doi:10.20944/preprints202103.0071.v1

Article

An Optimised Satellite Constellation for Bushfire Detection through Edge Computing

Minduli Wijayatunga [1,†,‡] and **Xiaofeng Wu** [1,‡]

[1] Affiliation 1; mwij6229@uni.sydney.edu.au
[2] Affiliation 2; xiaofeng.wu@sydney.edu.au
* Correspondence: mwij6229@uni.sydney.edu.au
† Current address: Camperdown NSW 2006, Australia

Abstract: The end of 2019 marked a bushfire crisis for Australia that affected more than 100 000 km^2 of land and destroyed more than 2000 houses. Here, we propose a method of in-orbit bushfire detection with high efficiency to prevent a repetition of this disaster. An LEO satellite constellation is first developed through NSGA-II (Nondominated Sorting Genetic Algorithm II), optimising for coverage over Australia. Then edge computing is adopted to run a bushfire detection algorithm using several constellation satellites as edge nodes to reduce fire detection time. A geostationary satellite is used for inter-satellite communications, such that an image taken by a satellite can be distributed among several satellites for processing. The geostationary satellite also maintains a constant link to the ground, so that a bushfire detection can be reported back without any significant delay. Overall, this system is able to detect fires that span more than 5 m in length, and can make detections in 1.39 s per image processed. This is faster than any currently available bushfire detection method.

Keywords: Bushfire Detection; Edge computing; Satellite constellation; Cubesats

Citation: Wijayatunga, M.; Wu, X. Edge Computing based Satellite Constellation Design for Bushfire Monitoring. *Preprints* **2021**, *1*, 0. https://doi.org/

Received:
Accepted:
Published:

1. Introduction

Bushfires have been associated with Australia annually from October through to March since breaking off from the Gondwana supercontinent [21]. However, the average surface air temperature of Australia has increased over the past century due to climate change [22]. Hence conditions that lead to higher bushfire risks such as high temperature, low humidity and low rainfall have all been on the incline since the turn of the century [22]. As a result, at end of 2019, Australia was in a bushfire crisis. Just within the state of New South Wales, the fires destroyed more than 2000 houses and forced many people to seek shelter elsewhere. By the end of 2019, the total area affected by bushfires were about 100,000 square kilometres, and the total number of deaths added up to thirty-three [24]. This implies a need for increased bushfire monitoring in the upcoming years.

As Australia has around 7.7 million km^2 of land area, it is difficult to detect small bushfires by ground-based observations alone [24]. Failure to detect smaller bushfires can often result in increased spread and damage, as a bushfire can spread at a rate of 2.5 m^2 s^{-1} [26]. Currently, global coverage remote sensing satellites such as Landsat, Aqua, Terra and

Sentinel can detect bushfires within Australia [27]. However, as the revisit times of these satellites are very high, their observations are unable to give real-time fire detections. The Himawari-8 satellite can provide fire detections in 10 minutes as it is in a geostationary orbit [28]. However, its sensor has a large resolution (5 km), thus can only see bushfires that are at least 5 km in length [1]. There is no current method of bushfire detection that is both able to detect small fires and to make detections under 10 minutes. Here we propose a constellation of remote sensing satellites for bushfire detection, equipped with a novel edge computing technology to achieve this capability.

1.1. Development of an Optimised Satellite Constellation

Here, we use a multivariable optimisation algorithm to find a constellation structure that can give us the lowest transit time and the highest coverage of Australia using the least number of satellites. To achieve this, we need to methodically determine the most suitable fitness function, imaging payloads and the particular optimisation algorithm to use.

1.2. Development of an Optimised Bushfire Detection Method

In most instances, bushfire detection and remote sensing are done by sending the satellite images to the groundstations to be processed [27]. The time taken to transmit and receive data is proportional to the amount of data in the images sent, as shown by the following Equation.

$$\text{Time taken} = \frac{\text{Data amount}}{\text{Data rate}}$$

As high resolution images contain a large amount of data, such processes tend to be time-consuming. By processing the images onboard, we can downlink only the coordinates of the locations of fires to the groundstation, reducing the data size and the time taken.

Currently, computing speeds on satellites are significantly lower compared to ground-based computers. This is because satellites are required to use radiation-hardened equipment [30]. As radiation hardening requires extensive testing and development, most advanced processors currently available on the ground do not have radiation-hardened equivalents. Thus if image processing is to utilise a single satellite-the same satellite that takes the image-, the time taken would not be less than the time taken for ground-based processing.

Edge computing using multiple nodes provides a solution to this problem.

> By definition, edge computing is the process of completing requested tasks closer to the requester to save time and storage [32].

Our aim here is to use edge computing to divide the task of processing an image among several satellites in the constellation in the interest of saving time. However, splitting up the image and distributing it to other satellites requires inter-satellite communication links. This can be a network of UAVs or geostationary satellites [32].

The following section describes the methodology we have used to create an edge computing satellite constellation while overcoming complications that arose at different stages of the project. The overall results are then analysed in the following section, followed by a discussion on how this research adds to the existing literature on constellation development and edge computing. Lastly, a conclusion is given to summarise the results obtained.

2. Methods

We start by designing our constellation structure, which consists of picking out suitable imaging sensors for satellites and constructing an optimisation process to obtain the orbits that give the best coverage and transit times. We then develop an edge computing-based on-orbit image processing algorithm to provide fast and efficient fire detections.

2.1. Development of an Optimised Satellite Constellation

The problem we wish to address in this section is the following.

What set of orbits provide the shortest revisit times and the most satellites over Australia at any given point in time?

2.1.1. Optimization to obtain the best coverage and transit time from the constellation

In this section, we use NSGA-II (Nondominated Sorting Genetic Algorithm II) for optimisation, as NSGA II is faster to run compared to other optimisation methods and seldom gets stuck in local optima [15]. The structure of the algorithm is given in Figure 1. Only the chromosome and fitness function vary from problem to problem when implementing NSGA-II. Thus, we have to wisely choose both before the start of the optimisation process.

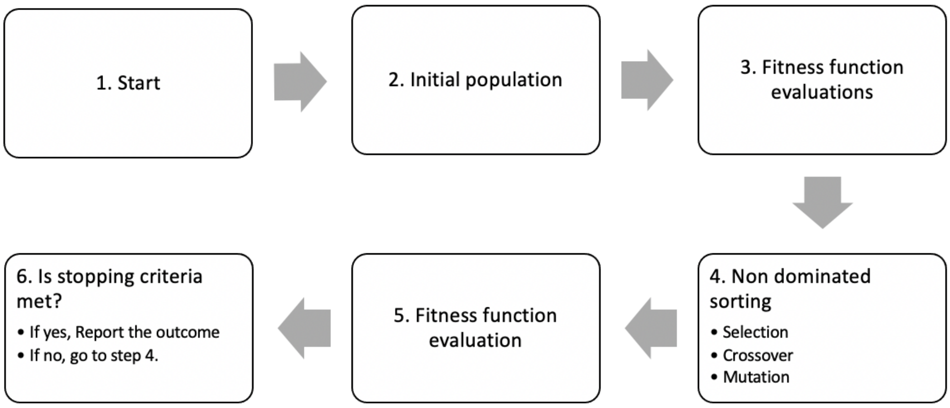
1. Start
2. Initial population
3. Fitness function evaluations
4. Non dominated sorting
• Selection
• Crossover
• Mutation
5. Fitness function evaluation
6. Is stopping criteria met?
• If yes, Report the outcome
• If no, go to step 4.

Figure 1. Flowchart of the NSGA-II algorithm [31]

Chromosome generation

doi:10.20944/preprints202103.0071.v1

After analysing the outcomes of representing the constellation as a whole and as individual satellites in the optimisation process, it can be found that representing the constellation as a whole can generate orbits with more coverage using less total number of satellites [15]. Thus, we use this technique to generate a chromosome. A chromosome (x) is then encoded to contain six genes in the order given in Table 1.

Table 1. Constellation information allocation in a chromosome

Parameter	Position in the chromosome
Semi major axis (a)	$x(1)$
Eccentricity (e)	$x(2)$
Inclination (i)	$x(3)$
Number of orbital planes (P)	$x(4)$
Relative spacing between satellites in adjacent planes (F)	$x(5)$
Number of satellites per plane (n)	$x(6)$

This method of constellation representation is often used to represent Walker constellations [17]. Ascending nodes of each plane of satellites is placed $360/P$. Thus, the right ascension of the ascending nodes (Ω) of each plane is $360/P$ degrees apart. In each plane, the satellites are separated $360/n$ degrees. Thus, the argument of periapsis (ω) of satellites in a single plane is $360/n$ degrees apart. Lastly, the true anomaly of equivalent satellites in adjacent planes is $360F/N$ degrees apart, where N is the total number of satellites, obtained by $N = nP$. Thus, the mean anomalies are also $360F/N$ degrees apart [16]. These relations allow us to convert these six parameters to Keplerian parameters.

The upper and lower limits of the genes in our chromosome also need to be set so that our solution space is in the expected region of space. We are searching for a low Earth orbit constellation; thus, we restrict the orbital altitudes to 200 km - 1000 km. Thus $(R_e + 200)\text{km} \leq a \leq (R_e + 1000)\text{km}$, where R_e is the radius of Earth and is taken to be 6371 km [14]. As a is relatively small, we cannot make our orbits highly eccentric without having their perigees be inside (or on the surface of) the planet. Thus we restrict our range of eccentricities to $0 \leq e \leq 0.05$. We also limit P to 1-100, n to 1-50 and F to 1-8 such that a constellation with at most 5000 satellites can be devised. It can be noted that as we can optimise P and n, we can optimise the total number of satellites in the constellation.

Fitness Function Determination

Now that we have our chromosomes encoded and the limits of each gene set, we go on to the derivation of the fitness function. Due to the nature of optimisation algorithms, the output of the fitness function must be a quantity to be minimised [18]. In our case, we need a fitness function output that is proportional to the total number of satellites and inversely proportional to the coverage over Australia.

A variable that represents the coverage is the number of satellites above Australia at a given time. We shall call this variable C from now on. The more satellites that are over the country, the greater proportion of the land is observed.

Another variable that also represents coverage is obtained by the process of constructing a 200 × 200 grid through our area of interest (shown in Figure 2). The latitude and longitude of each grid point are determined, and the number of grid points that the constellation covers within a day is counted. We call this parameter P (for points covered). As the middle of Australia does not have the conditions to grow forests, they seldom experience bushfires [24]. Thus, we exclude the grid points in the middle when calculating P.

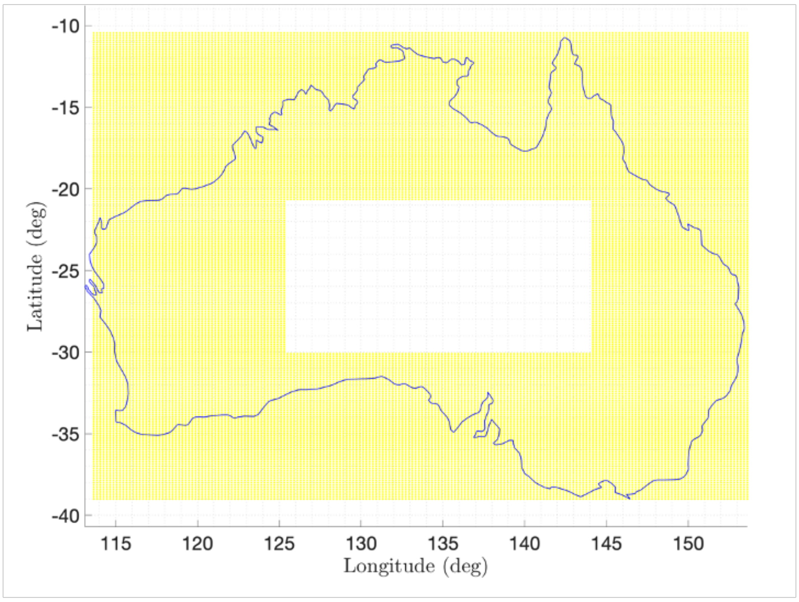

Figure 2. Area expected to be covered by the constellation within the Australian mainland. Each yellow point denotes a gridpoint.

A variable that is proportional to the revisit time is the average time between two satellite transits across the area of interest. We shall call this variable R. This does not represent the revisit time of an individual satellite, but the revisit time of the constellation.

We can expect that the coverage and revisit times would increase with the number of satellites [15]. However, as the launch and construction of satellites is an expensive process, we require the minimum number of satellites that still gives full coverage of the area of interest. Therefore, we must minimise the total number of satellites in the constellation (N).

Thus, the fitness function goes through the following steps to determine C, P and R and to combine them with N to form an output.

1. Create a matrix of instances(I) to store the position data created.
2. Create a variable P to store the number of grid points crossed as each satellite passes above the area of interest.
3. For satellite number $i = 1$ to $i = N$, do the following.
   (a) **Obtain the Earth-centred inertial (ECI) coordinates of the satellite for 1 Earth Day.**
   First, we calculate the mean anomalies (MA) within the satellite orbit using Equation 1.

$$\mathrm{MA} = \mathrm{MA} + \sqrt{\frac{\mu}{a^3}}t \tag{1}$$

Here, μ is the gravitational constant and t is the time. Then we use the Newton-Raphson method to obtain the Eccentric anomalies(E) from the Mean anomalies calculated, using Equation 2.

$$M = E - e \sin E \tag{2}$$

The true anomaly (f) is then calculated using Equation 3.

$$f = 2 \tan^{-1}\left(\sqrt{\frac{1+e}{1-e}} \tan \frac{E}{2}\right) \tag{3}$$

The radial distance to the satellite from the centre of the Earth at each point of its orbit is then calculated.

$$r = \frac{a(1-e^2)}{1+e\cos f} \tag{4}$$

Then the 2-D orbit coordinates are determined.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} R\cos\theta \\ R\sin\theta \\ 0 \end{bmatrix} \tag{5}$$

Finally, the ECEF coordinates are determined by multiplying the 2-D coordinates by the following conversion matrix.

$$C_{Orbit}^{ECI} = \begin{bmatrix} \cos\omega\cos\Omega - \sin\Omega\sin\omega\cos i & -\cos\Omega\sin\omega - \sin\Omega\cos\omega\cos i & \sin\Omega\sin i \\ \sin\Omega\cos\omega + \cos\Omega\sin\omega\cos i & -\sin\omega\sin\Omega + \cos\Omega\cos\omega\cos i & -\cos\Omega\sin i \\ \sin\omega\sin i & \cos\omega\sin i & \cos i \end{bmatrix} \tag{6}$$

(b) **Obtain the latitudes and longitudes traversed by the satellites through coordinate conversions.**

First, we have to convert the ECI coordinates to Earth-centred Earth Fixed(ECEF) coordinates by multiplying them by the following conversion matrix.
Given that:

$$\omega = 7.292 \times 10^{-5}$$

$$t = \text{Time since the vernal equinox}$$

$$C_{ECI}^{ECEF} = \begin{bmatrix} \cos\omega t & \sin\omega t & 0 \\ -\sin\omega t & \cos\omega t & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Then we convert the ECEF coordinates to longitude as follows.

$$\text{Longitude} = \tan^{-1}\left(\frac{y_{ECEF}}{z_{ECEF}}\right) \tag{7}$$

The latitude is then determined iteratively using Equation 8, where R_E is the radius of Earth.

$$\text{Latitude} = \tan^{-1}\left(\frac{z + e^2 R_E(\frac{z}{R_E(1-e^2)} + h)}{\sqrt{x^2 + y^2}}\right) \tag{8}$$

(c) If at time t, the latitude and longitude lies within the yellow section in Figure 2, denote:

$$\text{instance(t)} = 1$$

(d) Count the number of grid points crossed by the satellite once it is in the yellow region in Figure 2. Add this to P.

(e) If at time t, it does not lie within the yellow region, denote:

$$\text{instance(t)} = 0$$

(f) Now, for satellite(i), a vector of instances is generated. This vector has 1 for instances when the satellite can see the chosen area of interest, and 0 when that satellite cannot. Let's call this vector:

$$\text{instances}_i$$

4. The final I matrix should take the following form.

$$I = \begin{bmatrix}\text{instances}_1 & \text{instances}_2 & \dots & \text{instances}_i & \dots & \text{instances}_N\end{bmatrix}^T \tag{9}$$

5. To determine C, sum the instances matrix along its columns. This tells us how many satellites are visible at each time step. Taking the average of this gives us the C parameter.

$$C = avg(\sum_{1}^{j=j_{max}} I_{ij}) \tag{10}$$

6. To determine R, calculate the space between the consecutive 1s in the I matrix, and find the average. This gives the average time gap between transits.

7. We have to maximise C and P and minimise R and N. Thus, the fitness function used is the following. As all parameters are of equal importance, we need to ensure that they are all on the same scale. Thus, we introduce a factor of 0.01 in front of R.

$$\text{Fitness function} = \frac{1}{C} + 0.01R + \frac{1}{P} + N \tag{11}$$

2.1.2. Choosing Imaging Sensors for the Satellite Network

Being able to detect small scale fires requires high-resolution imaging (a pixel size ≤ 5m) [13]. However, high resolution leads to a lower field of view (hence, smaller swath width), which increases the number of satellites required to observe the area of interest fully [12].

Note that the 200 × 200 grid constructed by the fitness function allows the distance between any two grid points to be 22.2 km [8]. Thus, when all grid points are covered, there would be 100% coverage as long as the swath widths of the constellation satellites are greater than 44.4 km, as shown in Figure 3. Thus, we need the swath width of the satellites to be greater than 44.4 km.

Hence, an image sensor must be chosen such that its sensor parameters adhere to these conditions.

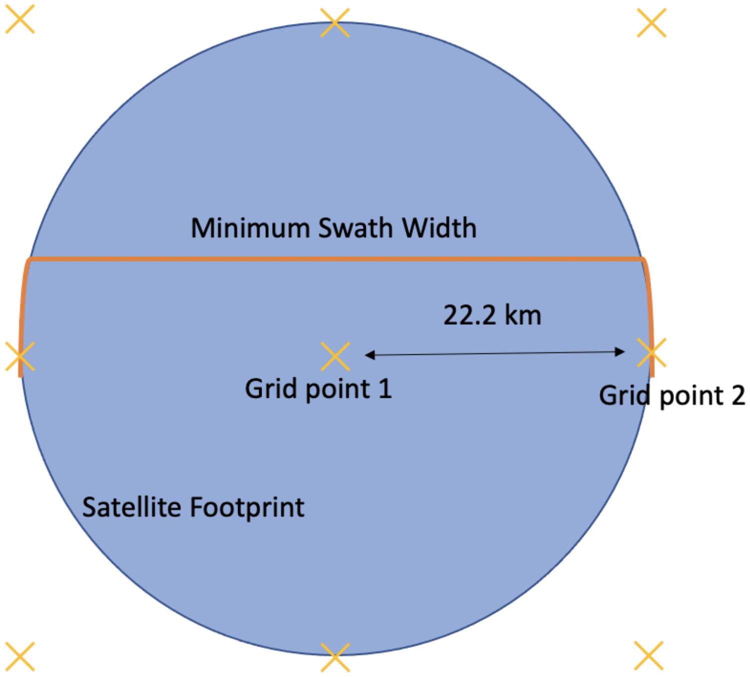

Figure 3. The distance between two grids being 22.2 km requires each satellite to have a swath width of at least 44.4 km (2 × 22.2 km) to ensure full visibility when all grid points (shown as yellow crosses) are covered.

2.2. Development of an Optimised Bushfire Detection Method

2.2.1. Construction of an Onboard Image Processing Algorithm for Bushfire Detection

In this section, we must first adopt a bushfire detection algorithm that can be conducted on-board the satellites. The Collection 6 bushfire detection algorithm is proven to provide the most accurate fire detections from images of Australia taken by the Terra satellite during the bushfire season [15]. The detailed algorithm is given in [13]. However, this algorithm is intended to be used for ground-based processing, as later parts of it rely on image comparisons. In this project, we have adapted parts of this algorithm and modified it such that it could be used onboard satellites [15].

2.2.2. Constructing an Edge Computing-based Image Processing Algorithm

As mentioned in the Introduction, computing speeds onboard satellites are much lower than ground-based computing speeds. One of the fastest processors that are considered to be space-grade at the moment is the RAD750, which has a maximum CPU clock rate of only 200 MHz [6]. Hence running the algorithm on a single satellite is inefficient. Thus, here we develop an innovative, edge-computing based method that uses several satellites for onboard bushfire detection. To do so, we require inter-satellite communication, which can be achieved by a geostationary satellite [4]. With its high altitude, a geostationary satellite can maintain communications with a large number of LEO satellites in the constellation at one time.

Choosing a Geostationary Orbit for Inter-satellite Communications

Here, we choose our inter-satellite link to be a satellite in orbit similar to Optus D3. As this is a geostationary orbit it has a constant view of Australia at all points in time [3]. The Keplerian coordinates of this satellite orbit are given in Table 2.

Table 2. Keplerian coordinates of the geostationary satellite selected for inter-satellite communications

Parameter	Value
Semi major axis (a)	42 165 km
Eccentricity (e)	0.0002541
Inclination (i)	0.0116°
RAAN (Ω)	48.4858°
Argument of perigee (ω)	135.8460°
Mean anomaly (MA)	294.4219°

Determining the LEO Satellite Visibility to the GEO Satellite

At any given point in time, we need to know what LEO satellites of the constellation are visible to the GEO satellite to know what satellites can be allocated processing tasks.

According to Figure 4, when the distance between an LEO satellite of the constellation and the GEO satellite is greater than x, the satellites would not be visible to each other. From Figure 4:

$$\alpha = \cos^{-1}(\frac{R_e}{a(LEO)}) \tag{12}$$

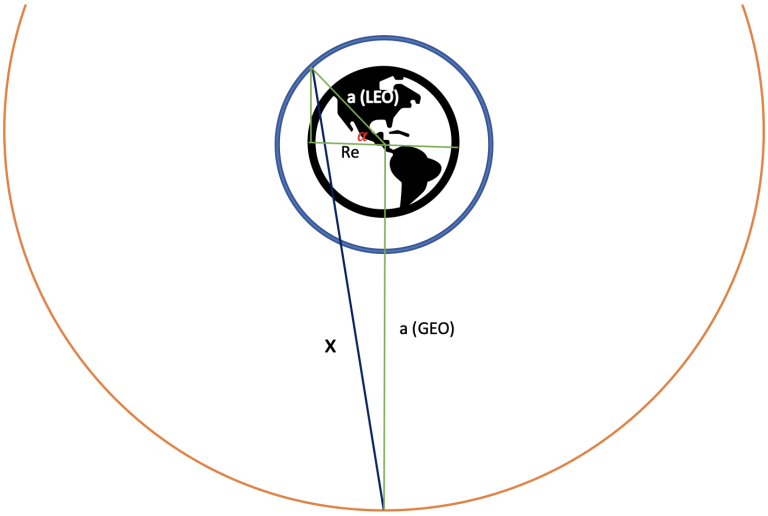

Figure 4. Visibility determination between LEO satellites and the GEO satellite. The LEO satellite orbit is shown in blue, while the GEO satellite orbit is shown in orange. x denotes the closest distance between the LEO and GEO satellites at a given time t.

Where R_e is the radius of Earth and $a(LEO)$ is the semi-major axis of the LEO orbit in question. We assume that both LEO and GEO orbits are circular to make our calculations simpler. This is a valid assumption as the eccentricities of all orbits involved are very low. Now by cosine rule:

$$x = \sqrt{a(LEO)^2 + a(GEO)^2 - 2a(LEO)a(GEO)\cos(90 + \alpha)} \tag{13}$$

If the distance between any LEO satellite and the GEO satellite is less than x, the satellites are visible to each other. Now we can use this to determine which LEO satellites are visible to the geostationary satellite so that inter-satellite communication links can be established between them.

Steps of Image Processing Task Allocation

The following actions must take place in the given order to successfully distribute the image data among the constellation satellites, to compile the results together, and to transmit the outcome back to Earth. These steps are also illustrated in Figure 5.

1. A satellite above Australia captures an image extracts the reflectance, radiance, latitude and longitude data for each pixel in the image and segments the image data to n parts.
2. The satellite sends (n-1) parts to the GEO satellite.
3. The GEO satellite receives the (n-1) segments and identifies which LEO satellites are visible to it using the calculation described in the previous section, and sends each segment to a different visible LEO satellite in the network.

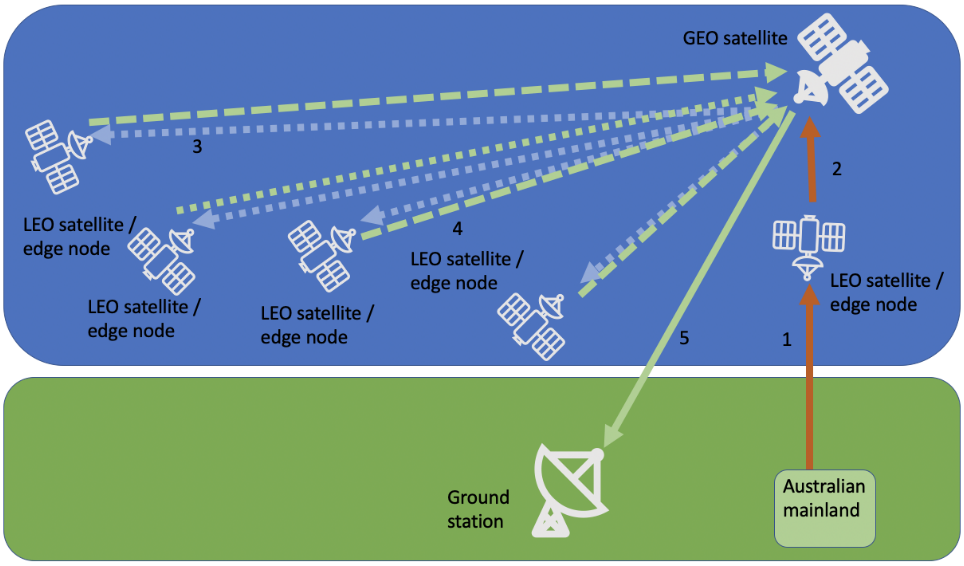
GEO satellite
3
2
LEO satellite / edge node
4
LEO satellite / edge node
LEO satellite / edge node
LEO satellite / edge node
LEO satellite / edge node
5
1
Ground station
Australian mainland

Figure 5. Inter-satellite comunication links established for the edge computing process

4. Each satellite receives its segment of data and processes it using the Collection 6 algorithm described previously. If a bushfire pixel is detected within the segment of data, the satellite relays the pixel's latitude and longitude back to the GEO satellite.
5. GEO satellite collects all bushfire coordinates it receives and sends the compilation down to an Australian groundstation. If no bushfire pixels are present, nothing is relayed to the ground, saving time and preventing redundant communications.

Each LEO satellite involved works as an edge node in this process, processing the data it receives through the GEO satellite. The GEO satellite works as the central edge node, allocating tasks to other edge nodes/satellites. What makes this an application of edge computing is the fact that none of the processing happens on the ground. Data is processed on the data receiver itself (and on other edge nodes).

3. Results

In this section, we provide the results obtained using the methodology discussed in the previous section. The final constellation structure, the results and performance of the onboard image processing algorithm, as well as the reduction in detection time due to the use of edge computing-based task distribution are discussed here.

3.1. Development of an Optimised Satellite Constellation

3.1.1. Choosing Imaging Sensors for the Satellite Network

As discussed in the previous section, we require high-resolution imaging with a pixel size of at least 5 m to detect small, initial-stage bushfires [13]. We also need swath widths that are greater than 44.4 km to satisfy our criteria for coverage. Thus, we choose the RapidEye satellite sensors to be onboard our satellites. These sensors have multi-spectral detection capabilities in the visible and infrared range, along with a field of view of 13.08 degrees [25].

3.1.2. Final Constellation Structure

The final constellation structure derived has 100% coverage over the area of interest and 75.4% coverage over the middle of Australia. Its parameters are given in Table 3.

Table 3. Final constellation orbits

Parameter	Value
Semi major axis (a)	7334.9 km
Altitude (h)	963.9 km
Eccentricity (e)	0.04
Inclination (i)	141.39°
Number of orbital planes (P)	95
Relative spacing between satellites in adjacent planes (F)	9
Number of satellites per plane (n)	42
Total number of satellites (N)	3990

Figure 6 shows how the number of satellites over Australia varies throughout the day for this constellation, while Figure 7 shows its coverage over Australia when only the minimum number of satellites are visible to the area of interest. Figure 8 shows the ground track of 80 randomly chosen satellites of the constellation.

This constellation generates a new image of the area of interest once every 6 minutes and can generate a new image of central Australia once every 10 minutes.

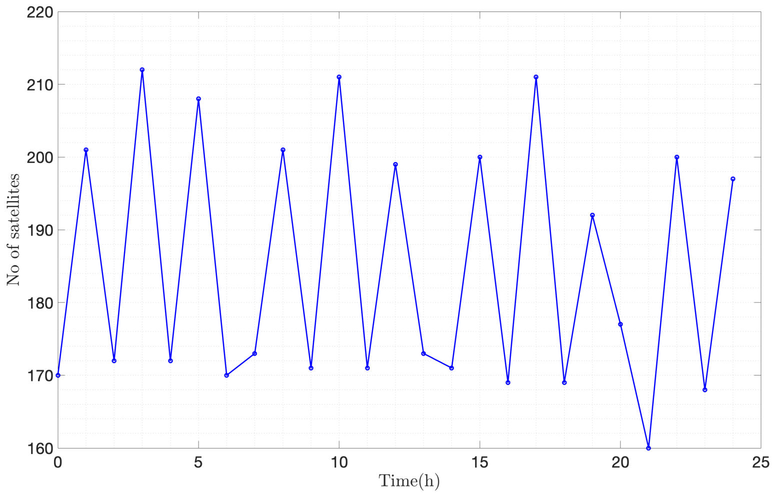

Figure 6. Number of satellite above Australia at each hour of the day

The Swath width of a satellite in the constellation can be calculated as follows using Figure 9.

$$\text{Swath Width} = 2h\tan\theta = 2 \times 963.9 \tan 6.99 = 236.4\,\text{km}$$

As this is greater than 44.4 km, the coverage criterion discussed in section 2.2.1 is fulfilled.

doi:10.20944/preprints202103.0071.v1

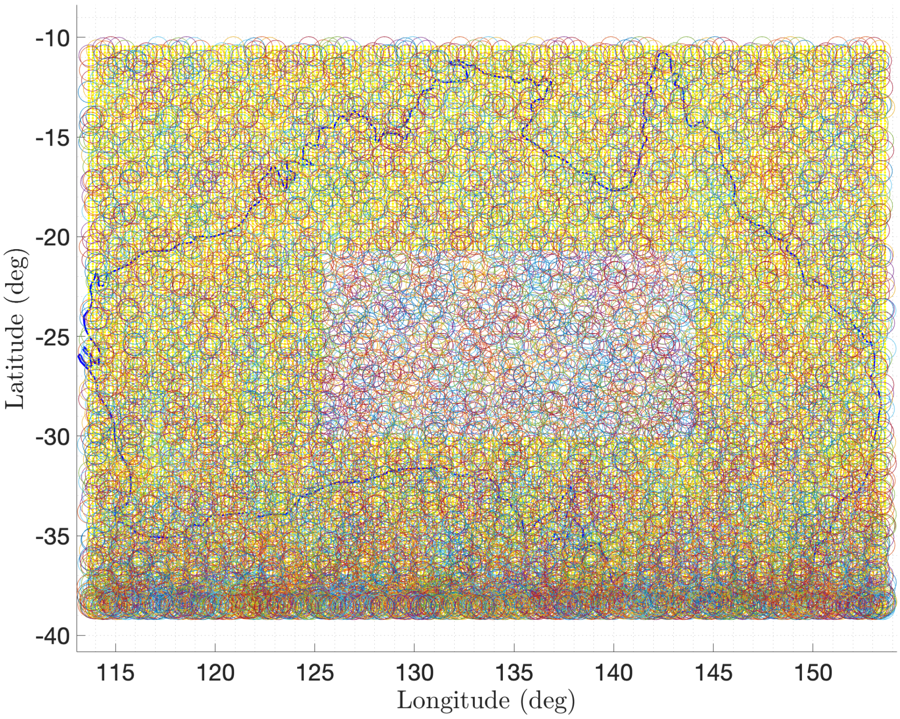

Figure 7. Coverage of the constellation (Area of interest is highlighted in yellow)

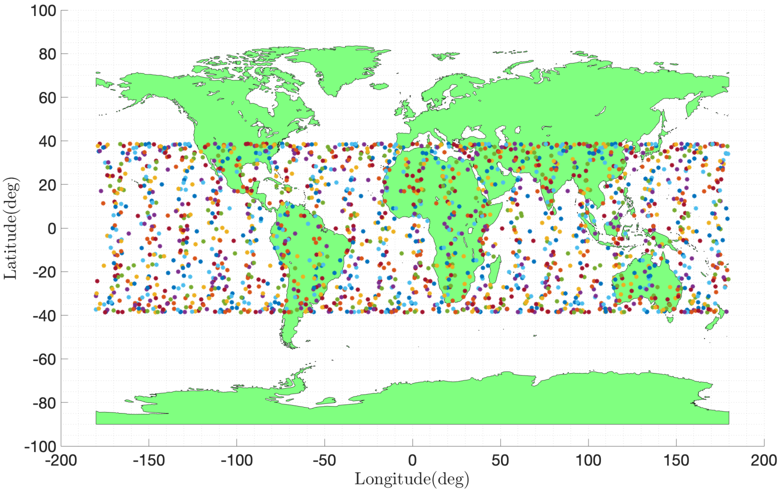

Figure 8. Ground track of the constellation

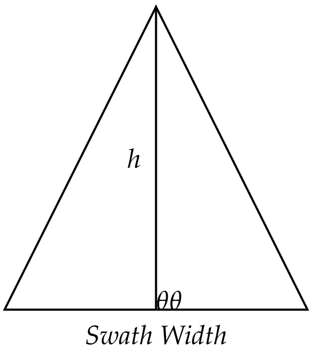

Figure 9. Swath width calculation using the field of view angle, given by 2θ

3.2. Development of an Optimised Bushfire Detection Method

3.2.1. Accuracy of the Onboard Image Processing Algorithm

In this section, we provide the results from the Collection 6 image processing algorithm discussed in the methodology section. Figures 10, 11 and 12 show the bushfires detected using this algorithm. The red spots indicate fires detected using our algorithm, while the blue spots indicate known fires from NASA's Fire Information for Resource Management System (FIRMS) database [11].

The three images processed have detection accuracies of 95.4%, 94.5% and 59.3%, respectively. Overall, the algorithm can predict fire locations with an average accuracy of 83.06%. The errors can be attributed to the fact that NASA's records of confirmed fires contain data from Terra, Aqua and other satellites, whereas our algorithm has only processed image data from the satellite Terra. Furthermore, NASA's records show fires happening throughout the day, while the image data processed by our algorithm is only relevant to a specific time of that day [11].

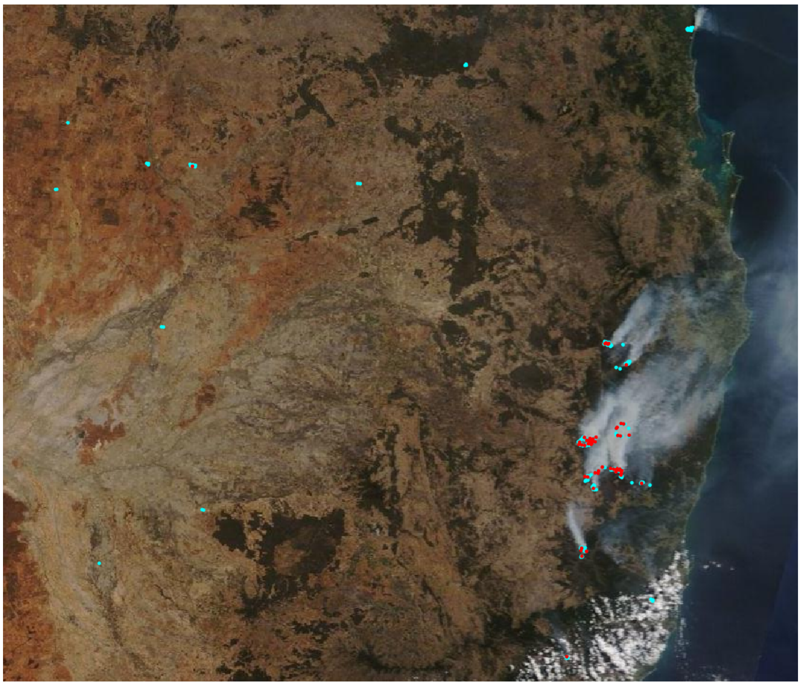

Figure 10. New South Wales as seen by Terra on 13th September 2019 [19], with the potential bushfire locations denoted in red. Detection accuracy is at 95.4%.

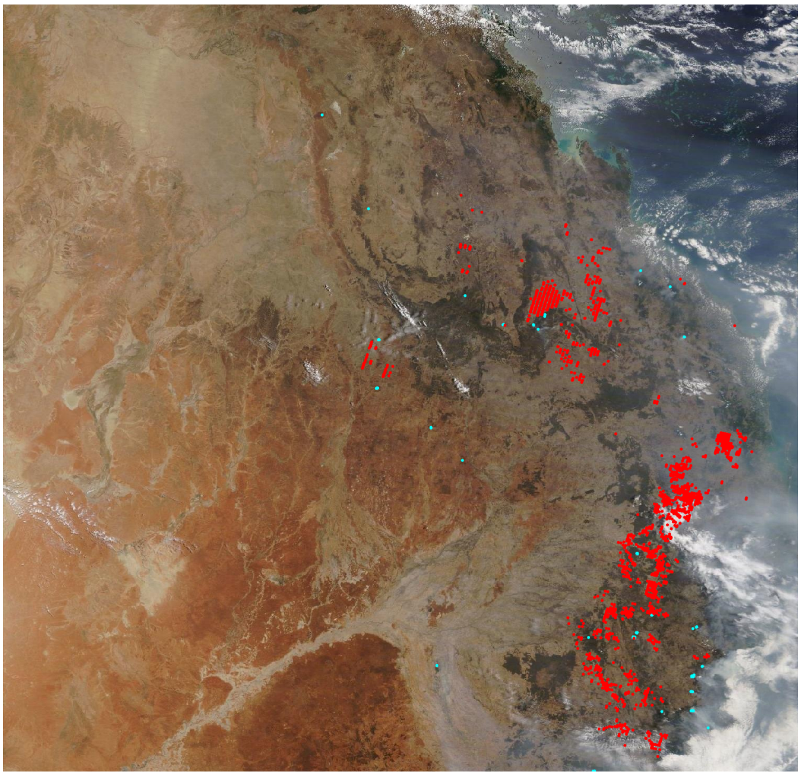

Figure 11. New South Wales as seen by Terra on 11th December 2019 [19], with the bushfires detected using the algorithm denoted in red. Detection accuracy is at 94.5%.

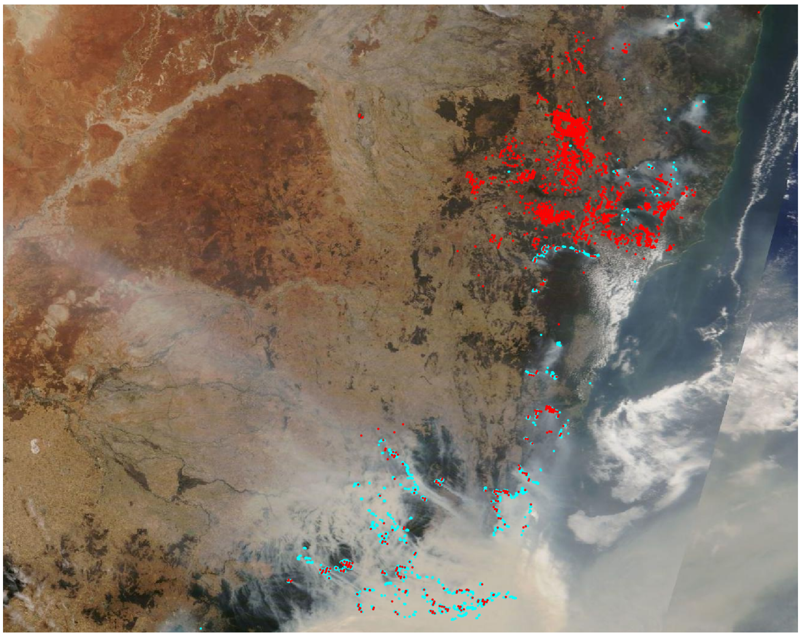

Figure 12. New South Wales as seen by Terra on 3rd January 2020 [19], with bushfire detections shown in red. Confirmed bushfires are shown in blue. Some bushfires have gone undetected in this image, showing that our preliminary method needs to be improved. Accuracy of the detections are at 59.3%.

3.2.2. Edge Computing-based Image Processing Task Allocation Results

In this section, we discuss the results obtained in the innovative process of using edge computing to allocate tasks of image processing among constellation satellites. Using

Equation 13, we find that at a given point in time, at least 95% of the LEO satellites (i.e. at least 3793 satellites) in the constellation are visible to the GEO satellite.

Now we calculate the time taken to complete the five steps mentioned in the previous section, to see if our algorithm can provide improved fire detection time compared to current bushfire detection techniques.

- t_1 - Time is taken by the onboard sensor to collect sufficient photons from the ground to achieve the required resolution. This is known as the dwell time [20]. It can be calculated using Equation 14.

$$t_1 = \text{Dwell Time} = \frac{\text{Down track pixel size/Orbital velocity}}{\text{Cross-track line width / Cross-track pixel size}} \tag{14}$$

Our satellites are to have similar sensor capabilities to the RapidEye satellites as mentioned before. Thus, it is sensible to use a similar Multispectral push broom sensor for our satellites with a down track pixel size of 5 m. We shall calculate the lowest orbital velocity so that we can get an upper estimate of t_1. For this, we use the largest radial distance possible under the limits imposed on the chromosome used for the optimisation process. $r^{sat}_{min} = 7375\,\text{km}$

$$\text{Max orbital velocity} = \sqrt{\frac{GM_{Earth}}{r^{sat}_{min}}} = \sqrt{\frac{6.673 \times 10^{-11}(5.98 \times 10^{24})}{7375 \times 10^{3}}} = 7355\,\text{m}\,\text{s}^{-1}$$

For a push broom sensor, Cross-track line width / Cross-track pixel size = 1 [20]. We assume an additional 1 second is taken for segmenting the data into n parts. Thus:

$$t_1 = \frac{5/7355}{1} + 1 = 1.000\,679\,7\,\text{s}$$

- t_2 - This is the time taken to transmit (n-1) segments of data from the LEO. It is sensible to assume that the data rate of our satellites would also be close to the RapidEye's data rate, which is 80 Mbit/s [34]. For both RapidEye and MODIS, a single output file of data contains 559.3 kbit of data. Thus:

$$t_2 = \text{Time taken to transmit data from LEO Satellite} = \frac{\text{Data amount}}{\text{LEO data rate}} = 0.006\,99\,\text{s}$$

- t_3 - This is the time taken for the signal to travel from the LEO satellite to the GEO satellite.

$$t_3 = \text{Time taken for the data to travel from LEO to GEO} = \frac{\text{Distance from LEO to GEO}}{\text{Speed of light}}$$

The GEO satellite selected is at a semi major axis of 42 165 km. The LEO satellites are at a semi major axis of 7375 km. Thus the distance between them is 34 790 km. Thus:

$$t_3 = 0.1160\,\mathrm{s}$$

- t_4 - This is the time taken for the GEO satellite to receive the data. For a satellite like Optus D3, the communication bandwidth is around 54 MHz [3]. Assuming our satellite has the same bandwidth, we can determine the maximum possible data rate by Shannon's theorem [29].

$$C = B\log_2(1 + SNR)$$

Here, C is the maximum data rate, B is the bandwidth, and SNR is the signal to noise ratio. The typical SNR for RapidEye satellite data (which is now going to the GEO satellite) is around 210 [13]. This gives us a data rate of :

$$C = B\log_2(1 + SNR) = 54 \times 10^6 \log_2(1 + 210) = 416\,939\,356\text{bps} \approx 416.9\text{Mbps}$$

$$t_4 = \text{Time taken for the GEO satellite to receive data}$$

$$t_4 = \frac{\text{Data amount}}{\text{GEO data rate}} = \frac{559300}{416939356} = 0.001\,34\,\mathrm{s}$$

- t_5 - This is the time taken for the GEO satellite to redistribute the data to other edge nodes for processing. We assume that the same amount of time is taken for the GEO satellite to redistribute the data to other edge nodes for processing.

$$t_5 = \text{Time taken for the GEO satellite to send away data} = t_4 = 0.001\,34\,\mathrm{s}$$

- t_6 - This is the time taken for the signals to travel from the GEO satellite to the chosen LEO satellites.We can assume this to be same as t_3.

$$t_6 = \text{Time taken for the data to travel from GEO to LEO} = t_3 = 0.1160\,\mathrm{s}$$

- t_7 - Now, each LEO satellite takes time to receive the data from the GEO, and the time taken depends on the amount of data received. Assuming we have n edge nodes available, we get the following.

$$t_7 = \text{Time taken for the LEO satellites to receive data} = \frac{\text{Data amount}}{\text{LEO data rate}} = \frac{\frac{559300}{n}}{80 \times 10^6}$$

- t_8 - This is the time taken for processing the image data. To get an accurate estimate of how much time our algorithm would take to run onboard a satellite, we convert the bushfire detection code to C, compile it and then count the number of lines in the generated assembly (.asm) file. Assuming that a single clock cycle processes a single line of assembly code, we can roughly estimate the time that the algorithm would take to run onboard a satellite [15].

Once the C code was compiled, it was noted that the generated assembly code had **241878560** lines. The fastest processor that is considered to be space-grade at the moment is the RAD750, which has a maximum CPU clock rate of only 200 MHz [6]. Thus, the time taken to run this algorithm can be calculated as follows:

$$t_8 = \text{Processing time} = \frac{1}{n}\frac{\text{Number of lines in the assembly code}}{\text{Clock rate}} = \frac{1}{n}\frac{241878560}{200 \times 10^6}$$

- t_9 - The LEO satellites need time to send the coordinates of bushfire pixels back to the GEO satellite. We will assume here that the data sent away is the same size as the data received (It is likely less than this).

$$t_9 = \text{Time taken for the LEO satellites to send data} = \frac{\text{Data amount}}{\text{LEO data rate}} = \frac{\frac{559300}{n}}{80 \times 10^6}$$

- t_{10} - Time is also taken for the signal to travel from LEO to GEO, we can again assume this to be the same as t_3.

$$t_{10} = \text{Time taken for the data to travel from LEO to GEO} = t_3$$

- t_{11} - Time is taken for the GEO satellites to receive data and then -if there are bushfire pixels- send data to the ground. Again assuming that the data received in this case is of the same size as the data received in t_5,

$$t_{11} = \text{Time taken for the GEO satellite to receive data} = t_5$$

- t_{12} - Time is taken for the GEO satellite to send the coordinates of fire pixels to the ground station. Again assuming size similarity, this time is equivalent to (and less than as the data size is likely smaller) t_4.

$$t_{12} = \text{Time taken for the GEO satellite to send data to ground} = t_4$$

- t_{13} - Lastly, time is also taken for the signal to travel to the ground.

$$t_{13} = \text{Time taken for the data to travel from GEO to ground}$$

$$t_{13} = \frac{\text{Distance from GEO to ground}}{\text{Speed of light}} = \frac{42\,165 \times 10^3}{c} = 0.140\,\text{s}$$

By summing up the above times for different values of n, we can see how increasing the number of edge nodes (satellites) speeds up the process of bushfire detection. For the constellation structure described in this chapter, Figure 13 illustrates how the detection time reduces and plateaus out as the number of edge node increases.

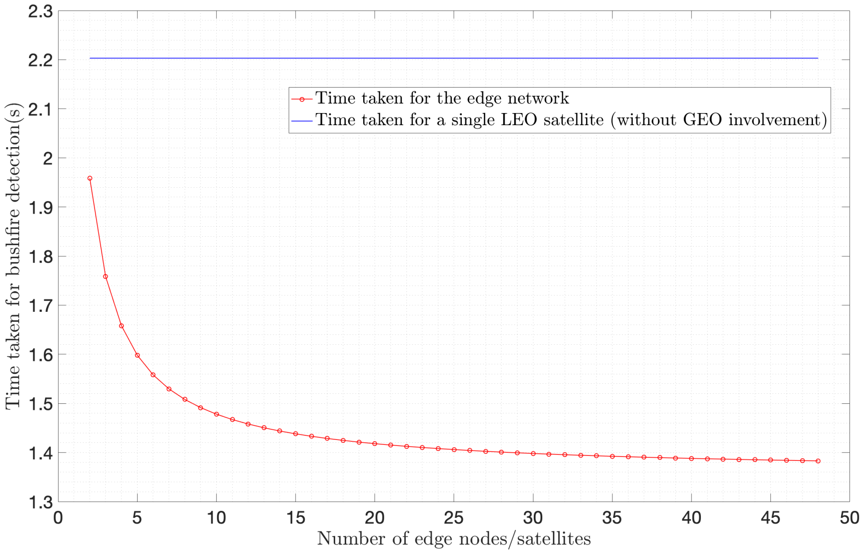

Figure 13. The relationship between the time taken to detect bushfires and the number of edge nodes used

In Figure 13, the blue line indicates the time taken to process an image onboard using just one satellite, without the use of the GEO satellite or other edge nodes. It is calculated by adding up the following times.

- t_1' - This is once again the dwell time and the time taken to extract the reflectance and radiance data from an image. Thus,

$$t_1' = t_1 = 1.000\,679\,6\,\text{s}$$

- t_2' - This is the time taken to process the data using just one satellite. Thus,

$$t_2' = \frac{\text{Number of lines in the assembly code}}{\text{Clock rate}} = 1.209\,\text{s}$$

- t_3' - This is the time taken for the LEO satellite to send the data to the ground if a bushfire is detected. Both RapidEye and Modis has output files of 559300 bytes of data per image, and the LEO data rate is again assumed to be the same as that of RapidEye [34].

$$t_3' = \frac{\text{Data amount}}{\text{LEO data rate}} = \frac{559300}{80 \times 10^6} = 0.006\,99\,\text{s}$$

- t'_4 - This is the time taken for the signal to travel from the LEO satellite to the ground station.

$$t'_4 = \frac{\text{Distance from LEO to Ground}}{\text{Speed of light}} = \frac{963.9 \times 10^3}{3 \times 10^8} = 0.003\,21\,\text{s}$$

Adding up the above times, we get the time taken for a single satellite to process an image on-orbit to be 2.22 seconds.

As seen in Figure 13, the reduction in time taken plateaus to 1.39 seconds around 35 edge nodes. Thus, there is no point in using more than 35 satellites in this process, as having more communication links than necessary could drastically reduce the power and resources of the satellites involved.

4. Discussion

4.1. The Optimised Satellite Constellation

The idea of using NSGA-II for orbit optimisation was inspired by a study on the construction of an LEO-MEO constellation by M. Asvial [10]. However, all orbits considered in this study are circular, and only one parameter-the cost of construction- is optimised. The study conducted by Tania Savitri on-orbit optimisation is more similar to this project, as the parameter optimised is coverage, although global [2]. Both studies only consider constellations with satellites at the same altitude and inclination, while our optimisation occupies a much larger range of orbits, including varying altitudes, inclinations as well as eccentricities. Thus, in many ways, our project is an extension of previously conducted orbit optimisation studies.

The RapidEye satellite sensor was chosen to be used onboard the satellites, allowing us to obtain a pixel size of 5m and a swath width of 236.4 km. This pixel size is far smaller than that of Aqua, Terra, Sentinel, Himawari-8 or Landsat satellites, as they all have pixel sizes that are upwards of 250 m [27] [28]. Hence, a bushfire must be at least 250 m wide for their detection. With the 5 m resolution, our constellation is capable of seeing bushfires that are 50 times smaller.

The final satellite constellation obtained using NSGA-II has 100% coverage over the area of interest and 75.4% coverage over central Australia. A new image can be taken in the area of interest once every 6 minutes and in central Australia once every 10 minutes. Thus, bushfires within the area of interest can be seen at any time by the constellation without significant delay.

This satellite constellation contains a large number of satellites, as monitoring the area of interest entirely at all times under 5 m resolution requires many satellites. However, considering that just the 2019-2020 bushfire season has estimated damages of 100 billion AUD [23], this would still be a feasible method of damage prevention.

4.2. The Optimised Bushfire Detection Method

4.2.1. Onboard Image Processing Algorithm for Bushfire Detection

The image processing outcomes show that a simplified version of the Collection 6 algorithm can be successfully conducted on-board satellites to detect bushfires. This technique reduces the time and cost of bushfire detections, as we no longer have to downlink high-resolution images.

As the FIRMS data source used to calculate the data accuracy of the detection algorithm contains fires detected throughout the day, comparing this data to the algorithm outcomes does not reveal much about its accuracy. Hence, while it can be calculated that the 83.06% of the bushfires detected by our algorithm is correct, several confirmed fires are not close to the detection algorithm results (See Figures 10, 11 and 12). These may be due to fires happening at other times of the day. There are also red dots far away from the blue ones, indicating that the algorithm can make infrequent false detections. These inaccuracies could potentially be eliminated by reinforcing the thresholding used in the algorithm with techniques such as machine learning methods (e.g. random forests).

4.2.2. Edge Computing-based Image Processing

Satellite constellations and the concepts of edge computing have been combined for efficiency in recent years. Several studies have been conducted on the use of edge computing for dynamic task allocation in satellite networks. For instance, the paper by Yongming He involves the use of edge computing on a satellite constellation to make observations upon user requests [32]. Another example is the paper by Feng Wang that looks into the use of edge computing to handle networking requests from users [4]. The edge computing algorithms involved in both these studies are much different from ours as they have different end goals in mind. Furthermore, there are no current studies conducted where computations are taking place on the edge nodes themselves. This concept is unique to our project.

Figure 13 shows that increasing the number of satellites/edge nodes decreases the overall time taken for bushfire detections. However, this decrease asymptotes out at around 35 satellites, to a value of 1.39 seconds. Thus, having more than 35 edge nodes does not affect the efficiency of bushfire detection.

The reason for this distribution of processing was the fact that using a typical processor onboard a satellite, a single satellite would take more than 2.2 seconds to detect bushfires out of a single image obtained. As seen in Figure 13, the time taken for the edge network-based processing is much less compared to this. Thus, this combination of force solves the issue of time taken for onboard processing.

With the use of a GEO satellite as the inter-satellite link, there is a constant ability to downlink data to the groundstation without delay. If the GEO satellite were not available, the LEO satellites would have to store the detection data until a groundstation becomes visible to them.

The 1.39 s detection time is far less than the 1.5 h required for Aqua, Terra, Sentinel and Landsat satellites working in combination [27]. It is even less compared to the 10 min detection time of the Himawari-8 satellite, which can only see bushfires that are at least 2 km long [28]. However, having to establish multiple communication links may drain the power of the satellites far quickly compared to a single ground-LEO communication link. Hence, before implementation, calculations need to be conducted to see if this process is also advantageous from a power perspective.

5. Conclusions

This project focuses on the construction of a network of satellites for bushfire detection in Australia. The constellation orbits are obtained through optimising the coverage, the total number of satellites and revisit times of the satellites. NSGA-II is chosen for the optimisation process, as it is more efficient compared to other optimisation algorithms. The constellation developed has 100% coverage over coastal Australia and 75% coverage over central Australia. It has 3990 satellites and can obtain an image of the coastal areas once every 6 minutes and of central areas once every 10 minutes. The constellation is at an altitude of 963.9 km, an eccentricity 0.04 and an inclination 141.39°. It has 95 orbital planes, 42 satellites per orbital plane and a relative phase separation of 9. All satellites are to be equipped with a crossbeam multispectral sensor similar to the RapidEye satellites, allowing them to provide real-time images with 5 m resolution in the visible and infrared wavelengths. The network is structured so that there are at least 160 satellites over coastal Australia at any given time. The satellites are also to be equipped with an on-orbit bushfire detection algorithm and processors with 200 MHz or higher clock rates. Edge computing is used to develop a novel method of distributed image processing, conducted through a GEO satellite in a similar orbit to Optus D3. Once a satellite above Australia takes an image, it is partitioned and sent to the geostationary satellite, which sends the segments off to 34 other constellation satellites to be processed. Each satellite processes its data segment, and if a bushfire pixel is detected, sends the fire coordinates to the GEO satellite. All bushfire coordinates are then compiled in the GEO satellite and downlinked to Earth. The entire process takes 1.39 seconds, leading to bushfire detections that are faster than any currently available method.

Author Contributions: Conceptualization, M.W. and X.W.; methodology, M.W.; software, M.W.; validation, M.W; formal analysis, M.W.; resources, X.W.; data curation, M.W.; writing—original draft preparation, , M.W.; writing—review and editing, M.W.; visualization, , M.W; supervision, , X.W.; project administration, X.W.;. All authors have read and agreed to the published version of the

manuscript.", please turn to the CRediT taxonomy for the term explanation. Authorship must be limited to those who have contributed substantially to the work reported.

Funding: This research received no external funding

Acknowledgments: I would first like to thank my supervisor, Dr Xiaofeng Wu , whose help was invaluable in the entire process of developing this thesis. Your faith in my ability to complete this thesis, your gentle guidance and helpful advice made this endeavour possible.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MODIS	Moderate Resolution Imaging Spectroradiometer
RAAN	Right Ascension of the Ascending Node
LEO	Low Earth Orbit
GEO	Geosynchronous Equatorial Orbit
NSGA II	Nondominated Sorting Algorithm II
SA	Simulated Annealing
MA	Mean Anomaly

References

1. De Sanctis, Mauro and Rossi, Tommaso and Lucente, Marco and Ruggieri, M. and Bruccoleri, Christian and Mortari, D. and Izzo, Dario,"Flower Constellations for Telemedicine Services," *International Journal of Applied Earth Observation and Geoinformation*, Vol. 91, 589-598, (01 2008).
2. Tania Savitri,"Constellation Orbit Design Optimization with Combined Genetic Algorithm and Semianalytical Approach," *International Journal of Aerospace Engineering*, Vol. 2017, No. 10, 205–222, (02 2017). Hindawi.
3. Optus,"Optus D3," , Available online: https://www.optus.com.au/about/network/satellite/fleet/d3, (2020).
4. Feng Wang,"A Dynamic Resource Scheduling Scheme in Edge Computing Satellite Networks," *Mobile Networks and Applications*, Vol. 8, 1052-1061, (01 2020).
5. C George,"Radiation hardening," , Available online: https://www.accessscience.com/content/566850#, (3 2020).
6. BAE Systems,"BAE RAD750 Radiation-Hardened SBCs Control WorldView-1 Satellite," , (10 2007).
7. Yongming He and Yingwu Chen and Jimin Lu and Cheng Chen and Guohua Wu,"Scheduling multiple agile earth observation satellites with an edge computing framework and a constructive heuristic algorithm," *Journal of Systems Architecture*, Vol. 95, 55 - 66, (2019).
8. National Hurricane Center,"Latitude/Longitude Distance Calculator," , Available online:https://www.nhc.noaa.gov/gccalc.shtml , (2020).
9. Robert A. Schowengerdt,"Remote Sensing: Models and Methods for Image Processing," , (2007).
10. M. Asvial,"," *IEE Proceedings - Communications*, Vol. 34, No. 10, 205–222, (07 2004). IEE.
11. NASA Official: Ed Masuoka,"Fire Information For Resource Management System (FIRMS)," , (08 2020).
12. David S. Johnson,"Optimisation by simulated annealing: An experimental evaluation," *Operations Research Society of America*, Vol. 37, No. 6, 806-866, (12 1989). Hindawi.

13. Louis Giglio and Wilfrid Schroeder and Christopher O. Justice,“The collection 6 MODIS active fire detection algorithm and fire products," *Remote Sensing of Environment*, Vol. 178, 31 - 41, (2016).
14. NASA,“The Earth," , Available online: https://imagine.gsfc.nasa.gov/features/cosmic/earth_info.html, (2020).
15. Wijayatunga, W, An Optimised Satellite Constellation for Bushfire Detection, *Bachelor's thesis*, University of Sydney, (2020) .
16. Stefama Cornara,“Satellite Constallation Mission Analysis and Design," , (2001).
17. Walker, J.,“Some circular orbit patterns providing continuous whole Earth coverage," *Interplanetary Society Journal*, Vol. 24, No. 6, 369–384.,(02 1971). J. Brit..
18. Karthik Sindhya,“Non-dominated Sorting Genetic Algorithm (NSGA-II)," , (2018).
19. NASA,“Earthdata Search," , Available online: https://search.earthdata.nasa.gov/ , (2020).
20. Oregon State University,‘Satellites and Sensors: How they work‘‘," ,Available online: http://www.geo.oregonstate.edu, (2003).
21. Monique Ross,“The history of fire in Australia — and how it can help us face the bushfires of the future," , (2020).
22. Rachel Westrate,“How Bushfires Will Affect Australia's Security," , (2020).
23. Misha Ketchell,“With costs approaching $100 billion, the fires are Australia's costliest natural disaster," , (2020).
24. McArthur, A. G.,“Bushfires in AustraliaS," *Australian Government Publishing Service*, Vol. 87, No. 1, (1978).
25. Satellite Imaging Cooperation,“RapidEye Satellite Sensors," , Available online: https://www.satimagingcorp.com/satellite-sensors/other-satellite-sensors/rapideye/, (3 2020).
26. Bushfire and Natural Hazards CRC,“Finding fires faster," , (12 2018).
27. European Space Agency,“Terra Mission (EOS/AM-1)," , Available online: https://earth.esa.int/web/eoportal/satellite-missions/t/terra, (2000).
28. Xu, Guang and Zhong, Xu,“Real-time wildfire detection and tracking in Australia using geostationary satellite: Himawari-8," *Remote Sensing Letters*, Vol. 8, 1052-1061, (11 2017).
29. van Lint, J. H., *Shannon's Theorem*, Springer Berlin Heidelberg, Available online: https://doi.org/10.1007/978-3-642-58575-3_2, (1999)
30. Robert A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*, Elsevier, (2007)
31. Lukic Natasa , Bozin-Dakic Marija , Grahovac Jovana , Dodic Jelena, Jokic Aleksandar, "Multi-objective optimization of microfiltration of baker's yeast using genetic algorithm",*Acta Periodica Technologica*, Vol 48, 211-220, (2017).
32. Eric Hamilton,“What is Edge Computing: The Network Edge Explained," *Cloudwards*, (10 2018).
33. Rutgers University,“Keplerian Elements," , Available online: https://marine.rutgers.edu/cool/education/class/paul/orbits.html, (2014).
34. Planet,“RapidEye Imagery Product Specifications," , Available online: https://www.satimagingcorp.com/satellite-sensors/other-satellite-sensors/rapideye, (2016).