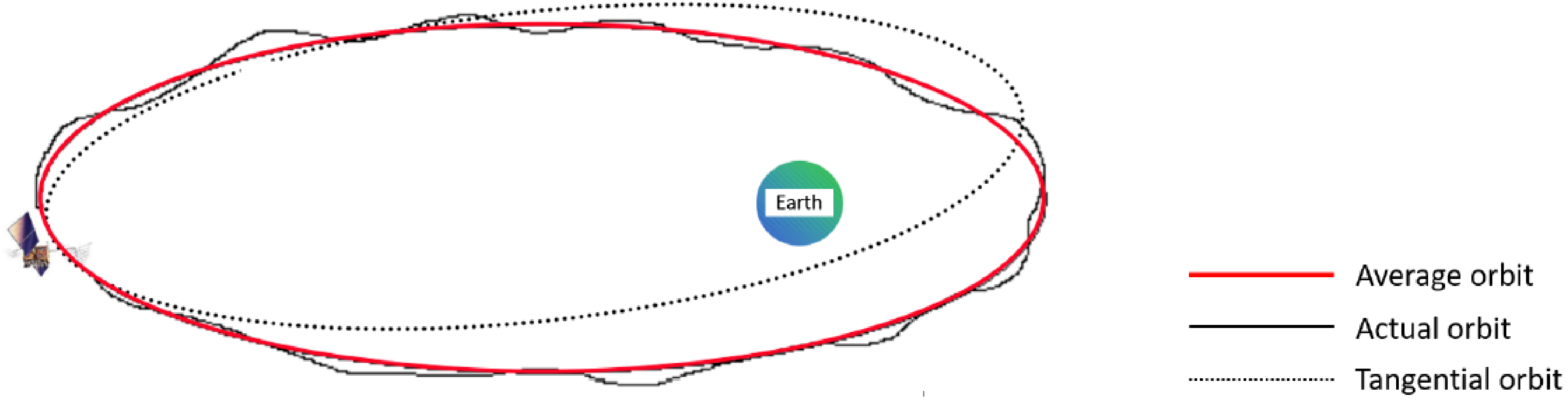


Initial Cataloguing



Catalogue update

Cunjunction Event Prediction - Uncertainties Everywhere



Osculating orbit

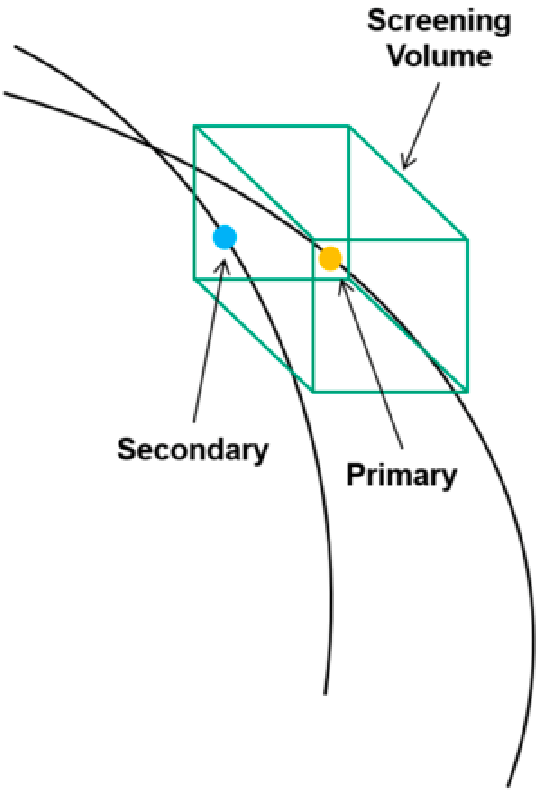
$a,\ e\ ,\ i,\ \Omega,\ \omega,\ M \rightarrow$ Osculating parameters

*Space Mechanics by B. Escudier & J.Y. Puillard

Cunjunction Event Prediction - Uncertainties Everywhere

- Size of orbital population is unknown, and time-varying
- Dynamics of orbital objects
  - Forces: Drag, Solar radiation pressure, Third-body, Earth potential $\cdots$
  - External effects: Solar activity, Atmospheric density $\cdots$
  - Behaviors: Unexpected manoeuvres of unknown nature
- Observation process of sensors
  - Measurement bias/noise, Atmospheric corrections
  - Missed detections, False positives
  - Origin object of measurement usually unknown

Close Approach Risk Assessment



- Using <u>projected</u> trajectories, assess if secondary object enters ***screening volume***
- Screening volume is centered on the primary object (i.e. satellite)
- ***<u>Probability of collision (Pc)</u>***: likelihood that the actual miss distance between the two satellites will be smaller than their combined size

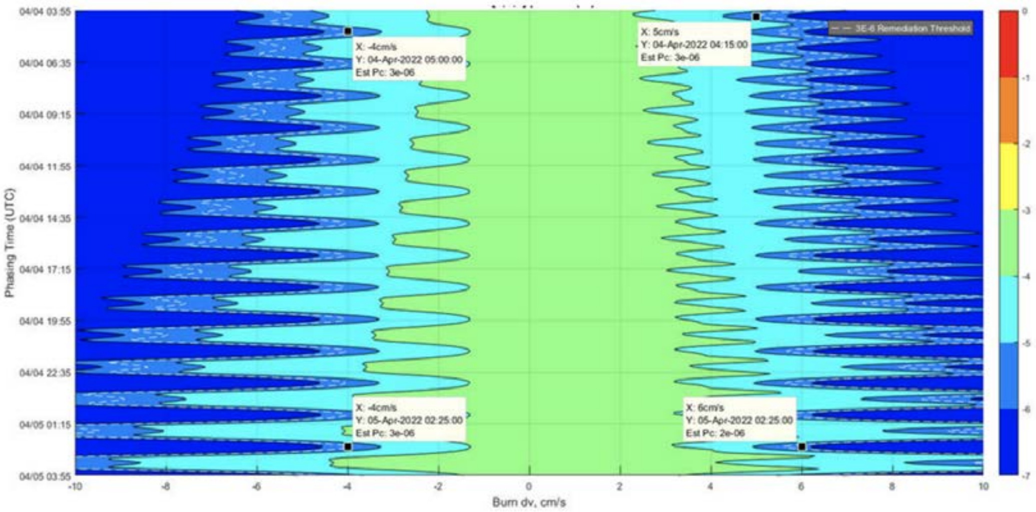
=> this likelihood calculation assumes a probabilistic form because the ephemerides used to identify the conjunction represent predictions with uncertainty

$$PoC = \frac{1}{2\pi\,|Det(C)|^{1/2}} \iint_{x^2+y^2\le d^2} \exp\left(-\frac{1}{2}(\mathbf{r}-\mathbf{r}_{S/P})^T C^{-1}(\mathbf{r}-\mathbf{r}_{S/P})\right)dxdy$$

- π = 3.141592653589793.
- C is the 2X2 projection of the combined 3X3 covariance at TCA onto the collision plane[22].
- $Det(C)$ is the determinant of C.
- d is the sum of the two object sizes[23].
- "exp" is the exponential function, i.e., e to the power in parentheses with e = 2.718281828459045.
- $\mathbf{r} = (x, y)^T$ is any point in the collision plane such that $x^2 + y^2 \le d^2$.
- $\mathbf{r}_{S/P} = (r_{S/P}, 0)^T$ is the position of the secondary relative to the primary along the x-axis in the collision plane.
- C^{-1} is the inverse of C.

https://nodis3.gsfc.nasa.gov/OCE_docs/OCE_51.pdf

Close Approach Risk Mitigation



Maneuver trade-space plot

- CDM sent 3 days prior
- Either *propulsion system* or *differential drag*

- Manoeuver required when PoC > *1E-04 [4]*

[4] A Parametric Analysis of Orbital Debris Collision Probability and Maneuver Rate for Space Vehicles

Existing data types

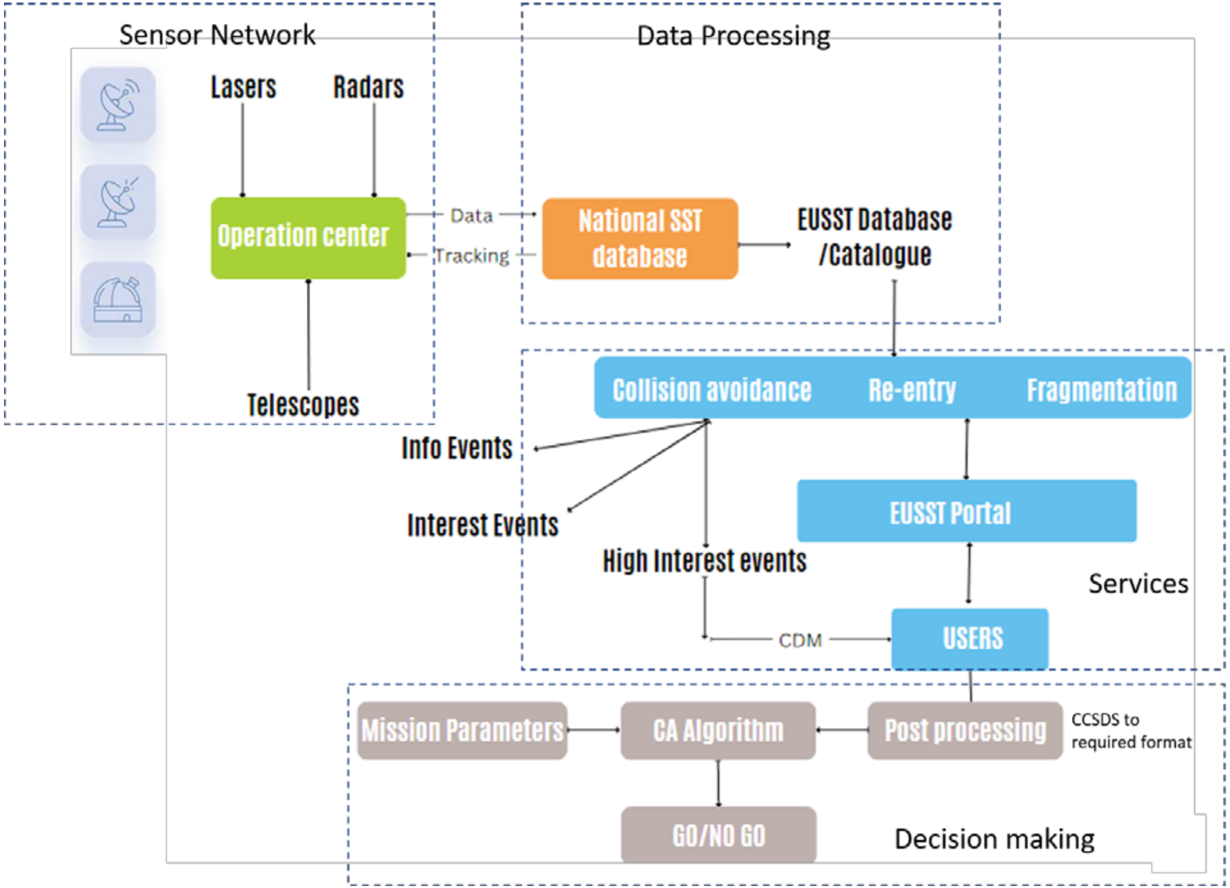
Data usage

TLE	CDM
Limited accuracy and precision	Provide precise and accurate information
Limited information	Detailed analysis of potential collision
Lack of real-time updates	Timely updates
Assumes perfectly spherical earth	Perturbations are included

Table: Comparison between TLE and CDM

[2] https://www.space-track.org/documentation#api-Overview

From sensory data to decision making



Data handling framework

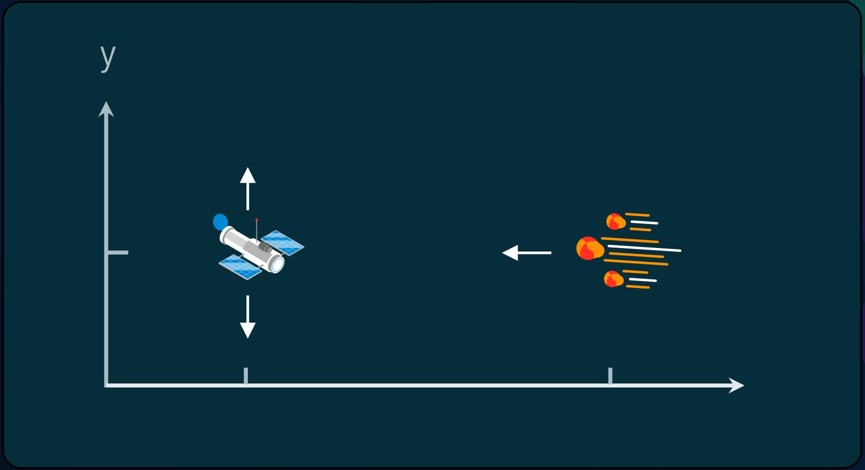
Automating Spacecraft Collision Avoidance

General Framework

Project Framework

Literature Review & Knowledge Gathering	Modeling
Gravitational Physics & Orbital Basics	POMDP CAM Model
Collision Avoidance Systems	Data generation using a simulator
Applications of POMDPs & Machine Learning for Collision Avoidance	Deep-learning to train the agent
Reinforcement Learning & Dynamic Programming Principles	Observation

Solving	Solving Methods Review	Model Simulation



POMDP:

<𝒮tates, 𝒜ctions, 𝒪bservations, 𝒯ransition, ℛeward, γ>

julia

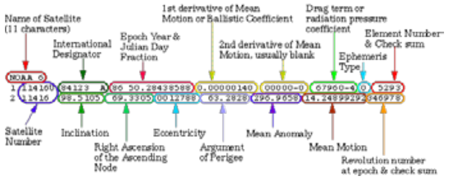
Collision avoidance

1. Data gathering

Ground based (USSPACECOM)

Signal Based (GNSS / GPS)

2. Data Format

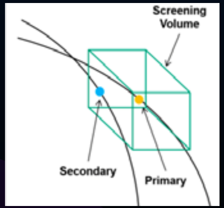


3. Orbital projection (SGP4 perturbator)

TLE
Date
→ SGP4 →
Position (x,y,z)
Velocity (dx,dy,dz)

Cara: Conjunction Assessment Risk Analysis

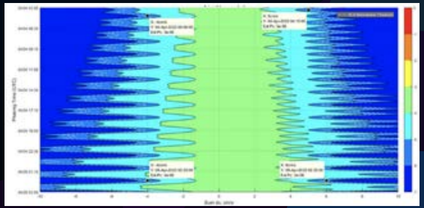
1. Conjunction Event Prediction



2. Close Approach Risk Assessment

Consequence
+
likelihood

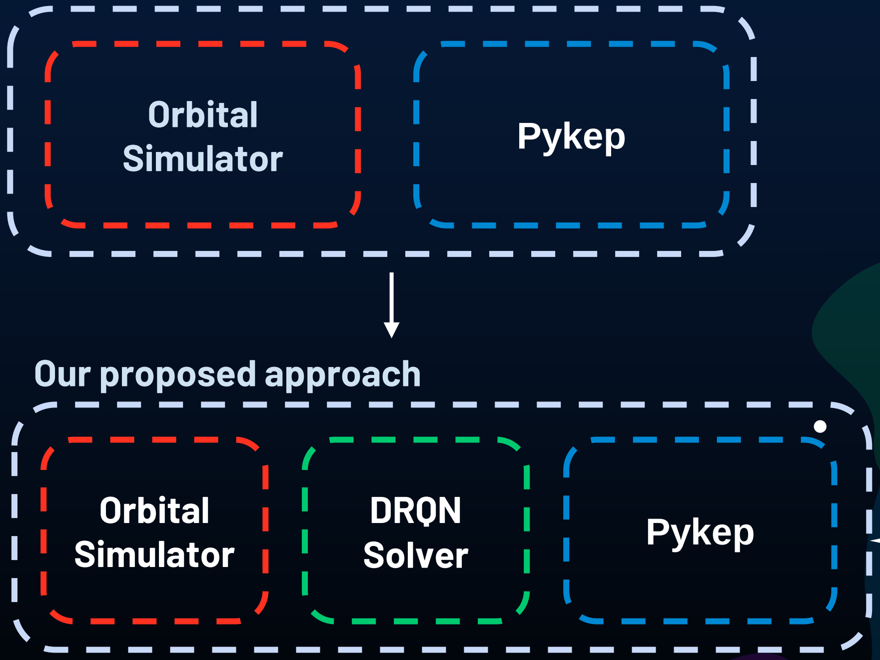
3. Close Approach Risk Mitigation



DATA FORMAT

- ✖ TLE
- ✖ Private databases
- ✖ CDM

Scarcity of collision data



Model: POMDP

<States , Actions , Observations, Transition , Reward , γ>

$\mathcal{S}$: Continuous
u, v : (x, y, z, dx, dy, dz)

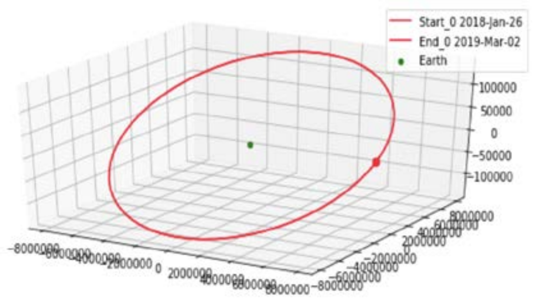
$\mathcal{A}$: Discrete (625 actions)
[0.0, 0.01, 0.05, 0.1, -0.05, time]

$\mathcal{T}$: Model Free

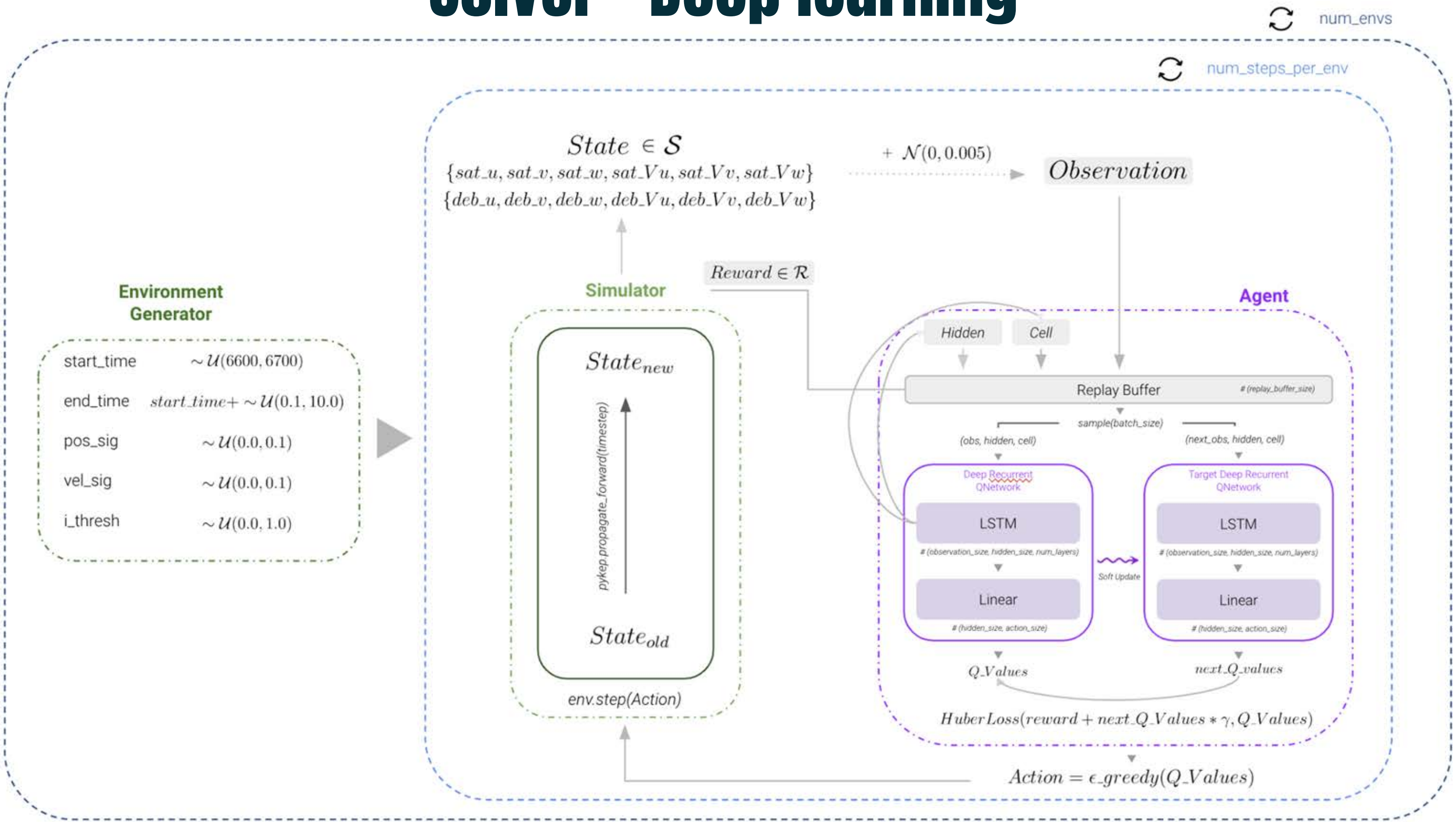
$\mathcal{O}$: Epistemic noise from SGP4
Noises from sensors

$\mathcal{R}$: Collision probability
Fuel level
Trajectory deviation

γ: Discounting rate



Solver - Deep learning



Dataset & Model

SIMULATOR

- Simple pipeline:

TLE Data Collection (SpaceTrack **API**/ Celestrack **JSON**)

↓

Data Preprocessing (**tle-tools/pandas**)

↓

Orbit Simulation (**poliastro**)

Next steps:
- *A more ready-to-use python simulator available ?*

DATASET

- Historical TLE dataset publicly available
- BUT no precision on conjunction events -> hard to train RL algo without conjunction situations

Next steps:
- *Generate artificial TLE ?*
- *Correlate CDM with TLE ?*

MODEL

- Actual simple pipeline helps understand the astrophysics -> helps to model the observations of our 3D problem
- BUT frequency of TLE data appears stochastic and object-dependent

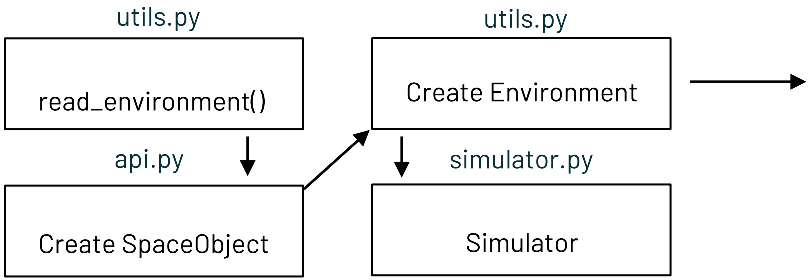
Next steps:
- *Propagate each received TLE with SGP4 to a state vector ?*

$$[(X, Y, Z), (X', Y', Z')]$$

- *Lambert method ?*
- *Use only TLE as observations ?*

Pipeline

Simulator



Agent

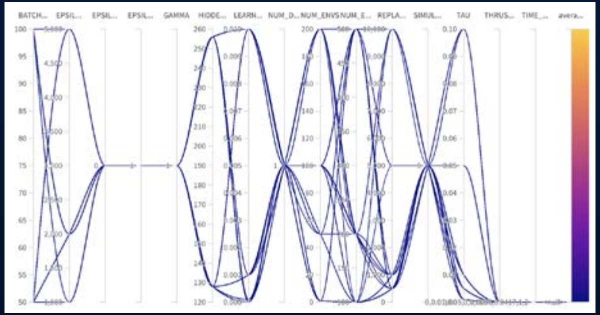
```python
class TableAgent(BaseAgent):

    def __init__(self, action_table=np.array([])):
        """
        Args:
            action_table (np.array with shape=(n_actions, 4) or (4) or (0)):
                table of actions with columns ["dVx", "dVy", "dVz", "time to request"].

        """
        self.action_table = adjust_action_table(action_table)
        self.action_idx = 0

    def get_action(self, state):
        """ Provides action for protected object.

        Args:
            state (dict): environment state
                {'coord' (dict):
                    {'st' (np.array with shape (1, 6)): satellite r and Vx, Vy, Vz coordinates.
                     'debr' (np.array with shape (n_items, 6)): debris r and Vx, Vy, Vz coordinates.}
                'trajectory_deviation_coef' (float).
                'epoch' (pk.epoch): at which time environment state is calculated.
                'fuel' (float): current remaining fuel in protected SpaceObject. }.

        Returns:
            action (np.array([dVx, dVy, dVz, time_to_req])):
                vector of deltas for protected object (m/s),
                step in time when to request the next action (mjd2000).

        """
        epoch = state["epoch"].mjd2000
        if self.action_table.size and self.action_idx < self.action_table.shape[0]:
            action = self.action_table[self.action_idx]
            self.action_idx += 1
        else:
            action = np.array([0, 0, 0, np.nan])
        return action
```

	time_mjd2000	epoch	protected_pos	protected_vel	debris_pos	debris_vel	alert
0	6600.0000	2018-Jan-26 00:00:00	[-4209358.575576871, 6393808.673674893, -1080565.7993943808]	[5942.404053175507, 3627.609290210152, -1726.9822249887982]	[[2713394.019835856, -5280644.467331182, 6687217.586841969], [-496957.2053770637, 5019026.037633914, 5679853.592008259]]	[[-5985.48861375989, -2476.5442529082225, 1818.0490459729094], [3360.027219101076, 4914.968848441329, -4147.126471761581]]	{'is_alert': False, 'info': {}}
1	6600.0007	2018-Jan-26 00:01:00.480000	[-3843511.679768749, 6603006.401406782, -1183254.428461531]	[6152.521714303881, 3288.493346969752, -1667.9141353355612]	[[2348766.3802139875, -5425029.346431939, 6790366.05281432], [-293028.64093000744, 5307767.429083958, 5419724.026258173]]	[[-6069.989596000446, -2297.889672149641, 1593.132833811351], [3381.777273102908, 4630.625922316658, -4452.743071270137]]	{'is_alert': False, 'info': {}}
2	6600.0014	2018-Jan-26 00:02:00.960000	[-3465541.322440386, 6791376.538647311, -1282210.7711207657]	[6343.208206753268, 2939.046604699317, -1603.592518872451]	[[1979444.158502754, -5558577.62091522, 6879950.104785391], [-88124.02781972336, 5578825.388436893, 5141538.5614142455]]	[[-6140.796644937557, -2118.2627835560493, 1369.5630667021776], [3392.288463640473, 4330.354465700382, -4744.037122299136]]	{'is_alert': False, 'info': {}}
3	6600.0021	2018-Jan-26 00:03:01.440000	[-3076640.9493011767, 6958327.315087309, -1377123.1453971958]	[6513.870295530154, 2580.3665031965206, -1534.2201291482263]	[[1606243.829369231, -5681243.215600596, 6956063.977468491], [117074.48328442452, 5831265.328088557, 4846194.911804498]]	[[-6198.299937592521, -1938.0794949060469, 1147.756761300748], [3391.4870592992083, 4015.1158741124027, -5019.9748159507635]]	{'is_alert': False, 'info': {}}

Agent(protected_pos, protected_vel, debris_pos, debris_vel)

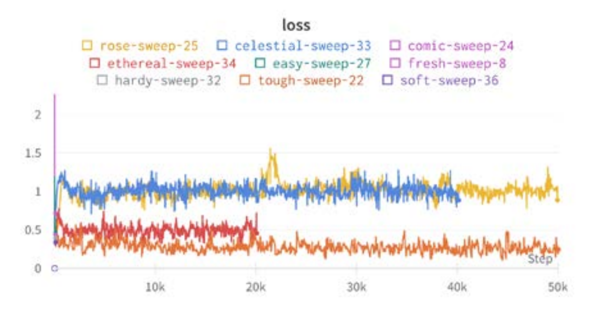
Experiments

Evaluating best hyper parameters

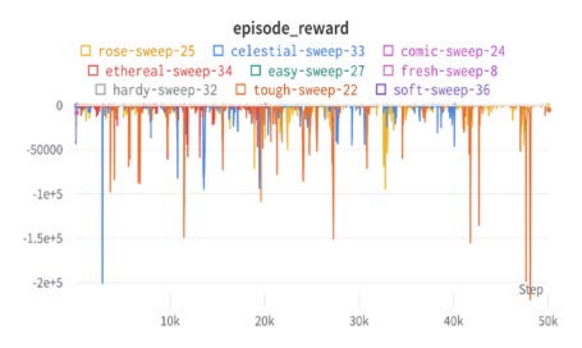


Best total reward sums

Best loss

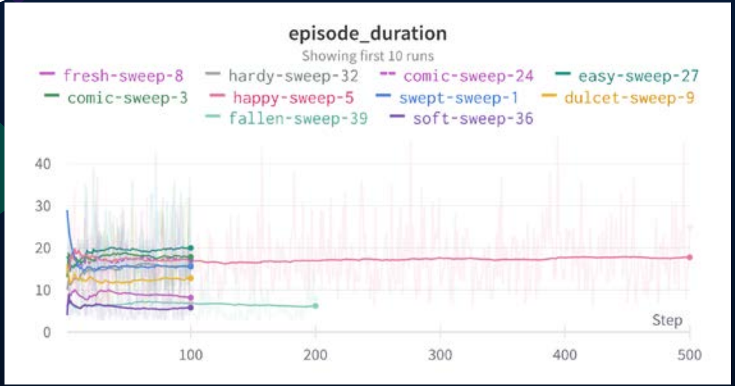


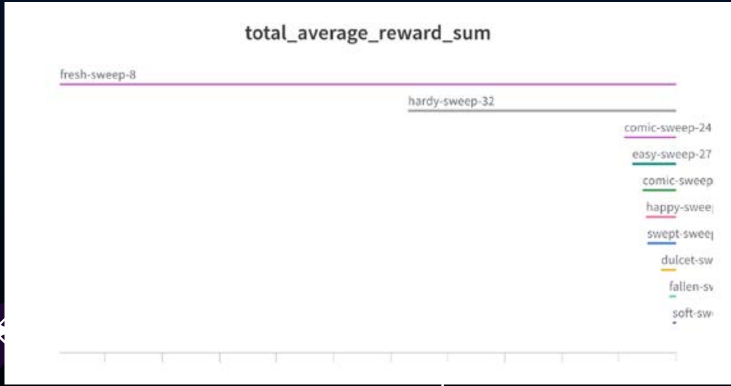
Best episode rewards



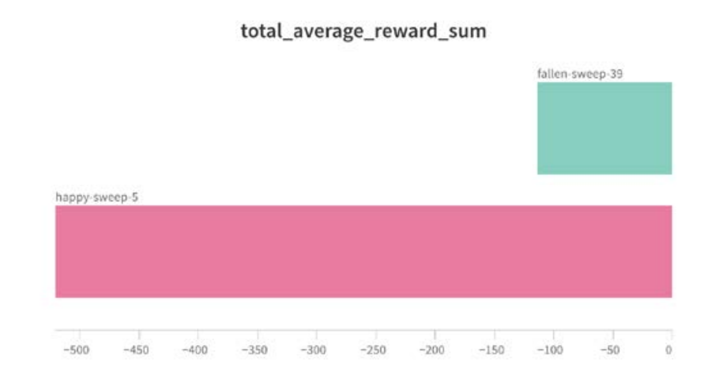
1 environment

All the runs



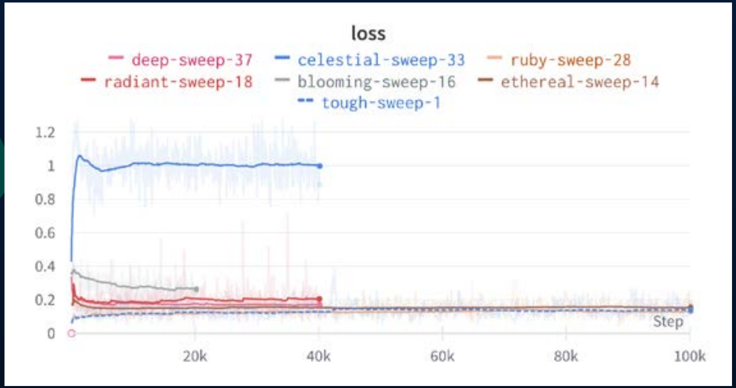


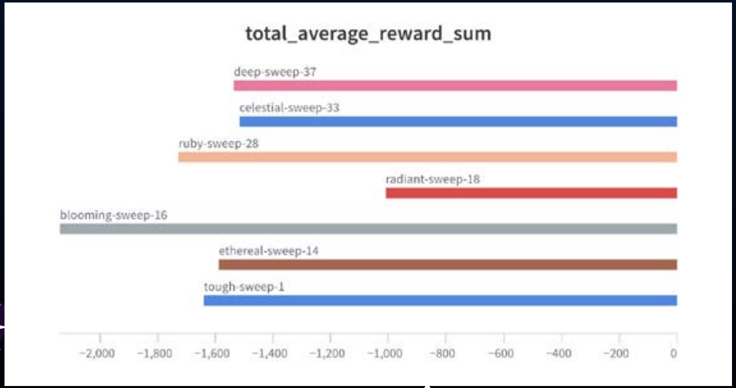
Converging vs diverging



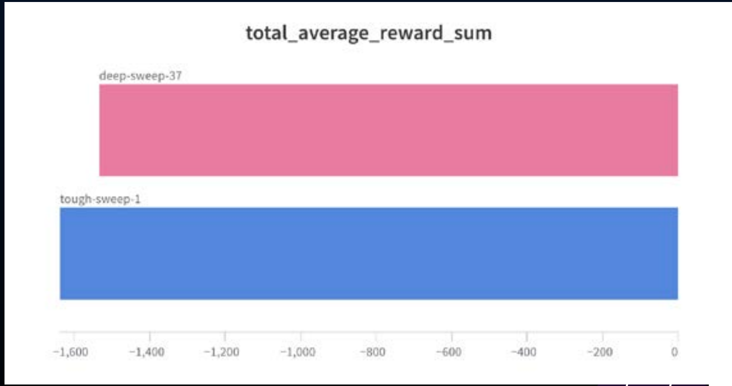
200 environments

All the runs





Converging vs diverging



CentraleSupélec

université PARIS-SACLAY

LGi Laboratoire Génie Industriel

Thank You

Asst. Prof. Adam Abdin – CentraleSupelec - @ESA_Lab CentraleSupelec

adam.abdin@centralesupelec.fr

Students :
Nicolas BOURRIEZ - Adrien LOIZEAU – Susmitha Patnala