RESEARCH REPORT

DEVELOPMENT OF A DECISION SUPPORT SYSTEM FOR MODELLING THE SPATIAL IMPACT OF INTERRELATED SOCIO-TECHNICAL CHALLENGES

FRANK RISSEEUW

UNIVERSITY OF TWENTE.

COLOPHON

Title:	Development of a Decision Support System for Modelling the Spatial Impact of Interrelated Socio-Technical Challenges
Report type:	Master Thesis
Version:	6 – Final
Date:	21-11-2024
Author:	Frank Risseeuw
Institution/Faculty:	University of Twente – Engineering Technology
Department:	Civil Engineering and Management
Master:	Construction Management and Engineering
Supervision team:	Léon olde Scholtenhuis - University of Twente
	Ramon ter Huurne - University of Twente
	Sebastiaan Olsman – Gemeente Enschede

PREFACE

November 2024

Around ten months ago, the graduation project orientation started. Initially, I aimed for a project on aligning planning data from underground utility owners, preferably from a municipal perspective. During conversations with the municipality of Enschede, it quickly turned out that no such data existed or was shared between utility owners; I had thought a couple of steps ahead. So, a project emerged where the focus was not on aligning planning data on an operational level, but spatial data on a strategic level. By having conversations and attending meetings with municipal staff, I gained a lot of insight into the complexity of organising public space. The most interesting lesson from this is that I, as an (almost) engineer, tend to only think from a ‘technical’ perspective (Does a city district heating network ‘fit’ in the underground? Is there a sufficient amount of water storage available during heavy rains?) whereas many more factors play part in decisions about physical space. For example, a city district heating network might be the only option in a neighbourhood were people cannot afford a heat pump or, climate adaptation measures may not be deployed on a paved square because cultural heritage is more important.

Although the tool I created is far from perfect, it was interesting to see that it succeeded in at least one of the goals, it ‘started the conversation’. And as said, a lot of conversations were needed to grasp the complexity of the municipal challenges. For this, I would like to thank all municipal staff from Enschede who have helped me and given me feedback during the process. Especially Sebastiaan, who made this research possible, guided me through the municipality, gave me feedback and acted as a regular sparring partner to have nice discussions with. Finally, I would like to thank my two university supervisors Léon and Ramon for the critical but ever so supportive feedback that was given and the sometimes tough but informal feedback sessions.

SUMMARY

Dutch municipalities are facing several transition challenges that have large impacts on the infrastructures and public spaces, both above and below ground. Think of building and renovating sustainable houses, upgrading the electricity grid, deploying alternative heating solutions, creating retention areas for extreme rainfall events, and creating charging networks for mobility transitions. All these transitions have a major impact on public space and need to be executed swiftly to attain sustainable development goals (SDGs). However, they are instigated and managed by different policymaking departments in municipalities. Also, there is a lack of understanding of how those distinctive challenges affect the Urban Underground Space (UUS). Let alone that decision makers within the different public and private organisations, like municipalities and grid operators, have a holistic overview of all challenges (i.e. insight into the combined spatial impact of the challenges and their interfaces and conflicts). Because of this lack of insight, these infrastructure owners are unable to prioritise the work that needs to be executed. Instead of assigning physical space on a first-come-fist-serve basis, it is desired that informed trade-offs between spatial claims of challenges can be made.

Literature suggests that the planning of USS should be carefully integrated into a city master plan. (Kuchler, Craig-Thompson, Alofe, & Tryggvason, 2024; von der Tann, Sterling, Zhou, & Metje, 2020). This strategic and integrated planning process depends on several meta-parameters (e.g. population density, GDP, land price) and comes with a lot of uncertainties (Lin, Zou, & Deng, 2023; Peng & Peng, 2018a, 2018b). For example, it is uncertain what additional physical space will be needed to fulfil the future needs of residents. This makes it hard to predict spatial claims and therefore plan the UUS. On an operational level, the uncertainties of spatial claims have been minimized. However, the fragmentation of information and the current way of siloed planning and execution of civil projects hampers an integrated planning process as well (Hehua Zhu, 2017; Kuchler et al., 2024). Currently, no mid-term, tactical-level, physical space modelling method exists.

The goal of this study therefore was to develop a decision support system (DSS) that models the spatial impact of interrelated sectoral challenges and supports different types of decision-makers in understanding and jointly prioritising distinct scenarios. To demonstrate the value of this tool, three sectoral challenges have been selected for this study: electrification, heat transition, and climate adaptation.

Following a design science research (DSR) methodology a prototype of this tool was developed. By interviewing experts on the sectoral challenges, combined with analysing policy documentation and literature, an architecture for the developed DSS was created. Then, the rules for defining the spatial claims of the three STCs were drafted. This was implemented in ArcGIS using Python to create a tool. Based on inputted data from the decision makers, it models various scenarios that capture the use of overground and underground space on a neighbourhood level. Overground space is visualized using polygon objects, while the use of underground space is conceptualized in a metric, which expresses the volume of used space per surface area in m^3/m^2.

The tool was evaluated during a two-hour workshop with experts from each STC. The session aimed to measure the 'value' of the tool i.e. to what extent is the developed tool valuable for aiding the decision-makers in making decisions. For this, a demonstration was given presenting several modelled scenarios for the city centre of Enschede, from which the results were discussed. Also, a survey was conducted focusing on three evaluation criteria (i.e. information quality, perceived usefulness and decision support satisfaction).

The decision-makers find the tool insightful in several ways. First, it shows them the complexity of the combined challenges by showing the (im)possibilities of different scenarios. Second, it

is insightful for the decision-makers that spatial conditionality is an important factor to take into account for prioritising. All in all, the tool aids in the joint understanding of the problem domain by the different municipal clients and ‘ignites’ the conversation among them on prioritising different solution alternatives.

Although the developed tool does gain insight into the complexity of the problem domain, it does not provide enough comprehensiveness to be used for prioritisation and thus decision-making. For this, additional data and decision-making factors should be included, STC models should be more accurate and visualisations should be more meaningful.

Also, the organisation of an integrated planning process should be improved. A tool as developed should be embedded into this process. This means that regular alignment meetings should take place in which various modelled options are presented. Based on this, the municipal clients and other decision-makers can discuss them. It would be valuable for such a process as this would be standardised for many municipalities using the same tool to learn from one another.

SAMENVATTING

Nederlandse gemeenten staan voor verschillende transitie-uitdagingen die grote gevolgen hebben voor de infrastructuur en openbare ruimte, zowel boven als onder de grond. Denk aan het bouwen en verduurzamen van huizen, het verzwaeren van het elektriciteitsnet, het inzetten van alternatieve verwarmingsoplossingen, het creëren van retentiegebieden voor extreme regenval en het aanleggen van oplaadnetwerken voor mobiliteitstransities. Al deze transities hebben een grote impact op de openbare ruimte en moeten snel worden uitgevoerd om de duurzame ontwikkelingsdoelstellingen te halen. Ze worden echter geïnitieerd en beheerd door verschillende beleidsafdelingen in gemeenten. Ook is er een gebrek aan inzicht in hoe deze specifieke uitdagingen de stedelijke ondergrondse ruimte beïnvloeden. Laat staan dat besluitvormers binnen de verschillende publieke en private organisaties, zoals gemeenten en netbeheerders, een holistisch overzicht hebben van alle uitdagingen (d.w.z. inzicht in de gecombineerde ruimtelijke impact van de uitdagingen en hun raakvlakken en conflicten). Door dit gebrek aan inzicht zijn de infrastructuureigenaren niet in staat het werk dat moet worden uitgevoerd te prioriteren. Dit betekent dat in plaats van dat fysieke ruimte toegewezen wordt op basis van wie het eerst komt, het eerst maalt, er een onderbouwde afweging wordt gemaakt tussen ruimtelijke claims van transities.

De literatuur suggereert dat de planning van de openbare ruimte zorgvuldig moet worden geïntegreerd in een city-masterplan (Kuchler et al., 2024; von der Tann et al., 2020). Dit strategische en geïntegreerde planningsproces is afhankelijk van verschillende meta-parameters (bijv. bevolkingsdichtheid, BBP, grondprijs) en gaat gepaard met veel onzekerheden (Lin et al., 2023; Peng & Peng, 2018a, 2018b). Het is bijvoorbeeld onzeker welke extra fysieke ruimte nodig zal zijn om aan de toekomstige behoeften van de bewoners te voldoen. Dit maakt het moeilijk om ruimtelijke claims te voorspellen en dus de openbare ruimte te plannen. Op operationeel niveau zijn de onzekerheden van ruimtelijke claims geminimaliseerd. Echter, de fragmentatie van informatie en de huidige manier van onafhankelijke planning en uitvoering van civiele projecten belemmert een geïntegreerd planningsproces (Hehua Zhu, 2017; Kuchler et al., 2024). Momenteel bestaat er geen methode voor het modelleren van de fysieke ruimte op tactisch niveau voor de middellange termijn.

Het doel van deze studie was dan ook om een beslissingsondersteunend systeem te ontwikkelen dat de ruimtelijke impact van onderling gerelateerde sectorale opgaven modelleert en verschillende soorten besluitvormers ondersteunt bij het begrijpen en gezamenlijk prioriteren van verschillende scenario's. Om de waarde van deze tool aan te tonen, zijn drie sectorale opgaven gekozen: elektrificatie, warmtetransitie en klimaatadaptatie.

Op basis van de design science research (DSR) methode is een prototype van deze tool ontwikkeld. Door experts van de sectorale opgaven te interviewen en beleidsdocumentatie en literatuur te analyseren, werd een architectuur voor het ontwikkelde beslissingsondersteunende systeem gecreëerd. Vervolgens werden de regels voor het definiëren van de ruimtelijke claims van de drie sectorale opgaven gedefinieerd. Dit werd geïmplementeerd in ArcGIS met behulp van Python om een tool te creëren. Op basis van ingevoerde gegevens van de besluitvormers worden verschillende scenario's gemodelleerd die het gebruik van bovengrondse en ondergrondse ruimte op buurtniveau weergeven. Bovengrondse ruimte wordt gevisualiseerd met polygonen, terwijl het gebruik van ondergrondse ruimte wordt weergegeven in het volume van de gebruikte ruimte per oppervlakte (m^3/m^2).

De tool werd geëvalueerd tijdens een twee uur durende workshop met deskundigen van elke sectorale opgave. Het doel van de sessie was om de 'waarde' van de tool te meten, d.w.z. in hoeverre is de tool waardevol voor het besluitvormingsproces. Hiervoor werd een demonstratie gegeven met verschillende gemodelleerde scenario's voor het stadscentrum van Enschede,

waarvan de resultaten werden besproken. Er werd ook een enquête gehouden die zich richtte op drie evaluatiecriteria: informatiekwaliteit, bruikbaarheid en tevredenheid over de beslissingsondersteuning.

De besluitvormers vinden de tool op verschillende manieren inzichtelijk. Ten eerste laat het hen de complexiteit van de gecombineerde uitdagingen zien door de (on)mogelijkheden van verschillende scenario's te tonen. Ten tweede is het voor de besluitvormers inzichtelijk dat de randvoorwaardelijkheid van ‘ruimte’ een belangrijke factor is om rekening mee te houden bij het stellen van prioriteiten. Al met al helpt de tool bij het gezamenlijk begrijpen van het probleemdomein door de verschillende gemeentelijke opdrachtgevers en wordt het gesprek tussen hen over het prioriteren van verschillende oplossingsalternatieven aangewakkerd.

Hoewel het ontwikkelde instrument inzicht geeft in de complexiteit van het probleemdomein, biedt is het niet accuraat en uitgebreid genoeg om gebruikt te worden voor prioritering en dus besluitvorming. Hiervoor moeten aanvullende gegevens en besluitvormingsfactoren worden opgenomen, moeten STC-modellen nauwkeuriger zijn en moeten visualisaties sprekender zijn.

Ook een geïntegreerd planningsproces moet verder worden uitgewerkt, waarbij een tool zoals ontwikkeld in dit proces moet worden ingebed. Dit betekent dat er regelmatig afstemmingsbijeenkomsten moeten plaatsvinden waarin verschillende gemodelleerde opties worden gepresenteerd, waarna de gemeentelijke en andere besluitvormers deze kunnen bespreken. Het zou waardevol zijn om een dergelijk proces te standaardiseren voor veel gemeenten zodat ze van elkaar kunnen leren.

TABLE OF CONTENTS

1 INTRODUCTION

The electricity grid in the Netherlands is nearly congested which hampers all kinds of developments such as the deployment of new housing, new businesses and windfarms (Generaal, 2024). Upgrading the grid to future specifications comes with a substantial additional physical claim (thicker cables, more transformation stations, etc.). Especially in the Urban Underground Space (UUS), this is problematic because of a scarcity of available space. In addition, electricity grid congestion is not the only challenge that has to be tackled in the urban context. Other examples are the urgency to deploy new housing and business parks, the planned transition to go from fossil fuels towards alternative forms of heating, the need for spatial adaptation to climate changes, the ongoing regular maintenance of infrastructure assets, and the need to accommodate the required infrastructure for alternative forms of transport such as electric charging facilities and mobility hubs. All these challenges can be defined as Socio-Technical Challenges (STC), being systematic changes that involve new technologies that cause changes in existing market dynamics, user practices, policies and public values (Geels, 2010).

Traditionally, physical space in the underground was claimed on a first-come-first-served basis, meaning that stakeholders claiming the underground first, receive priority on deploying their new infrastructure. This, however, leads to sub-optimal use of the underground space and impedes the decision-making on a strategic level in prioritising different ‘works’ (e.g. street works, road works, civil engineering projects, utility projects) that need to be executed. It is therefore desirable to take into account the longer-term and future developments that may have an impact on the physical space both above and below ground, to plan the underground in a more sustainable and resilient manner (Volchko et al., 2020). Although municipalities do not have jurisdiction over all underground utilities (such as electricity, gas, and water), they have a responsibility to keep the liveability to a certain level. Recently, this has motivated an increasing number of municipalities to better coordinate the deployment of interventions such as work on underground utilities.

Apart from these governance-related issues that complicate stakeholder alignment, there are technical restrictions to gaining a holistic insight into the STC-induced spatial claims on the underground. For one, different infrastructure-owning organisations tend to store data about their assets in different formats, logic, and locations, sometimes these approaches even differ within the same organisation. This lack of standardization in data storage makes exchanging and integrating all this data hard, which is, however, needed to gain insight into the interfaces of the different challenges. Second, organisations hardly create a long-term schedule on when to maintain and expand specific parts of their networks. On the contrary, even, most interventions take place on a reactive basis, meaning that works are performed only when the network malfunctions. Third, few tools successfully integrate heterogeneous data from different sectors. All this makes it hard for decision-makers to overlay data and prioritise and coordinate (subsurface) works.

The culmination of challenges for the development of public space is unprecedented. Consequently, **there is a lack of understanding of how those distinctive challenges affect the UUS. Let alone that decision makers within the different public and private organisations, like municipalities and grid operators, have a holistic overview of all challenges** (i.e. insight into the combined spatial impact of the challenges and their interfaces and conflicts). **Because of this lack of insight, these infrastructure owners are unable to prioritise the work that needs to be executed**. This means, that instead of assigning physical space on a first-come-fist-serve basis, informed trade-offs between spatial claims of STCs are made.

These problems are generic and apply to nearly any urban area. The municipality of Enschede is one such area in which the societal challenges manifest themselves; and where the lack of technologies for integrated planning creates issues for underground space coordination. This study focuses on the case of Enschede and addresses several neighbourhoods on how digital technologies can help to prioritise works within the scope of three critical socio-technical challenges of electricity grid congestion, heat transition and climate adaptation.

The remainder of the report is structured as follows. Section 2 introduces the case by the municipality of Enschede that was used. Next, the theoretical framework used is described in section 3 after which the research objective is defined in section 4. Section 5 describes the methodology of the study. Sections 6-10 present the results by respectively presenting the architecture of the tool, the knowledge base to build the modelling rules, the technical implementation into a modelling engine and finally a case demonstration. The evaluation process is described in section 11. Finally, the results are discussed and conclusions are drawn in sections 12-13.

2 CASE CONTEXT: THREE CHALLENGES FOR THE MUNICIPALITY OF ENSCHEDE

As explained in the introduction, the municipality of Enschede will be the case study for this research since this is a typical urban area where several socio-technical challenges pressure the underground. The scope of this research is limited to three urgent socio-technical challenges that the municipality of Enschede is facing, viz. electrification, heat transition, and climate adaptation. Although there is an urgent need for additional housing in almost all municipalities in the country, it has been decided to leave this out of the scope because of the modelling complexity and different nature of the challenge in comparison to the other three.

2.1.1 STC1: Electrification

In a governmental document on electricity grid congestion in the Netherlands the approach to tackle the electricity grid congestion contains three directions: expand the grid, steer grid usage and increase the flexible capacity (Rijksoverheid, 2022). For this research, the scope is on the physical work that is to be executed regionally. Netbeheer Nederland (2024), an association for all electricity grid operators in the Netherlands, states that this physical work includes the replacement or expansion of around 80.000 to 105.000 km of electricity cables; and the placement of around 37.000 to 54.000 low voltage distribution stations (LVDSs). This operation requires around 260 to 330 km^2 of additional space. Around a third of all streets need some sort of intervention (Liander, 2024). In the municipality of Enschede, this means adding 580 transformer houses on top of the existing 560 of which 340 need to be upgraded. Also, 600 km of cable is to be upgraded.

This transition will lead to not only a lot of temporary use of underground space during construction work of new lines but also to the placement of LVDSs at surface-level places that are currently used for different purposes. How this relates to other uses of the subsurface space and the ground surface above, has been mapped only to a limited extent.

2.1.2 STC2: Heat transition

All municipalities in the Netherlands were compelled by the national government to create a vision of transitioning towards heating the built environment without any natural gas by 2050. In the vision of the municipality of Enschede (2022), they proposed several solutions and assigned one or several of those solutions to all neighbourhoods in the municipality, see Figure 1. Most of the solutions include all-electric heating (heat pumps), a heat grid, either small (local source) or larger (industry source), or a combination of both. These solutions have an impact on the underground; the increasing amount of heat pumps pressurises the electricity grid which needs to be upgraded and is therefore highly related to the first challenge (STC1). The deployment of heat grids also has a tremendous impact on the underground since large transportation pipes are to be deployed.

While such new energy systems are proposed, changing laws, regulations and innovations lead to new insights and changes, with consequent ramifications on the use of public space. These may have effects at different spatial scales (i.e. house, street and neighbourhood level).

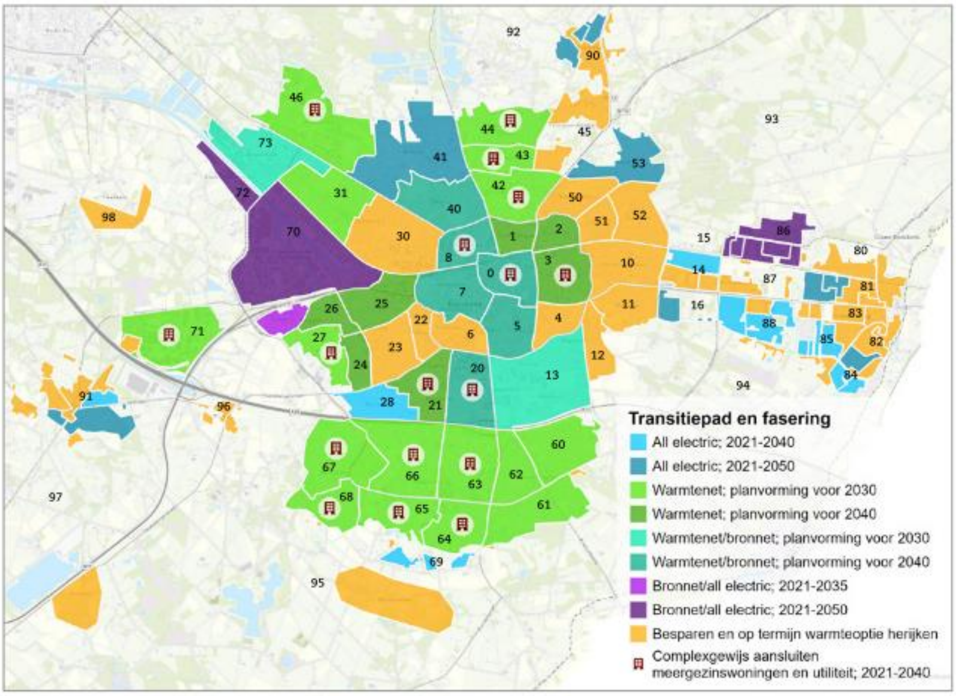


Figure 1: Opted heating alternatives per neighbourhood by the Municipality of Enschede (2022), reprint permission

2.1.3 STC3: Climate adaptation

Due to climate change, more extreme weather in the form of heat and heavy rainfall is expected. The city of Enschede needs to adapt to this weather to cope with the consequences in the form of heat stress and flooding or extreme droughts. Therefore, the municipality wrote a climate adaptation plan (Water- en Klimaatadaptatieplan Gemeente Enschede 2022-2026).

This plan aims to resolve the three issues aforementioned: flooding, drought and heat by increasing the 'vegetation percentage' and the 'water storage', the so-called 'green-blue' ambitions. These are the parameters that the municipality will focus on in assessing 'climate adaptiveness'. They created a table in which the levels are shown with their respective percentages and water storage capacity, see Table 1 (Gemeente Enschede, 2021). The municipality aims to increase the climate adaptation level by at least one step for every street when an intervention is done.

Table 1: Climate adaptiveness levels

Climate label	Percentage vegetation	Available water storage capacity
A	>20%	>55mm
B	15-20%	45-55mm
C	10-15%	35-45mm
D	5-10%	25-35mm
E	<5%	<25mm

2.1.4 Three challenges competing over space

The challenges described above claim physical space in the public domain. Especially in urban areas such as Enschede, this space is scarce. Often these challenges claim the same space (Figure 2) for the placement of additional objects e.g. expanding the electricity grid, deploying a city district heating (CDH) network or deploying climate adaptation measures. However, the urgency for climate adaptation measures depends on other factors (e.g. geographic location) than energy transitions do (e.g. electricity grid congestion or reducing use of fossil fuels). The prioritising of these transitions may therefore conflict with one another.

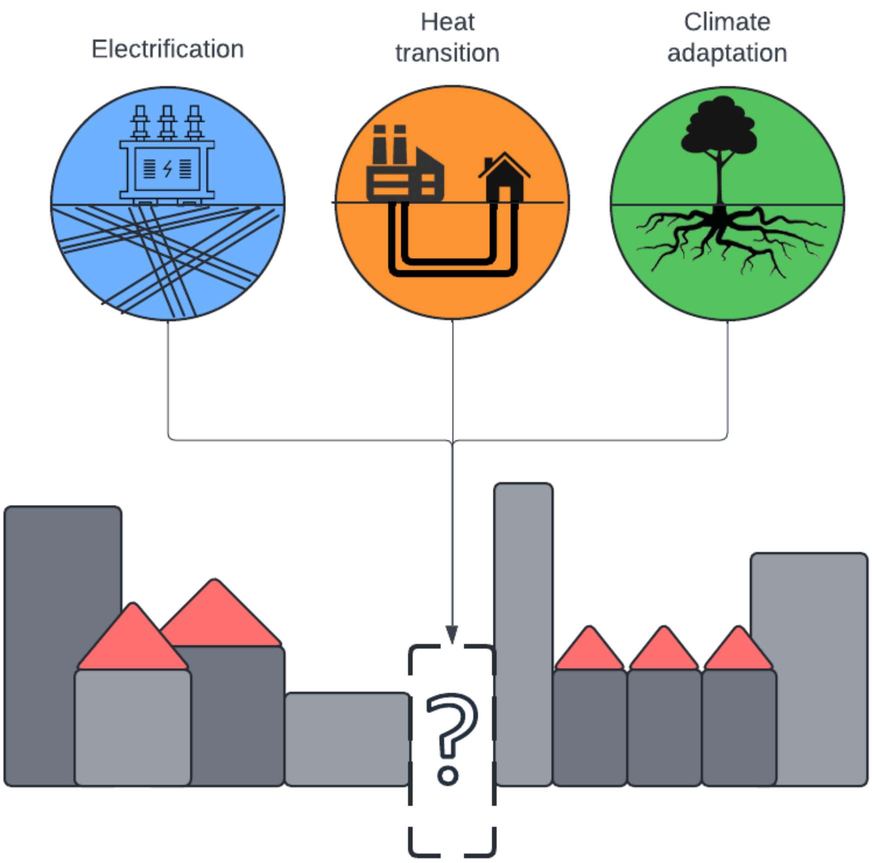


Figure 2: Three challenges competing over space

3 THEORETICAL BACKGROUND

This theoretical section is divided into three parts. First section 3.1 introduces the concept of underground urban space planning. Section 3.2 introduces currently existing spatial analysis tools. Finally, section 3.3 will summarize the research gap.

3.1 UNDERGROUND URBAN SPACE PLANNING

The urban underground space (UUS) is a scarce and non-renewable resource that is pressurized by rapid urbanization (Bobylev, 2009; Yuan, He, & Wu, 2019). For a long time, all kinds of utilities in the underground were deployed on a first-come-first-serve basis, resulting in an uncoordinated and chaotic underground. This approach is no longer viable because the increased number of applications of UUS combined with the sub-optimal deployment of structures that fulfil these applications leads to a shortage of underground space (Admiraal & Cornaro, 2016; Bobylev, 2009; Kuchler et al., 2024).

Different authors therefore argue that the UUS should be strategically planned in a city masterplan (Kuchler et al., 2024; von der Tann et al., 2020). Apart from planning the underground space in the long term, it should be approached holistically including different underground functions such as storage, industry, transport, utilities, public use and private use (Admiraal & Cornaro, 2016; Bobylev, 2009; Kuchler et al., 2024). This also means that this planning should be made on a societal level, instead of on the level of individual projects (Kuchler et al., 2024). Such a master plan should be flexible enough to account for future developments and needs (Zhao, Peng, Wang, Zhang, & Jiang, 2016), which means that it needs to enable prioritising different UUS structures by decision-makers (Bobylev, 2009). One of the problems in achieving this integrated approach to UUS planning is however the fragmentation of information and the current way of planning which happens separately for every sector and is carried out project-by-project (Hehua Zhu, 2017; Kuchler et al., 2024). This complicates the generation of a holistic insight and therefore hampers the process of prioritisation.

3.2 UNDERGROUND SPATIAL ANALYSIS TOOLS

A holistic spatial analysis of the UUS combines data from almost all possible underground uses in an urban space. This means that not only underground utilities are taken into account but also underground structures, groundwater, geothermal energy and geomaterial harvesting (Zerhouny, Fadil, & Hakdaoui, 2018). Although the urge for a holistic approach is acknowledged by several authors, there is little literature available on the interaction mapping of different underground structures. There are methods that analyse the underground volume that structures/utilities may take and calculate certain utilisation factors or occupation density from this (Bobylev, 2010; Peng & Peng, 2018a, 2018b). These factors, in turn, can then be used to see available space left and to classify this space according to usability (Peng & Peng, 2018a, 2018b). Other methods that use spatial indices are mostly aimed at gaining insight into the development potential of certain areas based on meta-parameters such as population density, GDP and land price (Lin et al., 2023; Peng & Peng, 2018a, 2018b).

3.3 RESEARCH GAP

The literature shows that planning and analysis of underground space currently takes place on two levels: operational and strategic. On the operational level, there are a lot of fixed parameters and only a specific sector is taken into account. This means that there is no integration of planning data from different sectors and analysis takes place only on 'as-is' data of structures in the underground, not taking into account possible changes to those structures in the future. On the other hand, literature suggests a more integrated way of strategically assessing the underground. However, the aim here is to evaluate the development potential of different neighbourhoods based on meta-parameters to prioritise those neighbourhoods in the long term.

What seems missing is a tactical level. This level should take into account multiple transitions that have spatial claims, i.e. the physical space needed for such a transition in public space. A tool or method is yet not developed but desired. This would ignite the conversation among decision-making authorities. It could be used to visualise individual STCs to create an understanding of one another's challenges. This would be a start to approach urban planning integrally. Consequently, there is no current knowledge and how such a tool would aid the process of prioritization of works resulting from contemporary sociotechnical challenges.

4 RESEARCH OBJECTIVE

Based on the problem space and research gap drafted above, the following research objective was defined, followed by the research questions.

> "**Design and validate a decision support system that models the spatial claims of interrelated socio-technical challenges, and supports different decision-makers in understanding and jointly prioritising distinct scenarios.**"

RQ1: Which spatial claims exist for each socio-technical challenge?

> **RQ 1.1:** What are the relevant parameters for defining the geometry of spatial claims?

RQ2: How to model spatial claims and their interrelations?

> **RQ2.1:** How can parameters be included in rules to define a spatial claim?
>
> **RQ2.2:** Which scenarios may evolve that cause interrelated spatial claims?
>
> **RQ2.3:** How to model these scenarios?

RQ3: How can modelled claims support decision-making about prioritising distinct scenarios?

> **RQ3.1:** How to visualise spatial claims?
>
> **RQ3.2:** How to model spatial conflicts of claims?
>
> **RQ3.3:** What is the value of the tool within the decision-making process based on information quality, perceived usefulness and decision support satisfaction?

5 METHODOLOGY

Combining the insights from the people of the municipality of Enschede with the current literature described in section 2 led to the conclusion that a tool that visualises spatial impact could benefit decision-makers. The combination of using an environment-defined problem and linking this with the current knowledge base in the design of an artefact fits the Design Science Research (DSR) framework (Brocke, Hevner, & Maedche, 2020). The six steps of this framework, which have been used as methodological guidance for this research, are elaborated on below.

5.1 STEP 1+2: PROBLEM IDENTIFICATION AND SOLUTION OBJECTIVES

The first two steps in the DSR process were on formulating and refining the problem and defining a solution for this problem. This iterative process emerged by having conversations with decision-makers within the municipality of Enschede and attending multi-sectoral meetings to grasp the hindrances of integral development processes. This led to the conclusion that a solution in the form of a GIS tool that would be able to model and visualise the interrelated spatial claims of STCs would be preferable.

5.2 STEP 3: DESIGN AND DEVELOPMENT

The development of the tool was divided into three stages/tasks. First, a system architecture was designed. Then a knowledge base was built in which all requirements, assumptions, and logic were gathered to model the STCs. Finally, this was technically implemented in a GIS environment for modelling and visualisation.

The creation of the architecture and knowledge base was done in parallel. This is because the architecture depended on the way the STCs would be modelled. For this, three semi-structured interviews of approximately one-hour were conducted with an expert on every STC, see Table 2. The questions focussed on three themes. First, gaining insight into the components that form the spatial claims of each STC. By asking questions such as: “What type of elements are to be (re)placed?” and “What is their spatial claim?” this could be established. Second, the parameters that influence the extent of the spatial claim and boundary conditions on their spatial claim were discussed. For example, it was asked what the boundary conditions are for placing electricity transformation stations or CDH pipes. Third, questions about the interaction between different STCs were asked such as “What dependencies exist with the other socio-technical challenges?”.

The results from the interviews were analysed and summarised based on the three themes mentioned above. This and other main take-away messages from the sessions are presented in Appendix II. Complementary, the interviewees provided documentation that assisted further in defining the rules to model the STCs. For example, a policy document was provided containing the defined dimensions of all public space entities (streets, sidewalks, ponds, etc.). Another document contained information and requirements for placing LVDSs such as its dimensions and required distance to the road. This information, complemented with additional sourced literature/policy was used to create a knowledge base with all assumptions, requirements and logic that was used in the model.

A two-hour session feedback session with technical experts of the municipality and electricity grid operators was organised to assess the preliminary knowledge base and architecture. Based on a presentation of the status-quo, the experts provided feedback and suggested improvements. This was used to create the definitive architecture and knowledge base. In Appendix III, the participants, their feedback and the revisions made are shown.

Finally, based on the architecture and knowledge base, the tool was built using Python and ArcGIS Pro. This step also included visualising the results. For overground structures, this was done using polygons. For underground structures, the UUS density metric (m^3/m^2) was used.

A remark is to be made that although the development process described above is presented chronologically, the steps have been executed in parallel in an iterative process. For example, when having modelled possible locations for climate adaptation measures in ArcGIS, it turned out that not all locations were viable. Therefore additional rules were defined to create a more realistic model. Such a rule was then added to the knowledge base.

Table 2: Interviewees

Function	STC expertise	Company
Senior Partner Energy transition	Electrification	Enexis (a major electricity grid operator in the Netherlands)
Project Manager	Heat transition	Ennatuurlijk (a major heat grid operator in the Netherlands)
Internal Client Physical Space	Climate adaptation	Municipality of Enschede

5.3 STEP 4: DEMONSTRATION

In the DSR process, the demonstration step is used to test the created artefact in a use case (Brocke et al., 2020). For this context, four scenarios were defined first that were used as input for an evaluation session which was performed during step 5.

The current dynamics within Enschede municipality in deciding on alternative heating solutions were used as a starting point for defining scenarios. The city centre was selected as a case study for the demonstration and evaluation since this is a typical area with high population density and scarcity of space.

Two heating alternatives, all-electric heating and CDH grid were modelled. Then, these had to be combined with the accommodation of space for climate adaptation measures. From this, four scenarios were created that prioritised installing heating solutions over climate adaptation measures and vice versa. The required parameters were defined for each of the four scenarios and they were modelled in the tool. The results were used as input for discussion for the evaluation session. An elaborate working out of these scenarios can be found in section 9.

1. All-electric heating, prioritising electricity grid expansion
2. All-electric heating, prioritising climate adaptation measures
3. CDH, prioritising electricity grid expansion and CDH deployment
4. CDH, prioritising climate adaptation measures

5.4 STEP 5: EVALUATION

The fifth step was to evaluate the artefact. For the created DSS this meant determining the extent of ‘value’ that the tool adds to the decision-making process (Frada Burstein, 2016). This value is determined by three evaluation criteria. First, is information quality which is the quality of the information that the DSS produces and delivers. Second, is perceived usefulness which is the degree to which a user believes that using the DSS would be free of effort. And third, is decision support satisfaction, which is the degree to which a user believes that the DSS is able to assist in the decision-making of the user’s job (Alshibly, 2015; Boukhayma & ElManouar, 2015). For this, a two-hour workshop was organised with internal clients from the municipality of Enschede and other relevant stakeholders, all participants are presented in Table 3. The workshop was based on a sequential evaluation approach in which all steps of the design process were evaluated. Because of the limited time for the session, a reduced version of the evaluation process by Borenstein was used. This consisted of three steps; 1) problem verification, 2) subsystem verification, and 3) user evaluation (Borenstein, 1998).

In the first step, the defined problem was presented to the participants after which this was verified using four statements in a survey. The participants were asked to identify themselves with one of the three STCs. Then, respond to the statements to the extent that they had insight into their 'own', the 'other' and the 'combined' challenge.

The next step consisted of presenting the participants with the way the tool was built and explaining all assumptions and requirements used to model the three STCs. Although this was already done in a feedback session with experts in the development step, room for a short discussion and additional comments was given.

In the last step of the evaluation, a demonstration of the tool was given by presenting and discussing the differences between scenarios 1 and 3. After this, another survey was held to measure the 'value' of the tool. First, by demonstrating how the scenarios were modelled and presenting the results of this test case, a discussion took place. For this two guiding questions were used: "What differences can be seen between the scenarios?" and "Can you decide for a specific scenario?"

At the end of this step, another survey was conducted based on the three evaluation criteria: information quality, perceived usefulness and decision support satisfaction. Besides the first three questions on insight into the spatial claims of their 'own', 'other' and 'combined' assignments were repeated to measure an increase or decrease. The action formulation of the questions was based on research by Borenstein (1998).

The recording of the session was used to cluster and summarise all relevant comments. The clusters were based on the evaluation criteria: information quality, perceived usefulness and decision support satisfaction. The survey is used as a framework for the analysis after which valuable comments and insights from the discussion were added to this.

Table 3: Evaluation session participants

Function	STC expertise	Company
Internal Client Physical Space	Climate adaptation	Municipality of Enschede
Internal Client Heat Grids	Heat transition	Municipality of Enschede
Manager	Electrification	GROND'G Foundation
Project Manager	Electrification	Municipality of Enschede

5.5 STEP 6: COMMUNICATION

The final step is about communication of the results and conclusions which is the purpose of this report.

6 RESULTS: SYSTEM ARCHITECTURE

In Figure 3, the system architecture is presented. It was based on a decision-support system (DSS) architecture by Mir and Quadri (1970). It contains three main components: data management, knowledge base and model engine. Usually, there is a user interface between the model engine and the user. For this research, the focus was mainly on establishing the knowledge base, therefore no customised user interface was created.

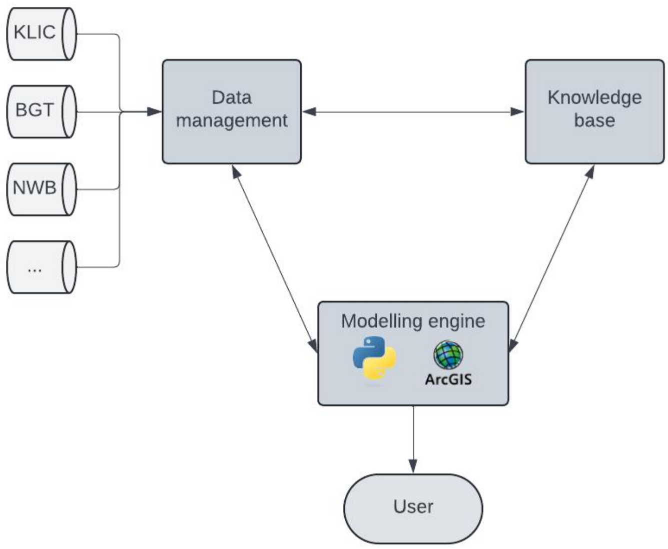


Figure 3: Tool's Architecture

The data management block handles the input data. Three Dutch cadastral datasets were used. Additionally, data from Enschede municipality containing the location and diameters of sewer pipes and a map with all parks have been used. The rules created in the knowledge base have been defined based on the specific input data. An elaboration on data management is given in section 7.

The knowledge base contains all STC modelling rules. This knowledge was gathered by processing information from the interviews, feedback sessions, policy documents and additional literature. This will be elaborated on in section 8.

The engine of the model translates the knowledge base rules into Python code. This is then used to automatically generate results in a GIS interface, for this research ArcGIS Pro has been used. With this tool, scenarios can be generated based on inputted planning parameters by decision-makers, these have to be hard-coded since no user interface is created. A further elaboration on this technical implementation is given in section 9.

7 RESULTS: DATA MANAGEMENT

This study aimed to use publicly available data combined with municipal data such that the approach used would be generically applicable to municipalities. For this study, three cadastral datasets were used KLIC (Cables and Pipes Information Centre), a dataset containing almost all underground utilities in polylines. BGT (Base registration Large-scale Topography) data was used to distinguish the different entities of overground public space. It contains polygons with the entities of buildings, roads, vegetation, etc. NWB (National Road Database) contains the centre-lines of all Dutch roads. Additionally, data from the municipality such as the sewerage network containing the diameters of sewer pipes were used.

The rules in the knowledge base were geared towards the available data and the format this data was retrieved. Also, the ArcGIS environment in which the modelling engine was created sometimes restricted the way data was handled and thus the way a rule had to be defined. As said, the process of managing data, creating a knowledge base and programming the modelling engine was an iterative process executed parallelly.

8 RESULTS: KNOWLEDGE BASE

It was decided that the tool would distinguish between overground and underground modelling for STCs. The main idea was to model an 'as-is' basemap above and below ground and build the respective claims of the three STCs on top of this. Overground models would consist of polygons representing a claim in public space. Underground objects were modelled using the UUS density metric; m^3/m^2. This concept is depicted in Figure 4.

The figure shows that all STCs have overground claims and electrification and heat transition also have claims on the underground. Above ground, Low Voltage Distribution Stations (LVDSs), Heat Distribution Stations (HDSs) and Climate Adaptation Surfaces (CASs) are modelled respectively for STCs 1-3. Below ground, the claims of additional electricity cables are modelled for STC1 and the claims for CDH pipes are modelled for STC2.

Furthermore, the arrow between the three STCs indicates that the order they are depicted is not necessarily the order they have to be modelled. On the contrary, this order can be chosen freely and therefore alters the priority.

Below the rules created for the modelling and the underlying assumptions and requirements are presented in sections 8.1.-8.4 for the 'as-is' basemap and respective STCs.

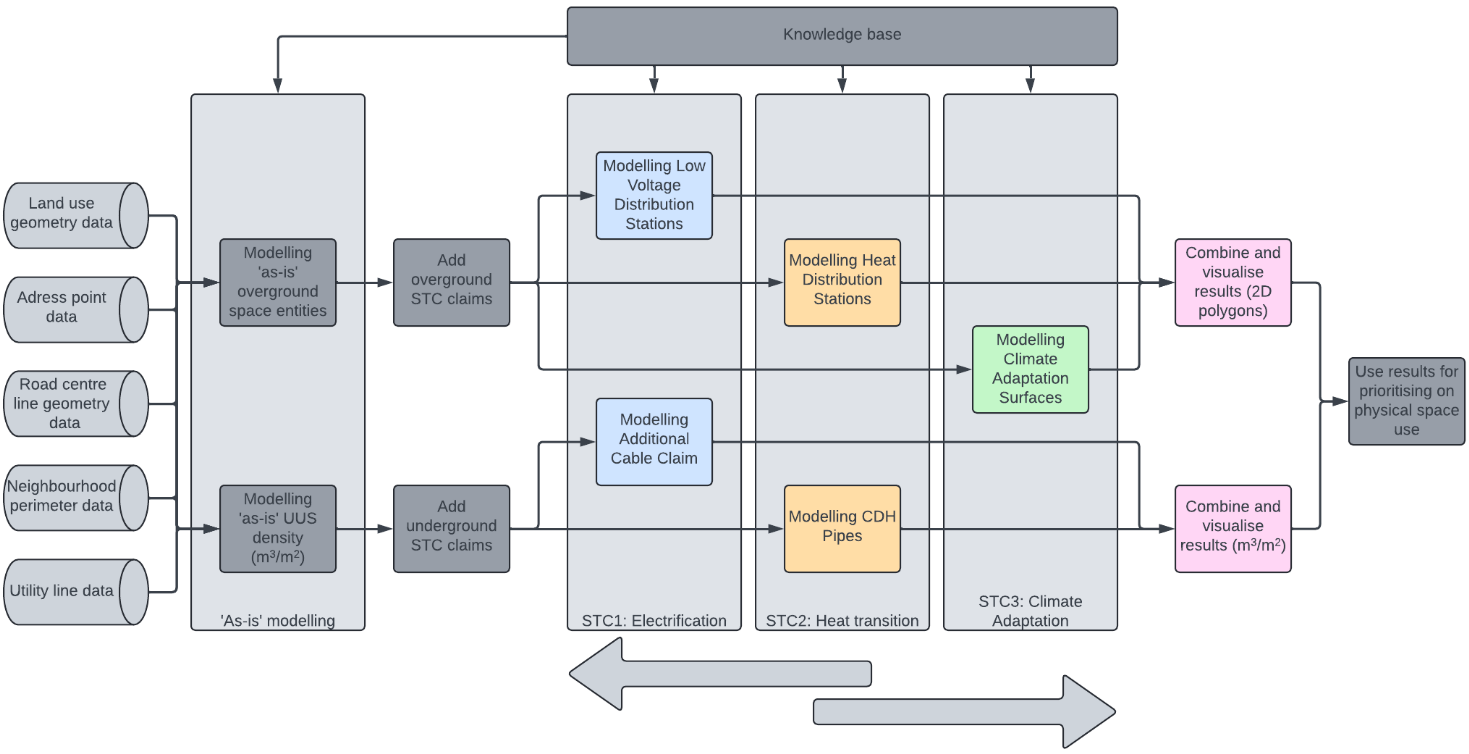


Figure 4: Conceptual design

8.1 'AS-IS'

For this tool, it was decided to model overground claims as 2D polygons on a map. To allocate additional components, it was needed to have information on the different 'entities' of public space surfaces. For this cadastral data was used (PDOK, 2024). This data represents public space in terms of polygons by their entity e.g. buildings, roads, vegetation areas, etc (Figure 5). The specific entity layers that have been used are presented in Table 4. For the STC models, this data could then be used to base allocation rules on, e.g., the placement of an electricity transformation station is only allowed on surfaces that have the entities of 'vegetation' or 'parking area' and are within a distance of 2 meters of a road.

Figure 5: BGT layers

For the underground basemap, it was decided that the level of detail of single utility cables and pipes would be unnecessary. As a matter of fact, for decision-making on this level, it was assumed that insight into the 'saturation' of the underground would be sufficient. Therefore it was chosen to model the occupation rate or UUS density in terms of m^3/m^2 as defined by Bobylev (2016). As a starting point, utility data was used, see Figure 6. This polyline data was converted into tiles displaying the volume of all utilities below such a tile in m^3/m^2 (Figure 7). This was done by multiplying all utilities with their respective trench footprint (see Table 4) and adding them.

Table 4: Rules - 'As-is'

ID	Rule description	Domain	Reference
0.1	The model captures the following spatial objects: location and surface areas of buildings (2D polygons), private grounds (2D polygons), vegetation areas (2D polygons), water (2D polygons), embankments (2D polygons), roads (2D polygons), and addresses (points).	Overground	BGT data gathered from the Dutch cadastre (PDOK, 2024)
0.2	The model should contain centre-line utility location information.	Underground	KLIC data provided by the municipality of Enschede

0.3	Utility geometries should be conceptually represented as a trench, where the width and depth are determined as follows per utility type:
	• Gas → 0.36*0.11 meters
	• Data/Telecom→ 0.35*0.04 meters
	• Electricity → 0.3*0.07 meters
	• Sewer → 1*0.45 meters
	• City District Heating
	  ○ 1.6*0.4 meters (transportation)
	  ○ 1*0.1 meters (distribution)
	• Water → 0.36*0.11 meters
		Underground	Depth and width from NEN7171 guideline (NEN, 2024).

0.4	Utilities that occur less frequently (other than energy, telecom, water, gas, and sanitation) are not included in the model.	Underground	Assumption

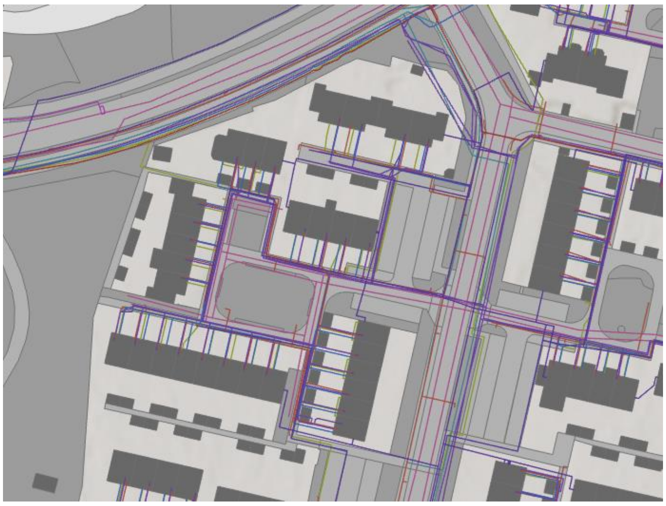


Figure 6: Utility polylines

Figure 7: Tiles representing combined utility volumes in m^3/m^2

8.2 STC1: ELECTRIFICATION

The upgrading of the electricity grid can be roughly divided into two tasks: additional placement of transformation stations (overground) and the replacement/expansion of current cables (underground). Since the tool focuses on a neighbourhood level it was chosen to only model the low-voltage distribution stations (LVDSs). The number of additional LVDSs needed was linked to the number of households based on figures by Netbeheer Nederland. They estimate that currently 167 households are connected to one LVDS and the number of LVDSs should increase between 15-75% based on the selected heating alternative i.e. CDH or all-electric heating (Netbeheer Nederland, 2019). For example, in an all-electric situation, this factor is 75% which means that a neighbourhood with 668 households is estimated to have 4 LVDSs currently and accounts for another 3 LVDSs additionally. Additional requirements that were used to model the LVDSs can be found in Table 5.

The underground model for STC1 consists of an additional volume claim in the underground. This can be seen as the claim that thicker or additional cables have on top of the current cables. The additional claim is modelled as a factor which varies, on the chosen heating alternative. Because all-electric heating or city district heating influences the number of LVDSs and the amount of additional cables. This factor, varying between 0.13-0.56, is multiplied by the current electricity cable claim (m^3/m^2). For example, a tile containing an electricity cable volume of 0.2 m^3/m^2 will have an additional volume claim of $0.56*0.2\ m^3/m^2 = 0.11\ m^3/m^2$. This yields an all-electric heating solution.

Figure 8: A LVDS

Table 5: Rules – Electrification

ID	Rule description	Domain	Reference
1.1	The model should only capture Low Voltage Distribution Stations (LVDSs).	Overground	Verified during feedback session
1.2	The tool models an LVDS for every 167 households in a neighbourhood.	Overground	(Enexis, 2023; Netbeheer Nederland, 2019)
1.3	The number of additional LVDSs is calculated by multiplying the current amount of LVDSs by a factor. This factor varies between 0.15 and 0.75. For scenarios were a CDH network is modelled 0.15 is chosen. For scenarios where an all-electric heating solution is modelled, 0.75 is chosen.	Overground	(Enexis, 2023; Netbeheer Nederland, 2019)
1.4	The tool models an LVDS as a 2D rectangle with a footprint of 7.5*4 meters.	Overground	(Enexis, 2023; Netbeheer Nederland, 2019)
1.5	The specific polygon types for placing an LVDS are 'vegetation' areas and 'parking' lots.	Overground	Verified during feedback session
1.6	An LVDS is placed within 2 meters of a truck-accessible road.	Overground	(Enexis, 2023; Netbeheer Nederland, 2019)
1.7	An LVDS cannot be placed on top of existing underground infrastructure.	Overground	Verified during feedback session
1.8	An LVDS is preferably within 100 meters of all households it's connected to.	Overground	Verified during feedback session
1.9	The number of additional HDSs is calculated by multiplying the current volume by a factor. This factor varies between 0.13 and 0.56. For scenarios were a CDH network is modelled 0.13 is chosen. For scenarios where an all-electric heating solution is modelled, 0.56 is chosen.	Underground	The use of a factor was verified during the feedback session, actual figures by Netbeheer Nederland (2019).

8.3 STC2: HEAT TRANSITION

A CDH system consists of a double underground pipe system (supply and return) that transports a hot fluid from a heat source e.g. a waste treatment plant to households. Large pipes transport the fluid to heat distribution stations (HDSs) above ground, whereafter smaller pipes bring the water to the household connections (Wartmenetwerk, 2021).

For the overground model, the tool models HDSs. Conceptually, this is modelled the same as LVDSs (rules 1.1-2, 1.4-7), only different parameters are used. A HDS serves about 300-500 households and needs a surface of around 20 m^2. Additional requirements that were used to model the HDSs can be found in Table 6.

To model the underground transportation and distribution pipes of a CDH system, it was chosen to use the current gas grid as a basis. It can reasonably be argued that a future CDH network could follow the same route towards all household connections. It was pointed out by one expert respondent that in reality, those pipes will not exactly overlay the gas pipes and that certain distances to other utilities such as waterpipes should be taken into account, but for this context the approach is viable. Also, since the final utility volumes will be converted to a volume underneath a virtual tile the exact location is not necessary for this application. For the modelling of the pipes, a distinction between 'high' and 'low' pressure gas pipes is made which are modelled respectively as 'transportation' and 'distribution' pipes. The trace of the gas grid was multiplied by the footprint of the CDH trenches and converted into tiles displaying its volume in m^3/m^2 (Table 6).

Table 6: Requirements and Assumptions - Heat transition

ID	Rule description	Domain	Reference
2.1	The model should only capture Heat Distribution Stations (HDSs)	Overground	Verified during feedback session
2.2	The tool models a HDS for every 167 households in a neighbourhood.	Overground	Heat transition interview
2.3	The tool models an HDS as a 2D rectangle with a footprint of 5*4 meters.	Overground	Heat transition interview
2.4	The specific polygon types for placing an HDS are 'vegetation' areas and 'parking' lots.	Overground	Verified during feedback session
2.5	An HDS is placed within 2 meters of a truck-accessible road.	Overground	Verified during feedback session
2.6	An HDS cannot be placed on top of existing underground infrastructure.	Overground	Verified during feedback session
2.7	For modelling the CDH network the tool models current gas utility lines as CDH pipes. A distinction is made between 'high-pressure' gas lines modelled as transportation pipes and 'low-pressure' gas lines modelled as distribution pipes. The trench dimensions that are used follow the definition in rule 0.3.	Underground	Verified during feedback session and expert opinion

8.4 STC3: CLIMATE ADAPTATION

The aim of the climate adaptation plan of the municipality is to increase the amount of vegetation area [%] and water storage [mm]. Measures to achieve this goal come in various forms; trees, parks, vertical vegetation solutions, wadi's etc. (Roijackers, 2018). For this tool, it was decided to focus on reducing currently paved areas in favour of an abstract form of CASs (CASs). This means that possible areas are sourced, but no specific function is assigned to them.

The general idea was to (re)model roads using their minimal required legal widths and subtract this from existing paving. These widths are determined based on the 'functional road class' assigned to a specific road, the corresponding road widths used come from a policy document (Gemeente Leiden, 2013). The surfaces that remain are then further filtered. First surfaces smaller than 10m^2 are removed, as the municipality of Enschede handles a minimum CAS area.

Second, during the modelling, it turned out that several paved surfaces were undesirable for turning into CAS (e.g. they are currently used as footpaths within a park). Because sidewalks could be partly turned into CAS, no filtering on entities could be performed. Therefore a rectangularity indicator was used since the undesirable entities most of the time had an 'odd' shaped polygon. Therefore a minimum bounding geometry was drawn and only polygons with a surface of at least half the bounding geometry were retained. The remaining surfaces can be used for climate adaptation measures. On top of that, parking places were modelled as semi-pavement. These assumptions are summarised in Table 7.

One of the assumptions is that no current cables or pipes can be below a climate adaptation measure except for sewer pipes. Together with the principle of the municipality of preferably deploying overground climate adaptation measures, it was decided not to model the underground claim of CASs.

Table 7: Requirements and Assumptions – Climate adaptation

ID	Rule description	Domain	Reference
3.1	Road sections are conceptually remodelled using a 'minimal required width' and are subtracted from the actual pavement representations. Surfaces that remain are defined as Climate Adaptation Surfaces (CASs).	Overground	Verified during feedback session
3.2	The model captures road centre lines for modelling 'minimal required width' roads.	Overground	NWB data was retrieved from the Dutch government (Wegverkeer, 2024)
3.3	The model should distinguish between different road 'entities' (e.g. main road, driveway, footpaths). This is done by using the functional road class (FRC)	Overground	
3.4	The FRC scores roads between 0-7, the minimal road widths used are: • FRC 0-6 → 4.5 meters • FRC 7 (footpaths) → 1.2 meters	Overground	The minimal road widths come from policy documents from the municipality of Leiden (2013)
3.5	The minimal surface for a CAS is 10m^2, smaller surfaces will be removed.	Overground	(Gemeente Enschede, 2023)
3.6	A rectangularity factor is used to remove polygons that have odd, non-rectangular shapes.	Overground	Assumption
3.7	All current parking areas are converted into CASs	Overground	Verified during feedback session
3.8	For the total amount of CASs, parking areas are accounted for 50% of their surface. This is	Overground	Verified during feedback session

	because they are assumed to be paved by semi-pavement.		
3.9	No underground representation is modelled for the underground volume of CASs.	Underground	Climate adaptation interview

9 RESULTS: MODELLING ENGINE

This section elaborates on the actual Python and ArcGIS implementation. In section 9.1, the translation of the knowledge base models into Python code is explained. Next, the running and interpretation of the model are discussed in section 9.2. The latter also includes an elaboration on the visualisation within the model.

9.1 STC MODELS

9.1.1 Electrification

Below the pseudocode for generating the LVDSs is presented including the default parameters used. First, all possible locations are generated by selecting the desired public space entities and making sure that the midpoint of an LVDS should fit at any point in the selected polygons (lines 1-3, Figure 9). Then, only locations within a vicinity of 2 meters of the road are kept (lines 4-6, Figure 10). After this, surfaces with cables and pipes below are removed. Then the remaining polygons are converted into points (lines 7-8, Figure 11). Next, only the points within a distance of 10 meters of the electricity grid are retained and the number of available locations is determined (lines 9-10). Finally, the number of required LVDSs is determined, randomly selected from the list of points and created by a rectangular polygon (lines 11-15, Figure 12). This is captured in the pseudocode below.

Electrification overground algorithm – Placement of LVDSs

Inputs:
A: Public space polygons (from ‘as-is’ map using BGT data)
B: Polygon types suitable for placing LVDSs (default = vegetation and parking places)
C: Memory layer
D: Footprint of an LVDS (default = 7.58*4.05 meters)
E: Polygon types suitable for driving (default = local road and driveway)
F: Max permitted distance LVDS to the road (default = 2 meters)
G: Utility polylines (from ‘as-is’ map using KLIC data)
H: Max permitted grid distance of LVDS (default = 10 meters)
I: Adress point map (from ‘as-is’ map using BGT data)
H: Number of household connections per LVDSs (default = 167)
I: Fraction of additional LVDS needed (default = 0.75)

Output: polygon layer displaying possible LVDS locations

		Intermediate Result (IR)
1	Copy polygons with characteristic **B** from **A**	**IR1**
2	Remove **C** from **IR1**	**IR2**
3	Create a negative buffer from **IR2** by the half diagonal of footprint **D**	**IR3**
4	Copy polygons with characteristic **E** from **A**	**IR4**
5	Create a positive buffer from **IR4** by the half diagonal of footprint **D** + distance **F**	**IR5**
6	Intersect **IR3** and **IR5**	**IR6**
7	Remove **G** from **IR6**	**IR7**
8	Convert remaining polygons (**IR7**) into points	**IR8**
9	Only retain points with distance **H** to current electricity cables	**IR9**

10	Determine the *number of points for possible LVDS locations* from **IR9**	**IR10**
11	Determine the *number of addresses in the current neighbourhood* from **I**	**IR11**
12	Calculate the *number of LVDSs needed*: **IR11/H*I**	**IR12**
13	IF *number of LVDSs needed >= number of possible LVDS locations* *number of LVDSs needed = number of possible locations*	
14	Select random locations: IF *number of LVDSs needed* = 1 Select a random point from **IR9** ELSE Select the *number of additional LVDSs needed* amount of points using the multivariate clustering algorithm from **IR9**	**IR14**
15	Convert selected point(s) (**IR13**) to rectangular polygons based on footprint **D**	

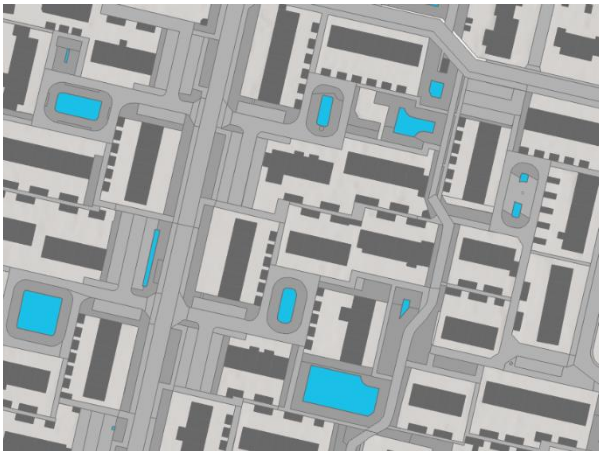


Figure 9: Determining possible midpoint locations for LVDSs

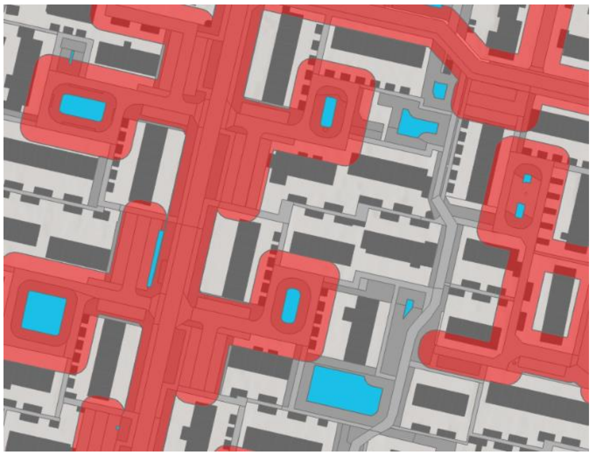


Figure 10: Intersecting areas in vicinity of roads

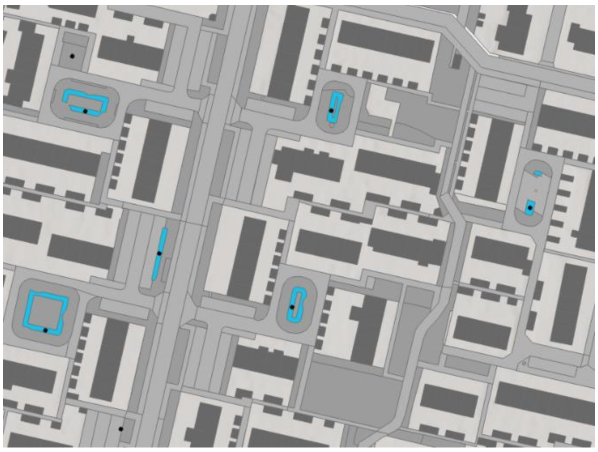


Figure 11: Transferring possible locations to points

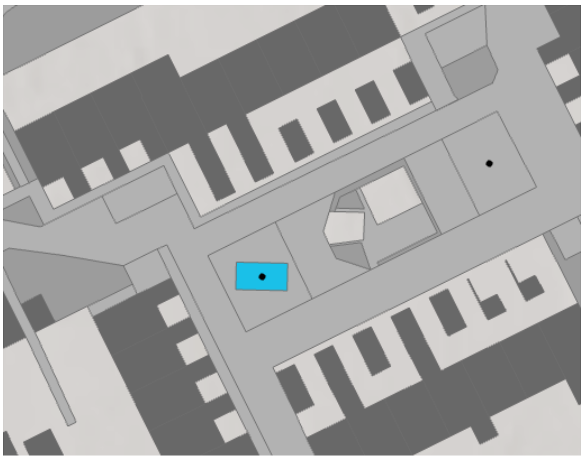


Figure 12: Transferring points to LVDSs

For the additional cable claim, the existing electricity cables are copied and rasterised by length (lines 1-3). Then the additional volume was calculated by multiplying the length by the cable trench footprint and the factor of additional cable claim (lines 3-5). This is captured in the pseudocode below.

Electrification underground algorithm – Additional cable claim

Inputs:

A: Utility polylines (from 'as-is' map using KLIC data)

B: Footprint of standard electricity cable trench (default = 0.3*0.07 meters)
C: Fraction of additional cable volume per added cable (default = 0.56)

Output: raster layer with additional surface occupied by new cables (m^3/m^2)

		Intermediate Result (IR)
1	Copy all polylines with characteristic ‘electricity cable’ from **A**	**IR1**
2	Rasterise the polylines from **IR1**	**IR2**
3	Calculate the *length* of every line section per tile from **IR2**	**IR3**
4	Calculate the additional volume of every line section based on formula: *length****B*****C**	**IR4**
5	Sum up volumes per tile	

9.1.2 Heat transition

The generation of HDSs works the same as the placement of LVDSs, for the pseudocode see section 8.1.1. There are two minor changes. First, there have been used two different parameters: the station footprint **B** (5*4 meters) and the number of connections per station **F** (400). Second, the calculation for the number of HDSs differs since no ‘additional’ stations are calculated as for the LVDSs but just the number of required HDSs. This is because it is assumed that no current stations are present. Therefore the calculation of line 12 becomes: Calculate the *number of HDSs needed*: **IR11**/**H**.

For the generation of the CDH pipes for calculating the underground claim, the polyline trace of the original gas network is used. For this, a copy of the polylines of the current gas network with a distinction between high and low pressure is made (line 1, Figure 13). Then, the polylines are converted into volumes by multiplying them by their respective footprints. Then their volumes are rasterised (lines 2-5).

Heat transition underground algorithm – Deploying transportation and distribution pipes

Inputs:
A: Utility polylines (from 'as-is' map using KLIC data)
B: Footprint of transportation pipes trench (default = 1.6*0.4 meters)
C: Footprint of distribution pipes trench (default = 1*0.1 meters)

Output: raster layer with additional surface occupied by CDH pipes (m^3/m^2)

		Intermediate Result (IR)
1	Copy all polylines with characteristic 'high-pressure gas' and 'low-pressure gas' from **A**	**IR1**
2	Rasterise the polylines from **IR1**	**IR2**
3	Calculate the *length* of every line section per tile from **IR2**	**IR3**
4	Calculate the additional volume of every line section based on formulas: 'high-pressure gas' *length**B** 'low-pressure gas' *length**C**	**IR4**
5	Sum up volumes per tile	

Figure 13: Copying current gas network, 'high' pressure (red) and 'low' pressure (yellow)

9.1.3 Climate adaptation

For modelling the CASs, first, the road centre lines from the National Road Database are copied and buffered by their minimal required widths based on the 'functional road class' (lines 1-4, Figure 14). Then, polygons above sewer lines are maintained, and other polygons above cables and pipes are removed. Next, polygons with a small area are removed as well. Also, polygons with a non-rectangular shape are removed. For this, a rectangular minimum bounding geometry is drawn and its surface is compared to the surface of the actual polygon. By default, the polygon is removed when the area of the bounding geometry is more than twice as large as the area of the polygon (lines 7-8, Figure 15). This is an assumption made to model more realistic results, its implementation is explained in section 8.4. Finally, calculations on the increase of CASs are carried out (lines 9-13).

Climate adaptation overground algorithm – Placement of climate adaptation measures

Inputs:
A: Road centre-polylines (from NWB)
B: FRC road widths
C: Public space polygons
D: Utility polylines, except for sewer pipes (from 'as-is' map using KLIC data)
E: Min permitted CAS (default = 10m^2)
F: Rectangularity threshold (default = 2)
G: Memory layer
H: Park layer

Output: polygon layer displaying possible CASs

		Intermediate Result (IR)
1	Copy the road centre lines from **A**	**IR1**
2	Add a column to **IR1** and assign a road width based on columns 'FRC' and **B**	**IR2**
3	Create a positive buffer from **IR1** by road widths assigned in **IR2**	**IR3**
4	Copy all polygons with characteristic 'pavement' from **C**	**IR4**
5	Subtract **IR3** from **IR4**	**IR5**
6	Remove **D** from **IR5**	**IR6**
7	Remove polygons from **IR6** that have an area below threshold **E**	**IR7**
8	Determine minimum bounding geometry for polygons from **IR7**	**IR8**
9	Remove polygons from **IR7** that have a bounding area above: **F****actual polygon surface*	**IR9**
10	Remove **G** from **IR9**	**IR10**
11	Determine the *area of all parks* within the neighbourhood	**IR11**
12	Determine the *current amount of vegetation (excluding parks* **H***)*	**IR12**
13	Calculate the *additional amount of vegetation (excluding parks* **H***)*	**IR13**
14	Calculate the *increase in vegetation* by the formula: *current amount of vegetation + additional amount of vegetation*)/*current amount of vegetation* * 100	**IR14**

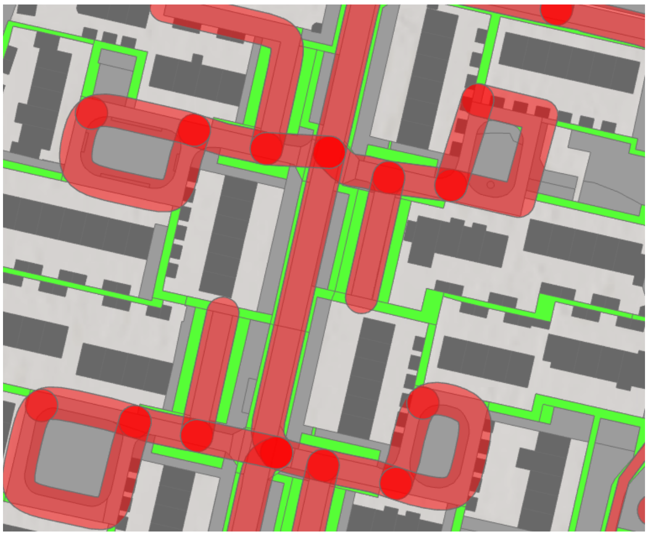


Figure 14: Subtracting roads with minimal widths (red) from current paving (green)

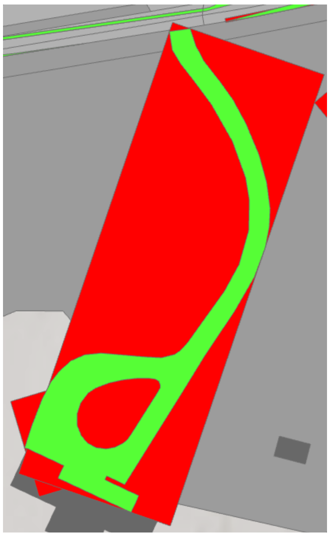


Figure 15: The minimum bounding geometry area red), is more than twice as large as the footpath polygon (green)

9.2 RUNNING AND INTERPRETING RESULTS

Before running the tool, the parameters for each STC have to be set. These parameters are shown in the pseudocode parts above. For example, for modelling the LVDSs the number of households per LVDS and the fraction of additional stations are to be set (parameters **H** and **I**, see 8.1.1). Moreover, at the end of the code, the prioritisation has to be set i.e. the sequence in which the STCs will be modelled. Changing this sequence influences the order in which the STCs are modelled. Since overground STC models take into account each other's claims to prevent double occupation, this means that when an STC is modelled earlier in the sequence, more space is available for the placement of objects.

Most conveniently, the tool is run via PyCharm, a Python IDE in which the tool was developed. By selecting the ArcGIS Pro Python interpreter, the code can be run and the results will be accessible via a pre-defined ArcGIS Pro project.

After the processing has been completed, the results can be shown in the pre-defined ArcGIS Pro document (Figure 16). Besides the code generates results in the console as the number of available locations for LVDSs and HDSs and the increase in vegetation area. On the left-hand side of the screen, all result layers are presented and grouped. They represent the following results:

- STC1_glayer: containing the polygon layer with LVDSs and the raster layer with the additional cable claim.
- STC2_glayer: containing the polygon layer with HDSs and the raster layer with the CDH pipe claim.
- STC3_glayer: containing the polygon layer with the CASs
- Volumes_glayer: containing the raster layers with the: KLIC volumes ('as-is'), KLIC+STC1 volumes, KLIC+STC2 volumes and KLIC+STC1+STC2 volumes
- KLIC_glayer: containing the polyline layers from the KLIC data
- BGT_glayer: containing the polygon layers used from BGT

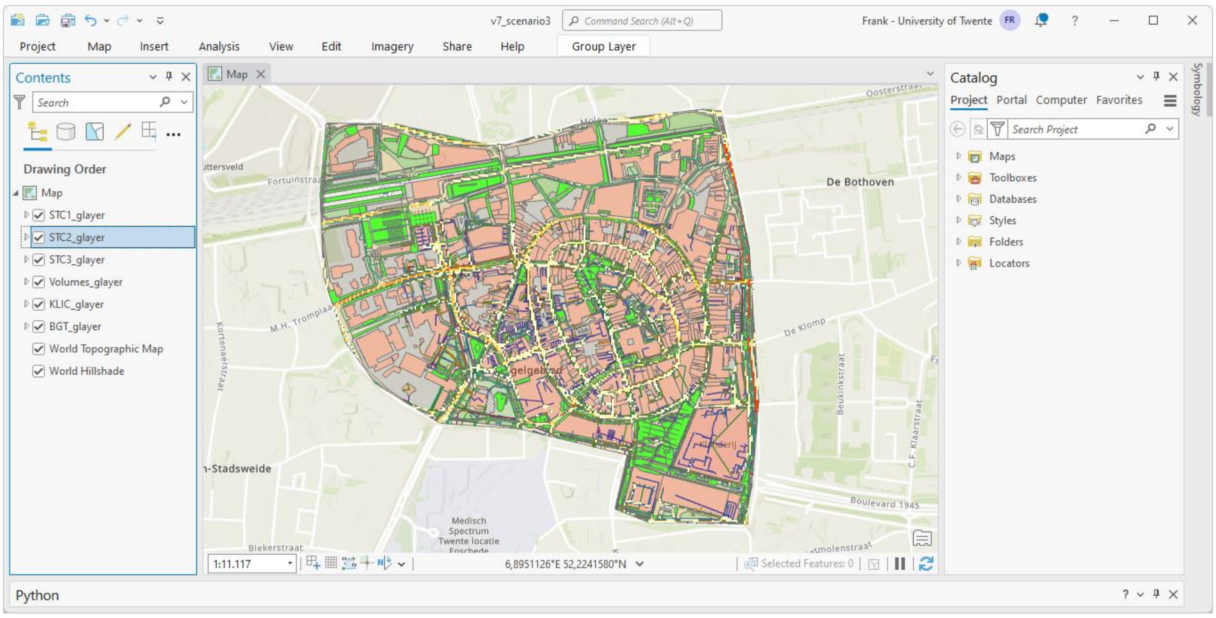


Figure 16: ArcGIS Pro output

For the visualisation of the underground models, the underground claims of the 'as-is' utilities, additional electricity cable claims and heat pipes had to be added. This is easily executed because of the raster layers that contain this data. After this, the challenge was to visualise this data in a meaningful way. This was done using a colour ramp ranging from yellow to red displaying occupation rates between 0.1-1.5 m^3/m^2. For the moment, the maximum of 1.5 m^3/m^2 has been loosely based on Figure 17 which displays the UUS density histogram of the 'as-is' map of the city-centre neighbourhood of Enschede. As can be seen, above 1.5 m^3/m^2 there are hardly any tiles having a higher occupation. From this, it was concluded that this 'maximum' is at least viable. Also, here it can be seen that most of the tiles have a very small occupation which troubles the view, therefore all values below 0.1 m^3/m^2 are made invisible.

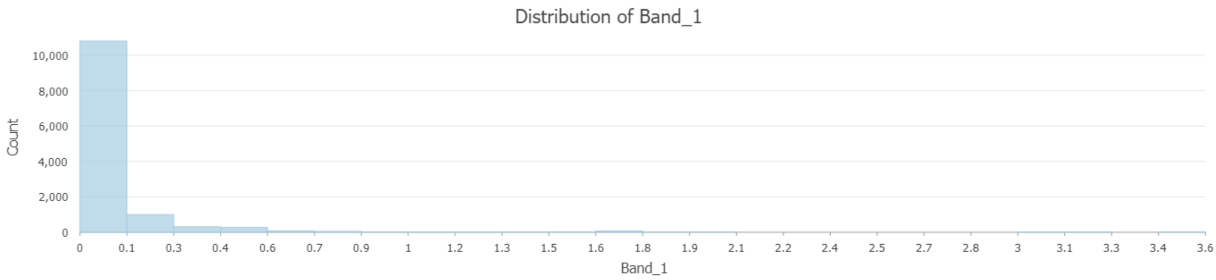


Figure 17: Distribution of UUS density for a neighbourhood in Enschede

10 RESULTS: CASE DEMONSTRATION

This chapter uses the rule-based tool to develop a range of scenarios, which can be generally applicable, and to show the results of a case study in the municipality of Enschede. For this section 10.1 defines the scenarios and section 10.2 presents the results of them.

10.1 SCENARIO DEFINITION

The scenarios were defined in the methodological part where a distinction between two heating alternatives i.e. all-electric heating and city district heating was made. Furthermore, the priority of the heating solutions (STC1 and STC2) and climate adaptation (STC3) was changed. Combining this gives a total of four scenarios.

Concretely, this leads to the use of different parameters for STC1 e.g. when opting for an all-electric solution, more LVDSs and additional cables are to be placed than if the city district heating solution is chosen. To quantify this, parameters **H** (overground electrification model) and **C** (underground electrification model,) model are changed according to Table 8. On top of that, the STC2 module in the tool was disabled for scenarios 1 and 2 since no CDH grid was to be generated, and enabled for scenarios 3 and 4. The inputs for STC3 were unchanged.

Table 8: Four scenarios and their STC settings

	Scenario 1: All-electric, prio. Electricity grid	Scenario 2: All-electric, prio. Climate adapatation	Sceneario 3: CDH, prio. Electricity grid/CDH	Scenario 4: CDH, prio. Climate adaptation
STC 1: Electrification parameters	Fraction of additional LVDSs (**H**): 0.75 Fraction of additional cables (**C**): 0.56	Fraction of additional LVDSs (**H**): 0.75 Fraction of additional cables (**C**): 0.56	Fraction of additional LVDSs (**H**): 0.15 Fraction of additional cables (**C**): 0.13	Fraction of additional LVDSs (**H**): 0.15 Fraction of additional cables (**C**): 0.13
STC2: Heat transition parameters	Disabled		Enabled	
STC3: Climate adaptation parameters	Default parameters		Default parameters	

10.2 SCENARIO RESULTS

The defined scenarios were run for the city centre of Enschede. To exemplify, part of the output map of scenario 3 is presented in Figure 18. All scenario maps are presented in appendix IV. A LVDS (blue) and HDS (orange) are highlighted. Furthermore, all possible CASs are displayed in green. The orange and yellow raster grid presents the UUS density in m^3/m^2.

Table 9 presents the results from the console of scenarios 1-4. Scenario 1 shows that very few locations for LVDSs are available and even fewer in scenario 2 where climate adaptation measures have priority. Scenarios 3 and 4 show that several HDSs cannot be placed. Also, it can be seen that the vegetation increase in all scenarios is 20%.

The actual impact of the differences between the scenarios will be discussed in the next section where these were discussed by an expert panel in an evaluation session.

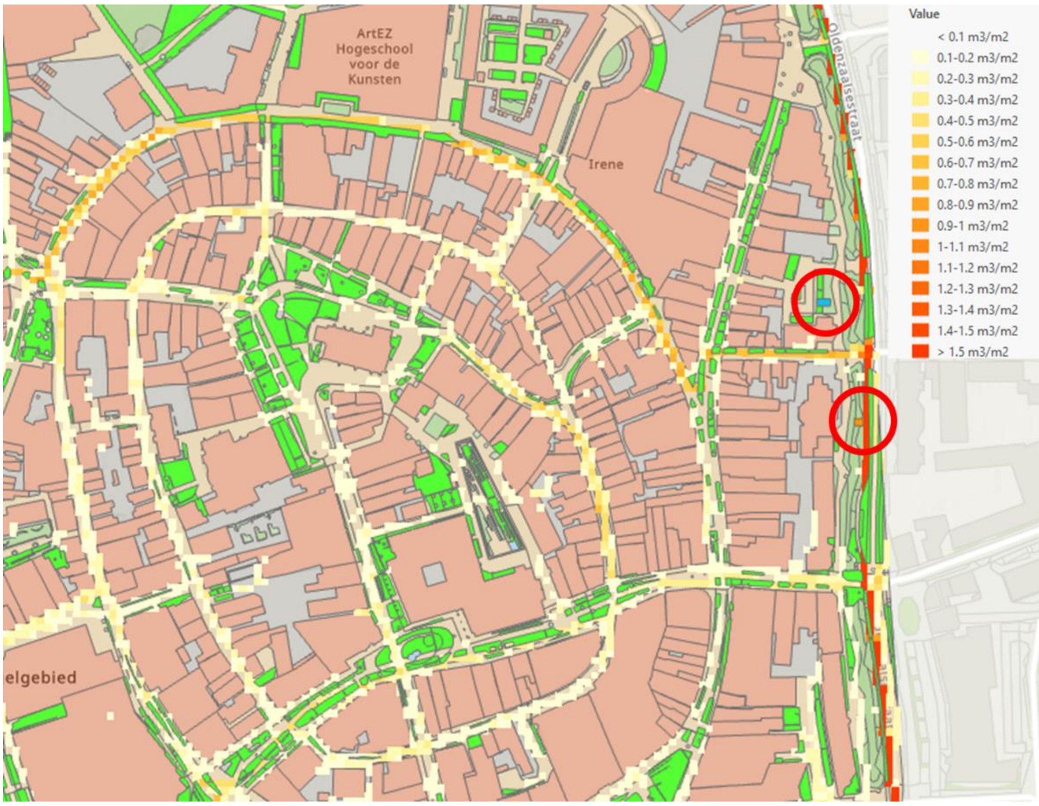


Figure 18: Visual results scenario 3

Table 9: Results scenarios 1-4

	Scenario 1: All-electric, prio. Electricity grid	Scenario 2: All-electric, prio. Climate adapatation	Sceneario 3: CDH, prio. Electricity grid/CDH	Scenario 4: CDH, prio. Climate adaptation
STC 1: Electrification parameters	15 LVDSs needed 4 available locations 11 cannot be placed	15 LVDSs needed 3 available locations 12 cannot be placed	3 LVDSs needed 4 available locations	3 LVDSs needed 3 available locations
STC2: Heat transition parameters	-	-	8 HDSs needed 2 available locations 6 cannot be placed	8 HDSs needed 2 available locations 6 cannot be placed
STC3: Climate adaptation parameters	13% vegetation increase (7→20%)	13% vegetation increase (7→20%)	13% vegetation increase (7→20%)	13% vegetation increase (7→20%)

11 EVALUATION

After the development period, a workshop with future users was organised to determine the value of the tool. The results from this session are discussed as follows. First, the acknowledgement of the problem space is discussed in 11.1. Then the three evaluation criteria on information quality (11.2), perceived usefulness and decision support satisfaction (11.3, summarised as 'usefulness') are discussed. Finally, a summary of the main insights from the session is presented in section 11.4.

11.1 PROBLEM SPACE

During the first part of the workshop, the problem space was presented to the participants together with the proposed solution of a tool. Four statements were used to measure the insight into the participants 'own', 'other' and 'combined' STC assignments (see questions 1-4 in Appendix V). This gave a good impression of the extent to which the participants acknowledged the defined problem.

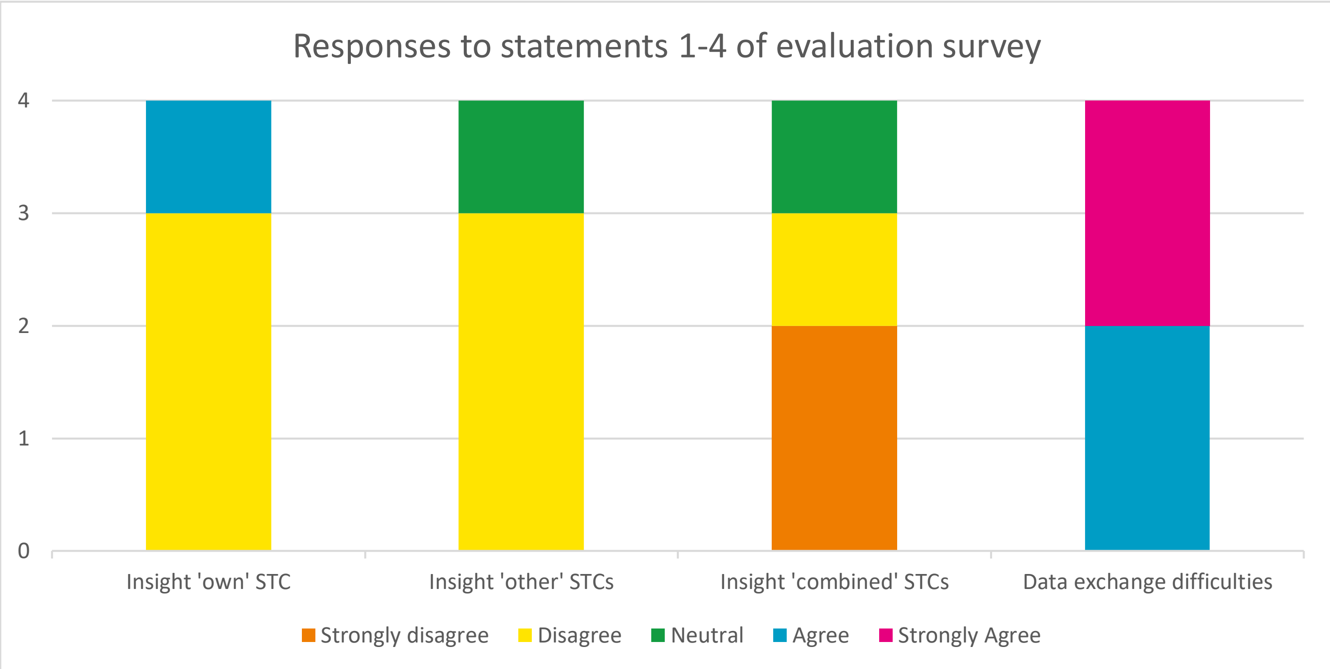


Figure 19: Responses to statements 1-4

As can be seen in Figure 19, three out of four respondents have given similar answers to the first three statements, and as expected, they recognise the problem of the lack of insight into the spatial claims of their 'own', 'other' and 'combined' assignment. A decrease in understanding can be seen towards 'other' and 'combined' STCs. Also, there is consensus on the fourth statement which confirms data exchange is a problem in gaining insight into the spatial claims of STCs. It can therefore be concluded that the identified problem has major relevance in a municipal context. A tool might be useful in increasing insight into the spatial claims of different STCs.

11.2 INFORMATION QUALITY

On the topic of information quality, two survey questions were asked on the extent and accuracy of the information. As can be seen in Figure 20, the information quality the tool presents is perceived as moderate to sufficient. During the discussion that took place, it turned out that participants could not decide between scenarios 1 and 3 based on the provided maps. Additional information such as the location of trees and their roots, other underground structures, maintenance data of underground structures and policy and functions of public space are desired.

Also, the participants question the realism of the STC models, e.g. "*The complexity of the electricity grid and how this is modelled should not be underestimated and is too simplified for the moment, this is a risk".*

However, the participants claim that the maps that were shown give insight into the complexity of all challenges that a municipality is facing e.g. *"It shows the complexity of the combined challenges very clearly"*. Especially the urgency of the challenges is displayed clearly. For example, the lack of available LVDS locations combined with the undesirability of most proposed locations (i.e. according to the participants) made participants realise that alternative locations are to be sourced on short notice.

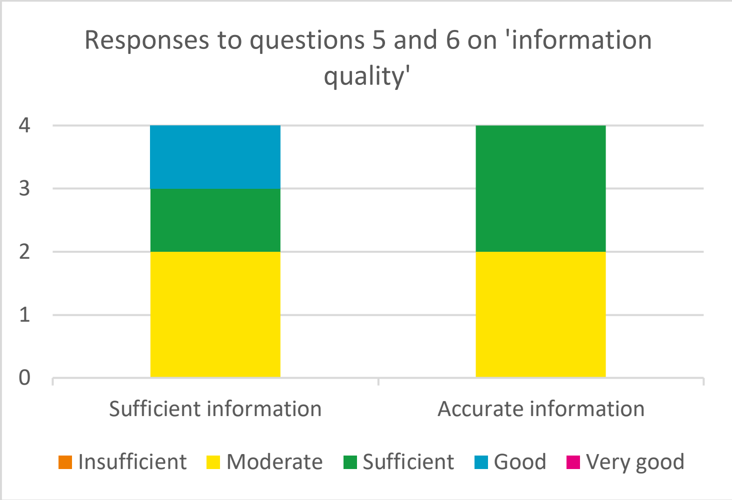


Figure 20: Responses to questions 5 and 6

Based on the comments it can be concluded that the tool does aid in understanding the combined problem and its complexities. It therefore 'ignites' the conversation among decision-makers. However, it cannot generate solutions that are realistic enough to implement. This is because of a lack of modelling accuracy and data in the model.

11.3 USEFULNESS

As can be seen in Figure 21, the participants claim moderate to sufficient usability for the tool and sufficient decision-support satisfaction. This can partly be explained by the reasons given above where participants desire more decision-making data into the tool and more accurate STC models.

However, during the discussion, questions were raised about the role the tool plays in a neighbourhood (re)development *"At what point in time do we want to use the information from the tool and what do we want to do with this data"* and "*How could such a system be further developed, deployed and maintained in the long term?".*

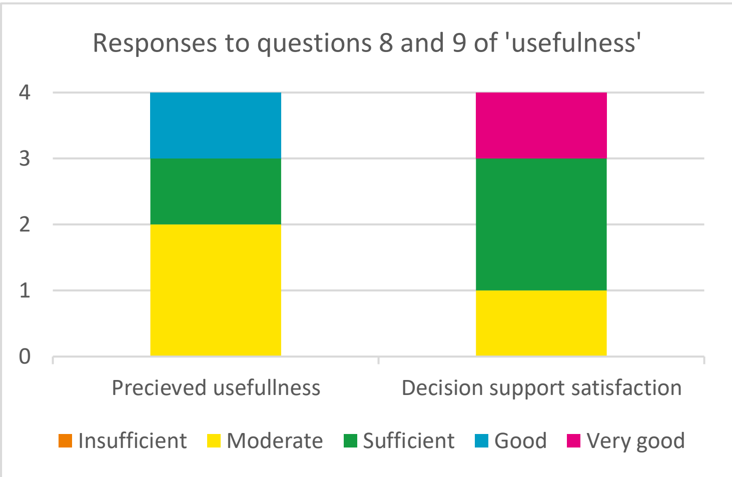


Figure 21: Responses to questions 8 and 9

From this can be concluded that not only does the tool itself need further development, but the entire process around it should be improved. Previously, there was little urge for integrated planning since space was available unlimited. However, with several STCs claiming public space at the same time, an integral planning process is required. A tool is to be incorporated and tailored to such a process.

11.4 INSIGHTS

The final survey questions repeated statements 1-3 to verify whether the insight had increased. The results are presented in Figure 22. It can in general be concluded that most participants feel an increased insight into their ‘own’, ‘other’ and ‘total’ assignments. The increase in the total assignment is slightly smaller than in the individual STCs. This increased insight can be summarised into three points;

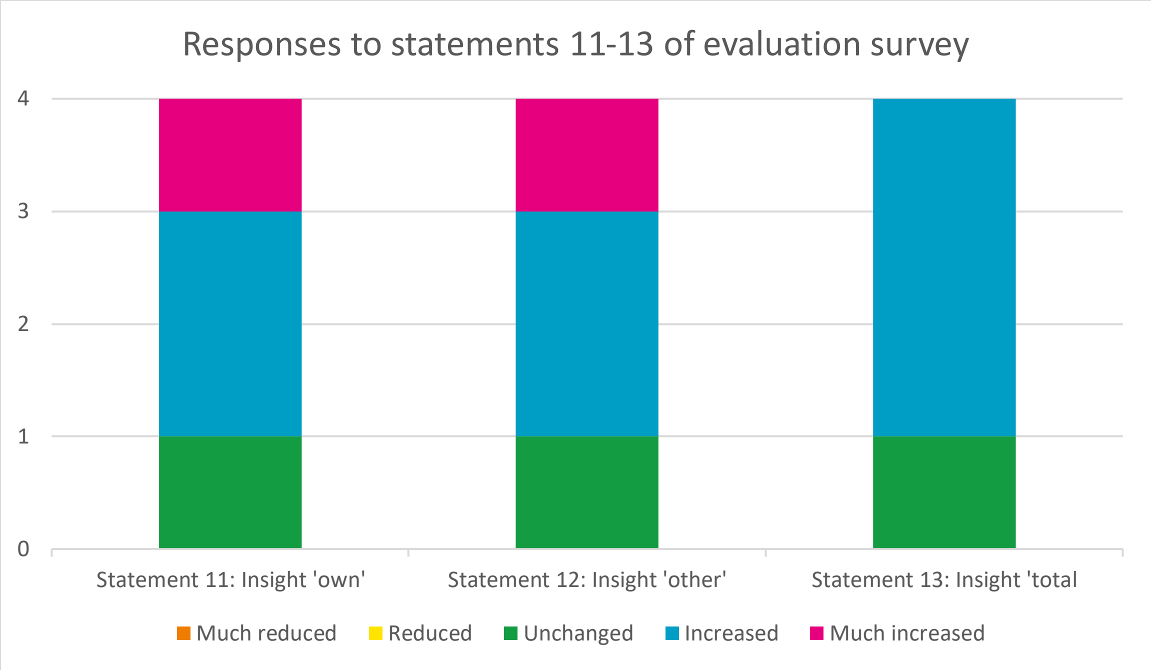


Figure 22: Responses to statements 11-13

First, the tool ‘ignites’ the conversation and displays the urgency of the challenges the municipality is facing. It is insightful for decision-makers to have a visual understanding of different alternatives/scenarios and their competing spatial claims in public space. Several discussions arose on the topic of placing overground structures and the impossibilities of the selected locations for LVDSs, HDSs and climate adaptation surfaces. The underground visualisations were received less useful because the colours in the UUS density rasters are unable to give a realistic view of the actual saturation below the surface and therefore cannot be used for decision-making.

During the discussion, the participants indicated that decisions on STC alternatives are most of the time not based on their spatial claim. For example, the decision for all-electric heating or a CDH network is currently based on factors such as degree of thermal insulation, type of housing (flat, detached house, etc.) and ownership (privately owned or by a housing corporation). The spatial impact of a solution is currently a ‘final check’. The second insight therefore is that the conditionality of the spatial claims is something to be assessed at the beginning of a development process. This means that earlier in the process it can be resorted to other or tailor-made solutions.

The final increased insight is that such a tool should be incorporated into a multi-disciplinary planning process. Also, then its place in this process is to be determined and it can be further developed (i.e. additional data, accurate models) according to the needs of its users. This insight also shows that the current way of working in planning and deploying measures by different municipal departments is no longer viable because of space scarcity. All measures to be deployed have a long lifespan and their locations should be chosen wisely i.e. their spatial impact will exist for many years. The social resources that are used, must be spent wisely.

12 DISCUSSION

The urge for an integral planning process is widely acknowledged by several authors (Kuchler et al., 2024; von der Tann et al., 2020). To date, this seemed to only have been done on a strategic level focusing not per se on spatiality but on factors as economy, society, environment, and culture (Zhao et al., 2016). Moreover, the importance of defining spatial interrelations of overground and underground infrastructure was stressed (2009), which has been attempted within this research. An observation on this end is that while the developed tool has a spatial perspective, the decision-makers require additional factors, as proposed in the literature, to be taken into account for decision-making. This means that the spatial claim of different STCs may become a factor to take into account beforehand and analyse the desirability of these claims afterwards, rather than planning objects based on several decision-making factors and finding 'space' for it afterwards, as is done currently.

The main practical contribution of this work is a case-based development and validation of a tool that models spatial claims of three socio-technical challenges. The results acknowledge the urge for an integrated planning process and show that a tool as created can be an aid within this process. Scientifically this contributes that the created tool increases the understanding of the complexity of the combined challenges and the spatial impact that they come with. Conversation is being 'ignited' and both the tool and the integral way of planning form a fruitful basis for further development.

The main limitation of this development process are the assumptions made for STC modelling. These assumptions found a basis in interviews, policy documentation and scientific literature, but they might be too generalised for certain situations. For example, the number of LVDSs placed is based on the number of households within a neighbourhood. However, the number of households may differ for every LVDS and the electricity grid operator does not model along the border of a 'neighbourhood'. Therefore, the amount of LVDSs within a neighbourhood may differ from the actual expected amount. Concretely, this means that the tool succeeds in presenting the urgency of the problem space but does not provide a sufficient basis for actual decision-making.

The tool can be improved on various ends, recommendations for doing this are:

- The incorporation of additional decision-making factors in the tool may enhance it towards a proper DSS. A suggestion on this end is to keep the spatial analysis as a basis but add a 'desirability score' to the different surfaces/locations found to present a more realistic result and add another dimension.
- Although a relation has been found by Bobylev between inhabitants per square kilometre and the average occupation rate in the underground, this data cannot be used to determine what value of occupation can be labelled as 'oversaturated' (Bobylev, 2016). Therefore, the colours and corresponding values to the UUS density metric should be worked out to give more meaningful insights. An insight into oversaturation would be useful for decision-making e.g. answering the question: "Does a city district heating grid 'fit' at location X?".
- The parameters used for modelling STCs could be modelled stochastically. By doing this the uncertainties within reality can be taken into account, this improves the current 'static' modelling rules, which possibly results in a more realistic model.

Besides a tool that is to be improved, an entire process should be developed to align all challenges that a municipality is facing and its associated interventions in physical space. This process should follow a strategic→tactic→operational path, in which a tool should be fitted in the most effective stage. For this research, the aim of this tool was on the tactic level, but it may be more desirable to adjust and use it on a longer (strategic) or shorter-term (operational).

13 CONCLUSIONS

There is a lack of insight among municipal decision-makers on how different socio-technical challenges (STCs) spatially affect public space. Let alone, that there is a holistic overview of the combination of challenges and their interfaces. Following a DSR methodology, a tool has been developed that models and visualises the spatial impact of interrelated STCs to support decision-makers in understanding and jointly prioritisation distinct scenarios. In the first step of the process, the problem has been defined which can be summarised as decision-makers having little insight into the spatial impact of their 'own', 'other' and 'total' STCs. After this, the solution, in the form of a tool integrating spatial data to gain more insight to better aid decision-makers in prioritising/deciding on different spatial solutions, was proposed and developed.

For this three STCs (electrification, heat transition and climate adaptation) have been selected and their spatial claims have been researched. To model these claims, several modelling rules have been defined. These claims have been defined above and below ground based on several parameters. Finally, four scenarios of different heating solutions were drafted and used as input for a workshop with representatives from every STC to evaluate the tool. From this, several conclusions can be drawn;

The decision-makers find the tool insightful in several ways. First, it shows them the complexity of the combined challenges by showing the (im)possibilities of different scenarios. Second, it is insightful for the decision-makers that spatial conditionality is an important factor to take into account for prioritising. All in all, the tool aids decision-makers in the joint understanding of the problem domain and 'ignites' the conversation on this topic.

Although the developed tool does gain insight into the complexity of the problem domain, it does not provide enough comprehensiveness to be used for prioritisation and thus decision-making. This has various reasons; first, it focuses on the spatial impact only whereas decisions are primarily based on other factors (social, political, technical). Therefore, these factors should be taken into account when alternatives are weighted. Second, the tool should be more accurate in its STC models. The possible locations for LVDSs and HDSs are currently carried out from a spatial perspective only, where it is desirable to combine this with the specific network perspectives of utility owners to model more realistic locations. Third, it is suggested that more data such as locations of trees and maintenance data on underground utilities is incorporated to gain a more holistic overview of a neighbourhood. Finally, the visualisation of the underground occupation is not sufficient for decision-making since no accurate saturation thresholds could be determined and no sense of depth is present in the model.

The organisation of an integrated planning process should be improved. A tool as developed should be embedded into this process. This means that regular alignment meetings should take place in which various modelled options are presented. Based on this, the municipal clients and other decision-makers can discuss the different outcomes. It would be valuable for such a process as this would be standardised for many municipalities using the same tool to learn from one another.

Although this integrated planning process is far from fully developed, the created tool has proven its positive effect on this process. Decision-makers have gained insight into the complexity of the problems and started discussing them integrally. This gives a fruitful basis for further expansion of this collaboration between departments within municipalities and utility owners.

REFERENCES

Admiraal, H., & Cornaro, A. (2016). Why underground space should be included in urban planning policy – And how this will enhance an urban underground future. *Tunnelling and Underground Space Technology, 55*, 214-220. doi:https://doi.org/10.1016/j.tust.2015.11.013

Alshibly, H. H. (2015). Investigating decision support system (DSS) success: A partial least squares structural equation modeling approach. *Journal of Business Studies Quarterly, 6*(4), 56.

Bobylev, N. (2009). Mainstreaming sustainable development into a city's Master plan: A case of Urban Underground Space use. *Land Use Policy, 26*(4), 1128-1137. doi:https://doi.org/10.1016/j.landusepol.2009.02.003

Bobylev, N. (2010). Underground space in the Alexanderplatz area, Berlin: Research into the quantification of urban underground space use. *Tunnelling and Underground Space Technology, 25*(5), 495-507. doi:https://doi.org/10.1016/j.tust.2010.02.013

Bobylev, N. (2016). Underground space as an urban indicator: Measuring use of subsurface. *Tunnelling and Underground Space Technology, 55*, 40-51. doi:https://doi.org/10.1016/j.tust.2015.10.024

Borenstein, D. (1998). Towards a practical method to validate decision support systems. *Decision Support Systems, 23*(3), 227-239. doi:https://doi.org/10.1016/S0167-9236(98)00046-3

Boukhayma, K., & ElManouar, A. (2015, 14-16 Dec. 2015). *Evaluating decision support systems.* Paper presented at the 2015 15th International Conference on Intelligent Systems Design and Applications (ISDA).

Brocke, J. v., Hevner, A., & Maedche, A. (2020). Introduction to Design Science Research. In (pp. 1-13).

Enexis. (2023). *Menukaart elektriciteitsstations*. Retrieved from

Frada Burstein, C. H. (2016). *Handbook on Decision Support Systems 2*.

Geels, F. W. (2010). Ontologies, socio-technical transitions (to sustainability), and the multi-level perspective. *Research Policy, 39*(4), 495-510. doi:https://doi.org/10.1016/j.respol.2010.01.022

Gemeente Enschede. (2021). *Water- en Klimaatadaptatieplan Gemeente Enschede 2022-2026*. Retrieved from

Gemeente Enschede. (2022). *Transitievisie Warmte Enschede*. Retrieved from

Gemeente Enschede. (2023). Toetstingkader Openbare Ruimte.

Gemeente Leiden. (2013). *Handboek kwaliteit openbare ruimte - DEEL II Inrichtingsprincipes*. Retrieved from

Generaal, T. K. d. S. (2024). Krapte op het elektriciteitsnet [Press release]. Retrieved from https://www.tweedekamer.nl/debat_en_vergadering/uitgelicht/krapte-op-het-elektriciteitsnet

Hehua Zhu, X. L., Fangle Peng, Xiaojun Li, Chunyan Liu. (2017). Development Strategy on Urban Underground Space Planning in China. *Strategic Study of Chinese Academy of Engineering, 19*(6), 12-17. doi:10.15302/j-sscae-2017.06.003

Kuchler, M., Craig-Thompson, A., Alofe, E., & Tryggvason, A. (2024). SubCity: Planning for a sustainable subsurface in Stockholm. *Tunnelling and Underground Space Technology, 144*. doi:10.1016/j.tust.2023.105545

Liander. (2024). 1 op de 3 straten open. Retrieved from https://www.liander.nl/stroom-vooruit/1-op-de-3-straten-open

Lin, Q., Zou, B., & Deng, Y. (2023). Evaluation of the Development Potential of Urban Underground Space Using Set Pair Analysis and the Comprehensive Weight Method. *Sustainability, 15*(7), 6255. Retrieved from https://www.mdpi.com/2071-1050/15/7/6255

Mir, S., & Quadri, S. (1970). Decision Support Systems: Concepts, Progress and Issues – A Review. In (pp. 373-399).

NEN. (2024). *NEN 7171-1 Ordening van ondergrondse netten – Deel 1: Criteria*. Retrieved from

Netbeheer Nederland. (2019). Basisdocument over energie-infrastructuur.

Netbeheer Nederland. (2024). Gezamenlijk actieprogramma tegen netcongestie op laagspanningsnet. Retrieved from https://www.netbeheernederland.nl/artikelen/nieuws/gezamenlijk-actieprogramma-tegen-netcongestie-op-laagspanningsnet
PDOK. (2024). *Dataset: Basisregistratie Grootschalige Topografie (BGT)*. Retrieved from: https://www.pdok.nl/introductie/-/article/basisregistratie-grootschalige-topografie-bgt-
Peng, J., & Peng, F.-L. (2018a). A GIS-based evaluation method of underground space resources for urban spatial planning: Part 1 methodology. *Tunnelling and Underground Space Technology, 74*, 82-95. doi:https://doi.org/10.1016/j.tust.2018.01.002
Peng, J., & Peng, F.-L. (2018b). A GIS-based evaluation method of underground space resources for urban spatial planning: Part 2 application. *Tunnelling and Underground Space Technology, 77*, 142-165. doi:https://doi.org/10.1016/j.tust.2018.03.013
Planbureau voor de Leefomgeving, CBS, & TNO. (2023). REFERENTIEVERBRUIK WARMTE WONINGEN.
Rijksoverheid. (2022). *Landelijk Actieprogramma Netcongestie*. Retrieved from
Roijackers, V. (2018). *nature-based solutions inspiratieboek*. Retrieved from
Volchko, Y., Norrman, J., Ericsson, L. O., Nilsson, K. L., Markstedt, A., Öberg, M., . . . Tengborg, P. (2020). Subsurface planning: Towards a common understanding of the subsurface as a multifunctional resource. *Land Use Policy, 90*, 104316. doi:https://doi.org/10.1016/j.landusepol.2019.104316
von der Tann, L., Sterling, R., Zhou, Y., & Metje, N. (2020). Systems approaches to urban underground space planning and management – A review. *Underground Space, 5*(2), 144-166. doi:https://doi.org/10.1016/j.undsp.2019.03.003
Wartmenetwerk. (2021). *NATIONAAL WARMTENET TRENDRAPPORT 2021*. Retrieved from
Wegverkeer, N. D. (2024). Overige attributen. Retrieved from https://docs.ndw.nu/handleidingen/nwb/nwb-basisstructuur/overige-attributen/
Yuan, H., He, Y., & Wu, Y. (2019). A comparative study on urban underground space planning system between China and Japan. *Sustainable Cities and Society, 48*, 101541. doi:https://doi.org/10.1016/j.scs.2019.101541
Zerhouny, M., Fadil, A., & Hakdaoui, M. (2018). Underground Space Utilization in the Urban Land-Use Planning of Casablanca (Morocco). *Land, 7*(4), 143. Retrieved from https://www.mdpi.com/2073-445X/7/4/143
Zhao, J.-W., Peng, F.-L., Wang, T.-Q., Zhang, X.-Y., & Jiang, B.-N. (2016). Advances in master planning of urban underground space (UUS) in China. *Tunnelling and Underground Space Technology, 55*, 290-307. doi:https://doi.org/10.1016/j.tust.2015.11.011

APPENDIX I – STATEMENT ON AI TOOLS USED DURING THE WORK

During the preparation of this work, the author used ChatGPT and Claude in order to assist in programming the created tool in Python in the form of coming up with (alternative) coding constructions and assisting when debugging. After using this tool/service, the author reviewed and edited the content as needed and take(s) full responsibility for the content of the work.

APPENDIX II – INTERVIEWS

ELECTRIFICATION

Date: 07-07-2024

Interviewee: Senior Partner Energy Transition, Enexis

Questions

The input for the conversation was based on several questions;

1. What type of elements are to be (re)placed?
2. What variations exist for those elements?
3. What is the spatial claim for every variation?
4. What parameters influence the necessity for a certain type/variation?
5. What boundary conditions exist for the placement of cables and transformation stations?
6. What dependencies exist with the other socio-technical challenges (heat transition, climate adaptation)?
7. How to estimate the electricity demand per neighbourhood?
8. Is the existing grid rerouted for new transformation stations?

Take-Away Messages

- For the first couple of questions, a document with technical requirements for the upgrading of the electricity grid was handed out, which covered questions 1-5 (Enexis, 2023).
- The additional/thicker cables needed for the upgrading of the electricity grid cannot be neglected and have a significant spatial claim on the underground.
- There will not be unlimited capacity on the grid, even when upgraded.
- Currently, houses are rated at 1kW, for the future this becomes 5kW.
- Currently, 500 houses could be fed by 1 mid/low voltage distribution station, in the future this becomes 50-150.
- Currently, the distance to a distribution station is 300 meters at max, for the future this reduces to 100 meters.
- The existing grid will preferably not be rerouted for additional transformation stations.
- Storage of energy has a spatial claim.

HEAT TRANSITION

Date: 10-6-2024

Interviewee: Project Manager, Ennatuurlijk

Questions

The input for the conversation was based on several questions;

1. What type of elements are to be (re)placed?
2. What variations exist for those elements?

3. What is the spatial claim for every variation?
4. What parameters influence the necessity for a certain type/variation?
5. What does the design process look like?
6. What boundary conditions exist for the placement of pipes and distribution stations?
7. What dependencies exist with the other socio-technical challenges (electrification, climate adaptation)?
8. Is there a way to abstractify city district heating networks?
9. How are different heating solutions addressed within one neighbourhood?
10. Are old gas pipes removed from the ground?

Take-Away Messages

- A distribution station roughly needs a space of 4x5 meters, depending on the number of households connected to it this is on average 300-500 for this size.
- A higher temperature drop from the heat source to the household connection means that smaller pipes can be used.
- When a low-temperature network is used, this means that there is a lower temperature drop and thus bigger pipes are required.
- Important parameters are the number of households connected within a neighbourhood and the heat demand of households, the latter can be assumed by the year of construction.
- When implementing a low-temperature heat grid this means that an additional heat pump is required meaning a higher pressure on the electricity grid.
- There are conflicting claims regarding the climate adaptation measures in terms of underground space e.g. pipes vs. trees.
- Currently, a grid design is made based on KLIC data.
- A heat grid is designed for the number of households that will connect to it.
- There is a referred to several norms that state the spatial integration of heat networks being CROW and NEN7171.
- Most important is the requirement that a heat pipe cannot lay close to a water pipe by the risk of legionella
- Most favourably, pipes are deployed in current vegetation areas
- For the determination of heat demand per type of house, standard figures from the PBL are used (Planbureau voor de Leefomgeving, CBS, & TNO, 2023).

CLIMATE ADAPTATION

Date 10-06-2024

Interviewee: Internal Client Physical Space, Gemeente Enschede

Questions

The input for the conversation was based on several questions;

1. What type of elements are to be (re)placed?
2. What variations exist for those elements?
3. What is the spatial claim for every variation?
4. What parameters influence the necessity for a certain type/variation?
5. What does the design process look like?
6. What boundary conditions exist for the placement of pipes and distribution stations?
7. What dependencies exist with the other socio-technical challenges (electrification, heat transition)?
8. Is there a way to abstractify climate adaptation measures?
9. How do you measure when there are ‘enough’ measures in a certain area?

Take-Away Messages

- A referral to the TOR (Toetsingskader Openbare Ruimte or Assessment Framework Physical Domain) is made to grasp the type of measures and their dimensions, requirements, etc (Gemeente Enschede, 2023).
- For all streets in the city, a quick scan is made of what challenge there is in terms of water storage and what the possibilities are in that certain area.
- Above-ground solutions such as wadis are always preferred above underground solutions such as larger sewer pipes.
- Mobility, vegetation and water solutions always have conflicting interests.

APPENDIX III – FEEDBACK SESSION

Table 10: Validation session participants

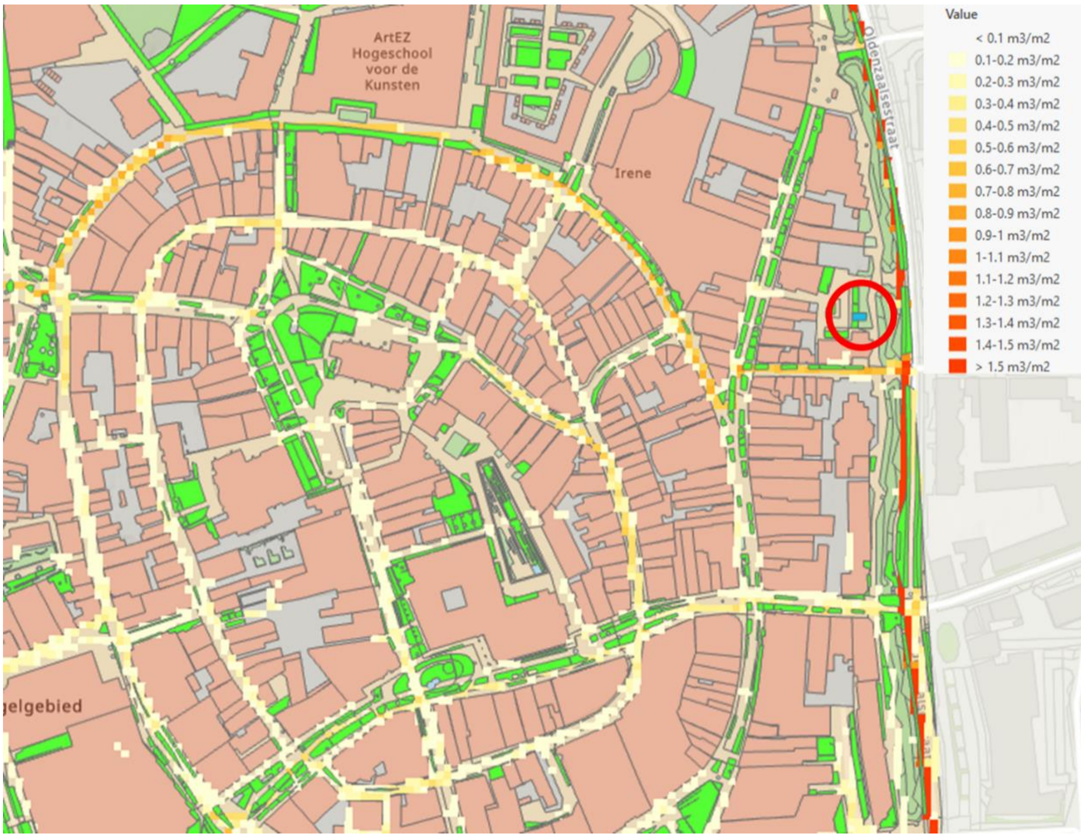
Function	STC expertise	Company
Urban Designer	General	Municipality of Enschede
Senior Consultant Urban Water	Climate adaptation	Municipality of Enschede
Senior Partner Energy transition	Electrification	Enexis (a major electricity grid operator in the Netherlands)
Project Manager	Electrification	Municipality of Enschede
Consultant Underground Infrastructure	Heat transition	Municipality of Enschede

Category	Expert Feedback	Change made?
General	**Underground modelling** • Take into account buffers around utilities, not only modelling diameters. • Take into account the volume claims of trees. • Sewer volumes cannot be estimated based on KLIC, suggested using the accurate municipal sewer data.	• Yes, based on NEN7171 • No, because of the lack of easily accessible data. • Yes
	Aim of the tool • The focus is on modelling preliminary strategies/designs, not creating fully worked-out designs. • It would be insightful to see impossibilities in public space so tailor-made solutions can be made quicker.	
	Visualisation • Household connections should be removed within the visualisation.	• Yes, by implementing a threshold of 0.1 m^3/m^2
	Additional STC • The placement of additional housing is an important STC left out. However, participants state that this is out of the scope of this research.	• No, defined as out of scope. This was also acknowledged by the participants.
Electrification	**Overground modelling** • Only modelling LVDSs is a valid assumption. • The placement of LVDSs on vegetation and parking areas is a valid assumption.	• Yes • Yes
	Underground modelling • Additional cables cannot be neglected and should be modelled.	• Yes, by increasing the current cable volume with a

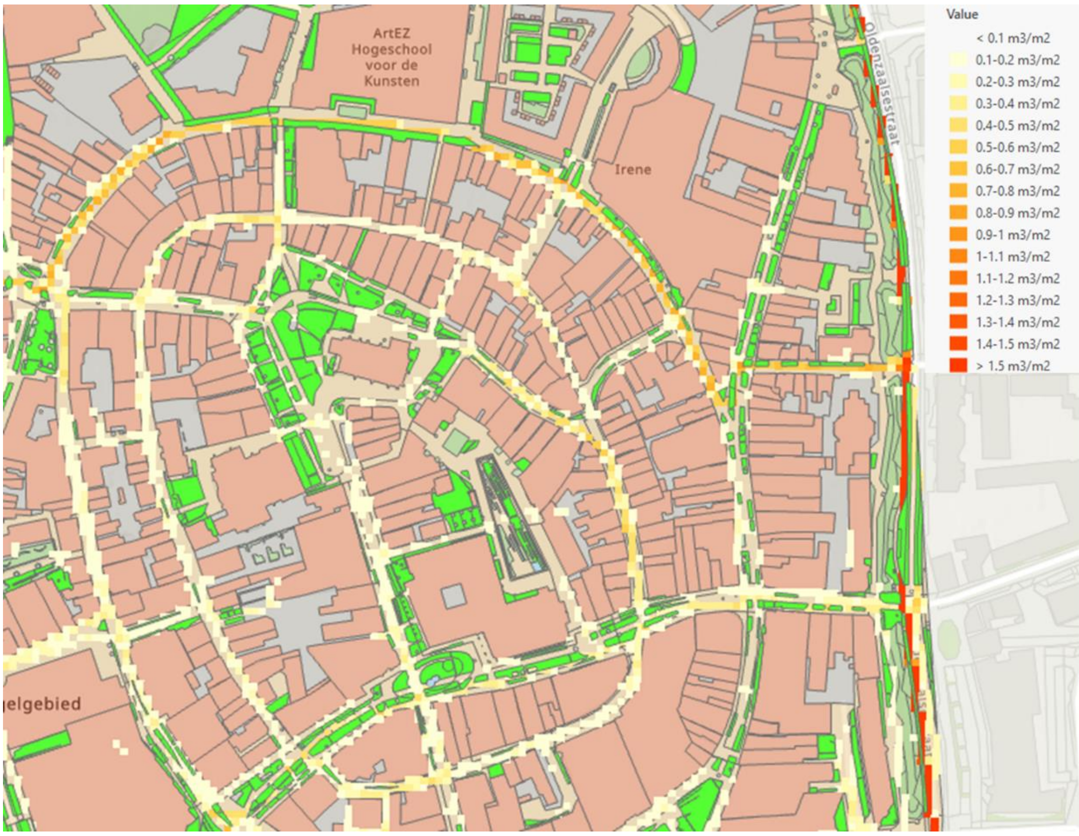
		factor (Netbeheer Nederland, 2019).
Heat transition	**Overground modelling** • It might be beneficial to see the spatial effect of more/fewer households participating in a CDH initiative.	• No, the way of modelling the CDH network using the current gas network complicated this too much.
	Underground modelling • Large transportation pipes should be taken into account because of their large claim.	• Yes, by modelling high-pressure gas pipes as heat transportation pipelines.
Climate adaptation	**Overground modelling** • When deploying climate adaptation measures, the municipality prefers above-ground solutions and the experts therefore suggest only modelling those. • It would be useful to implement climate stress tests to assign CASs more intelligently. • The experts suggest assuming that no cables and pipes are below climate adaptation measures except for sewer pipes. • Maximizing CASs is a valid approach.	• Yes, only above-ground solutions are modelled. • No, since the maximum amount of measures was modelled, no optimization was implemented. • Yes • Yes

APPENDIX IV – SCENARIO MAPS

SCENARIO 1: ALL-ELECTRIC PRIO. ELECTRICITY GRID

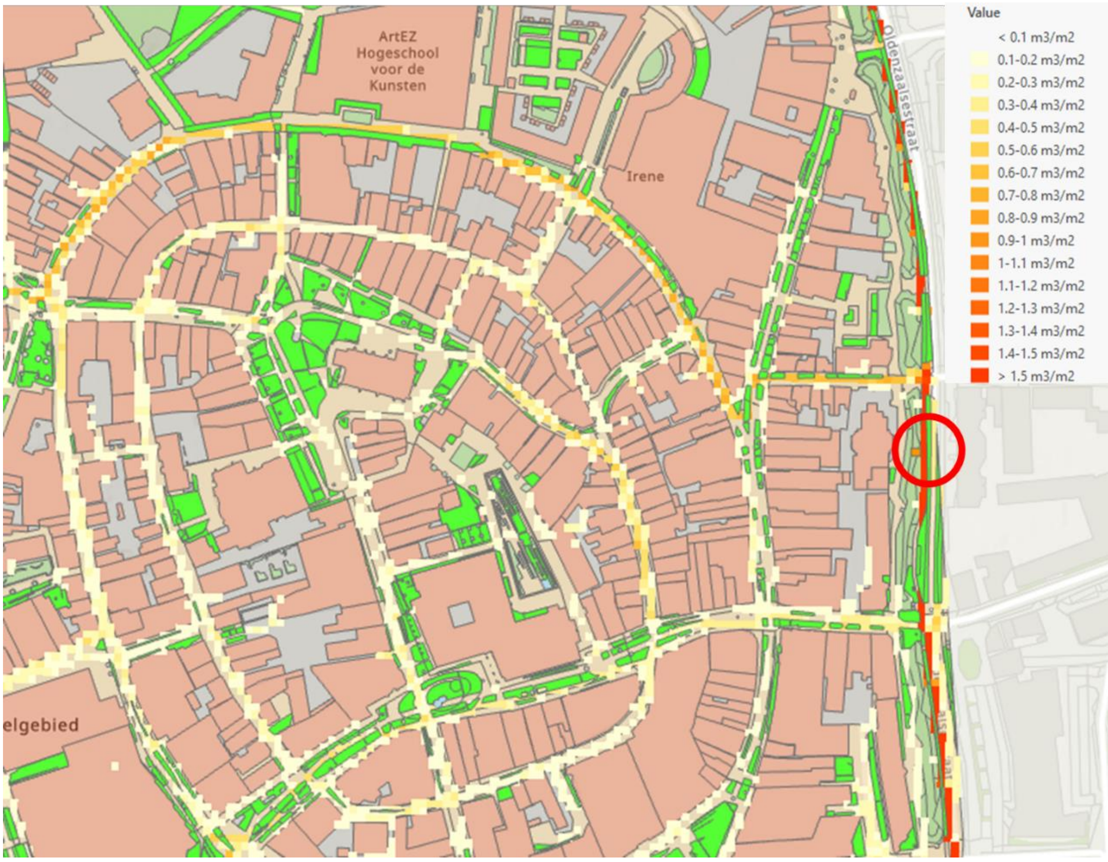


SCENARIO 2: ALL-ELECTRIC PRIO. CLIMATE ADAPTATION



SCENARIO 3: CDH PRIO. ELECTRICITY GRID/CDH

SCENARIO 4: CDH PRIO. CLIMATE ADAPTATION



APPENDIX V – EVALUATION SURVEY QUESTIONS

1. “I have insight into the spatial claim of my STC”
2. “I have insight into the spatial claim of other STCs”
3. “I have a holistic insight into the combined spatial claims of all STCs”
4. “The lack of data(exchange) makes it hard to gain a holistic insight into the spatial claims of all STCs”
5. To what extent is the presented information sufficient to make an informed decision between scenarios 1-4?
6. To what extent is the presented information accurate to make an informed decision between scenarios 1-4?
7. What information/factors are missing to make an informed decision?
8. To what extent is the tool usable for its designed purpose? (modelling and visualising the spatial impact of STCs to assist decision-makers in prioritising those challenges)
9. To what extent does the tool suit the needs for decision-making/prioritising on STCs?
10. What additional comments do you have regarding the tool?
11. After demonstrating the tool, how do you feel now about the following statement; “I have insight into the spatial claim of my STC”?
12. After demonstrating the tool, how do you feel now about the following statement; “I have insight into the spatial claim of other STCs”?
13. After demonstrating the tool, how do you feel now about the following statement; “I have a holistic insight into the combined spatial claims of all STCs”?

APPENDIX VI – PYTHON SCRIPT

```
#%% Part 1 - Initializing code
#import the packages that are needed to run the code
import arcpy
from arcpy.ia import *
import numpy
from arcpy.sa import *
import os
import numpy as np
import pandas as pd
import math
import time
import random

#start measuring time
start_time = time.time()

#%% Part 2 - Prepare and load data
#load project filepath and name
fp_project = r"C:\Users\frank\OneDrive - University of
Twente\Utwente\Afstuderen\2. Execution phase\Overdracht tool\Test"
project_name = "Test"

#load all requiered data into the tool
fp_KLIC_directory = r"C:\Users\frank\OneDrive - University of
Twente\Utwente\Afstuderen\2. Execution phase\Data\KLIC\220128108_1\.kkv_"
#KLIC folder containing .JSON files
fp_buurt = r"C:\Users\frank\OneDrive - University of
Twente\Utwente\Afstuderen\2. Execution
phase\Data\WijkBuurtkaart_2020_v3\Stroinkslanden_Noord_West.shp"
#neighbourhood polygon
fp_NWB = r"C:\Users\frank\OneDrive - University of
Twente\Utwente\Afstuderen\2. Execution phase\Data\01-12-
2023\Wegvakken\Wegvakken.shp" #road centerlines dataset
fp_Riool = r"C:\Users\frank\OneDrive - University of
Twente\Utwente\Afstuderen\2. Execution
phase\Data\Riolering\Kikker_lijnen_lijn.shp" #minicipal sewer dataset
(including dimensions)
fp_Parken = r"C:\Users\frank\OneDrive - University of
Twente\Utwente\Afstuderen\2. Execution
phase\Data\Parken\parkenkaart_3_3_300.shp" #municipal parks polygons

#load all BGT layers
fp_BGT_pa =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BGT/BGT_obj
ecttypen/FeatureServer/39"
fp_BGT_ot =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BGT/BGT_obj
ecttypen/FeatureServer/49"
fp_BGT_bt =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BGT/BGT_obj
ecttypen/FeatureServer/50"
fp_BGT_wa =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BGT/BGT_obj
ecttypen/FeatureServer/47"
fp_BGT_we =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BGT/BGT_obj
ecttypen/FeatureServer/45"
```

```
fp_BGT_hu =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BAG/BAGv3/F
eatureServer/0"
fp_BGT_ow =
"https://basisregistraties.arcgisonline.nl/arcgis/rest/services/BGT/BGT_obj
ecttypen/FeatureServer/48"
fp_BGT =
[fp_BGT_pa,fp_BGT_ot,fp_BGT_bt,fp_BGT_wa,fp_BGT_we,fp_BGT_hu,fp_BGT_ow]
BGT_layer_names = ["pand",
"onbegroeidterreindeel","begroeidterreindeel","waterdeel","wegdeel","huisnu
mmers","ondersteunendwaterdeel"]


#%% Part 3 - Define all parameters
gs = 4 #gridsize fishnet/raster

#trench widhts and areas (width*heigth) of different utility types
w_spanning = 0.3
A_spanning = w_spanning*0.07
w_data = 0.34
A_data = w_data*0.04
w_water = 0.36
A_water = w_water*0.11
w_gas = 0.36
A_gas = w_gas*0.1
# w_riool = 1
# A_riool = w_riool*0.45
w_else = 0.001
A_else = w_else*0

#STC1 parameters
ts_l = 7.58 #length transformation station
ts_b = 4.05 #width transformation station
ts_rd = 2 #distance to road
ts_qi = 167 #number of connections per station
ts_ex = 0.15 #fraction of additional stations
ts_dg = 10 #max. distance to grid
c_ac = 0.13 #fraction of additional cable claim

#STC2 parameters
ds_l = 5 #lengh of distribution station
ds_b = 4 #width of distribution station
ds_rd = 2 #distance to road (same taaken as electricity)
ds_qi = 400 #number of connections per station
ds_dg = 10 #max. distance to grid
A_warmte_trans = 1.6*0.4 #trench area of transportation pipes
A_warmte_dist = 1*0.1 #trench area of transportation pipes

#STC3 parameters
th_area_small = 10 #threshold for small areas to remove
th_area_large = 100 #overshoot percentage weird road shapes to remove
frc_width = {0:4.5, 1:4.5, 2:4.5, 3:4.5, 4:4.5, 5:4.5, 6:4.5,7:1.5} #road
width based on functional road class
ps_area = 12.5 #parking place area (2.5*5)

#%% Part 4 - Prepare and load databases
aprx =
arcpy.mp.ArcGISProject(os.path.join(fp_project,f"{project_name}.aprx"))
map = aprx.listMaps()[0]

#if the geodatabase already exists; delete it, otherwise; create it
```

```
if arcpy.Exists(os.path.join(fp_project,"Buurt.gdb")):
    arcpy.Delete_management(os.path.join(fp_project,"Buurt.gdb"))
Buurt_gdb = arcpy.CreateFileGDB_management(fp_project, "Buurt.gdb")
fp_Buurt_gdb = Buurt_gdb.getOutput(0)

if arcpy.Exists(os.path.join(fp_project,"Results.gdb")):
    arcpy.Delete_management(os.path.join(fp_project,"Results.gdb"))
Results_gdb = arcpy.CreateFileGDB_management(fp_project, "Results.gdb")
fp_Results_gdb = Results_gdb.getOutput(0)

if arcpy.Exists(os.path.join(fp_project,"BGT.gdb")):
    arcpy.Delete_management(os.path.join(fp_project,"BGT.gdb"))
BGT_gdb = arcpy.CreateFileGDB_management(fp_project, "BGT.gdb")
fp_BGT_gdb = BGT_gdb.getOutput(0)

for layer in map.listLayers():
    if layer.name == "BGT_glayer" and layer.isGroupLayer:
        map.removeLayer(layer)
        break
BGT_glayer = map.createGroupLayer("BGT_glayer")

if arcpy.Exists(os.path.join(fp_project,"KLIC.gdb")):
    arcpy.Delete_management(os.path.join(fp_project,"KLIC.gdb"))
KLIC_gdb = arcpy.CreateFileGDB_management(fp_project, "KLIC.gdb")
fp_KLIC_gdb = KLIC_gdb.getOutput(0)

for layer in map.listLayers():
    if layer.name == "KLIC_glayer" and layer.isGroupLayer:
        map.removeLayer(layer)
        break
KLIC_glayer = map.createGroupLayer("KLIC_glayer")

if arcpy.Exists(os.path.join(fp_project,"STC1.gdb")):
    arcpy.Delete_management(os.path.join(fp_project,"STC1.gdb"))
STC1_gdb = arcpy.CreateFileGDB_management(fp_project, "STC1.gdb")
fp_STC1_gdb = STC1_gdb.getOutput(0)

for layer in map.listLayers():
    if layer.name == "STC1_glayer" and layer.isGroupLayer:
        map.removeLayer(layer)
        break
STC1_glayer = map.createGroupLayer("STC1_glayer")

if arcpy.Exists(os.path.join(fp_project,"STC2.gdb")):
    arcpy.Delete_management(os.path.join(fp_project,"STC2.gdb"))
STC2_gdb = arcpy.CreateFileGDB_management(fp_project, "STC2.gdb")
fp_STC2_gdb = STC2_gdb.getOutput(0)

for layer in map.listLayers():
    if layer.name == "STC2_glayer" and layer.isGroupLayer:
        map.removeLayer(layer)
        break
STC2_glayer = map.createGroupLayer("STC2_glayer")

if arcpy.Exists(os.path.join(fp_project,"STC3.gdb")):
     arcpy.Delete_management(os.path.join(fp_project,"STC3.gdb"))
STC3_gdb = arcpy.CreateFileGDB_management(fp_project, "STC3.gdb")
fp_STC3_gdb = STC3_gdb.getOutput(0)

for layer in map.listLayers():
    if layer.name == "STC3_glayer" and layer.isGroupLayer:
```

```
        map.removeLayer(layer)
        break
STC3_glayer = map.createGroupLayer("STC3_glayer")

for layer in map.listLayers():
    if layer.name == "Volumes_glayer" and layer.isGroupLayer:
        map.removeLayer(layer)
        break
Volumes_glayer = map.createGroupLayer("Volumes_glayer")

la_2DfeaturesCombined =
arcpy.management.CreateFeatureclass(fp_Results_gdb,"FeaturesCombined2D",spa
tial_reference=arcpy.SpatialReference(28992))

#%% Part 5 - Visualisation functions
# Function: Visualise 2D
def visualise2D(input_layer,group_layer,R,G,B,T):
    layer = map.addDataFromPath(input_layer)
    symbology = layer.symbology
    symbology.renderer.symbol.color = {'RGB': [R, G, B, T]}
    layer.symbology = symbology
    map.addLayerToGroup(group_layer, layer)
    map.removeLayer(layer)

# Function: Visualise 3D
def visualise3D(input_layer,group_layer):
    layer = map.addDataFromPath(input_layer)
    symbology = layer.symbology
    symbology.updateColorizer('RasterClassifyColorizer')
    colorizer = symbology.colorizer
    colorizer.classificationMethod = "ManualInterval"
    colorizer.breakCount = 16

    colour_classes = [
        {"label": "< 0.1 m3/m2","value": 0.1, "color": [0, 0, 0, 0]},
        {"label": "0.1-0.2 m3/m2", "value": 0.2, "color": [255, 255, 215,
100]},
        {"label": "0.2-0.3 m3/m2", "value": 0.3, "color": [255, 250, 180,
100]},
        {"label": "0.3-0.4 m3/m2", "value": 0.4, "color": [255, 240, 145,
100]},
        {"label": "0.4-0.5 m3/m2", "value": 0.5, "color": [255, 225, 110,
100]},
        {"label": "0.5-0.6 m3/m2", "value": 0.6, "color": [255, 210, 80,
100]},
        {"label": "0.6-0.7 m3/m2", "value": 0.7, "color": [255, 195, 60,
100]},
        {"label": "0.7-0.8 m3/m2", "value": 0.8, "color": [255, 180, 45,
100]},
        {"label": "0.8-0.9 m3/m2", "value": 0.9, "color": [255, 165, 35,
100]},
        {"label": "0.9-1 m3/m2", "value": 1, "color": [255, 150, 25, 100]},
        {"label": "1-1.1 m3/m2", "value": 1.1, "color": [255, 135, 20,
100]},
        {"label": "1.1-1.2 m3/m2", "value": 1.2, "color": [255, 120, 15,
100]},
        {"label": "1.2-1.3 m3/m2", "value": 1.3, "color": [255, 105, 10,
100]},
        {"label": "1.3-1.4 m3/m2", "value": 1.4, "color": [255, 90, 5,
100]},
        {"label": "1.4-1.5 m3/m2", "value": 1.5, "color": [255, 75, 0,
```

```
100]},
        {"label": "> 1.5 m3/m2", "value": 1000, "color": [255, 60, 0,
100]},
    ]

    for row, column in enumerate(colour_classes):
        colorizer.classBreaks[row].label = column["label"]
        colorizer.classBreaks[row].upperBound = column["value"]
        colorizer.classBreaks[row].color = {'RGB': column["color"]}

    layer.symbology = symbology
    map.addLayerToGroup(group_layer, layer)
    map.removeLayer(layer)

# Function: Calculate 3D claim
def claim3D(name,gdb_name,claim):
    arcpy.env.workspace = fp_KLIC_gdb
    utility = arcpy.ListFeatureClasses(wild_card=f'*{name}*In')
    arcpy.management.Merge(utility,gdb_name)
    arcpy.CalculateField_management(gdb_name, "Segment_Volume",
                                    f'(!Shape_Length! *
{claim})/(math.pow({gs},2))')
    arr = arcpy.da.TableToNumPyArray(gdb_name,("FID_buurt_fishnet",
"Segment_Volume"))
    df = pd.DataFrame(arr, columns=["FID_buurt_fishnet", "Segment_Volume"])
    df["Segment_Volume"] = df["Segment_Volume"].str.replace(',', '.')
    df["Segment_Volume"] = df["Segment_Volume"].astype(float)
    return(df)

#%% Part 6 - Create 'as-is' map

#create buurt layer from buurt shapefile and convert into fishnet
la_buurt_org = arcpy.MakeFeatureLayer_management(fp_buurt, "buurt")
desc_buurt_org = arcpy.Describe(la_buurt_org)
la_buurt_fishnet =
arcpy.CreateFishnet_management(os.path.join(fp_Buurt_gdb,"buurt_fishnet"),
str(desc_buurt_org.extent.lowerLeft), str(desc_buurt_org.extent.XMin) + " "
+ str(desc_buurt_org.extent.YMax), gs, gs, None, None,
str(desc_buurt_org.extent.upperRight), "NO_LABELS", "#", "POLYGON")
desc_buurt_fish = arcpy.Describe(la_buurt_fishnet)

#create layers from BGT and clip them by the buurt layer
list = []
fms = arcpy.FieldMappings()
fm_functie = arcpy.FieldMap()

for layer, layer_name in zip(fp_BGT,BGT_layer_names):
    arcpy.MakeFeatureLayer_management(layer,layer_name)
    out_feature_class1 = os.path.join(fp_BGT_gdb,layer_name+"_Cl")
    arcpy.analysis.PairwiseClip(layer_name, la_buurt_org,
out_feature_class1)


    if layer_name == "huisnummers":
        #nothing needs to be done to this layer
        pass
    elif layer_name != "wegdeel":
        #copy different road functions to seperate features in feature
class "functie"
        arcpy.management.AddField(out_feature_class1, "functie", "TEXT")
```

```
        with arcpy.da.UpdateCursor(out_feature_class1, "functie") as
cursor:
            for row in cursor:
                row[0] = layer_name
                cursor.updateRow(row)
        list.append(out_feature_class1)
        fm_functie.addInputField(out_feature_class1,"functie")
    else:
        #create feature class with name of class to "functie"
        #this will be used to merge layers later on
        list.append(out_feature_class1)
        fm_functie.addInputField(out_feature_class1, "functie")

    if layer_name == "pand":
        visualise2D(out_feature_class1, BGT_glayer, 249, 65, 8, 30)
    elif layer_name == "onbegroeidterreindeel":
        visualise2D(out_feature_class1, BGT_glayer, 140, 132, 130, 30)
    elif layer_name == "begroeidterreindeel":
        visualise2D(out_feature_class1, BGT_glayer, 112, 188, 82, 30)
    elif layer_name == "ondersteunendwaterdeel":
        visualise2D(out_feature_class1, BGT_glayer, 125, 218, 88, 30)
    elif layer_name == "waterdeel":
        visualise2D(out_feature_class1, BGT_glayer, 41, 166, 240, 30)
    elif layer_name == "wegdeel":
        visualise2D(out_feature_class1, BGT_glayer, 232, 173, 78, 30)
    else:
        pass

name = fm_functie.outputField
name.name = "functie"
fms.addFieldMap(fm_functie)
out_feature_class2 = os.path.join(fp_BGT_gdb,"CombinedFeatures")
arcpy.management.Merge(list, out_feature_class2, fms, None)

#create NWB layer
la_NWB = arcpy.MakeFeatureLayer_management(fp_NWB, "NWB")
out_feature_class1 = os.path.join(fp_BGT_gdb,"NWB_Cl")
arcpy.analysis.PairwiseClip(la_NWB, la_buurt_org, out_feature_class1)

#create parken layer
la_Parken = arcpy.MakeFeatureLayer_management(fp_Parken, "Parken")
arcpy.analysis.PairwiseClip(la_Parken, la_buurt_org,
os.path.join(fp_Buurt_gdb,"Parken"))
arcpy.management.Dissolve(os.path.join(fp_Buurt_gdb,"Parken"),os.path.join(
fp_Buurt_gdb,"Parken_Di"), None, None, 'SINGLE_PART')

#create empty lists
KLIC_list_STC1 = []
KLIC_list_STC3 = []

#create riool layer
arcpy.management.DefineProjection(fp_Riool, arcpy.SpatialReference(28992))
la_Riool = arcpy.MakeFeatureLayer_management(fp_Riool, "Riool")
out_feature_class1 = os.path.join(fp_KLIC_gdb,"Riool_Cl")
arcpy.analysis.PairwiseClip(la_Riool, la_buurt_org, out_feature_class1)
out_feature_class2 = os.path.join(out_feature_class1 + "_In")
arcpy.analysis.PairwiseIntersect([out_feature_class1,
os.path.join(fp_Buurt_gdb, "buurt_fishnet")],out_feature_class2, None,
None, "LINE")
arcpy.CalculateField_management(out_feature_class2,
"Segment_Volume",f'(!Shape_Length! * (!Breedte!/1000+0.5)*(!Hoogte!/1000))
```

```
/ (math.pow({gs},2))')

out_feature_class3 = os.path.join(out_feature_class1 + "_Bu")
arcpy.CalculateField_management(out_feature_class1,
"Buffer_width",f'(!Breedte!/1000+0.5)')
arcpy.analysis.Buffer(out_feature_class1, out_feature_class3,
"Buffer_width")
KLIC_list_STC3.append(out_feature_class3)

layer_added = map.addDataFromPath(out_feature_class1)
map.addLayerToGroup(KLIC_glayer, layer_added)
map.removeLayer(layer_added)

#create KLIC layers
#only copy the files that contain "ligging" in their name and have a .json
extensionand put them in a list
KLIC_layer_names = [f for f in os.listdir(fp_KLIC_directory) if "ligging"
in f and f.endswith('.json') and "riool" not in f]

fp_KLIC = {}
df1 = pd.DataFrame()

#copy names of files to use as layer names
for row in range(0,len(KLIC_layer_names)):
    fp_KLIC[row] = os.path.join(fp_KLIC_directory,KLIC_layer_names[row])
    KLIC_layer_names[row] = os.path.splitext(KLIC_layer_names[row])[0]

#convert all layers that were selected above to a feature layer
for row in range(0,len(KLIC_layer_names)):
    arcpy.conversion.JSONToFeatures(fp_KLIC[row], os.path.join(fp_KLIC_gdb,
KLIC_layer_names[row]))
    arcpy.MakeFeatureLayer_management(os.path.join(fp_KLIC_gdb,
KLIC_layer_names[row]), KLIC_layer_names[row])
    desc = arcpy.Describe(os.path.join(fp_KLIC_gdb, KLIC_layer_names[row]))

    if desc.shapeType == 'Polyline':
        #only use the polyline layers so no polygon layers (sometimes they
are present in the KLIC database)
        #then: clip layer --> inersect layer (by fishnet) to determine
utility length per cell
        out_feature_class1 = os.path.join(fp_KLIC_gdb,
KLIC_layer_names[row] + "_Cl")
        arcpy.analysis.PairwiseClip(KLIC_layer_names[row], la_buurt_org,
out_feature_class1)
        layer_added = map.addDataFromPath(out_feature_class1)
        map.addLayerToGroup(KLIC_glayer,layer_added)
        map.removeLayer(layer_added)

        out_feature_class2 = os.path.join(out_feature_class1 + "_In")
        arcpy.analysis.PairwiseIntersect([out_feature_class1,
os.path.join(fp_Buurt_gdb, "buurt_fishnet")],out_feature_class2, None,
None, "LINE")
        KLIC_list_STC1.append(out_feature_class2)

        #this if-statement is used to select the correct surface of the
utility cable/pipe
        if "spanning" in KLIC_layer_names[row]:
            A = A_spanning
            w = w_spanning/2
        elif "data" in KLIC_layer_names[row]:
            A = A_data
```

```
            w = w_data/2
        elif "water" in KLIC_layer_names[row]:
            A = A_water
            w = w_water/2
        elif "gas" in KLIC_layer_names[row]:
            A = A_gas
            w = w_gas/2
        # elif "riool" in KLIC_layer_names[row]:
        #     A = A_riool
        #     w = w_riool/2
        else:
            A = A_else
            w = w_else/2

        #create buffer of width for all utilities
        out_feature_class3 = os.path.join(out_feature_class1 + "_Bu")
        arcpy.analysis.Buffer(out_feature_class1, out_feature_class3, f"{w}
Meters")
        KLIC_list_STC3.append(out_feature_class3)

        #create new field an calculate  m3/m2 per cell
        arcpy.CalculateField_management(out_feature_class2,
"Segment_Volume",f'(!Shape_Length! * {A})/(math.pow({gs},2))')

        #convert to numpy in order to add upp all values of the files
        arr = arcpy.da.TableToNumPyArray(out_feature_class2,
("FID_buurt_fishnet", "Segment_Volume"))
        df2 = pd.DataFrame(arr, columns=["FID_buurt_fishnet",
"Segment_Volume"])
        df1 = pd.concat([df1, df2])
        df2 = []
        arcpy.Delete_management(os.path.join(fp_KLIC_gdb,
KLIC_layer_names[row]))

    else:
        print(f"{KLIC_layer_names[row]} is not a correct layer")
        arcpy.Delete_management(os.path.join(fp_KLIC_gdb,
KLIC_layer_names[row]))


#combine with Riool data
arr = arcpy.da.TableToNumPyArray(os.path.join(fp_KLIC_gdb,"Riool_Cl" +
"_In"), ("FID_buurt_fishnet", "Segment_Volume"))
df2 = pd.DataFrame(arr, columns=["FID_buurt_fishnet", "Segment_Volume"])
df1 = pd.concat([df1, df2])
df2 = []

#some data conversions in order to create raster layer and align with
original map
df1["Segment_Volume"] = df1["Segment_Volume"].str.replace(',','.')
df1["Segment_Volume"] = df1["Segment_Volume"].astype(float)
df1 = df1.groupby("FID_buurt_fishnet").sum()
KLIC_volume =
pd.DataFrame(0,index=np.arange(1,int(arcpy.management.GetCount(os.path.join
(fp_Buurt_gdb,"buurt_fishnet")).getOutput(0))+1),columns=['empty'])
KLIC_volume = KLIC_volume.join(df1)
KLIC_volume = KLIC_volume.sum(axis=1,skipna=False)
KLIC_volume = KLIC_volume.to_numpy()
KLIC_volume = np.reshape(KLIC_volume, (int((desc_buurt_fish.extent.YMax -
desc_buurt_fish.extent.YMin) / gs), -1))
KLIC_volume = np.flipud(KLIC_volume)
```

```
KLIC_volume = arcpy.NumPyArrayToRaster(KLIC_volume,
arcpy.Point(desc_buurt_fish.extent.XMin, desc_buurt_fish.extent.YMin), gs,
gs)
KLIC_volume.save(os.path.join(fp_KLIC_gdb, "KLIC_volumes"))
KLIC_volume_Cl = ExtractByMask(KLIC_volume,la_buurt_org)
KLIC_volume_Cl.save(os.path.join(fp_KLIC_gdb, "KLIC_volumes"+"_Cl"))

visualise3D(os.path.join(fp_KLIC_gdb, "KLIC_volumes"+"_Cl"),Volumes_glayer)

# Calculations
# perform some statistical calculations
number_of_households =
int(arcpy.management.GetCount(os.path.join(fp_BGT_gdb,"huisnummers"+"_Cl"))
.getOutput(0))
print(f"There are {number_of_households} households")

#%% Part 7 - Function: STC1

def STC1():
    #calculate additional cable claim
    df2 = claim3D("spanning", os.path.join(fp_STC1_gdb, "STC1_Volume_1"),
A_spanning * c_ac)
    df2 = df2.groupby("FID_buurt_fishnet").sum()
    Cable_volume = pd.DataFrame(0, index=np.arange(1,
int(arcpy.management.GetCount(os.path.join(fp_Buurt_gdb,
"buurt_fishnet")).getOutput(0)) + 1), columns=['empty'])
    Cable_volume = Cable_volume.join(df2)
    Cable_volume = Cable_volume.sum(axis=1, skipna=False)
    Cable_volume = Cable_volume.to_numpy()
    Cable_volume = np.reshape(Cable_volume,
(int((desc_buurt_fish.extent.YMax - desc_buurt_fish.extent.YMin) / gs), -
1))
    Cable_volume = np.flipud(Cable_volume)
    Cable_volume =
arcpy.NumPyArrayToRaster(Cable_volume,arcpy.Point(desc_buurt_fish.extent.XM
in, desc_buurt_fish.extent.YMin), gs, gs)
    Cable_volume.save(os.path.join(fp_STC1_gdb, "STC1_Volume_2"))
    Cable_volume_Cl = ExtractByMask(Cable_volume, la_buurt_org)
    Cable_volume_Cl.save(os.path.join(fp_STC1_gdb, "STC1_Volume_final"))
    visualise3D(os.path.join(fp_STC1_gdb, "STC1_Volume_final"),
STC1_glayer)

    #determine transformation station locations
    ts_d = math.sqrt((math.pow(ts_l, 2) + math.pow(ts_b, 2)))
    ts_d2 = ts_d / 2

    #determine free spaces based on BGT and copy and merge those features
    #then they are negatively buffered with the half diagonal of the
transformation station
    la_CombinedFeatures =
arcpy.MakeFeatureLayer_management(os.path.join(fp_BGT_gdb,
"CombinedFeatures"))
    arcpy.management.SelectLayerByAttribute(la_CombinedFeatures,
'NEW_SELECTION',
                                            "functie IN
('parkeervlak','begroeidterreindeel')")
    arcpy.management.CopyFeatures(la_CombinedFeatures,
os.path.join(fp_STC1_gdb, "STC1_Locations_1"))
    arcpy.management.Dissolve(os.path.join(fp_STC1_gdb,
"STC1_Locations_1"),
                              os.path.join(fp_STC1_gdb,
```

```
"STC1_Locations_2"), None, None, 'SINGLE_PART')

    #remove memory layer (for prioritization)
    arcpy.analysis.Erase(os.path.join(fp_STC1_gdb, "STC1_Locations_2"),
la_2DfeaturesCombined,
                         os.path.join(fp_STC1_gdb, "STC1_Locations_3"))

    arcpy.analysis.Buffer(os.path.join(fp_STC1_gdb, "STC1_Locations_3"),
os.path.join(fp_STC1_gdb, "STC1_Locations_4"),
                          f"-{ts_d2} Meters")

    # determine road distance by selecting road types (rijbaan lokale weg
and inrit)
    # and dissolving and buffering them by the road distance and the half
diagonal of the station
    # then create intersection between two layers above; possible locations
and distance to road
    arcpy.management.SelectLayerByAttribute(la_CombinedFeatures,
'NEW_SELECTION',
                                            "functie IN ('rijbaan lokale
weg','inrit')")
    arcpy.management.CopyFeatures(la_CombinedFeatures,
os.path.join(fp_STC1_gdb, "STC1_Locations_5"))
    arcpy.management.Dissolve(os.path.join(fp_STC1_gdb,
"STC1_Locations_5"),
                              os.path.join(fp_STC1_gdb,
"STC1_Locations_6"), None, None, 'SINGLE_PART')
    arcpy.analysis.Buffer(os.path.join(fp_STC1_gdb, "STC1_Locations_6"),
os.path.join(fp_STC1_gdb, "STC1_Locations_7"),
                          f"{ts_rd + ts_d2} Meters", None, None, 'ALL')
    arcpy.analysis.Intersect(
        [os.path.join(fp_STC1_gdb, "STC1_Locations_4"),
os.path.join(fp_STC1_gdb, "STC1_Locations_7")],
        os.path.join(fp_STC1_gdb, "STC1_Locations_8"))

    #delete overlapping parts with cables by merging and buferring (by half
diagonal) of the polygon utility layers and delete this from the above
result
    arcpy.management.Merge(KLIC_list_STC1, os.path.join(fp_STC1_gdb,
"STC1_Locations_9"))
    arcpy.analysis.Buffer(os.path.join(fp_STC1_gdb, "STC1_Locations_9"),
os.path.join(fp_STC1_gdb, "STC1_Locations_10"),
                          f"{ts_d2} Meters", None, None, 'ALL')
    arcpy.analysis.Erase(os.path.join(fp_STC1_gdb, "STC1_Locations_8"),
os.path.join(fp_STC1_gdb, "STC1_Locations_10"),
                         os.path.join(fp_STC1_gdb, "STC1_Locations_11"))
    arcpy.management.FeatureToPoint(os.path.join(fp_STC1_gdb,
"STC1_Locations_11"),
                                    os.path.join(fp_STC1_gdb,
"STC1_Locations_12"), 'INSIDE')

    #find locations within a certain distance of the grid
    KLIC_list_STC3_electricity = [f for f in KLIC_list_STC3 if "spanning"
in f]
    arcpy.management.Merge(KLIC_list_STC3_electricity,
os.path.join(fp_STC1_gdb, "STC1_Locations_15"))
    la_STC1_locations =
arcpy.MakeFeatureLayer_management(os.path.join(fp_STC1_gdb,
"STC1_Locations_12"))
    arcpy.management.SelectLayerByLocation(la_STC1_locations,
"WITHIN_A_DISTANCE", os.path.join(fp_STC1_gdb, "STC1_Locations_15"),
```

```
f"{ts_dg} Meters",
                                                "NEW_SELECTION", "INVERT")
    arcpy.management.DeleteFeatures(la_STC1_locations)

    # Calculations
    number_of_STC1_additional = round(number_of_households * ts_ex / ts_qi)
    number_of_STC1_locations = int(
        arcpy.management.GetCount(os.path.join(fp_STC1_gdb,
"STC1_Locations_12")).getOutput(0))

    print(f"There are {number_of_STC1_additional} additional transformation
stations needed")
    print(f"There are {number_of_STC1_locations} possible locations for
additional transformation stations")

    if number_of_STC1_additional > number_of_STC1_locations:
        print(f"{number_of_STC1_additional-number_of_STC1_locations}
additional transformation stations could not be placed")
        number_of_STC1_additional = number_of_STC1_locations
    else:
        pass

    # Select random locations
    if number_of_STC1_additional == 1:
        STC1_locations_ids =
random.sample(range(number_of_STC1_locations),number_of_STC1_additional)
    else:
        arcpy.analysis.Near(os.path.join(fp_BGT_gdb, "huisnummers" +
"_Cl"),
                            os.path.join(fp_STC1_gdb, "STC1_Locations_12"))

arcpy.stats.SpatiallyConstrainedMultivariateClustering(os.path.join(fp_BGT_
gdb, "huisnummers" + "_Cl"),

os.path.join(fp_STC1_gdb, "STC1_Locations_13"),
                                                               'NEAR_FID',
                                                               None, None,
None, None, number_of_STC1_additional)

        arr = arcpy.da.FeatureClassToNumPyArray(os.path.join(fp_STC1_gdb,
"STC1_Locations_13"),
                                                 ["NEAR_FID", "CLUSTER_ID"])
        arr = pd.DataFrame(arr, columns=["NEAR_FID", "CLUSTER_ID"])
        STC1_locations_ids = arr.groupby("CLUSTER_ID").sample()
        STC1_locations_ids = STC1_locations_ids["NEAR_FID"]

        while STC1_locations_ids.is_unique == False:
            STC1_locations_ids = arr.groupby("CLUSTER_ID").sample()
            STC1_locations_ids = STC1_locations_ids["NEAR_FID"]

        STC1_locations_ids = STC1_locations_ids.to_list()

    la_STC1_locations =
arcpy.MakeFeatureLayer_management(os.path.join(fp_STC1_gdb,
"STC1_Locations_12"))
    for row in STC1_locations_ids:
        arcpy.management.SelectLayerByAttribute(la_STC1_locations,
'ADD_TO_SELECTION', f"OBJECTID = {row}")
    arcpy.management.CopyFeatures(la_STC1_locations,
os.path.join(fp_STC1_gdb, "STC1_Locations_14"))
```

```
    #create rectangles to represent LVDS
    polygons = []
    sr = arcpy.Describe(os.path.join(fp_STC1_gdb,
"STC1_Locations_14")).spatialReference
    with arcpy.da.SearchCursor(os.path.join(fp_STC1_gdb,
"STC1_Locations_14"), "SHAPE@",
                               spatial_reference=sr) as cursor:
        for row in cursor:
            polygon = arcpy.Polygon(arcpy.Array(
                [arcpy.Point(row[0].centroid.X - ts_l / 2,
row[0].centroid.Y + ts_b / 2),
                 arcpy.Point(row[0].centroid.X - ts_l / 2,
row[0].centroid.Y - ts_b / 2),
                 arcpy.Point(row[0].centroid.X + ts_l / 2,
row[0].centroid.Y - ts_b / 2),
                 arcpy.Point(row[0].centroid.X + ts_l / 2,
row[0].centroid.Y + ts_b / 2)]), sr)
            polygons.append(polygon)

    arcpy.CopyFeatures_management(polygons, os.path.join(fp_STC1_gdb,
"STC1_Locations_final"))

    layer_added = map.addDataFromPath(os.path.join(fp_STC1_gdb,
"STC1_Locations_final"))
    symbology = layer_added.symbology
    symbology.renderer.symbol.color = {'RGB': [26, 191, 232, 100]}
    layer_added.symbology = symbology
    map.addLayerToGroup(STC1_glayer, layer_added)
    map.removeLayer(layer_added)

    arcpy.management.Append(os.path.join(fp_STC1_gdb,
"STC1_Locations_final"), la_2DfeaturesCombined)


#%% Part 8 - Function: STC2
def STC2():
    #underground claim of pipes
    df2 = pd.concat([claim3D("gasLageDruk", os.path.join(fp_STC2_gdb,
"STC2_Volume_1_1"), A_warmte_dist),
                     claim3D("gasHogeDruk", os.path.join(fp_STC2_gdb,
"STC2_Volume_1_2"), A_warmte_trans)])

    df2 = df2.groupby("FID_buurt_fishnet").sum()
    CDH_volume = pd.DataFrame(0, index=np.arange(1,
int(arcpy.management.GetCount(
        os.path.join(fp_Buurt_gdb, "buurt_fishnet")).getOutput(0)) + 1),
columns=['empty'])
    CDH_volume = CDH_volume.join(df2)
    CDH_volume = CDH_volume.sum(axis=1, skipna=False)
    CDH_volume = CDH_volume.to_numpy()
    CDH_volume = np.reshape(CDH_volume, (int((desc_buurt_fish.extent.YMax -
desc_buurt_fish.extent.YMin) / gs), -1))
    CDH_volume = np.flipud(CDH_volume)
    CDH_volume = arcpy.NumPyArrayToRaster(CDH_volume,

arcpy.Point(desc_buurt_fish.extent.XMin, desc_buurt_fish.extent.YMin), gs,
gs)
    CDH_volume.save(os.path.join(fp_STC2_gdb, "STC2_Volume_2"))
    CDH_volume_Cl = ExtractByMask(CDH_volume, la_buurt_org)
    CDH_volume_Cl.save(os.path.join(fp_STC2_gdb, "STC2_Volume_final"))
```

```
    visualise3D(os.path.join(fp_STC2_gdb, "STC2_Volume_final"),STC2_glayer)


    #placement of distribution stations
    ds_d = math.sqrt((math.pow(ds_l, 2) + math.pow(ds_b, 2)))
    ds_d2 = ds_d / 2

    # determine free spaces based on BGT for now, it was chosen to set
'parkeervlak' and 'green area' as possible locations for these housing
    # those features are copied and merged into full polygons using
dissolve, then they are negatively buffered with the half diagonal of the
transformation station
    la_CombinedFeatures =
arcpy.MakeFeatureLayer_management(os.path.join(fp_BGT_gdb,
"CombinedFeatures"))
    arcpy.management.SelectLayerByAttribute(la_CombinedFeatures,
'NEW_SELECTION',
                                            "functie IN
('parkeervlak','begroeidterreindeel')")
    arcpy.management.CopyFeatures(la_CombinedFeatures,
os.path.join(fp_STC2_gdb, "STC2_Locations_1"))
    arcpy.management.Dissolve(os.path.join(fp_STC2_gdb,
"STC2_Locations_1"),
                              os.path.join(fp_STC2_gdb,
"STC2_Locations_2"), None, None, 'SINGLE_PART')

    arcpy.analysis.Erase(os.path.join(fp_STC2_gdb, "STC2_Locations_2"),
la_2DfeaturesCombined,
                         os.path.join(fp_STC2_gdb, "STC2_Locations_3"))

    arcpy.analysis.Buffer(os.path.join(fp_STC2_gdb, "STC2_Locations_3"),
os.path.join(fp_STC2_gdb, "STC2_Locations_4"),
                          f"-{ds_d2} Meters")

    # determine road distance by selecting road types (rijbaan lokale weg
and inrit)
    # and dissolving and buffering them by the road distance and the half
diagonal of the station
    # then create intersection between two layers above; possible locations
and distance to road
    arcpy.management.SelectLayerByAttribute(la_CombinedFeatures,
'NEW_SELECTION',
                                            "functie IN ('rijbaan lokale
weg','inrit')")
    arcpy.management.CopyFeatures(la_CombinedFeatures,
os.path.join(fp_STC2_gdb, "STC2_Locations_5"))
    arcpy.management.Dissolve(os.path.join(fp_STC2_gdb,
"STC2_Locations_5"),
                              os.path.join(fp_STC2_gdb,
"STC2_Locations_6"), None, None, 'SINGLE_PART')
    arcpy.analysis.Buffer(os.path.join(fp_STC2_gdb, "STC2_Locations_6"),
os.path.join(fp_STC2_gdb, "STC2_Locations_7"),
                          f"{ds_rd + ds_d2} Meters", None, None, 'ALL')
    arcpy.analysis.Intersect(
        [os.path.join(fp_STC2_gdb, "STC2_Locations_4"),
os.path.join(fp_STC2_gdb, "STC2_Locations_7")],
        os.path.join(fp_STC2_gdb, "STC2_Locations_8"))

    # delete overlapping parts with cables by merging and buferring (by
half diagonal) of the polygon utility layers and delete this from the above
result
```

```
    arcpy.management.Merge(KLIC_list_STC1, os.path.join(fp_STC2_gdb, 
"STC2_Locations_9"))
    arcpy.analysis.Buffer(os.path.join(fp_STC2_gdb, "STC2_Locations_9"), 
os.path.join(fp_STC2_gdb, "STC2_Locations_10"),
                          f"{ds_d2} Meters", None, None, 'ALL')
    arcpy.analysis.Erase(os.path.join(fp_STC2_gdb, "STC2_Locations_8"), 
os.path.join(fp_STC2_gdb, "STC2_Locations_10"),
                         os.path.join(fp_STC2_gdb, "STC2_Locations_11"))
    arcpy.management.FeatureToPoint(os.path.join(fp_STC2_gdb, 
"STC2_Locations_11"),
                                    os.path.join(fp_STC2_gdb, 
"STC2_Locations_12"), 'INSIDE')

    KLIC_list_STC3_heatgrid = [f for f in KLIC_list_STC3 if "gas" in f]
    arcpy.management.Merge(KLIC_list_STC3_heatgrid, 
os.path.join(fp_STC2_gdb, "STC2_Locations_15"))
    la_STC2_locations = 
arcpy.MakeFeatureLayer_management(os.path.join(fp_STC2_gdb, 
"STC2_Locations_12"))
    arcpy.management.SelectLayerByLocation(la_STC2_locations, 
"WITHIN_A_DISTANCE", os.path.join(fp_STC2_gdb, "STC2_Locations_15"), 
f"{ds_dg} Meters",
                                            "NEW_SELECTION", "INVERT")
    arcpy.management.DeleteFeatures(la_STC2_locations)

    # Calculations
    # more calculations on the STC1 subject
    number_of_STC2_additional = round(number_of_households / ds_qi)
    number_of_STC2_locations = int(
        arcpy.management.GetCount(os.path.join(fp_STC2_gdb, 
"STC2_Locations_12")).getOutput(0))

    print(f"There are {number_of_STC2_additional} distribution stations 
needed")
    print(f"There are {number_of_STC2_locations} possible locations for 
distribution stations")

    if number_of_STC2_additional > number_of_STC2_locations:
        print(f"{number_of_STC2_additional-number_of_STC2_locations} 
distribution stations could not be placed")
        number_of_STC2_additional = number_of_STC2_locations
    else:
        pass

    # Select random locations
    if number_of_STC2_additional == 1:
        STC2_locations_ids = random.sample(range(number_of_STC2_locations), 
number_of_STC2_additional)
    else:
        arcpy.analysis.Near(os.path.join(fp_BGT_gdb, "huisnummers" + 
"_Cl"),
                            os.path.join(fp_STC2_gdb, "STC2_Locations_12"))

arcpy.stats.SpatiallyConstrainedMultivariateClustering(os.path.join(fp_BGT_
gdb, "huisnummers" + "_Cl"),

os.path.join(fp_STC2_gdb, "STC2_Locations_13"),
                                                                'NEAR_FID', 
                                                                None, None, 
None, None, number_of_STC2_additional)
```

```
        arr = arcpy.da.FeatureClassToNumPyArray(os.path.join(fp_STC2_gdb,
"STC2_Locations_13"),
                                                ["NEAR_FID", "CLUSTER_ID"])
        arr = pd.DataFrame(arr, columns=["NEAR_FID", "CLUSTER_ID"])
        STC2_locations_ids = arr.groupby("CLUSTER_ID").sample()
        STC2_locations_ids = STC2_locations_ids["NEAR_FID"]

        while STC2_locations_ids.is_unique == False:
            STC2_locations_ids = arr.groupby("CLUSTER_ID").sample()
            STC2_locations_ids = STC2_locations_ids["NEAR_FID"]

        STC2_locations_ids = STC2_locations_ids.to_list()


    la_STC2_locations =
arcpy.MakeFeatureLayer_management(os.path.join(fp_STC2_gdb,
"STC2_Locations_12"))
    for row in STC2_locations_ids:
        arcpy.management.SelectLayerByAttribute(la_STC2_locations,
'ADD_TO_SELECTION', f"OBJECTID = {row}")

    arcpy.management.CopyFeatures(la_STC2_locations,
os.path.join(fp_STC2_gdb, "STC2_Locations_14"))

    polygons = []
    sr = arcpy.Describe(os.path.join(fp_STC2_gdb,
"STC2_Locations_14")).spatialReference
    with arcpy.da.SearchCursor(os.path.join(fp_STC2_gdb,
"STC2_Locations_14"), "SHAPE@",
                               spatial_reference=sr) as cursor:
        for row in cursor:
            polygon = arcpy.Polygon(arcpy.Array(
                [arcpy.Point(row[0].centroid.X - ds_l / 2,
row[0].centroid.Y + ds_b / 2),
                 arcpy.Point(row[0].centroid.X - ds_l / 2,
row[0].centroid.Y - ds_b / 2),
                 arcpy.Point(row[0].centroid.X + ds_l / 2,
row[0].centroid.Y - ds_b / 2),
                 arcpy.Point(row[0].centroid.X + ds_l / 2,
row[0].centroid.Y + ds_b / 2)]), sr)
            polygons.append(polygon)

    arcpy.CopyFeatures_management(polygons, os.path.join(fp_STC2_gdb,
"STC2_Locations_final"))

    layer_added = map.addDataFromPath(os.path.join(fp_STC2_gdb,
"STC2_Locations_final"))
    symbology = layer_added.symbology
    symbology.renderer.symbol.color = {'RGB': [255, 153, 0, 100]}
    layer_added.symbology = symbology
    map.addLayerToGroup(STC2_glayer, layer_added)
    map.removeLayer(layer_added)

    arcpy.management.Append(os.path.join(fp_STC2_gdb,
"STC2_Locations_final"), la_2DfeaturesCombined)


#%% Part 9 - Function: STC3
def STC3():
    #copy road centerlines and buffer them by minimal road distance
    arcpy.management.CopyFeatures(os.path.join(fp_BGT_gdb, "NWB_Cl"),
```

```
os.path.join(fp_STC3_gdb, "STC3_Locations_1"))
    arcpy.management.AddField(os.path.join(fp_STC3_gdb,
"STC3_Locations_1"), "buffer_dist", 'TEXT')

    with arcpy.da.UpdateCursor(os.path.join(fp_STC3_gdb,
"STC3_Locations_1"), ["FRC", "buffer_dist"]) as cursor:
        for row in cursor:
            frc_value = int(row[0])
            row[1] = f"{frc_width[frc_value]} Meters"
            cursor.updateRow(row)

    arcpy.analysis.Buffer(os.path.join(fp_STC3_gdb, "STC3_Locations_1"),
os.path.join(fp_STC3_gdb, "STC3_Locations_2"),
                          "buffer_dist")

    arcpy.management.CopyFeatures(os.path.join(fp_BGT_gdb, "wegdeel_Cl"),
os.path.join(fp_STC3_gdb, "STC3_Locations_3"))
    arcpy.analysis.Erase(os.path.join(fp_STC3_gdb, "STC3_Locations_3"),
os.path.join(fp_STC3_gdb, "STC3_Locations_2"),
                         os.path.join(fp_STC3_gdb, "STC3_Locations_4"))


    #remove all cables and pipes below remaining polygons (except for
sewer)
    KLIC_list_STC3_nosewer = [f for f in KLIC_list_STC3 if "Riool" not in
f]

    arcpy.management.Merge(KLIC_list_STC3_nosewer,
os.path.join(fp_STC3_gdb, "STC3_Locations_5"))
    arcpy.management.Dissolve(os.path.join(fp_STC3_gdb,
"STC3_Locations_5"),
                              os.path.join(fp_STC3_gdb,
"STC3_Locations_6"))
    arcpy.analysis.Erase(os.path.join(fp_STC3_gdb, "STC3_Locations_4"),
os.path.join(fp_STC3_gdb, "STC3_Locations_6"),
                         os.path.join(fp_STC3_gdb, "STC3_Locations_7"))
    arcpy.management.MultipartToSinglepart(os.path.join(fp_STC3_gdb,
"STC3_Locations_7"),
                                           os.path.join(fp_STC3_gdb,
"STC3_Locations_8"))
    la_STC3_locations_8 =
arcpy.MakeFeatureLayer_management(os.path.join(fp_STC3_gdb,
"STC3_Locations_8"))

    # delete features that have too small area
    arcpy.management.SelectLayerByAttribute(la_STC3_locations_8,
'NEW_SELECTION', f"Shape_Area <= {th_area_small}")
    arcpy.management.DeleteFeatures(la_STC3_locations_8)

    # delete features that have an odd shape and therefore cannot determine
width
    arcpy.management.MinimumBoundingGeometry(la_STC3_locations_8,
os.path.join(fp_STC3_gdb, "STC3_Locations_9"),

mbg_fields_option='MBG_FIELDS')
    arcpy.env.workspace = fp_STC3_gdb
    arcpy.management.JoinField("STC3_Locations_8", "OBJECTID",
"STC3_Locations_9", "ORIG_FID",
                               ["MBG_Width", "MBG_Length", "Shape_Area"])
    arcpy.CalculateField_management(os.path.join(fp_STC3_gdb,
"STC3_Locations_8"), "perc_overshoot",
```

```
                                    '((!Shape_Area_1!/!Shape_Area!)-
1)*100', field_type='Float')
    arcpy.management.SelectLayerByAttribute(la_STC3_locations_8,
'NEW_SELECTION', f"perc_overshoot >= {th_area_large}")



    arcpy.management.DeleteFeatures(la_STC3_locations_8)
    arcpy.analysis.Erase(la_STC3_locations_8, la_2DfeaturesCombined,
os.path.join(fp_STC3_gdb, "STC3_Locations_10"))

    field_mapping = arcpy.FieldMappings()
    for field_name in ["Shape_Area", "Shape_Length"]:
        field_map = arcpy.FieldMap()
        field_map.addInputField(os.path.join(fp_STC3_gdb,
"STC3_Locations_10"), field_name)
        field_mapping.addFieldMap(field_map)

    arcpy.conversion.FeatureClassToFeatureClass(os.path.join(fp_STC3_gdb,
"STC3_Locations_10"), fp_STC3_gdb,
                                                "STC3_Locations_final",
field_mapping=field_mapping)
    arcpy.management.Append(os.path.join(fp_STC3_gdb,
"STC3_Locations_final"), la_2DfeaturesCombined)

    layer_added = map.addDataFromPath(os.path.join(fp_STC3_gdb,
"STC3_Locations_final"))
    symbology = layer_added.symbology
    symbology.renderer.symbol.color = {'RGB': [86, 254, 54, 100]}
    layer_added.symbology = symbology
    map.addLayerToGroup(STC3_glayer, layer_added)
    map.removeLayer(layer_added)

    # Calculations
    area_STC3_parken = 0
    with arcpy.da.SearchCursor(os.path.join(fp_Buurt_gdb,"Parken_Di"),
"shape_Area") as cursor:
        for row in cursor:
            area_STC3_parken = area_STC3_parken + row[0]

    area_STC3_total = 0
    with arcpy.da.SearchCursor(os.path.join(fp_BGT_gdb,
"CombinedFeatures"), "shape_Area") as cursor:
        for row in cursor:
            area_STC3_total = area_STC3_total + row[0]

    area_STC3_current = 0
    with arcpy.da.SearchCursor(os.path.join(fp_BGT_gdb,
"begroeidterreindeel_Cl"), "shape_Area") as cursor:
        for row in cursor:
            area_STC3_current = area_STC3_current + row[0]

    with arcpy.da.SearchCursor(os.path.join(fp_BGT_gdb,
"ondersteunendwaterdeel_Cl"), "shape_Area") as cursor:
        for row in cursor:
            area_STC3_current = area_STC3_current + row[0]

    perc_STC3_current = round((area_STC3_current-area_STC3_parken) /
(area_STC3_total-area_STC3_parken) * 100)

    print(f"Currently {perc_STC3_current} percent of the total area is
```

```
vegetation")

    area_STC3_added = 0
    area_STC3_parking = 0

    with arcpy.da.SearchCursor(os.path.join(fp_STC3_gdb,
"STC3_Locations_8"), ["functie", "shape_Area"]) as cursor:
        for row in cursor:
            if row[0] == "parkeervlak":
                area_STC3_parking = area_STC3_parking + row[1]/2
                area_STC3_added = area_STC3_added + row[1]/2
            else:
                area_STC3_added = area_STC3_added + row[1]

    perc_STC3_added = round(area_STC3_added / (area_STC3_total-
area_STC3_parken) * 100)
    print(f"This can be raised by {perc_STC3_added} percent to a total of
{perc_STC3_current + perc_STC3_added}")
    perc_STC3_parking = round(area_STC3_parking / (area_STC3_total-
area_STC3_parken) * 100 )
    number_of_parking = round(2* area_STC3_parking / ps_area)
    print(f"This {perc_STC3_added} percent consists of {perc_STC3_parking}
percent parking places, which roughly accounts for {number_of_parking}
parking places")



#%% Part 10 - Run STC's (choose desired order)
STC1()
STC2()
STC3()

#%% Part 11 - Visualise results
#sum original situation with STC1 results
KLIC_STC1 = CellStatistics([os.path.join(fp_KLIC_gdb,
"KLIC_volumes_Cl"),os.path.join(fp_STC1_gdb,"STC1_Volume_final")],'SUM','DA
TA')
KLIC_STC1.save(os.path.join(fp_STC1_gdb,"KLIC_STC1"))
visualise3D(os.path.join(fp_STC1_gdb,"KLIC_STC1"),Volumes_glayer)

#sum original situation with STC2 results
KLIC_STC2 = CellStatistics([os.path.join(fp_KLIC_gdb,
"KLIC_volumes_Cl"),os.path.join(fp_STC2_gdb,"STC2_Volume_final")],'SUM','DA
TA')
KLIC_STC2.save(os.path.join(fp_STC2_gdb,"KLIC_STC2"))
visualise3D(os.path.join(fp_STC2_gdb,"KLIC_STC2"),Volumes_glayer)

#sum original situation with all results
KLIC_STC1STC2 = CellStatistics([os.path.join(fp_KLIC_gdb,
"KLIC_volumes_Cl"),os.path.join(fp_STC2_gdb,"STC2_Volume_final")],'SUM','DA
TA')
KLIC_STC1STC2.save(os.path.join(fp_Results_gdb,"KLIC_STC1STC2"))
visualise3D(os.path.join(fp_Results_gdb,"KLIC_STC1STC2"),Volumes_glayer)

aprx.save()

end_time = time.time()
print(f"the elapsed time is {(end_time - start_time)/60} minutes")
```