Acta Astronautica 198 (2022) 179–193

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Design and analysis of flexible multi-layer staged deployment for satellite mega-constellations under demand uncertainty

Joshua F. Anderson [a], Michel-Alexandre Cardin [a,*], Paul T. Grogan [b]

[a] *Dyson School of Design Engineering, Imperial College London, United Kingdom*
[b] *School of Systems and Enterprise, Stevens Institute of Technology, USA*

ARTICLE INFO

Index Terms:
Global broadband internet systems
Mega-constellations
Multi-layer satellite constellations
Optimisation
Real options
Staged deployment

ABSTRACT

Internet satellite constellations are expected to play an important role in accommodating the rising global demand for internet access. Such rise in demand, however, is highly uncertain. Staged deployment is an approach that provides flexibility to tackle demand uncertainty by enabling the real option to reconfigure a constellation if demand changes. Advancements in satellite technology have led to the emergence of multi-layered constellations. This opens the opportunity to enhance staged deployment by enabling an additional real option: adding a new layer to a constellation. This real option has no associated reconfiguration costs, and therefore has the potential to reduce the cost of staged systems deployment. This paper proposes a framework to design multi-layer staged deployment systems and analyse their effectiveness in modern mega-constellations under global demand uncertainty. The framework is applied to four case studies based on market projections. Results show that multi-layer staged deployment decreases the expected life-cycle cost (ELCC) by 42.8% compared to optimal traditional single-layer deployment. Multi-layer staged deployment is more cost effective than single-layer staged deployment in all practical cases, which decreases ELCC by 22.9% compared to traditional deployment. Several cost altering mechanisms in staged deployment are identified. The results and analysis provide improved economic performance and better resource utilization, thus contributing in the long term to improved sustainability and market resilience. An accompanying decision support system provides system engineers with valuable insights on how to reduce deployment costs using the proposed multi-layered staged strategy.

1. Introduction

Providing internet from space has the potential to enable anyone to access the internet from any location, at any time. Low Earth Orbit (LEO) satellite constellations are expected to play a key role in accommodating the rising demand for internet access, with global demand expected to rise to 15 million by 2030 [1].

Satellite constellations are traditionally designed to be deployed in one stage to meet an expected future demand. Future demand for such rapidly evolving market, however, is highly uncertain. This uncertainty poses a huge financial risk for the traditional approach: If demand is lower than expected, the reduced revenues will not sustain the deployment costs, which could lead to economic failure such as the Iridium and Globalstar constellations in the early 2000s [2]. If demand is higher than expected, then the opportunity for increased revenues is missed.

Flexibility is a paradigm for managing uncertainty and risk in early conceptual design phases. It provides a system with the ability to adapt and evolve to deal with uncertainty and risks in a cost-effective, value-enhancing manner. Embedding flexibility into a system provides two advantages: Firstly, flexible systems can be deployed in stages that respond to the current market conditions, which reduces uncertainty and hence risk of unused capacity – and therefore makes better use of limited financial and material resources. Secondly, the cost of deployment is spread throughout time, which discounts the cost in the present. When flexibility is used in a system design, two key elements need to be considered: Firstly, the design of a flexible strategy (i.e. *how* the system adapts to uncertain conditions). Secondly, the method for assessing the economic value that the flexible strategy provides for the system. Understanding the economic value of a flexible strategy allows designers to identify the best flexible strategies, and the maximum cost that a designer should be willing to pay to embed a flexible strategy into a system. An example method for valuing flexibility is Real Options Analysis, which quantifies the value of flexibility available on real irreversible investment projects [3].

Staged deployment is an approach that uses flexibility to mitigate the

* Corresponding author.
E-mail address: m.cardin@imperial.ac.uk (M.-A. Cardin).

https://doi.org/10.1016/j.actaastro.2022.05.022
Received 6 December 2021; Received in revised form 23 March 2022; Accepted 15 May 2022
Available online 30 May 2022

Nomenclature

a	Altitude (km)
$\boldsymbol{\alpha}$	Orbital Shell $= [\boldsymbol{\lambda} \quad \boldsymbol{\kappa}] = [D \quad P \quad f \quad I \quad a \quad e]$
B_G	Guard bandwidth (GHz)
B_{sat}	Satellite frequency bandwidth (GHz)
B_T	TDMA carrier bandwidth (GHz)
γ_f γ_p	Angle between footprint centres, angle between adjacent planes (°)
C_I C_P C_L C_{OM} C_R	Cost components (Initial development, production, launch, onboard & maintenance, reconfiguration)
cap_{act} cap_{tot}	Active capacity, total capacity (Subscribers)
D	Antenna Diameter (m)
d_f d_p	Distance between footprint centres, distance between adjacent planes (km)
dem_{start} dem_{exp}	Starting demand, expected future demand (Subscribers)
e	Minimum Elevation Angle (°)
E_b/N_0	Energy per bit to noise power spectral density ratio
$EIRP$	Effective isotropic radiated power (dB)
$ELCC$	Expected Lifecycle Cost ($B)
f	Downlink frequency (GHz)
F	MF-TDMA framing bits
G/T	Antenna gain-to-noise-temperature (dB/K)
I	Inter-satellite-link topology (None, Ring, Mesh)
J	Capacity Jump (%)
$\boldsymbol{\kappa}$	Formation $= [a \quad e]$
k	Boltzmann Constant
K	Cluster size
LCC	Lifecycle cost ($B)
L_m	Max Layers (Layers)
L^{tot}	Total loss (dB)
$\boldsymbol{\lambda}$	Satellite Design $= [D \quad P \quad f \quad I]$
$MLSD$	Multi-layer staged deployment
η	Nadir Angle (°)
N^s N^p N^{tot}	Number of satellites per plane, number of planes, total number of satellites
n_{bits}	Number of bits per time slot in MF-TDMA
ρ	Relative packing distance
P	Transmitter Power (W)
PV	Present Value ($B)
r	Discount rate
R_b	TDMA carrier data rate (b/s)
r_{foot} r_{earth}	Satellite footprint radius, Earth radius (km)
rec	Reconfiguration cost (% of C_P /reconfiguration)
S	Deployment stage
$SLSD$	Single-layer staged deployment
σ	Percentage volatility in demand (%/year)
t	Current time (years from initial deployment)
T	Simulation time (years)
δt	Simulation time step (years)
T_f	MF-TDMA time frame duration (seconds)
T_g	MF-TDMA guard time (seconds)
τ	Reconfiguration capacity
μ	Percentage growth rate in demand (%/year)
U_s	Global system utilization (%)
VoF	Value of Flexibility (% reduction in ELCC compared to traditional strategy)
$\boldsymbol{\xi}$	Deployment path
$\boldsymbol{\Xi}$	Set of all deployment paths (that satisfy the deployment path constraints)
$\boldsymbol{\pi}_T$ $\boldsymbol{\pi}_S$ $\boldsymbol{\pi}_M$	Strategies (Traditional, single-layer, multi-layer)
$\boldsymbol{\pi}_T^*$ $\boldsymbol{\pi}_S^*$ $\boldsymbol{\pi}_M^*$	Optimal strategies (Traditional, single-layer, multi-layer)
$\boldsymbol{\psi}$	Deployment Strategy Design Vector $= [\boldsymbol{\lambda} \quad J \quad L_m]$
z	Perpendicular distance of satellite footprint to centre of Earth (km)
Z	Number of cells in a satellite's footprint

financial risks of demand uncertainty by deploying a satellite constellation in stages. Whenever demand exceeds the current capacity of the constellation, the constellation evolves to the next stage, which has a higher capacity, based on some decision criteria. Staged deployment has one real option to evolve the constellation to the next stage: reconfiguration. During an evolution, new satellites are deployed, and existing satellites can reconfigure their orbits to accommodate these new satellites. The most significant cost to embed flexibility in staged deployment is the reconfiguration cost, materializing as additional chemical fuel propellant and fuel tank space to enable reconfigurations [4]. The standard method for valuing flexibility in the staged deployment literature is to compare the expected lifecycle cost (ELCC) of the optimum staged deployment strategy to the optimum traditional strategy. Staged deployment has been shown to reduce the ELCC of a constellation by up to 28.9% compared to traditional deployment [4–6].

The technology underpinning satellite constellations has advanced significantly since staged deployment of satellite constellations was first studied in early 2000. In particular, the CubeSat revolution has led to smaller satellites being deployed [7], and the cost to launch into space is falling in the advent of reusable rockets [8]. These advancements have resulted in the emerging design of 'mega-constellations' which contain far more satellites than previous generations of constellations. Companies are beginning to apply staged deployment to mega-constellations: SpaceX is deploying Starlink in two stages comprising thousands of satellites [9] and Amazon is deploying Kuiper in five stages [10].

Advancements in satellite technology have made it economically feasible to deploy satellites into multiple layers of orbital altitudes. This is seen in the most advanced mega-constellations filed by the FCC, *Starlink* and *Kuiper*, who have planned to deploy eight and three layers, respectively [9,10]. The ability to arrange satellites into multiple layers enables a new real option for staged deployment: adding a new orbital layer. When a new layer is deployed, no existing satellites need to reconfigure their orbits. Multi-layer staged deployment (MLSD) therefore provides an opportunity to minimise reconfiguration costs, and the cost of embedding flexibility in the design, as compared to traditional single layer systems, and as compared to single-layer staged deployment (SLSD). From this point onwards, 'Traditional' staged deployment involving only one layer, as first studied by de Weck et al. [4], will be referred to by the retronym *single-layer staged deployment* (SLSD) to distinguish it from multi-layer staged deployment (MLSD).

2. Literature review

The research on Flexibility in engineering design has emerged over recent decades from the field of Real Options Analysis (ROA) [11]. Like standard ROA, the field aims to develop new methods or adapt existing ones to quantify the benefits associated to flexibility in irreversible investment projects [12,13]. It builds upon techniques such as dynamic programming, decision analysis, simulation, and robust optimisation to identify stochastically optimal solutions under uncertainty, rank ordered using primarily economic and risk tolerance metrics. Unlike standard ROA, the research aims to develop new methods to support the engineering design process more systematically for flexibility, with the goal of improving expected economic performance, sustainability, and resilience in the face of uncertainty [14]. It builds upon and integrates knowledge from engineering design theory, creativity, optimisation, stochastic modelling, and systems engineering to support early conceptual activities for flexibility.

Most relevant to this paper is the work done to quantify and optimize the value of flexibility (VoF) in engineering systems design. One method is Decision Analysis (DA) [15], which relies on decision trees and dynamic programming to compare different design alternatives as a structured sequence of decisions and uncertainty realizations. The decisions emulate those that are made by system operators based on uncertainty realizations, depending on adaptation capabilities embedded in the design to deal with changing conditions [16]. DA, however, is limited by the curse of dimensionality, as decision trees can grow exponentially with increasing decisions and uncertainty nodes, making them more difficult to use and interpret in practice. A similar approach is based on binomial lattice analysis, a simplified formulation of the Black-Scholes formula used to quantify the value of financial options [17]. Such approach is also limited as it assumes path independence, which is valid in the context of financial options, but not for real engineering systems.

A more prevalent method to quantify the value of flexibility combines Monte Carlo simulations with decision rules. Decision rules are akin to triggering conditions that must be met for a system to adapt, evolve, or reconfigure in the light of uncertainty realizations. This method elegantly combines design and managerial considerations to model more directly the design and decision-making process in operations. A simple decision rule is *'If demand is higher than capacity by threshold Δ, increase capacity by amount Φ, else stay the same'*. This method is comparable to standard ROA in its ability to value flexibility, and it has been shown to handle multiple decision rules and uncertainty sources simultaneously [18]. The approach can be combined with stochastic programming or robust optimisation to identify the best decision variables to design and exercise flexibility in operations e.g., Δ and Φ in the example above, and more complex decision rules as well. Important benefits compared to traditional ROA is that decision rule-based ROA does not require any assumption on market supply and demand, the existence of a portfolio of replicating project cash flows, or path independence, to name a few. It is therefore more amenable to analyse and quantify the benefits of flexibility in complex engineered systems, such as here.

Chaize [19] first proposed a flexible approach to SLSD for satellite constellation design in 2003. A framework was created that included a method for designing SLSD strategies, and a method for valuing their flexibility using DA and ROA. Flexibility was valued by comparing the ELCC of the optimum SLSD strategy, $\boldsymbol{\pi}_S^*$, to the optimum traditional strategy, $\boldsymbol{\pi}_T^*$. de Weck et al. [4] extended this framework to a case study, revealing a reduction in ELCC of up to 20% in $\boldsymbol{\pi}_S^*$ compared to $\boldsymbol{\pi}_T^*$. de Weck et al. acknowledged the potential for *MLSD*, encouraging research in this area. Lee et al. [5] analysed the value of flexibility in 2-layer MLSD under *regional* demand uncertainty, finding an ELCC reduction for the optimum MLSD strategy, $\boldsymbol{\pi}_M^*$, compared to $\boldsymbol{\pi}_T^*$ of up to 28.9%. Bosomworth and Grogan [6] used Chaize's framework to analyse the value of two-layer MLSD for modern constellations. Results showed an ELCC reduction for $\boldsymbol{\pi}_M^*$ compared to $\boldsymbol{\pi}_T^*$ of up to 19.5%. Bosomworth and Grogan's results also indicated that $\boldsymbol{\pi}_M^*$ is more cost effective than $\boldsymbol{\pi}_S^*$. Both Lee et al. and Bosomworth and Grogan only considered the simplest case of MLSD, with two layers, and this is not representative of emerging constellation proposals to the FCC, which contain up to eight layers [9]. Therefore, there is a gap in the literature for analyzing the economic value that flexibility provides in n-layer MLSD for emerging mega-constellations. This will become increasingly important to understand as more multi-layer constellations are approved by the FCC. It is hypothesized that the value of flexibility for MLSD is greater than SLSD because the new layer real option does not have reconfiguration costs.

The primary objective of this paper is to create a staged deployment valuation framework for multi-layer mega-constellations under global demand uncertainty. By applying the framework to four case studies comparing traditional, SLSD and MLSD systems, the paper has three secondary objectives: Firstly, to uncover the value of flexibility in MLSD for modern mega-constellations under global demand uncertainty. Secondly, to understand the characteristics of optimal deployment strategies. And finally, to uncover the mechanisms that can alter the effectiveness of these strategies. These contributions aim to help constellation engineers incorporate MLSD strategies into their future designs.

3. Methodology: Flexibility valuation framework

This paper proposes a staged deployment valuation framework for multi-layer mega-constellations under global demand uncertainty. It includes novel methods for modelling and comparing three types of systems: 1) traditional, 2) SLSD, and 3) MSLD. The framework also enables analysing the value of flexibility in SLSD and MLSD, which is an important metric for quantifying the benefits and costs of the different strategies.

3.1. Terminology

Explaining the mechanics of the framework requires introducing the following terminology:

1. An *orbital shell* is a formation of satellites, in circular orbits, that share the same orbital altitude [20] (Fig. 1A). An example of an orbital shell would be the GPS constellation: A formation of 31 satellites, orbiting Earth at an altitude of 20,180 km [21]. A constellation can include multiple *layers* of orbital shells (Fig. 1B). For example, Starlink's phase-2 constellation proposal has 8 orbital shells, whose altitudes range from 570 km to 335 km [20,22].
2. In a satellite constellation staged deployment strategy, a *deployment stage* refers to the constellations current state.
3. A *deployment path* is a sequence of deployment stages (Fig. 1C and 1. D). In a staged deployment strategy, a constellation progresses through its planned deployment path. In SLSD, all stages consist of a single orbital shell (Fig. 1C). In MLSD, a stage can have multiple *layers* of orbital shells (Fig. 1D).

3.2. Framework overview

The framework is split into three parts (Fig. 2):

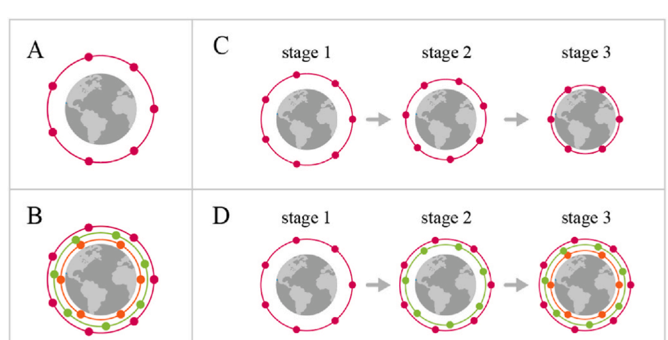

Fig. 1. Diagrammatic representations of A) a single orbital shell, B) three layers of orbital shells, C) a three-stage, single-layer deployment path, and D) a three-stage, multi-layer deployment path.

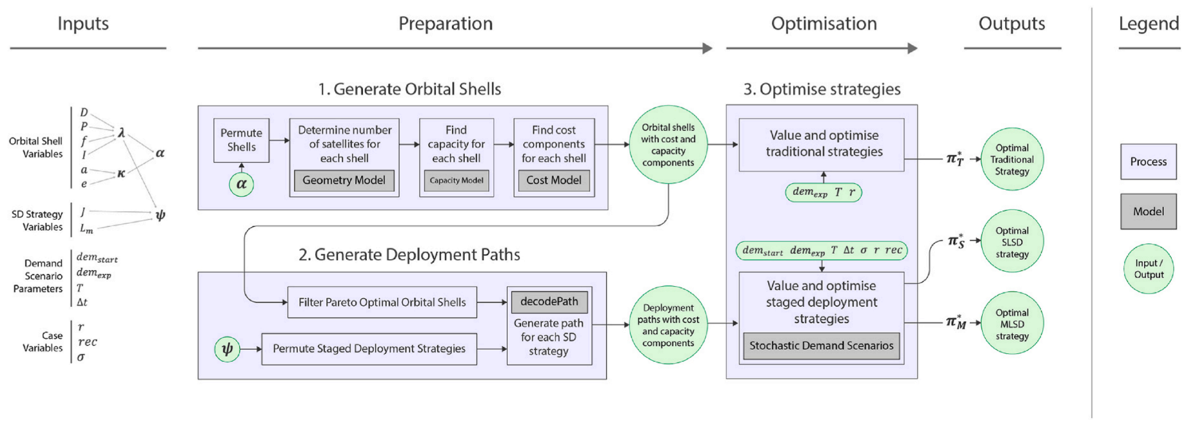

Fig. 2. High level overview of the proposed staged deployment valuation framework.

1. **Generating orbital shells.** Orbital shells are generated to serve as building blocks for traditional and staged deployment strategies. The cost components and capacity for each orbital shell is calculated, which is used for the valuation of traditional strategies and the generation of deployment paths.
2. **Generating deployment paths.** Each staged deployment strategy is modelled as a deployment path. The cost components and capacity for each stage in each path are calculated, which is used for the valuation of staged deployment strategies.
3. **Optimizing deployment strategies.** Traditional and staged deployment strategies are optimised through a full exploration of the problem space. The value of flexibility of SLSD and MLSD can then be determined by comparing the ELCC of the optimum SLSD and optimum MLSD strategies to the optimum traditional strategy.

3.2.1. *Generating orbital shells*

An orbital shell, $\boldsymbol{\alpha}$, is modelled using the design vector:

$$\boldsymbol{\alpha} = [D \quad P \quad f \quad I \quad a \quad e] \tag{1}$$

The design variables are the satellite antenna diameter, D, satellite transmitter power, P, downlink frequency, f, inter-satellite-link topology, I, orbital altitude, a, and minimum elevation angle, e. This design vector can be split into two sub-vectors:

1. Satellite design $\boldsymbol{\lambda} = [D \quad P \quad f \quad I]$. This vector represents the satellite design for all the satellites in the orbital shell.
2. Formation $\boldsymbol{\kappa} = [a \quad e]$. This vector represents the geometric formation of satellites in the orbital shell.

An orbital shell can therefore also be equivalently defined in terms of $\boldsymbol{\lambda}$ and $\boldsymbol{\kappa}$:

$$\boldsymbol{\alpha} = [\boldsymbol{\lambda} \quad \boldsymbol{\kappa}] \tag{2}$$

Orbital shells are assumed to have global coverage, which significantly simplifies modelling and analysis. To achieve global coverage, orbital shells are assumed to have a polar inclination of 97.6°. This inclination is chosen because it is used by the polar orbital shells in the Starlink constellation [22], which matches the papers scope of satellite internet mega-constellations and global demand for satellite internet.

A traditional deployment strategy, $\boldsymbol{\pi}_T$, is modelled as a single orbital shell (Equation (3)) (Fig. 1A). Iridium and Globalstar are examples that used this strategy: deploying a formation of satellites, in circular orbits, sharing same orbital altitude.

$$\boldsymbol{\pi}_T = \boldsymbol{\alpha} \tag{3}$$

A *tradespace* of orbital shells (the $\boldsymbol{\alpha}$ tradespace) is generated to serve as building blocks for deployment strategies. Each of the six individual components of the design variable, $\boldsymbol{\alpha}$, take a range of values as an input. The tradespace contains all permutations of values for the six design variables.

3.2.1.1. *Geometry model.*

The geometry model calculates the total number of satellites in an orbital shell, N^{tot}, based on its altitude, a, and minimum elevation angle, e. N^{tot} is then used in cost and capacity calculations.

Firstly, the radius of a satellite's footprint, r_{foot}, is calculated. r_{foot} is then used to derive the number of satellites per plane, N^s, and number of planes, N^p. This determines the total number of satellites, N^{tot}. More details on the geometry model can be found in Appendix C.

3.2.1.2. *Capacity model.*

The capacity model calculates the collective capacity of all satellites in an orbital shell based on its satellite design, λ, formation, $\boldsymbol{\kappa}$, and total number of satellites, N^{tot}. The capacity model first calculates the data rate, R_d, using a link budget equation (Equation (4)) adapted from the Space Mission Engineering Textbook [23] and Bosomworth and Grogan [6]. The Starlink satellite design is used as a benchmark to calibrate the link budget equation, with parameter values sourced from Portillo, Cameron, and Crawley [24] (Appendix D). Satellites are assumed to use multi-frequency time-division multiple access (MF-TDMA), as used in previous staged deployment frameworks [6,19], and is the case for Starlink. This enables the use of Chang and de Weck's capacity calculation (Equation (5)) [25] to derive the total capacity of the constellation, cap_{tot}, from the data rate. Other parameters in Equation (5) are sourced from Portillo, Cameron, and Crawley [24], and FCC filings [9].

$$R_d = EIRP - G/_T - L^{tot} - k - E_b/_{N_0} \tag{4}$$

$$cap_{tot} = n_{sats} \cdot \frac{Z}{2K} \frac{B_{sat}}{B_T + B_G} \frac{R_b T_f - F}{n_{bits} + R_b T_G} \tag{5}$$

The active capacity of the constellation, cap_{act}, is determined by the global percentage utilization of the orbital shell, U_S, which represents the percentage of satellites serving users at any given time. This considers satellites flying over uninhabited areas such as the ocean or mountains. With no inter-satellite links (ISLs), U_S is around 10% [26]. This increases to 70% for a ring ISL topology [6], and to 90% for a mesh ISL topology [6]. 'Constellation capacity' from this point onwards refers to cap_{act}.

$$cap_{act} = cap_{tot} \cdot U_S \tag{6}$$

3.2.1.3. *Cost model.*

The cost model calculates cost components for each orbital shell, which are used to calculate the ELCC for each deployment strategy. The cost components are the initial development cost, C_{ID}, production cost, C_P, launch cost, C_L, onboard and maintenance cost, C_{OM}, and reconfiguration cost, C_R. There are three ways in which a constellation can incur costs:

1. Initial Deployment: $Cost = C_{ID} + C_P + C_L + C_R$
2. Maintenance: $Cost = C_{OM}$
3. Evolving to the next stage: $Cost = C_P + C_L + C_R$

The Small Satellite Cost Model (SSCM) [27] is used to derive four of the cost components: C_{ID}, C_P, C_L, and C_{OM}. The SSCM uses the total dry mass, m_{tot}, of a satellite to calculate cost components. m_{tot} is estimated using the input parameters D, P, and I (Equation (7)). The Starlink satellite design is used as a benchmark for estimating dry mass because it resembles the satellite designs used in the papers case studies and is well documented. The benchmark constants used are the Starlink satellite's total dry mass, $m_d = 260kg$, transmitter power, $P_d = 2200W$, and antenna diameter, $D_d = 3.5m$. The total dry mass is assumed to scale linearly with power, and quadratically with antenna diameter. Finally, the presence of ISLs is assumed to scale the dry mass. This is captured through m_I, which has the value of 1, 1.05 or 1.2 for no ISLs, ring ISLs, and mesh ISLs, respectively.

$$m_{tot} = m_d \cdot \left(\frac{P}{P_d}\right) \cdot \left(\frac{D}{D_d}\right)^2 \cdot m_I \tag{7}$$

The SSCM uses m_{tot} to derive 26 separate costs. These costs are used to derive the following cost components for each orbital shell: C_{ID} is calculated by summing all non-recurring costs. C_P is calculated by summing all structural recurring costs. C_{OM} is equal to the Launch & Orbital Operation Support (LOOS) recurring cost. C_L is calculated by multiplying the cost of a Falcon 9 rocket launch by the number of required launches.

The SSCM is selected over the other publicly available cost model, the Unmanned Space Vehicle Cost Model (USCM) [28]. This is because unlike USCM, the SSCM is scoped for satellites with a dry mass <1000 kg, which is typical for satellites used in mega-constellations.

The reconfiguration cost, C_R, cannot be derived from the available cost models, therefore a new method for calculating C_R is proposed: C_R is assumed to be the cost for additional fuel and tank space required to enable reconfigurations. This cost is incurred when the satellite is produced, not during a reconfiguration itself. The cost of additional fuel and tank space is modelled as a percentage of the production cost, rec, multiplied by the reconfiguration capacity of a satellite, τ.

$$C_R = C_P * rec * \tau \tag{8}$$

The reconfiguration capacity of a satellite is determined by the staged deployment strategy, and is discussed in B.2.2.

Whenever a cost C is incurred, its present value PV is calculated based on the discount rate r at the time t (years) the cost is incurred, starting from $t = 0$:

$$PV(C) = \frac{C}{(1+r)^t} \tag{9}$$

Discounting is used to account for the time value of money. The value of r can only be estimated and is different between applications and industries.

3.2.2. Generating deployment paths

A staged deployment strategy is modelled as a deployment path, $\boldsymbol{\xi}$ (Fig. 1C and D): a constellation follows the deployment path from the first stage, S_1, to the Nth stage, S_N:

$$\boldsymbol{\xi} = [\boldsymbol{S}_1 \quad \cdots \quad \boldsymbol{S}_N] \tag{10}$$

A stage, S, is comprised of layers of orbital shells. The number of layers in a given stage, L, is between one and a specified maximum, L_m:

$$\boldsymbol{S} = [\boldsymbol{\alpha}_1 \quad \cdots \quad \boldsymbol{\alpha}_L] \text{ where } 1 \le L \le L_m \tag{11}$$

Staged deployment strategies include SLSD strategies and MLSD strategies. For an SLSD strategy, $\boldsymbol{\pi}_S$, all stages must contain only one orbital shell (Equation (12)). For an MLSD strategy, $\boldsymbol{\pi}_M$, a stage can contain multiple layers of orbital shells (Equation (13)). Examples of SLSD and MLSD strategies using this notation are shown in Fig. 3.

$$\boldsymbol{\pi}_S = \boldsymbol{\xi}, \text{ where } L_m = 1 \tag{12}$$

$$\boldsymbol{\pi}_M = \boldsymbol{\xi}, \text{ where } L_m > 1 \tag{13}$$

3.2.2.1. Deployment path constraints. The following constraints are applied to deployment paths to ensure that only practical deployment paths are generated:

$$cap(x+1) \ge cap(x) \cdot J \quad \text{where } J \ge 1 \tag{C1}$$

$$e_{layer\ i} \ge e_{initial} \forall i \tag{C2}$$

$$\lambda_{layer\ i} = \lambda_{initial} \ \forall i \tag{C3}$$

$$L \le L_m \tag{C4}$$

Constraint (C1) states that the next stage $(x+1)$ must have a higher constellation capacity than the current stage (x). This is because if the next stage has a smaller constellation capacity than the current stage, it implies that satellites will be prematurely de-commissioned – this is more expensive to do than simply leaving satellites in orbit, and is thus economically disadvantageous. This highlights that staged deployment is only flexible in one direction – as demand increases. The variable J is explained in section B.2.2. Constraint (C2) states that the minimum elevation angle, e, cannot decrease below the initial e. This rule guarantees that all receivers from the initial deployment will have a constant signal. Constraint (C3) specifies that all satellites in the constellation share the same satellite design throughout the constellation's lifetime. This constraint is used in prior literature to acknowledge that once deployed, the satellite design is difficult to change for physical or strategic reasons [4,6,17]. Secondly, this constraint enables considerable simplifications in the implementation of the framework. Constraint (C4) states that the number of layers in the current stage, L, cannot exceed the maximum layers, L_m, of this staged deployment strategy.

3.2.2.2. Reducing the problem space. Previous frameworks find the globally optimum staged deployment strategy by searching through the entire set of all possible deployment paths that satisfy the deployment constraints, Ξ. In the context of modelling multi-layer constellations, this approach yields an impractically large problem space (For example, there are $1.4 \cdot 10^{553}$ possible deployment paths in Ξ for this papers primary case study). This necessitates a new method for finding optimal deployment paths.

A more direct search algorithm for finding optimal deployment paths

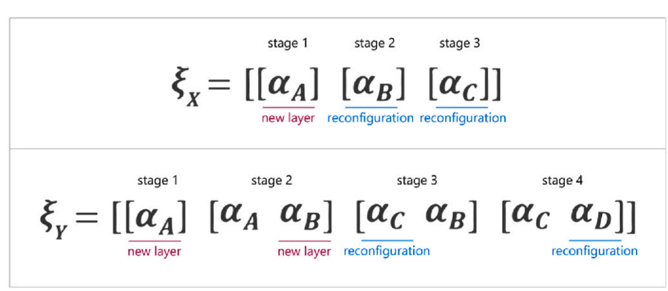

Fig. 3. Examples of deployment paths in their mathematical representation, including their orbital shells, stages, and evolution types (new layer or reconfiguration). Top: An SLSD deployment path ($L_m = 1$). Bottom: An MLSD deployment path, where.$L_m = 2$

has not been implemented in prior literature that uses deployment paths because there is no clear way to compare similarity between deployment paths in Ξ (Fig. 4, right): This makes it difficult to infer a route to an optimal solution, for example using a genetic algorithm.

The proposed search method maintains the full search of a set of deployment paths, however the set it searches is significantly reduced compared to Ξ as used in prior literature. This is achieved through two procedures:

1. For each family of orbital shells sharing the same satellite design,[1] all non-Pareto optimal orbital shells in that family are removed (A Pareto optimal orbital shell maximises capacity and minimizes ELCC). This stems from the observation that the optimal deployment path for a given family is always composed of Pareto optimal orbital shells in that family. This means that non-Pareto optimal orbital shells will never contribute to an optimal strategy, and therefore can be removed.
2. Staged deployment strategies are represented by a new staged deployment design vector, $\boldsymbol{\psi}$. This vector is decoded into a close-to-optimal deployment path, $\xi^{\sim}$. The design variables in $\boldsymbol{\psi}$ are chosen such that:
   a. The $\boldsymbol{\psi}$ tradespace contains a practical number of staged deployment strategies to be explored (45,000 compared to $1.4 \cdot 10^{553}$ in Ξ for the papers primary case study).
   b. A wide range of deployment strategies is explored.
   c. A staged deployment strategy can be summarised in a simpler, more concrete way than as a deployment path.

$\boldsymbol{\psi}$ is composed of three design variables: $\boldsymbol{\psi} = [\lambda \quad J \quad L_m]$.

1. Satellite design, λ, is a fundamental property of a deployment path, because of the constraint (C3) that all satellite designs must be the same for a given deployment path.
2. Capacity jump, J, specifies the minimum jump in capacity between stages (As shown in rule C1). Capacity jump is a chosen variable because it presents an important trade-off involving flexibility and reconfiguration costs: When J is low, there are many stages in ξ. This enables high flexibility, but high reconfiguration costs. Whereas when J is high, flexibility[1] is low, but so are the reconfiguration costs.
3. Maximum layers, L_m, specifies the maximum number of layers allowed in any given stage in ξ. L_m is a chosen variable because it can be used to assess the effectiveness of increasing layers on ELCC, and it differentiates between $\boldsymbol{\pi}_S$ (where $L_m = 1$) and $\boldsymbol{\pi}_M$ (where $L_m > 1$).

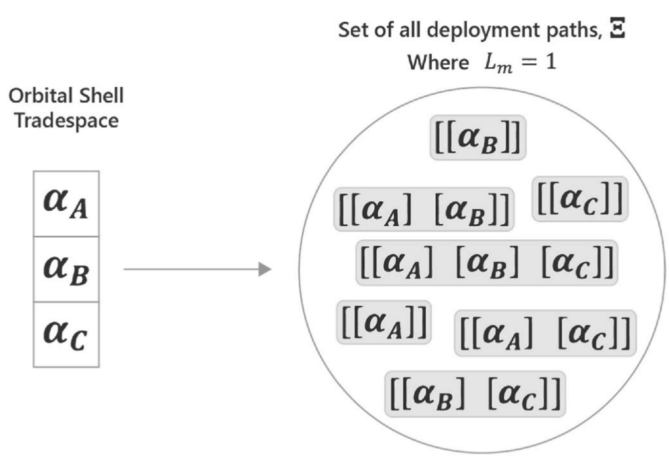

Fig. 4. An example of the creation of the set of all deployment paths (that satisfy the deployment path constraints), Ξ, from an orbital shell tradespace containing three orbital shells. The deployment paths in Ξ are limited to a single layer ($L_m = 1$). The number of deployment paths in Ξ increases hyper-exponentially with 1) the number of orbital shells and 2) the maximum layers, L_m.

$\boldsymbol{\psi}$ is converted into $\xi^{\sim}$ using a fixed function *decodePath*, which has the following high-level steps:

1. For the first stage, select the Pareto optimal orbital shell with the lowest constellation capacity that is greater than or equal to $dem_{start} \cdot J$. This step replicates a characteristic of optimal paths observed from Chaize's results [19].
2. A new layer is added in each subsequent stage, until L_m layers of orbital shells have been deployed. When adding a new layer, select the Pareto optimal orbital shell with the lowest capacity that satisfies the deployment path constraints.
3. The remaining stages are deployed as reconfigurations. For each subsequent stage, select the layer and Pareto optimal orbital shell to reconfigure to which results in the smallest jump in capacity that satisfies the deployment path constraints.
4. Finish the path once the constellation capacity is greater than dem_{exp}.

It is not guaranteed that any of the paths encoded in the $\boldsymbol{\psi}$ tradespace are globally optimal because not every path containing only Pareto optimal shells is explored. However, a suite test cases for small Ξ show that the ELCC of the optimal single-layer strategy found using *decodePath* is on average 0.41% higher than the ELCC doing a full search of Ξ. This demonstrates that *decodePath* is capable of producing close-to-optimal paths in a significantly reduced problem space.

3.2.3. Optimizing deployment strategies

The optimal strategies $\boldsymbol{\pi}_M^*$, $\boldsymbol{\pi}_S^*$ and $\boldsymbol{\pi}_T^*$ are strategies that minimise ELCC in their respective deployment approaches. A strategy's ELCC is calculated using an objective function that simulates the strategy. The optimisation is performed by running a full search on all possible strategies that can be created from the $\boldsymbol{\alpha}$ and $\boldsymbol{\psi}$ tradespaces.

$\boldsymbol{\pi}_T^*$ is found by minimising the objective function $ELCC_{\pi_T}(\boldsymbol{\pi}_T)$, where $\boldsymbol{\pi}_T$ is the variable to optimize. $\boldsymbol{\pi}_T^*$ is constrained by $cap(\boldsymbol{\pi}_T) > dem_{exp}$, meaning that a traditional strategy must be an orbital shell with a capacity greater than the expected future demand after a specified timeframe, dem_{final}. This constraint models the traditional approach of deploying an orbital shell upfront with a capacity that accommodates the expected future demand, and has been modelled previously [6,19]. In $ELCC_{\pi_T}(\boldsymbol{\pi}_T)$, the initial deployment cost is incurred at the start of the simulation. The onboard and maintenance cost is then incurred at every time step.

$\boldsymbol{\pi}_S^*$ and $\boldsymbol{\pi}_M^*$ are both found by minimising the objective function $ELCC_{\psi}(\boldsymbol{\psi})$. $\boldsymbol{\pi}_S^*$ is constrained by $L_m = 1$. $\boldsymbol{\pi}_M^*$ is constrained by $L_m > 1$. In $ELCC_{\psi}(\boldsymbol{\psi})$, a staged deployment strategy, $\boldsymbol{\psi}$, is decoded into its corresponding close to optimal deployment path, $\xi^{\sim}$. This path is then simulated through pre-computed demand scenarios. The initial development cost is incurred at the start of the simulation. Every time the current demand exceeds the constellation capacity of the current stage in $\xi^{\sim}$, the constellation evolves to the next stage in $\xi^{\sim}$. The corresponding evolution cost is then incurred. The onboard and maintenance cost for the current stage is incurred at every time step.

In both objective functions, increasing the expected final demand, dem_{exp}, will increase ELCC since larger, more expensive constellations will have to be deployed to accommodate the demand.

3.2.3.1. Demand model.
The demand model generates stochastic demand scenarios, which are used to value the ELCC of staged deployment strategies. Demand scenarios are modelled using geometric Brownian

[1] The requirement to segment into families stems from deployment path constraint C3.

motion (GBM) (Fig. 5), which simulates growth as a percentage drift, μ, with percentage volatility, σ:

$$dem_t = dem_{start} \cdot e^{\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right)} \quad (14)$$

where dem_t is the demand at time t, dem_{start} is the starting demand, and W_t is the Wiener process. μ is calculated based on dem_{start}, dem_{final}, and total simulation time, T:

$$\mu = \frac{ln\left(\frac{dem_{exp}}{dem_{start}}\right)}{T} \quad (15)$$

10,000 stochastic demand scenarios are generated to ensure that strategies are assessed against a representative sample of possible scenarios.

Note that traditional strategies are not simulated against stochastic demand scenarios. They are instead simulated based on the future expected demand, dem_{exp}. This is because their optimal strategy, and hence ELCC, only depends on dem_{exp}. The interpretation of this is that traditional strategies deploy to meet the expected demand, and don't adapt, regardless of the demand scenario, thus offering no flexibility.

3.3. Case studies

Four case studies are analysed to achieve the secondary objectives of this paper. The first case study analyses the characteristics of optimal MLSD strategies. The subsequent three case studies analyse the value of flexibility (VoF) in SLSD and MLSD, and the mechanisms which can alter it.

VoF represents the expected savings from using the proposed optimal staged deployment strategy over an optimal traditional strategy, used as benchmark. It gives system engineers an indication of how much they should be willing to pay to embed this flexibility into the system design, since this design will typically differ significantly from a traditional design approach. If the cost is less than the expected savings, a staged deployment approach should be favored over a traditional design.

All case studies share the same tradespaces. The orbital shell tradespace (Table 1) is based off Bosomworth and Grogan [6], who explored mega-constellation orbital shells. The 20 km interval of orbital altitude reflects the smallest orbital intervals currently approved by FCC (Kuiper Constellation [29]). There are a total of 65,880 orbital shells in the tradespace.

The staged deployment strategy tradespace (Table 2) is designed to provide a range of staged deployment strategies. Capacity jump ranges from 1.2, enabling very high flexibility, to 15, which results in a 2–3 stage deployment strategy. Maximum layers, L_m, ranges from 1 (covering SLSD), to 5 (the average number of layers in the largest modern mega-constellations currently approved by the FCC [9,29]).

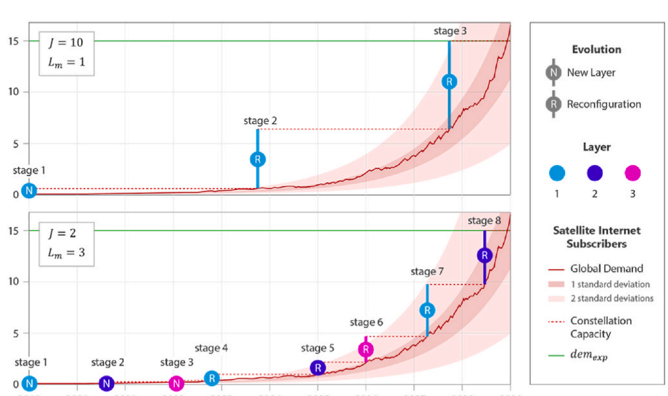

Fig. 5. Example of a demand scenario, with a 3-stage SLSD strategy (top) and an 8-stage, 3-layer MLSD strategy (bottom).

Table 1
Orbital shell, α, tradespace ranges.

Variable	Units	Bounds	Step	Discrete Values
Antenna Diameter, D	m	2–4	0.5	–
Transmitter Power, P	W	200–2200	400	–
Frequency Band, f	GHz	–	–	15, 30
ISL Geometry, I	–	–	–	None, Ring, Mesh
Altitude, a	km	400–1600	20	–
Min Elevation, e	degrees	10–60	10	–

Table 2
Staged deployment strategy, ψ, tradespace ranges.

Variable	Range	Step
Capacity Jump, J	1.2–15	0.2
Max Layers, L_m	1–5	1
Satellite Design, λ	(Based on Orbital Shell Tradespace Ranges)	

All fixed demand scenario parameters (Table 3) are shared between case studies (Table 4), and are chosen to reflect demand forecasts from 2019 to 2029 by Northern Sky Research (NSR) [1]. A fine time step of 1/48 years (≈ 2 weeks) results in detailed demand scenarios, which allows a wide range of possible scenarios to be explored.

Case studies differ in their values for three scenario parameters: the discount rate, r, reconfiguration cost, rec, and volatility, σ. These parameters are chosen because they are each expected to affect the VoF in a staged deployment strategy. A nominal value is chosen for each parameter to estimate realistic scenarios: $r = 10\%$ is a commonly chosen value for space based projects. $\sigma = 30\%$ produces demand scenarios that fluctuate within the upper and lower bounds of demand forecasts by NSR [1]. A nominal value for rec is harder to determine since rec is a novel concept in this framework, and therefore no nominal value of rec exists in prior literature. It is estimated by finding a value for rec that minimizes the discrepancy in results to prior literature, in particular, the value of flexibility, VoF, and reconfiguration capacity, τ. When benchmarked against case studies in prior literature, the value of rec which minimized differences in VoF and τ was 22% : This value leads to a VoF = 20.2% for SLSD, in line with de Weck [4] (where VoF = 20%). It also leads to $\tau = 3$ in 2-layer MLSD, in line with Bosomworth and Grogan [6] (where $\tau = 3$), and $\tau = 1$, approaching Lee et al. [5] (where $\tau = 0$).

The first case study simulates a single case with nominal parameter values. The subsequent three case studies each simulate 50 cases that explore a range of values for one of the parameters, keeping the other parameter values nominal.

A primary application of the framework is to analyse VoF for SLSD and MLSD as compared to a benchmark traditional system (Equations (16) and (17)). It quantifies the reduction in ELCC from using an optimal flexible strategy as compared to the optimal traditional benchmark strategy.

$$VoF_{SLSD} = \frac{ELCC_{\pi_T^*} - ELCC_{\pi_S^*}}{ELCC_{\pi_T^*}} \quad (16)$$

$$VoF_{MLSD} = \frac{ELCC_{\pi_T^*} - ELCC_{\pi_M^*}}{ELCC_{\pi_T^*}} \quad (17)$$

Table 3
Fixed demand scenario parameters.

Variable	Units	Value
Simulation Time, T	Years	10
Time step, δt	Years	1/48
Starting demand, dem_{start}	Subscribers (M)	0.05
Expected future demand, dem_{exp}	Subscribers (M)	15

Table 4
Case studies and their parameter values.

Case Study	Discount Rate, *r* (%)	Reconfiguration Cost, *rec* (%)	Volatility, σ (%)
1 – Nominal	10	22	30
2 – Variable Discount Rate	0–20	22	30
3 – Variable Reconfiguration Cost	10	0–50	30
4 – Variable Volatility	10	22	0–60

4. Results and discussion

The proposed framework and case studies in section III are used to analyse the value of flexibility, VoF, for SLSD and MLSD systems. This is because VoF can be affected significantly by prior modelling assumptions about discount rate, *r*, reconfiguration cost, *rec*, and volatility, σ. This analysis also helps uncover important design mechanisms to deal with uncertainty, as discussed below.

The results of the nominal case study (Table 5) show that the VoF for MLSD, 42.8%, is almost double the VoF for SLSD, 22.9%, indicating that cost savings could approach twice that of MLSD compared to SLSD, both in respect of a traditional design. The nominal case study also indicates characteristics of the optimal strategies for the traditional, SLSD, and MLSD approaches. The optimal strategies share the same values for three of the four satellite design variables: $D = 2m$, $f = 30GHz$, $I = Mesh$. *P* is low, ranging from $200 - 250W$. Strategies with these satellite designs have a set of orbital shells which are close to or on the Pareto front. These results indicate that optimum strategies favour small satellites with higher downlink frequencies, consistent with the findings in Bosomworth and Grogan [6]. MLSD has a smaller capacity jump than SLSD because the cost savings from reducing reconfigurations permits more evolutions, which is achieved by smaller jumps in capacity.

The results of case studies 2–4 (Fig. 6) show how the VoF for SLSD and MSLD changes with changing discount rate, reconfiguration cost, and volatility.

Case study 2 (Fig. 6A) shows that the VoF for both SLSD and MLSD increases with increasing discount rate, *r*. This is because future costs are more heavily discounted with increasing *r*, thereby providing additional benefits for more modular flexible design strategies that deploy capacity later. The rate of increase in VoF slows because the initial development cost remains undiscounted, which proportionally takes up more of the ELCC as *r* increases. The fluctuations in VoF for case studies 2–4 are a result of noise from the stochastic demand scenarios. This can be reduced by simulating more scenarios for each case.

Case study 3 (Fig. 6B) shows that the VoF for MLSD is highly resilient to changes in reconfiguration cost, *rec*, with VoF = 42.8%. The VoF for SLSD however decreases linearly with increasing *rec*, where VoF = 46.5% when *rec* = 0%, and VoF = −4.9% when *rec* = 50%. A negative VoF means that the value of embedding flexibility in the design is superseded by its cost – which is rare here, and as confirmed in many other studies [13]. The VoF for MLSD is resilient to reconfiguration costs after *rec* = 3.1% because in every subsequent case, π_M^* has sufficiently high values for L_m and *J* that there are no reconfigurations in the deployment path, which makes VoF independent of *rec*. MLSD is less cost effective than SLSD when *rec* is very low (*rec* < 1.8%). The mechanism explaining this is in sub-section A.

Case study 4 (Fig. 6C) shows that the VoF for both SLSD and MLSD increases with increasing volatility, σ. This is consistent with prior literature on real options and flexibility in design [3,12,14], where optionality is increasingly valuable when a system is facing increasing volatility. It reinforces the intuition that in a world fraught with irreducible uncertainty, flexibility can be extremely valuable.

Table 5
Optimal strategies in nominal case study.

Variable	Units	Approach		
		Traditional	SLSD	MLSD
Antenna Diameter, D	m	2	2	2
Transmitter Power, P	W	200	200	250
Frequency Band, f	GHz	30	30	30
ISL Geometry, I	–	Mesh	Mesh	Mesh
Altitude, a	km	520	–	–
Min Elevation, e	degrees	50	–	–
Capacity Jump, *J*	%	–	9.2	5.4
Maximum Layers, L_m	Layers	–	1	5
Expected Lifecycle Cost, ELCC	$M	1088	839	622
Value of Flexibility, VoF	%	–	22.9	42.8

4.1. Mechanisms that affect the value of flexibility in staged deployment

It is important for system engineers to understand the mechanisms that affect the VoF in flexible SLSD and MLSD systems to make the most informed design decisions. Six such mechanisms have been identified using this framework.

Firstly, low-capacity orbital shells (LCOs) are less cost effective than high-capacity orbital shells (HCOs). LCOs have high altitudes and low minimum elevation angles, which both increase the path distance between a satellite and a receiver. Increasing path distance increases signal attenuation from space (path loss), which lowers the capacity per satellite. This makes LCOs cost ineffective compared to HCOs. The traditional approach always deploys an HCO with a capacity greater than dem_{exp}, whereas the staged deployment approaches initially deploy LCOs. This makes the staged deployment approaches less cost effective than the traditional approach initially, which is further penalised by the discount rate.

Secondly, when *J* is small, launch costs are disproportionately high. Each evolution requires at least one rocket to launch new satellites. A small jump in capacity, however, results in a small payload relative to the payload capacity of the rocket. This leads to significant empty space during launches, which greatly increases the cost to launch per satellite for small values of *J*. This mechanism is becoming less significant as modern ride-sharing services such as the *SmallSat Rideshare Program* by SpaceX are becoming more widespread [30].

Thirdly, as *J* decreases, the reconfiguration cost increases. This is because decreasing *J* increases the number of evolutions, which increases the required reconfiguration capacity per satellite.

The following two mechanisms increase the VoF for MLSD by reducing the required reconfiguration capacity of satellites:

Increasing L_m decreases the number of reconfigurations in a deployment path. This is because a deployment path has L_m new layer evolutions, and the remaining evolutions (if any) are reconfigurations. Decreasing reconfigurations in the deployment path reduces the required reconfiguration capacity per satellite.

Increasing L_m decreases the number of reconfigurations per layer. This is because during a reconfiguration evolution, only one layer is reconfigured. Therefore, the more layers there are, the fewer reconfigurations are required per layer. This decreases the required reconfiguration capacity per satellite, τ.

The final mechanism decreases the VoF for MLSD relative to SLSD. For early stages in MLSD, capacity is increased by deploying an LCO as a new layer, while for early stages in SLSD, capacity is increased by reconfiguring into a higher capacity orbital shell. This initially makes VoF_{MLSD} lower than VoF_{SLSD}. If *rec* = 0%, this mechanism leads to VoF_{MLSD} being lower than VoF_{SLSD} overall.

4.2. Study limitations

The framework is limited to modelling orbital shells with a *Streets-of-Coverage* (SoC) design. The most common design used in mega-

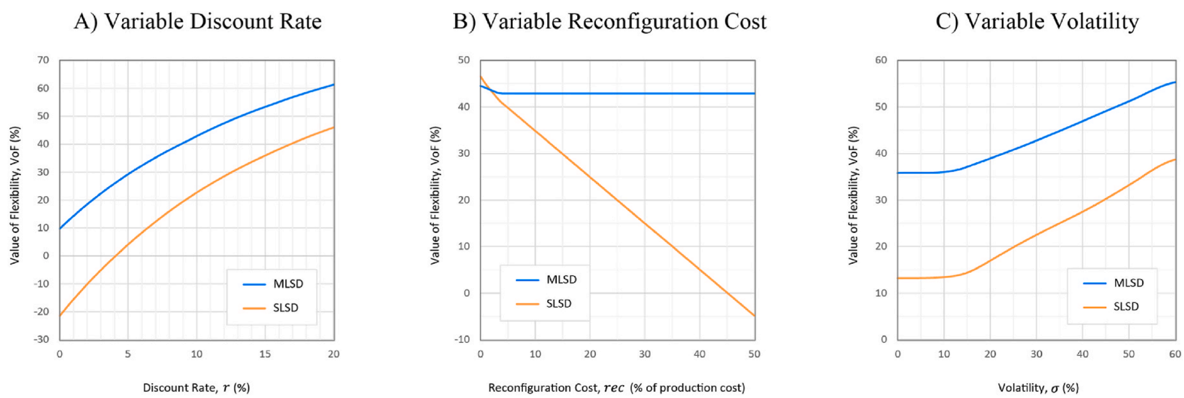

Fig. 6. Value of Flexibility in MLSD and SLSD for under variable A) reconfiguration cost, B) discount rate, and C) volatility.

constellations is the *Walker* design, followed by the SoC design. Walker designs are not modelled because they often have non-polar orbital inclinations (unlike SoC) - This is problematic for two reasons: Firstly, the geometry model becomes increasingly inaccurate at deriving the total number of satellites in an orbital shell with low inclinations. Secondly, the inclination of a Walker design can make a huge difference to how many people are able to access its services. This is not factored into the current framework. Factoring in the percentage of global demand that can access the Walker designs shell based on its inclination adds a significant layer of complexity that is outside the scope of this paper, - but would nonetheless be an interesting avenue for future research.

The case study has some limitations that also offer avenues for future research. Firstly, only costs are considered, not revenues from subscribers. This is sufficient to show the relative cost effectiveness of the three deployment approaches, but insufficient to verify their economic feasibility.

Secondly, the reconfiguration cost is assumed to scale linearly with the number of required reconfigurations to minimise complexity. A linear relationship may not be the case: Parametric cost models such as the SSCM and USCM use cost estimating relationships (CERs) to generate cost estimates [31]. CERs are calculated by analysing historic data, which often reveal logarithmic, quadratic, and exponential relationships.

Thirdly, the reconfiguration cost is assumed to be incurred whenever a new satellite is produced. However, designing a satellite to enable reconfigurations will also incur cost during the initial development, which is not considered in the framework to minimise complexity. This decreases the reliability of ELCC comparisons between π_M^* and π_S^*.

4.3. Significance of outcomes

Results of the nominal case study show that flexible single-layer and multi-layer staged deployment are cost effective compared to traditional deployment, reducing ELCC by an average of 22.9% and 42.8%, respectively. This is in line with previous results in the literature [4–6].

The cost effectiveness of multi-layer compared to single-layer staged deployment is less clear because the reconfiguration cost is not known. However, multi-layer staged deployment is likely to increase in cost effectiveness in the near future for three reasons: firstly, launch costs are continuing to decrease [32]. Secondly, smaller rockets are emerging on the market, enabling small jumps in capacity to be more cost effective. Thirdly, reconfiguration costs are unlikely to decrease in the near future because recent advancements in propulsion technology such as solar sails are unsuitable for the application of orbital reconfiguration.

Flexible multi-layer staged deployment therefore holds potential to significantly reduce ELCC in modern constellations. This has both economic and social implications. If this is achieved, the market for space-based internet systems is likely to grow. Such approach would make better use of limited financial and material resources (i.e., deploy additional capacity only if and when it is needed), thus contributing in the long term to better financial and environmental sustainability, reduced pollution from launches, and space junk. As shown in Fig. 6, it would contribute better resilience in the face of global demand uncertainty. Internet access could become more widely available for users in remote areas with limited Internet access, thus contributing to wider access to knowledge, and education. Satellite internet could also become more competitive with terrestrial networks, making internet access more affordable for all users. In a world where connectivity is increasingly important for society and the economy, multi-layer staged deployment could play an important role in enabling such new global connectivity.

5. Conclusion

5.1. Overview and contributions

Multi-layer staged deployment (MLSD) is a promising approach to deploy satellite constellations that has emerged from recent advances in satellite technology.

This paper's primary contribution is the proposal of a staged deployment valuation framework for flexible multi-layer mega-constellations under global demand uncertainty. The framework has a novel approach for modelling staged deployment strategies to reduce the large problem space whilst exploring a variety of strategies. The framework's approach for valuing flexibility in staged deployment strategies builds on previous methods by incorporating a model for reconfiguration cost. A decision support system is designed and implemented to rapidly prototype, test and optimize staged deployment strategies[2] The framework and decision support system are applied to four case studies based on market projections. The results and discussion of these studies present the following contributions which reflect the papers secondary objectives:

1. Flexible MLSD is shown to decrease the expected lifecycle cost compared to traditional deployment by 42.8%. MLSD is shown to be more cost effective than single-layer staged deployment (SLSD), even with low reconfiguration costs, confirming the papers hypothesis. A sensitivity analysis reveals that MLSD has improved resilience to uncertain parameters compared to SLSD.
2. Optimal MLSD strategies have small, low power satellite designs, which are deployed into many layers.
3. Six cost altering mechanisms are identified, which explain how and why the expected lifecycle cost of flexible strategies can change.

Framework limitations and significance of outcomes are discussed.

[2] Details on the decision support system can be found in the appendix.

The framework, decision support system, and cost altering mechanisms introduced provide tools for constellation designers to reduce constellation costs by exploring multi-layered staged deployment. In doing so, it is hoped that the market for satellite internet will continue to grow, providing internet access for anyone, anywhere, at any time. It will also contribute to better financial and environmental sustainability in the long term, reduced pollution, and space junk.

5.2. Future work

The framework could be further developed by modelling Walker constellation designs. Doing so would allow the framework to more accurately model modern mega-constellations and analyse their effectiveness. Modelling satellite lifetime into the framework could improve its fidelity: some low Earth orbit constellations such as Lynk Smallsat System [33] only have a lifetime of 2–3 years. In this case, there is an opportunity for future work to explore a replenishment strategy over a reconfiguration strategy. Other potential strategies could be explored: a 'combine' evolution where two layers reconfigure into one could improve cost effectiveness without requiring any launch costs. A detailed demand threshold could be investigated, where evolutions are only triggered when demand reaches a percentage of capacity *and* for a set duration. This could avoid triggering evolutions for temporary peaks in demand. Triggering evolutions based off an exponential trendline of demand could help model the trajectory of demand more accurately. A future study could explore how revenues affect the economic feasibility of staged deployment: Modelling a starting budget with revenues could dismiss strategies that run out of budget during a deployment mission.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We are thankful to Jordan Stern and Josue Tapia Tamayo for providing useful resources and directions on cost and capacity modelling in the early stages of the study.

Appendix

A. Decision Support System

A Decision Support System (DSS) was built to enable rapid testing for the proposed framework. All the results in this paper were calculated using the DSS. It is written in JavaScript, and consists of 5000+ lines of code, across 300+ functions. It allows constellation designers to rapidly model and assess the value of single-layer and multi-layer staged deployment strategies for mega-constellations (Fig. 7). The DSS contains the following features:

1. Constellation visualiser: A detailed visualisation suite enables system designers to see a visual depiction of constellation strategies. Strategies can be played and paused in time for randomly generated demand scenarios. Approaches can be directly compared for the same demand scenario to uncover design insights. A strategy's deployment path is visualised and graphed against the current demand scenario to see how capacity increases through time.
2. Tradespace visualiser: The orbital shell tradespace can be visualised to see the relationships between orbital shells in terms of cost and capacity.
3. Simulation configuration: All scenario parameters and tradespace ranges can be adjusted to run specific tests.
4. Results summary: Optimal results are displayed for each case analysed in the simulation.
5. Heatmap generator: Heatmaps can be heavily customised through a wide range of input and output options. These can reveal hidden relationships between two variables.
6. Save and load results: Results are compressed using a custom compression algorithm to minimise file size.
7.Loading screen: During simulation, a visual loading screen shows progress, current case and strategy are displayed, and the estimated time remaining is calculated

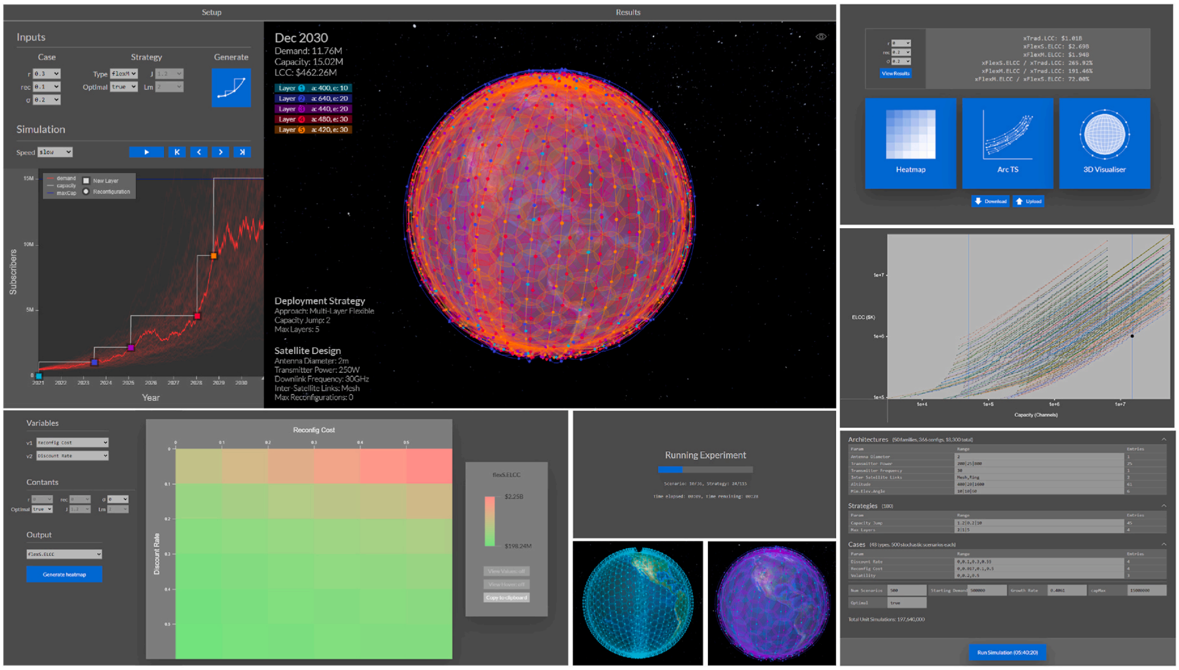

Fig. 7. A collection of screenshots from the developed Decision Support System. Spiralling clockwise from top left corner: visualisation suite, results panel, orbital shell tradespace explorer, configuration panel, constellation visual comparisons, heatmap generator, and loading screen

B. Performance Optimisation in DSS

An important consideration when designing the DSS was the optimisation performance. Increasing performance allows more tests, results, and insights. The following performance enhancing strategies improved the speed of optimisation by a factor of over 3000 compared to the original design. Two key performance enhancers were implemented:

Firstly, discount values were pre-calculated. Every time a cost is incurred after $t = 0$, it needs to be discounted using Equation (9). Discounting requires exponentiation, a relatively costly operation computationally when compared to the other mathematical operators. Discounting is the most repeated calculation in the optimisation, for the papers case study, it is executed over 5 billion times. To stop the same discount values being recalculated every demand scenario, the discount value at every time step was pre-calculated at the beginning of the optimisation and stored in a lookup table. Discounting now only requires multiplying a cost by the discount value at the corresponding time step. Pre-calculating discount values improved performance by a factor of 100.

Secondly, *memoization* was used for calculating C_{OM} costs. Calculating C_{OM} is amongst the most repeated calculations in the optimisation. To calculate C_{OM}, the recurring *LOOS* cost needs to be exponentiated using a learning factor L, based on the current number of satellites in the constellation. The number of satellites only changes when an evolution occurs, which is in the order of 1% of steps. Therefore, many calculations are repeated. This situation is a good candidate for memoization. Memoization is an optimisation technique where results for an expensive function are stored in a cache. If a given input has been calculated before, the corresponding output is taken from the cache, otherwise the output is calculated and added to the cache. Using memoization improved performance by a factor of 20.

Other smaller performance enhancers were implemented, including calculating deployment paths per case instead of per demand scenario, bulk calculating C_{OM} costs, and only running one demand scenario when $\sigma = 0$.

C. Geometry Model Details

The geometry model calculates the orbital arrangement of satellites based on an orbital shell's formation $\boldsymbol{\kappa} = [a \quad e]$. The formation is defined in terms of the number of orbital planes, N^p, and satellites per plane, N^s. These determine the number of satellites in the constellation, which is used by the cost and capacity models. The geometry model first calculates the dimensions of a single satellite's footprint. This is then used to determine the smallest arrangement of satellites that achieves global coverage.

Calculating satellite footprint, $[z \quad r_{foot}]$, from minimum elevation angle e and altitude a.

A satellite's footprint is the area of coverage provided by the satellite. It can be defined geometrically as a circle, with a distance from the centre of the Earth z, and footprint radius, r_{foot}. These variables can be derived from the minimum elevation angle, e, altitude a, nadir angle, η, and earth central angle, γ. The relationship between these variables is shown in Fig. 8.

The sine rule can be used to derive η:

$$\frac{\sin(\eta)}{r_{earth}} = \frac{\sin(e + \pi/2)}{r_{earth} + a} \tag{18}$$

$$\eta = \sin^{-1}\left(\frac{r_{earth}}{r_{earth} + a}\cos(e)\right) \tag{19}$$

η and e can be used to derive γ:

$$\gamma = \pi/2 - e - \eta \quad (20)$$

γ can then be used to derive r_{foot} and z:

$$r_{foot} = r_{earth} \cdot \sin(\gamma) \quad (21)$$

$$z = r_{earth} \cdot \cos(\gamma) \quad (22)$$

Calculating orbital arrangement, $[N^s \quad N^p]$, from $[z \quad r_{foot}]$:

There are two widely used constellation designs: Streets-of-Coverage, and Walker [34]. Both are defined with the notation $i: \ t/ \ p/ \ f$, where i is inclination, t is the total number of satellites, p is the number of equally spaced planes, and f is the relative spacing between satellites in adjacent planes. Satellites have a circular orbit, and are evenly spaced in a plane. Planes are evenly spaced across a span around the equator. In a Streets-of-Coverage constellation, this span is approximately 180°. In a Walker constellation, this span is 360°. Streets-of-Coverage constellations are typically only configured with near-polar inclinations, while Walker constellations are capable of having both polar and non-polar inclinations.

An orbital arrangement is defined in the geometry model by the number of orbital planes, N^p, and the number of satellites per plane, N^s. To calculate N^p and N^s, constellations are assumed to have a *Streets-of-Coverage* design.

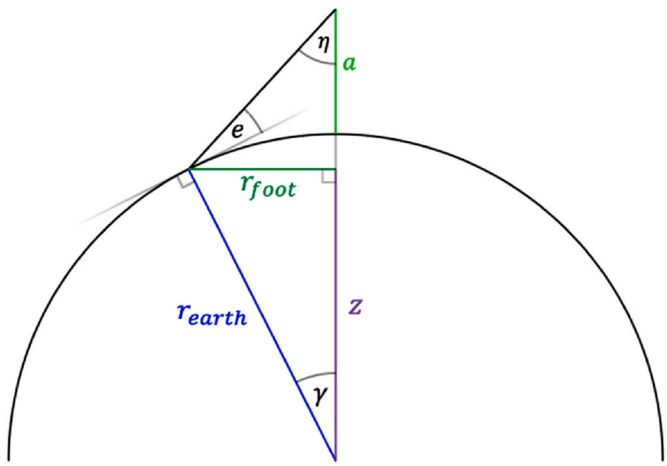

Fig. 8. Relationship between z, r_{foot}, r_{earth}, e, a, η, and γ

Constellations generated with the geometry model have the form $97.6° : \ SP/P/2$. Relative spacing is fixed to simplify modelling and analysis. It is fixed to 2 because $f = 2$ is commonly used for Streets-of-Coverage constellations, and it enables the following derivation.

N^p and N^s are calculated by assuming the unit arrangement of adjacent satellites is an equilateral triangle ($f = 2$) (Fig. 9). This is the most efficient packing of circular satellite footprints to achieve global coverage. In this arrangement, if the distance between adjacent footprint centres, d_f, is 2 units, then the distance between adjacent planes, d_p, is $\sqrt{3}$ units (Fig. 9A). That is to say, the ratio of $d_f : d_p$ is $2 : \sqrt{3}$.

One might think to pack the footprints edge to edge (Fig. 9A), however, this would lead to spots of no coverage, which would result in temporary but frequent connection drops. Therefore, the distance between footprint centres needs to be decreased until the spots with no coverage are eliminated. To regulate the relative distance between footprint centres, a variable is proposed: *relative packing distance*, ρ. The distance between adjacent footprint centres, d_f, and the distance between adjacent planes, d_p, is calculated as follows:

$$d_f = 2 \cdot r_{foot} \cdot \rho \quad (23)$$

$$d_p = \sqrt{3} \cdot r_{foot} \cdot \rho \quad (24)$$

When $\rho = 1$, footprints meet edge to edge (Fig. 9A). When $\rho < 1$, footprints overlap. $\rho = \sqrt{3}/2 \approx 0.866$ is the largest value of ρ that ensures there is global coverage (Fig. 9B). Any value of ρ smaller than this would result in more overlapping than necessary to achieve global coverage.

N^s is derived from angular distance between adjacent footprint centres, γ_f (Fig. 10), which is calculated as follows:

$$\gamma_f = 2\sin^{-1}\left(\frac{\frac{1}{2}d_f}{r_{earth}}\right) \quad (25)$$

$2\pi/\gamma_f$ will return the minimum number of satellites per plane to achieve global coverage. However, this number is non-integer, and an integer number of satellites is required. Therefore, this number is rounded up to ensure global coverage:

$$N^s = \frac{2\pi}{\gamma_f} \quad (26)$$

Finding the number of planes, N^p, follows the same process as finding sat_p, except it is based of d_p rather than d_f:

$$\gamma_p = 2\sin^{-1}\left(\frac{\frac{1}{2}d_p}{r_{earth}}\right) \tag{27}$$

$$N^p = \frac{2\pi}{\gamma_p} \tag{28}$$

D. Link Budget Equation for Capacity Model

Table 6 shows the link budget used in the framework. The data in the table is sourced from Portillo, Cameron, and Crawley [24].

Table 6
Beam Link Budget for Starlink Downlink.

Parameter	Value	Units
Tx Diameter	3.5	m
Tx Frequency	13.5	GHz
Tx Efficiency	60	%
Tx Gain	18.6	dBi
Tx Backoff and Line Loss	5	dB
EIRP	36.7	dBW
Path Distance	1684	Km
Space Loss	179.6	dB
Atmospheric Loss	0.53	dB
Rx Diameter	0.7	m
Rx Efficiency	55	%
Rx Gain	46.4	dBi
Rx Line Loss	2	dB
System Noise Temperature	27.3	dB-K
Rx Antenna Gain-to-noise Temperature	19.3	dB-K
Required Nb/N0	12.3	dB
Data Rate	9416	Mbps

E. Orbital Shell Tradespace Visualised

An impression of the number of orbital shells in the orbital shell tradespace from Table 1 can be observed in Fig. 11. The Pareto front belongs almost entirely to the same satellite design. This design is used by the optimal traditional and SLSD strategies (Table 5).

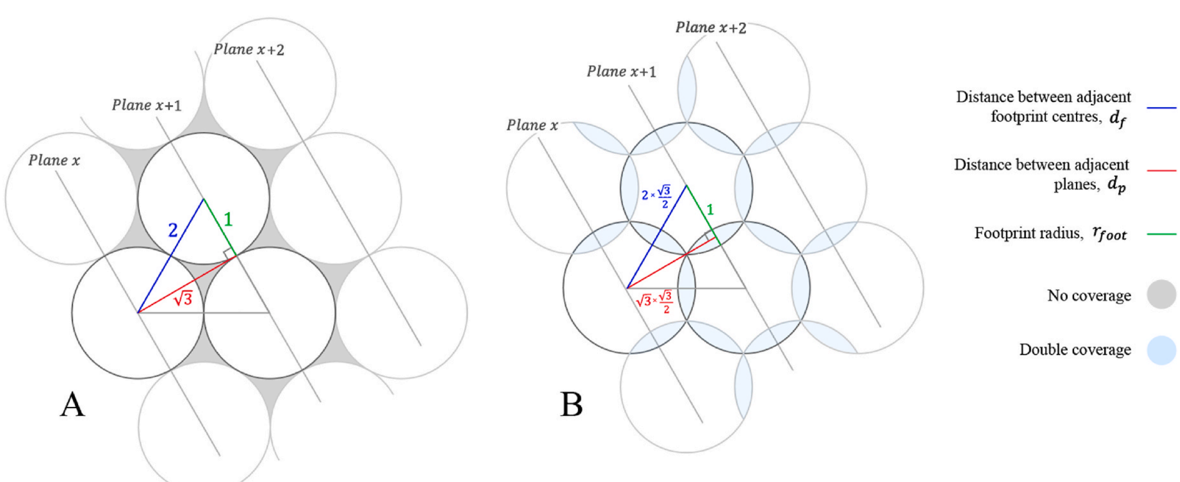

Fig. 9. Satellite footprint packing patterns and corresponding distances for footprints with radius, $r_{foot} = 1$, subject to relative packing distance, ρ. A: $\rho = 1$. B: $\rho = \sqrt{3}/2 \approx 0.866$.

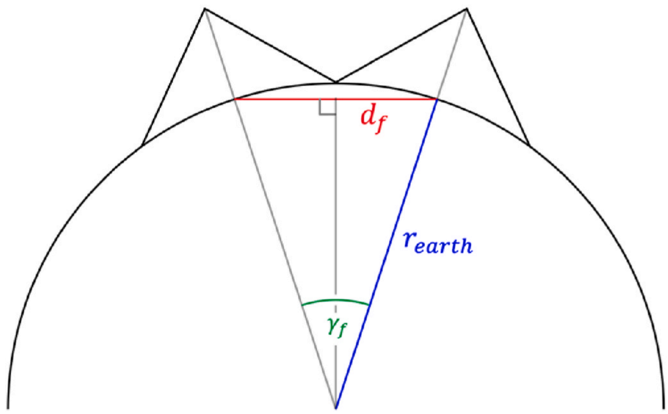

Fig. 10. Relationship between γ_f (angular distance between adjacent footprint centres), d_f (cartesian distance between adjacent footprint centres), and r_{earth} (radius of Earth).

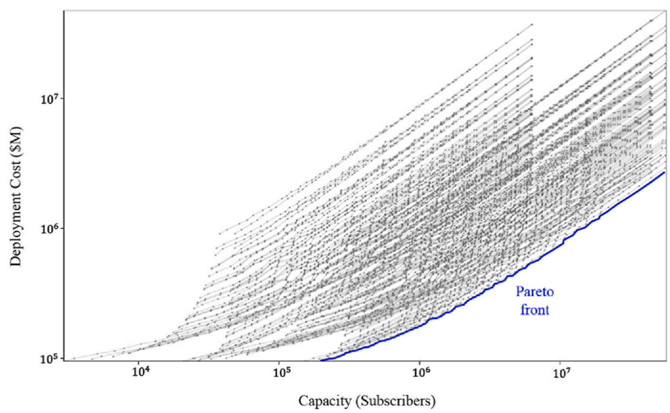

Fig. 11. Orbital Shell Tradespace from Table 2, represented in terms of capacity and deployment cost. Each dot is an orbital shell, and each line connects orbital shells which share satellite design.

References

[1] NSR, VSAT and Broadband Satellite Markets, seventeenth ed., North. Sky Res., Jan. 2019.
[2] O.L. de Weck, R. de Neufville, D. Chang, M. Chaize, Communications satellite constellations: technical success and economic failure [Online]. Available: https://www.mit.edu/~deweck/research_files/comsats_2004_001_v10/Unit1%20Success%20and%20Failure/unit1_summary.htm, Oct. 14, 2003.
[3] L. Trigeorgis, Real Options: Managerial Flexibility and Strategy in Resource Allocation, MIT Press, Cambridge, MA, 1996.
[4] O.L. de Weck, R. de Neufville, M. Chaize, Staged deployment of communications satellite constellations in low earth orbit, J. Aero. Comput. Inf. Commun. 1 (3) (Mar. 2004) 119–136, https://doi.org/10.2514/1.6346.
[5] H.W. Lee, P.C. Jakob, K. Ho, S. Shimizu, S. Yoshikawa, Optimization of satellite constellation deployment strategy considering uncertain areas of interest, Acta Astronaut. 153 (Dec. 2018) 213–228, https://doi.org/10.1016/j.actaastro.2018.03.054.
[6] S. Bosomworth, P. Grogan, Effects of Staged Deployment on the Economics of Global Broadband Internet Satellite Constellations, School of Systems and Enterprises, Stevens Institute of Technology, 2019.
[7] D. Selva, A. Golkar, O. Korobova, I.L. i Cruz, P. Collopy, O.L. de Weck, Distributed earth satellite systems: what is needed to move forward? J. Aero. Inf. Syst. 14 (8) (Aug. 2017) 412–438, https://doi.org/10.2514/1.I010497.
[8] G. Autry, The Next Economic Revolution Just (re)Launched: Congratulate SpaceX, Thank NASA, Forbes, Apr. 01, 2017 [Online]. Available: https://www.forbes.com/sites/gregautry/2017/04/01/the-next-economic-revolution-just-relaunched-congratulate-spacex-thank-nasa/?sh=2ac1da7c8c7e.
[9] L.S.E. Holdings, SpaceX Ka-band NGSO constellation FCC filing sat-LOA-20161115-00118, no. January 4th,. [Online]. Available: http://licensing.fcc.gov/myibfs/forwardtopublictabaction. do?file number=SATLOA2 016111500118, 2021.
[10] L.K. Systems, Application for fixed satellite service by Kuiper systems LLC, no. January 4th,. [Online]. Available:2021, https://fcc.report/IBFS/SAT-LOA-20190704-00057.
[11] T. Lenos, Real Options: Managerial Flexibility and Strategy in Resource Allocation, MIT Press, Cambridge, MA, 1996.
[12] R. de Neufville, S. Scholtes, Flexibility in Engineering Design, MIT Press, Cambridge, MA USA, 2011.
[13] M.-A. Cardin, Enabling flexibility in engineering systems: a taxonomy of procedures and a design framework, J. Mech. Des. 136 (1) (2014), https://doi.org/10.1115/1.4025704.
[14] M.-A. Cardin, A. Mijic, J. Whyte, Flexibility and real options in engineering systems design, in: A. Maier, J. Oehmen, P. Vermaas (Eds.), Handbook of Engineering Systems Design, Springer, 2022.
[15] Raiffa, Decision Analysis: Introductory Lectures on Choices under Uncertainty. Reading, Mass, Addison-Wesley, 1968.
[16] M.-A. Cardin, S.J. Steer, W.J. Nuttall, G.T. Parks, L.V.N. Gonçalves, R. de Neufville, Minimizing the economic cost and risk to Accelerator-Driven Subcritical Reactor technology. Part 2: the case of designing for flexibility, Nucl. Eng. Des. 243 (2012) 120–134.
[17] F. Black, M. Scholes, The pricing of options and corporate liabilities, J. Polit. Econ. 81 (3) (May 1973) 637–654, https://doi.org/10.1086/260062.
[18] M.-A. Cardin, Q. Xie, T.S. Ng, S. Wang, J. Hu, An approach for analyzing and managing flexibility in engineering systems design based on decision rules and multistage stochastic programming, IISE Trans. 49 (2017) 1–12.
[19] M. Chaize, Enhancing the Economics of Satellite Constellations via Staged Deployment and Orbital Reconfiguration, PhD Thesis, 2003 [Online]. Available: http://hdl.handle.net/1721.1/28285.
[20] SpaceX Non-Geostationary Satellite System, Attachment A, technical information to supplement schedule S." US federal communications commission, Accessed: Feb. 28, 2022. [Online]. Available: https://fcc.report/IBFS/SAT-MOD-20200417-00037/2274316.pdf, Apr. 17, 2020.
[21] N. Samama, Global Positioning: Technologies and Performance, John Wiley & Sons, Hoboken, New Jersey, 2008.
[22] V.- SpaceX, Band Non-geostationary Satellite System, Attachment A, Technical Information to Supplement Schedule S." US Federal Communications Commission, Mar. 01, 2017. Accessed: Feb. 28, 2022. [Online]. Available: https://fcc.report/IBFS/SAT-LOA-20170301-00027/1190019.pdf.
[23] J.R. Wertz, D.F. Everett, J.J. Puschell, Space Mission Engineering: the New SMAD, Microcosm Press, Hawthorne, CA, 2011.
[24] I. del Portillo, B.G. Cameron, E.F. Crawley, A technical comparison of three low earth orbit satellite constellation systems to provide global broadband, Acta Astronaut. 159 (Jun. 2019) 123–135, https://doi.org/10.1016/j.actaastro.2019.03.040.

[25] D.D. Chang, O.L. de Weck, Basic capacity calculation methods and benchmarking for MF-TDMA and MF-CDMA communication satellites, Int. J. Satell. Commun. Netw. 23 (3) (May 2005) 153–171, https://doi.org/10.1002/sat.812.
[26] E. Lutz, M. Wener, A. Jahn, Satellite Systems for Personal and Broadband Communications, Springer, Berlin, Heidelberg, 2000.
[27] SSCM, no. January 4th,. [Online]. Available: https://aerospace.org/sscm, 2021.
[28] P. Nguyen, et al., Unmanned space Vehicle cost model, US Air Force Space Missile Syst. Cent. SMCFMC 2430 (1994) 90245–94687.
[29] FCC, Kuiper Systes LLC: Application for Authority to Deploy and Operate a Ka-Band Non-geostationary Satellite Orbit System, Jul. 30, 2020.
[30] J. Foust, Strong interest for SpaceX smallsat rideshare launch services, Accessed: Sep. 27, 2021. [Online]. Available: https://spacenews.com/spacex-sees-strong-demand-for-smallsat-rideshare-launch-services/, Feb. 09, 2021.
[31] V.L. Foreman, J.L. Moigne, O.L. de Weck, A Survey of Cost Estimating Methodologies for Distributed Spacecraft Missions, Sep. 2016 [Online]. Available: https://ntrs.nasa.gov/citations/20160011343.
[32] H. Jones, The Recent Large Reduction in Space Launch Cost, NASA Ames Research Center, Moffett Field, CA, Jul. 2018.
[33] T. Speidel, Application for Mobile Satellite Service by Lynk Global, Inc, Lynk Global, Inc., Technical Narrative SAT-LOA-20210511-00064, 2021 [Online], https://fcc.report/IBFS/SAT-LOA-20210511-00064/7345368. (Accessed 27 September 2021).
[34] J.J. Davis, D. Mortari, Reducing Walker, Flower, and streets-of-coverage constellations to a single constellation design framework, Am. Astronaut. Soc. 143 (Jan. 2012) 697–712.