Dynamic Context-Sensitive Deliberation for Social Simulations: Balancing Scalability and Realism

Maarten Jensen

Doctoral Thesis, April 2025
Department of Computing Science
Umeå University
Sweden

Department of Computing Science
Umeå University
SE-901 87 Umeå, Sweden

maartenj@cs.umu.se

ISBN 978-91-8070-704-6 (print)
ISBN 978-91-8070-705-3 (digital)
ISSN 0348-0542
UMINF 25.06
The cover image was designed with the help of Gemini AI tool.

Printed by UmU Print Service, Umeå University, 2025

Abstract

There exists a realism and scalability trade-off in modelling agent-based simulations. As an example, one could create a simpler model with a simpler behavioural model that allows to simulate many agents. However, this simple model can miss important details of the real world. Incorporating these aspects of the real world can increase realism, however, it can come at the cost of scalability [43]. If more aspects of the real world are incorporated into the model, the behavioural model should be tied to these aspects. However, an interdependent behavioural system that considers all information at each time step is usually poorly scalable in terms of deliberative aspects [63, 42]. This poor scalability hinders the expansion of other desirable properties in the model, such as aspects of life and the number of agents.

We propose a context-sensitive deliberation framework that could help increase the scalability of the deliberation without losing behavioural realism. The framework is inspired by Kahneman's concept of thinking fast and thinking slow [46]. On the one hand, it will be capable of fast deliberation that is efficient (scalability). On the other hand, it can sometimes perform slower deliberation that can solve complex situations (behavioural realism). Rather than switching between these two modes, the framework slides, gradually incorporating more information into the deliberation. This is the complexity by need principle. The framework needs to be aware of the context to determine what kind of information to use and what kind of information to deliberate on.

Whether context-sensitive deliberation can increase scalability while retaining realism will be evaluated with a use-case simulation, the Agent-Based Social Simulation of the Coronavirus Crisis (ASSOCC). Context-sensitive deliberation is implemented in the ASSOCC framework. The Original ASSOCC framework is then compared with the context-sensitive ASSOCC variant. The results show that deliberation is no longer the bottleneck, since context-sensitive deliberation achieved a roughly 16-17 times speed-up over the original ASSOCC deliberation model. This speed-up was retained with higher agent numbers, and it can be expected that if deliberation contains more aspects, context-sensitive deliberation will be capable of an even greater speed-up. The behavioural and infection curves were similar between the two models, thus the realism of the model is retained. In conclusion, the work shows that context-sensitive deliberation can increase scalability and retain realism in agent-based simulations.

Sammanfattning

Inom agentbaserad modellering och simulering finns en inneboende avvägning mellan skalbarhet och realism. Till exempel kan man skapa en simplare modell med en mer simpel beteendemodell, vilket möjliggör simulering av många agenter. En simpel modell kan dock missa viktiga detaljer från den verkliga världen. Att integrera dessa aspekter av verkligheten kan öka realismen, men det kan ske på bekostnad av modellens skalbarheten [43]. Om fler aspekter av verkligheten inkorporeras i modellen, bör beteendemodellen kopplas till dessa aspekter. Ett ömsesidigt beroende beteendesystem som beaktar all information vid varje tidpunkt är dock vanligtvis dåligt skalbart när det gäller deliberativa aspekter [63, 42]. Denna bristande skalbarhet försvårar vidareutvecklingen av andra önskvärda egenskaper i modellen, såsom livsaspekter och antalet agenter.

Vi presenterar ett kontextkänsligt beslutsfattningsramverk som kan bidra till att öka skalbarheten hos beslutsfattningen utan att förlora beteenderealism. Ramverket är inspirerat av Kahneman's system koncept om att tänka snabbt och långsamt [46]. Å ena sidan möjliggör det snabb beslutsfattning som är effektiv (skalbarhet). Å andra sidan kan det ibland utföra ett mer deliberativt beslutsfattande som kan lösa komplexa situationer (beteenderealism). Istället för att växla mellan dessa två lägen, fungerar ramverket med en glidande övergång, där mer information gradvis införlivas i beslutsfattning. Detta kallas för komplexitet-efter-behov-principen. Ramverket måste vara medvetet om kontexten för att kunna avgöra vilken typ av information som ska användas och vad som bör beaktas i beslutsfattning.

Huruvida kontextkänslig beslutsfattning kan öka skalbarheten samtidigt som realismen bibehålls, kommer att utvärderas med hjälp av en användningsfall: Agent-Based Social Simulation of the Coronavirus Crisis (ASSOCC). Ramverket för kontextkänslig beslutsfattning har implementerats i ASSOCC-modellen. Den ursprungliga ASSOCC-modellen jämförs därefter med den kontextkänsliga ASSOCC-varianten. Resultaten visar att beslutsfattning inte längre är flaskhalsen, eftersom kontextkänslig beslutsfattning uppnådde en hastighetsökning på ungefär 16–17 gånger jämfört med det ursprungliga beslutsfattningsmodellen i ASSOCC. Denna hastighetsökning bibehölls vid högre agentantal, och det kan förväntas att om fler aspekter inkluderas i beslutsfattningen, kan kontextkänslig beslutsfattning uppnå ännu större förbättringar. Beteende- och infektionskurvorna var liknande mellan de två modellerna, vilket innebär att modellens realism bevarades. Sammanfattningsvis visar detta arbete att kontextkänslig beslutsfattning kan öka skalbarheten och samtidigt bevara realismen i agentbaserade simuleringar.

Samenvatting

Er bestaat een afweging tussen realisme en schaalbaarheid bij het modelleren van agentgebaseerde simulaties. Zo kan men bijvoorbeeld een eenvoudiger model maken met een simpel gedragsmodel, waarmee veel agenten gesimuleerd kunnen worden. Dit eenvoudige model kan echter belangrijke details van de echte wereld missen. Het opnemen van deze aspecten van de echte wereld kan het realisme verhogen, maar dit kan ten koste gaan van de schaalbaarheid [43]. Wanneer meer aspecten van de echte wereld in het model worden opgenomen, moet het gedragsmodel hieraan gekoppeld worden. Een onderling afhankelijk gedragssysteem dat alle informatie bij elke tijdstap in overweging neemt, is echter meestal slecht schaalbaar op het vlak van deliberatie [63, 42]. Deze beperkte schaalbaarheid belemmert de uitbreiding van andere wenselijke eigenschappen in het model, zoals aspecten van het dagelijks leven en het aantal agenten.

Wij stellen een contextgevoelig deliberatie framework voor dat kan helpen om de schaalbaarheid van deliberatie te vergroten zonder gedragsrealisme te verliezen. Het framework is geïnspireerd op Kahneman's concept van snel- en langzaam denken [46]. Enerzijds is het in staat tot snelle deliberatie die efficiënt is (schaalbaarheid). Anderzijds kan het in bepaalde situaties ook trager delibereren om complexe situaties op te lossen (gedragsrealisme). In plaats van te schakelen tussen deze twee modi, maakt het framework een geleidelijke overgang, waarbij stap voor stap meer informatie wordt meegenomen in de deliberatie. Dit is het complexiteit naar behoefte-principe. Het framework moet zich bewust zijn van de context om te kunnen bepalen welk soort informatie relevant is in de specifieke situatie.

Of contextgevoelige deliberatie de schaalbaarheid kan vergroten en tegelijk het realisme behouden, wordt geëvalueerd met een casestudie: de Agent-Based Social Simulation of the Coronavirus Crisis (ASSOCC). Contextgevoelige deliberatie is geïmplementeerd binnen het ASSOCC-framework. Het originele ASSOCC-framwork is vervolgens vergeleken met de contextgevoelige variant. De resultaten laten zien dat deliberatie niet langer de bottleneck is, aangezien het contextgevoelige deliberatie model zestien a zeventien keer sneller werd dan het originele ASSOCC deliberatiemodel. Deze snelheidswinst bleef behouden bij hogere aantallen agenten, en men kan verwachten dat bij uitbreiding met meer gedragsaspecten, de contextgevoelige deliberatie nog grotere snelheidswin-

sten kan behalen. De gedrags- en infectiecurves waren vergelijkbaar tussen de twee modellen, wat erop wijst dat het realisme behouden blijft. Concluderend toont dit werk aan dat contextgevoelige deliberatie de schaalbaarheid kan vergroten en het realisme in agentgebaseerde simulaties kan behouden.

Acknowledgements

Going abroad for my PhD was not only a scientific journey, it has also been a cultural journey. Not only because of getting to know much about the Swedish culture, but also because of the people I met from all over the world. Since I learned so much about different cultures I thought it would be cool to share the nationality of my friends and colleagues in this acknowledgement. Here is a thank you for all people that were part of this research journey!

Umeå University

First, I would like to thank the socially aware AI group in Umeå University. This group started when I started my PhD in October 2019. Thanks to Ree (Dutch) for the interesting and deep conversations about many aspects of life. Christian (German) for the support, especially when we both went back to our home countries. Cezara (Romanian) for the feedback and the humour. Eren (Turkish) for the interesting discussions.

I would like to thank my supervisors, Frank (Dutch) and Loïs (French). Frank, thank you for your supervision and support. Every time I got stuck with research, how hopeless the situation may have seemed, you almost always had a solution. It always amazed me how sometimes you could just clear all my worries with a single sentence or idea. I also appreciated the research retreats we did with the research group, I think it really brought the group closer together. Lois (French), I appreciate to have had you as a supervisor and that you were critical on the scalability aspects of my PhD, this in the end made it a solid story. We also had some adventures before the supervision started. I will never forget the fish we tried to cook with Mehari and Ravi. We started a fire in a grilling hut, but quickly the hut filled with smoke. After half an hour of running in and out to supply the fire with oxygen. We found that the entire roof, including the exhaust, was covered in snow. It was the best fish I have ever eaten!

I would like to thank my colleagues from the computing science department and fellow PhD students at Umeå University. Romuald (Cameroonian), Arka (Indian), Divya (Indian), Ayush (Indian), Saloni (Indian), Younes (Maroccan), Sudipta (Indian), for the fun we had in the corridor and outside of work. Arvid (Swedish) for helping understanding the social sciences better and discussing context. Lena (Austrian) for the deep conversations and for digging out the hole in the quinzhee. Hannah (Swedish), Hannah (Irish/North-American), Annakarin (Swedish), Frank (German), Nina (Ukrainian), Kai-Florian (German), Virginia (Portuguese), Julian (Argentinian), Luis Gustavo (Brazilian), Andrea (Spanish), Andreas (Greek), Roshni (Indian), Kaan, Diego, and Stella (Spanish). Special thanks to my Indian colleagues who showed me how to celebrate Holi! Also special thanks to the tech support at computing science: Mattias (Swedish), Tomas (Swedish), Bertil (Swedish), and Andreas (Swedish). I always felt welcome and thank you for the annual Christmas decorations bringing some light in the darkness.

I also would like to thank some institutes or groups that I got in contact with during my PhD in Sweden. Thanks to Umeå Naturvetar- och Teknologkår (NTK), especially Erik Steinvall (Swedish), for supporting PhD's and organising activities for us. Thanks to the Shut Up and Write (SUAW) group at the Umeå University library, lead by Annika (Swedish). It was really helpful and fun to write together weekly with other PhD students. Thanks to IKSU, the sport center and also IKSU Sarek for organising outdoor activities. I would like to thank Wallenberg AI, Autonomous Systems and Software Program (WASP) for funding the PhD and giving the opportunity to meet fellow AI PhD's in Sweden. The European Social Simulation Association (ESSA) and the Social Simulation Conferences they organised. I always thought it was great to be there, mainly due to the people. To me the people were very supportive and respectful towards each other, an example of how conferences should be. Thanks to Bruce, Harko, Nanda, Wander, Fabian (German), Rui (Portuguese), Inês (Portuguese), Ivan, Christa, Jason, and Rajith. Mikeal (French/English) for your honesty and the great conversations late at night.

Sweden, Umeå

During my stay in Umeå I also met many people outside of the work environment. Erik (Swedish) who welcomed me in many aspects of Swedish life. Together with some of your other friends, Fredrik (Swedish), Anton (Swedish), Niklas (Swedish) we did things like a skiing, eating surströmming and julbord (I prefer the latter), and picking mushrooms.

David (Hong Kongese), I would actually consider you my first friend in Umeå. We met through our joined interest in starting a language cafe. I am really happy that we managed to create Language Cafe Umeå (together with Mayumi (Japanese)). Mehari (Eritrean) and your family, I would like to thank you, for the complex scientific discussions we had, but also the times we just relax and sit outside the whole day. Ramzi (Syrian) and Hwaida (Syrian) it was very special to be part of your Syrian wedding in Gothenburg. Ravi (Indian), thanks for your support and talking me out of my worries.

Ruslan (Kyrgyz), it is great to see how you managed to complete your studies and now found a job as a windmill engineer. It was great to have you in the Netherlands and you are always welcome again. Malaz (Syrian), I am happy to have you as a friend! We really had great times in Umeå, the Netherlands and Germany, as we would always come up with something fun to do! Maryam (Iranian), I really appreciated that we met and liked our joined interests. I really liked working together at Kulturbageriet, and when I look back I think those were among my most fun and productive working days. Sofia (Swedish) and Samir (French/Algerian) for the support and great adventures together. Anastasia N. (Russian), I am happy that I met you, thank you for the support during part of this journey.

Thanks to the people I met at Language Cafe Umeå. Especially, Mimmi (Finnish), Atanas (Bulgarian), Eduardo (Spanish), Irina (Russian), Ezis (Latvian) for helping to organise and maintain Language Cafe after David left Umeå. Eduardo (Spanish) and Zeynep (Turkish), It was very special to see your wedding in Turkey. In addition, I would like to thank Marco (South-African/German), Alejandra (Columbian), Joseph (Syrian), Rickard (Swedish), Liesbeth (Dutch), Maria (Spanish), Julio (Mexico), Jose (Mexico), Negar (Iranian), Julia (German), Alva (Swedish), Sasha (Swedish/Dutch), Elliot (Swedish), Guillaume (French), Sarah (North-American), Marcus (Swedish), Isolde (Dutch), Karolina (Swedish), Samaneh (Iranian), Anastasia M. (Russian), Amber (Belgium), and Sandra (Dutch).

I would like to thank Carlos (Spanish), Josh (English), Daniel (Italian) for being amazing housemates the last part of my stay in Umeå. I really enjoyed living in the yellow house: the spontaneous hangouts, the cooking together, and the games of Mario kart. Carlos and Daniel, good luck with the last part of your PhD's! And Josh good luck with your job in the South of Sweden!

Utrecht University

During the PhD I frequently visited Utrecht University when I got the opportunity. I always liked to come back and work together with Mijke (Dutch) and Alexander (Dutch). Despite the distance, I really appreciated that we stayed in contact, especially during those SSC deadlines. Alexander, you did a very good job last year finishing your thesis! And Mijke I really hope you get to finish your PhD this year, I believe in you! At Utrecht University I would also like to thank the others at the AI and Police labs. Pinar (Turkish), Sjoerd (Dutch), Bas (Dutch), Davide (Italian), Michiel (Dutch), Roos (Dutch), Anastasia (Turkish), and Emre (Turkish).

Radboud University

When I came back to the Netherlands to finish the last part of my PhD research, it did not take too long before I connected with Radboud University again. I would like to thank the Radboud Sport and Culture centre. I would also like to thank the PhD Organisation Nijmegen (PON), of which I happily have been a member of the past one and a half year. PON helped me connect with the PhD community in Nijmegen and gave me a lot of motivation to continue the

last part of the PhD. Special thanks to the other PON board: Jurgen (Dutch), Kim (Dutch), Pauline (French), Krishna (Indian), Ayşegül (Turkish), Debbie (Dutch), Daniel (Dutch), Xinyu (Chinese), Matteo (Italian), Josi (German), and other members.

In particular Funda (Turkish), together we started the PON writing group, it was a lot of fun to organise both writing days for the PhD's in Nijmegen. In addition, I thank Charlotte (Dutch), Mariska (Dutch), Felicitas (German), Julian (Dutch) who frequently wrote together with me. We could share feedback with each other, but also emotionally support each other. I would also like to thank the Writing Lab, and in particular the writing coach, Sarah (Dutch). Thank you for the discussions which helped me set daily achievable goals instead of getting overwhelmed by the thesis as a whole.

Also, thanks to the Artificial Cognitive Systems group at the Donders Institute. Thanks to Marcel (Dutch), Benjamin (Israeli), David (Dutch), Sigur (Dutch), Marzieh (Iranian), and the other people in the lab. Muhammad (Iranian) for the first barbecue of the year. Special thanks to Siddharth (Indian) for our interesting discussions on scalability in agent-based simulations, sometimes during salsa parties.

The Netherlands

I would also like to thank my friends I met in the Netherlands. Many of which I already knew before this PhD journey. Thanks to Stefan (Dutch), Thijs (Dutch), Kirsten (Dutch), Stan (Dutch), Dave (Dutch) and Ting Ting (Chinese), Remi (Dutch), Timothy (Dutch), Nina (German), Lucas (Dutch), Thomas (Dutch), Philip (North-American), Roberto (Spanish), Lauraine, Rafael, Ron, Willeke, Alex, Martine, Beppie, Jan, Janneke, and Frank.

Special thanks to Philip (Dutch), who came to visit me in Umeå multiple times and supported me often. I appreciate that I could always call you if there was a personal or research wise difficulty. Jasper (Dutch), I appreciate your support throughout the last years and the years before. I appreciate that we can open up to each other and be honest about what we feel. Marieke (Dutch) for being good friends and for organising the language café in Utrecht which has been the inspiration for Language Cafe Umeå. Daan (Dutch) for the support, working together, and fun night of intense board games where David (Dutch) always won anyway. Peter (Dutch) for the support and spontaneous adventures. Mattijn (Dutch), Stephanie (Dutch), Mattijn, before I went to Umeå you showed me how it was to camp with -18 degrees in the Norwegian winter. Since I really liked our trip, this was one of the reasons I moved to Sweden. Samaneh (Iranian) and Naveed (Iranian), I am happy to have met you and Sameneh you really showed how cool science can be when working together! Paulina (Ecuadorean) and your family, working together in Wageningen was a lot of fun and I really appreciated the talks we had. I also would like to thank the people in Alverna, my home village, with whom I frequently talked with when walking the dog.

I want to thank my entire family for supporting me through these years and before. Even when times were difficult, you were always there for me and I am really happy you were able to visit me so often in Umeå. Lykke (Dutch/Danish) for reminding me to relax. Lasse (Dutch/Danish) we did not meet that often the last years, since I was in Sweden and you were in Australia. However, I was happy that you were here in the Netherlands in the last months of thesis writing and could be part of this journey as well. My father, Ole (Danish) and Julia (English) thank you for the support, and Ole thanks for coming to Umeå, it was great to do the trip towards Abisko together. Kenai (dog) for the walks and the hugs. My uncle Jos (Dutch) for reminding me to not take life too seriously. Min bästefar Jens-Christian (Dansk), jeg vil gerne takke dig for at spørge om mit thesis og forsøge at forstå, selvom det er et svært emne. Fester Bente (Dansk), tak för at lave rigtig god mad, hver gang vi kommer hen til jer. Også tak til de øvrige danske familiemedlemmer.

My mother, Marja (Dutch), who was one of the greatest help to me. Not only the emotional support and financial support for the last part of the PhD in the Netherlands. Thank you for that and also for putting up with me, my sister, and my brother all being back home. I really appreciate that that was possible, so thank you so much!

Conclusion

Although there were many happy and good moments during this PhD, it did not come without struggles. However, despite what happened, I am happy that I always followed my heart wherever I went. This is something I would recommend to everyone at all stages of life, but perhaps mostly to fellow PhD students. Remember that it is your PhD, and you should do your PhD the way you want to!

For me, I always wanted to connect and share ideas with other people. And that is what became possible thanks to you. I think with everyone mentioned here I have had a conversation about the thesis at some point. In this sense everyone contributed a little bit, and therefore I would like to thank all of you again, I could not have done this without you!

Thank you, Merci, Danke, Dankjewel, Gracias, Grazie, Tak, Xiè xie, Shkran, Spasiba, Teşekkür ederim, Tack!!

Publications

This chapter lists all the publications to which the author contributed. Paper I [45] and Paper II [41] are the basis of Chapter 3. Paper III [42] presents a prototype context-sensitive deliberation implementation, this is not directly used in the thesis, but provided inspiration for the model in this thesis. Paper IV [43] discusses some requirements for realism which are mentioned in the Introduction. Other publications are not directly included in the thesis but contributed to the intuitions and background of the thesis.

Paper I *Maarten Jensen et al. "Towards Efficient Context-Sensitive Deliberation". In: Advances in Social Simulation. Springer, 2022, pp. 409–421.*

Paper II *Maarten Jensen, Loïs Vanhée, and Frank Dignum. "Dynamic Context-Sensitive Deliberation". In: Multi-Agent-Based Simulation, MABS 2023. 2023.*

Paper III *Maarten Jensen, Loïs Vanhée, and Frank Dignum. "Dynamic Context-Sensitive Deliberation for Scalability in Realistic Social Simulations". In: Advances in Social Simulation. Springer, 2023.*

Paper IV *Maarten Jensen, Loïs Vanhée, and Frank Dignum. "Towards Realism for Policy Testing". In: Advances in Social Simulation. arXiv, 2024.*

Other Publications

At the beginning of the PhD project, two publications were made.

Paper V *Cezara Pastrav, Maarten Jensen, René Mellema, Christian Kammler, and Frank Dignum. "Sharing Green Spaces at the Umea Academic Campus". In: International Workshop on Agent-Based Modelling of Urban Systems (ABMUS). 2020, p. 47.*

Paper VI *René Mellema, Maarten Jensen, and Frank Dignum. "Social rules for agent systems". In: International Workshop on Coordination, Organizations, Institutions, Norms, and Ethics for Governance of Multi-Agent Systems. Springer. 2020, pp. 175–180.*

During the PhD the author worked together with other researchers on the Agent-based Social Simulation of the Coronavirus Crisis (ASSOCC) framework. This lead to a book, *Social Simulation for a Crisis* [19], of which the author contributed to the following chapters and some paper publications.

Ch. III *Maarten Jensen, Loïs Vanhée, and Christian Kammler. "Social Simulations for Crises: From Theories to Implementation". In: Social Simulation for a Crisis. Springer, 2021, pp. 39–84.*

Ch. IV *Cezara Păstrăv, Maarten Jensen, René Mellema, and Loïs Vanhée "Social simulations for crises: from models to usable implementations". In: Social Simulation for a Crisis: Results and Lessons from Simulating the COVID-19 Crisis. Springer, 2021, pp. 85–117.*

Ch. VII *Maarten Jensen, Fabian Lorig, Loïs Vanhée, and Frank Dignum. "Deployment and effects of an app for tracking and tracing contacts during the COVID-19 crisis". In: Social simulation for a crisis: Results and lessons from simulating the COVID-19 crisis. Springer, 2021, pp. 167–188.*

Ch. XII *Fabian Lorig, Maarten Jensen, Christian Kammler, Paul Davidsson, and Harko Verhagen. "Comparative Validation of Simulation Models for the COVID-19 Crisis". In: Social Simulation for a Crisis: Results and Lessons from Simulating the COVID-19 Crisis. Springer, 2021, pp. 331–352.*

Ch. XIV *Maarten Jensen, Frank Dignum, Loïs Vanhée, Cezara Păstrăv, and Harko Verhagen. "Agile social simulations for resilience". In: Social Simulation for a Crisis: Results and Lessons from Simulating the COVID- 19 Crisis. Springer, 2021, pp. 379–408.*

Ch. XV *Frank Dignum, Maarten Jensen, Christian Kammler, Alexander Melchior, and Mijke van den Hurk. "Challenges and Issues for Social Simulations for Crises". In: Social Simulation for a Crisis: Results and Lessons from Simulating the COVID-19 Crisis (2021), pp. 409–426.*

Paper VII *Frank Dignum et al. "Analysing the combined health, social and economic impacts of the corovanvirus pandemic using agent-based social simulation". In: Minds and Machines 30 (2020), pp. 177–194.*

Paper VIII *Christian Kammler et al. "Social Simulations for Intelligently Beating COVID-19". In: AI for Social Good Workshop, virtual, July 20-21, 2020. 2020.*

Paper IX *Christian Kammler et al. "Towards an Agent-based Platform for Crisis Management". In: Review of Artificial Societies and Social Simulation (2023).*

Contents

Chapter 1

Introduction

In the daily life of a person, decisions can require different types of information. When buying ingredients for your favourite pasta recipe, the process will usually be quite automatic. You go to the local store, know where the products are and buy them. Compare this with the decision on which house to buy for yourself. Most people will consider all kinds of information before selecting their house. People usually extensively check their finances, check multiple houses, ask family and friends, visualise how it will be to live in the house. From these examples it should become clear that what determines the information we would consider depends a lot on the decision situation, in other words the context.

Understanding the daily decisions people make is especially important when making models of society. If a model has clearly wrong assumptions about the behaviour of people, the outcomes can also be wrong. Let us illustrate this with a model about the COVID-19 pandemic, where we want to test the effect of national lockdown on the infection curve. People generally follow their daily life schedule of sleeping at home, going to work, buying groceries, and seeing friends once in a while. When people gather in places, the infected people can infect the healthy people with the virus. Now imagine that the model assumes that all people will follow the governmental rules 100% of the time. If national lockdown is introduced, this means that all people will stay at home 100% of the time. In the model, the people will therefore not meet other people. This will stop the spread of the virus completely. However, this does not align with what was happening in almost every country in the world. During the pandemic, national lockdowns may have contributed to flattening the infection curve, but these national lockdowns have hardly ever completely eradicated the virus. Thus, this particular example model does not simulate the effect of the policy well and, therefore, cannot be seen as realistic enough. This is caused by the assumption that people follow the rules 100% of the time.

The behavioural model needs to be improved to obtain model outcomes that are better matched with the real world. As an example, let us use the same model as before, but now the deliberation model is changed. Instead of people following the rules 100% of the times people stay inside during lockdown, they will break the rule with a 10% probability and go either grocery shopping or see friends. This model can be expected to show a flattened infection curve during lockdown that could rise again after lockdown. This outcome matches better with what happened in real life than the model discussed previously. However, it missed explicit motivations for breaking the rule since in this model this is based on probability. It would then be unclear whether they break the rule for grocery shopping or to see friends. And if a policy developer would like to test whether it is effective to deliver food to people during the lockdown, this model would not be of much use. We argue that explicit motivations have to be added to evaluate the effect of policies or combinations of policies.

A pandemic framework that explicitly models motivations is the Agent-Based Social Simulation of the Coronavirus Crisis (ASSOCC) framework [19]. It is a framework that simulates the spread of the Covid-19 virus in a population. The population had access to schools, work, shops, leisure activities, a hospital, and more. It included motivation in the form of needs, these needs can be individual like the food and sleep need, but also more societal related like conformity and compliance. This model combined many behavioural aspects to support the use of needs, it contained scheduled behaviour, norm following, social networks, and more. Since the model contained explicit motivations, it was capable of not only testing the effects of various policies but also explaining why the agents had chosen to break those policies. It would become clear whether the agents would break lockdown to go to a grocery shop or to see friends. This could be valuable information if one wants to understand what combinations of policies could work better or worse.

The incorporation of many different behavioural aspects came at a cost. All these aspects were combined into a single deliberation system. This is the need-based deliberation system, which required information from almost all parts of the model to calculate the need satisfaction levels. It would for each action available to the agent, calculate the optimal action according to all available information. This system worked but also had its limits. Since the deliberation considers all the information every single time step, the deliberation became very slow. The deliberation became the main bottleneck of the simulation [63] and limited further expansion of the simulation. Adding more aspects to the model, would mean the deliberation had to be expanded as well which was not practically possible. This brings us to the main problem, where, by making the model fit to reality more, the model became less scalable.

1.1 Problem Statement: Scalability vs Realism

As became clear from the introduction, there exists a scalability vs. realism trade-off when modelling agent-based simulations. When incorporating more aspects in a model to make it more similar to the real world, the model usually becomes less scalable. Scalability does not only relate to the number of sub models included in the model. It can also be related to the number of agents that can be simulated [53, 52, 3] or to the deliberative aspects taken into account in the deliberation [63]. It can be said that if a model is not scalable, at least one of these aspects cannot be easily expanded without increasing the execution time of the simulation to impractical levels. This thesis will mainly focus on scalability in the sense of adding deliberative aspects.

Realism in agent-based simulations is a complicated topic. In a sense, every simulation attempts to at least simulate reality to some extent. From the most abstract models to the most complex models, almost every model takes at least some theory, data, or assumption from the real world. However, incorporating and evaluating the right aspects in a model can be difficult, as illustrated in [27]. In the end, what is necessary for an agent-based simulation to be useful can depend a lot on the goal, and there can be many goals [31]. The work in this thesis will mainly focus on simulations for policy testing. These are the less abstract models, such as the ASSOCC framework [19], which incorporate many behavioural aspects to be able to simulate the effects of multiple policies. As discussed in Jensen [43], to achieve realism for policy testing, the model requires many aspects of life, interdependent deliberation, and the capability of simulating sufficient agents. As discussed before, the ASSOCC framework suffers from scalability and could practically not be extended further due to deliberation being the main bottleneck. The deliberation could not easily be made more efficient, by taking out aspects as then the model would become less realistic. It suffered from the scalability and realism trade-off. The work in this thesis will aim to achieve both realism and scalability in the deliberation of agent-based simulation.

1.2 The Goal: Context-Sensitive Deliberation

To achieve both realism and scalability in agent deliberation, perhaps the literature on human decision making can be of help. The human brain can efficiently make decisions about daily life, for example, through the use of heuristics [34], and could perhaps provide some inspiration for modelling agents' deliberation. Kahnemann [46] suggests that humans have a fast thinking system that is used most of the time and a slow thinking system that is used occasionally. Although this concept of two distinct system has been outdated. The concept of human decision making not always using the same process still remains. Based on the thinking fast and thinking slow concept, a more efficient deliberation system could be created. One that could use fast decision making most of the time, which would make it scalable. And use slow decision making when necessary, which could make it realistic.

When to use faster or more complex deliberation is not clear as it depends on the situation. In some situations it is enough to use little information, for example when buying food for your favourite meal. However, some situations require more information and perhaps other decision processes, such as buying a house for the first time. As discussed in the beginning of this chapter, it depends on the context. If a model were to be made that could switch deliberation type, it needs to be explicitly aware of the context to choose the right deliberation type. In other words, the deliberation should be context sensitive.

The goal of this thesis is to evaluate whether using context-sensitive deliberation can in fact lead to more scalability while retaining the realism in an agent-based simulation. As far as we know, there are no existing context-sensitive deliberation models that fit this purpose. Therefore, we will develop, formalise, implement, and evaluate such a deliberation framework.

1.3 Research Questions

The goal of this thesis is to evaluate whether the use of context-sensitive deliberation can lead to increased scalability while retaining realism in an agent-based simulation. Achieving this can be done by answering the following research question:

"Can context-sensitive deliberation increase scalability while retaining realism in agent-based simulations?"

One could think of splitting this research question into two questions, one question focussing on realism and one on scalability; however, this would undermine the connection these two concepts have. If one were to ask *"Can context-sensitive deliberation increase scalability in agent-based simulations?"*. One could just create a very simple deliberation model that performs prescheduled actions mapped to the time. This would be very fast, however, it will lack realism as it will not adapt to other aspects such as needs, being sick, or the laws.

The same applies to asking *"Can context-sensitive deliberation retain realism in agent-based simulations?"*. Imagine that one would make context-sensitive deliberation model that is basically a copy of need-based deliberation in ASSOCC and has the same number of variables to consider. This would give very similar behaviour; however, might also require the same amount of computational resources. Thus in that case it cannot be said to increase scalability. Due to this interdependency, the research question has to incorporate both scalability and realism. To answer the main research question a couple of steps are necessary which are explained in the upcoming section.

1.3.1 Supporting Research Questions

To answer the main research question a study needs to be performed that consists of three distinct steps. First we need to create and formalise the context-sensitive deliberation framework. After formalising context-sensitive deliberation model it should be implemented in an existing simulation. If this is completed experiments can be performed evaluating the trade-offs between realism and scalability in the context-sensitive deliberation model. The next section divides these steps into distinct supporting research questions.

Formalising Context-Sensitive Deliberation

A very naive context-sensitive deliberation model would determine the full context beforehand. Based on the full context it could choose an appropriate deliberation type, starting with the most simple deliberation type. One could at first expect this model to be efficient. However, this type of model just moves the complexity of the model to another point. The complexity is not directly in the deliberation, but it is now in the context determination. This type of model can thus still be expected to be inefficient. Since this type of model is not efficient and other useful models could not be found in the literature, another type of context-sensitive deliberation should be developed. This brings us to the following research question:

RQ1: How to formalise context-sensitive deliberation?
This question is investigated in Chapter 3, Framework.

This research question can be split into two sub research questions. To understand how this model should be formalised the required aspects of the model should be known. For example, it should at least have an explicit context module. However, as explained by the naive context-sensitive deliberation model, this context module should not completely fix context determination. Then it becomes the question what should happen instead and which elements should take over deliberation. These aspects will be investigated by answering the following research question:

RQ1.1: What are the main aspects necessary for context-sensitive deliberation?
This question is investigated in the first part of Chapter 3, Framework.

Just having an abstract representation of the aspects of context-sensitive deliberation will not be enough to implement the framework. While determining the main aspects of a model that represents some part of human behavioural theory is one of the first steps, it is usually not enough to perform agent actions. In the agent-based simulation literature there are quite some abstract frameworks that describe deliberation for agents based on sociological theories [21, 51, 20]. Although this is important work, these frameworks are usually not directly implementable in agent-based simulations. They miss an intermediate step of describing how an agent can select actions by using the framework. To implement context-sensitive deliberation it needs to be formalised for action taking in agents:

RQ1.2: How to formalise context-sensitive deliberation as framework for agent action taking in agent-based simulations?
This question is investigated in the second part of Chapter 3, Framework.

Implementing Context-Sensitive Deliberation

For evaluating the main research question, context-sensitive deliberation should be implemented and tested in an existing agent-based simulation. Moving from formalisation to implementation is a challenging step and can be done in many different ways. It does matter with which algorithm context-sensitive deliberation is implemented. If functions in the programming language are used that are very slow, the deliberation may be sound compared to the conceptual model, but in practice it may not be efficient enough. The chosen implementation can also affect the agent's behaviour, as shown in [54]. To investigate how to implement context-sensitive deliberation, the following question has been added.

RQ2: How can context-sensitive deliberation be implemented taking both efficiency and realism into account?
This question is investigated in Chapter 5, Implementation.

Evaluating Context-Sensitive Deliberation

During this study, the context-sensitive deliberation model should strike the right balance between realism and scalability. If it leaves out crucial information in a decision, it may be fast but could portray less realistic behaviour. This happens, for example, when the agents follow all the laws of the governments 100% of the time. This simplification could lead to unrealistic results, as it leaves out motivations of people to break out of lockdown. If the model uses all the laws and motivations available to the agent all the time, it probably makes more realistic choices for the agents. However, this comes at the cost of its scalability. This trade-off between scalability and realism does not only apply when deliberating about laws. It could also be applied when considering other behavioural aspects. The model needs to strike a balance between realism and scalability. To investigate this trade-off, the following research question will be investigated.

RQ3: What are the trade-offs between scalability and realism in the deliberation of an agent-based simulation?
This question is investigated in Chapter 6, Evaluation.

1.4 Thesis Structure

The thesis is structured as follows. Chapter 2, Background, will analyse literature on context, agent-based simulation, agent-based deliberation systems, and scalability aspects of agent-based simulations.

Chapter 3, Framework, will formalise context-sensitive Deliberation. This chapter will answer **RQ1** by answering two sub research questions, **RQ1.1** and **RQ1.2**. The first part of the chapter will explain what aspects are necessary for context-sensitive deliberation [45]. The second part of the chapter will explain how to conceptualise context-sensitive deliberation for agent action taking [41]. This part presents the Dynamic Context-Sensitive Deliberation (DCSD) framework which is a context-sensitive deliberation framework for agent action taking.

Chapter 4, Methodology, will explain how the main research question will be answered. It proposes to use the ASSOCC framework [19] as a use case to validate the DCSD. That is, the Original ASSOCC version will be compared with a Dynamic Context-Sensitive Deliberation version of ASSOCC. It describes how to measure whether the DCSD implemented in ASSOCC retains the realistic properties. It also describes how to measure scalability in deliberation.

Chapter 5, Implementation, explains how the DCSD framework is implemented in the ASSOCC framework. It applies the context-sensitive deliberation framework from Chapter 3 to the ASSOCC framework. The relevant information for the deliberation of ASSOCC agents is categorised. It is determined when the DCSD should use different information and deliberation type. The algorithm used to implement DCSD in ASSOCC is based on decision trees. This chapter answers **RQ2** about the implementation of context-sensitive deliberation.

Chapter 6, Evaluation, shows the results of comparing Original ASSOCC and DCSD ASSOCC. It starts the comparison by evaluating the retention of realism. The DCSD is gradually expanded over five sections, those are 1) habitual behaviour, 2) strategic behaviour, 3) normative behaviour, 4) social behaviour and 5) the DCSD ASSOCC model, which is further optimised version. The sixth section assesses whether DCSD ASSOCC has increased scalability compared to Original ASSOCC. This chapter answers the **RQ3** on the trade-offs in realism and scalability in modelling deliberation.

Chapter 7, Conclusion, summarises the most important results and reflects on the research. The chapter answers the main research question using the answers for the supporting research question, shows limitations and future work. Chapter 8, Appendix, shows additional tables and figures that support the arguments in the thesis but are not essential to understand the arguments.

Chapter 2

Background

In this chapter, we first define decision context based on existing literature. Understanding what kind of decision context we are dealing with brings some implications for Context-Sensitive Deliberation. After providing the definition, we will discuss agent-based simulations and some staple examples within that field. This is followed up Section 2.3 about Multi-Agent Systems (MAS) describing agents that can plan and coordinate. MAS agents lack handles for implementing irrational and social aspects seen in human behaviour. The agent-based simulations community came up with a couple of adaptive frameworks such as Consumat [40], Humat [1] and CAFCA [30] that incorporate multiple types of behaviour. The CAFCA framework is particularly useful for context-sensitive deliberation as it gives handles on how to deal with all kinds of different decision situations. Finally, a section on scalability within agent-based simulations, where three directions of scaling agent-based simulations are discussed. Agents can be scaled on the number of agents, sub models and deliberative aspects. The latter is the most important throughout this thesis.

2.1 Context-Sensitive Deliberation in Human Decision Making

The importance of context in modelling human decision making has been indicated by a vast amount of literature [45, 41]. The work by Edmonds attempts to model context recognition and usage in agent reasoning [26, 28, 25, 24]. Rato describes social context for social agents [56]. Kokinov investigates how context can influence decision making in humans [48]. However, our definition of context differs from the definitions in the literature mentioned. There is no agreement on a single definition of context. The existing definitions provide some useful concepts, but do not exactly cover what we think context entails in our research. We will explain our definition of context by looking at a number of different definitions and arguing why they are not suitable for our purpose.

Dey [16] defines **context** as:

'Context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.'

This definition originates from studying context in users and software applications. Our definition differs in two aspects 1) we consider the digital world where entities are agents, not the human world. 2) information considered is not only external to the agent but also relates to the internal state of the agent. This definition will be specified in the following paragraphs.

Traditionally context is often studied from the perspective of an entity in the real world. The definition by Dey [16] is based on a software system in the real world and states that all information can be potentially part of the context. Zimmermann [67] further categorises the concept of context by splitting it into five categories (see Figure 2.1).

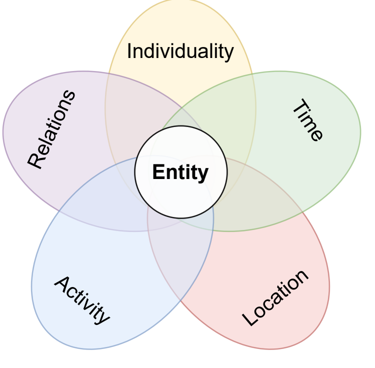


Figure 2.1: Categories of context according to Zimmermann [67]

As described in Jensen et al. [45], to give a better understanding of the categories, we give a couple of examples. *Time* can be a specific point in time, but also a period, it can relate to for example seconds, minutes, days (even working days or weekdays), years, centuries, etc. The *location* can be a physical place, with variety in size, for example larger geographical, building, complex, town, region or country. The *activity* indicates what is done in the context, alone or together, grocery shopping, playing football, having dinner, in a formal meeting or non-formal. The *relations* include the aspects of the context related to other people, groups or institutes. It also includes theory of mind, that can for example relate to goals, intentions, social norms, values of other entities. The *individuality* contains the characteristics of an entity itself, its current interests and goals, value priorities, experience (is the situation known or not?), needs/motives and more. It should become clear from these examples that context can basically be any kind of information. This type of context as described in Dey [16] and Zimmermann [67] is infinite.

Edmonds [24] adds to this that perhaps humans do not even have the capability of understanding all information in the universe and therefore, cannot get a full grasp on context in the real world. The context for agent deliberation within a simulation is however, different. Rather than context in a system, it is context of an agent within a system (the simulation). For example, a house could be implemented as an object that contains a location, can hold a number of agents and stores food. Meaning those objects are the only contextual information that can be considered from the house within the simulation. The context within a simulation is pre-defined and all the information is readily available which simplifies the information to be considered drastically.

Rato [56] defines social context where a distinction is made between the external state (social context) and the internal state of an agent. The individuality concept from Zimmermann, seems at first glance to relate to the internal state of an entity, however, it is defined as 'information that can be observed about the state of the entity' [67]. Thus, this does not strictly mean the internal state of an entity, since when considering a human as entity, internal motivations, goals, needs, and other social aspects cannot easily be observed (unless asked). In a social simulation, the internal state of an agent is accessible as these aspects such as motivations, goals, needs are represented by variables. In decision making, not only the external state of the environment but also the internal state of the agent play a role. Therefore, our definition of context should consider both the external and the internal state as relevant information.

To avoid confusion of our definition of context with the other definitions of context, we will from now on talk about *decision context*, rather than just context. Since decision context captures the information that is to be considered better. Thus, based on the previous two paragraphs, our definition of the **decision context**, as defined in Jensen et al. [41] will be the following:

"The decision context is any information that can be used in the decision making of an agent in a social simulation. Any information is information internal to the agent, external to the agent (i.e. the simulation environment), and also includes other agents' internal states."

This definition narrows down the information considerably compared to the earlier definition of context by [16]. As we focus on context-sensitive deliberation, the agents we want to develop should comply with this definition. In other words, we want agents that can be aware of their internal state, but also aware of other agents and their surrounding environment when this is relevant to their decision making situation. The external information will not only be the actions other agents perform but also social concepts such as the intentions or needs of other agents, social norms or groups. These agents should thus be able to incorporate different types of information and deal with it accordingly in their decision making. The following sections will discuss agent frameworks and to what extend these existing frameworks could handle contexts according to our definition of context. In the next section we discuss agent-based simulations.

2.2 Agent-Based Simulations

Agent-based simulations are used to study societal phenomena. In contrast to compartmental models, which are mathematical models that can represent a homogeneous population through mathematical representations [7], agent-based simulations are capable of representing a heterogeneous population. Having heterogeneity in a model is important as some kinds of social behaviour are driven by specific individuals [60]. An agent-based simulation consists of multiple agents that can represent individuals or other aspects of society (groups, buildings, institutes, other objects) [1]. Generally, agents in agent-based simulations can autonomously perform actions; these actions manipulate the state of the simulation. Through the interactions of the agents, interesting phenomena can emerge. The next section will explain the emergence in a staple agent-based simulation model, i.e. the Schelling Segregation Model.

2.2.1 An Example of Agent-Based Simulation

The Schelling segregation model [58] is considered one of the first agent-based simulations, dating from 1969, created by Thomas Schelling. The model studies segregation by dividing agents into two groups (indicated by two different colours) and giving them a satisfaction rule. The agents are randomly placed on a grid and have a mild satisfaction rule. The rule is based on the groups to which the neighbours belong. The neighbours are agents on one of the eight cells adjacent to the agent's cell. If from the neighbours at least $1/3$ of the agents belong to the same group as the agent in the middle, the agent is satisfied. The agent will not move if satisfied. If the agent is unsatisfied, when only $1/3$ or lower amount of agents belong to the same group, then the individual agent will move randomly to a new position. The Schelling model shows that even with mild in-group preferences, segregation is still likely to occur in a population.

In this example, the decision context of the agent is basically its own colour, the number of neighbours, and the number of neighbours of the same colour. This is enough information to determine whether the agent is happy or not, which results in, respectively, not moving or moving to a different location on the board. This type of model is obviously very simple and, when compared with our decision context definition, it does lack many of the requirements. The agents lack planning, normative reasoning, group behaviour, just to name a few things. However, in agent-based simulations, there are more sophisticated simulations available, one of which is a pandemic model, which we show in the next section.

2.2.2 Testing Policies in a more Complicated Model

While the Shelling [58] example was created to challenge a very specific thought among policy makers or the public, there are models that can actually test the effect of multiple types of policies. During the Covid-19 pandemic, many models have been produced that simulate aspects of the pandemic [49]. Although many models were relatively simple, there are a couple of models that could be used for multiple types of policies. For example, the agent-based social simulation of the Coronavirus Crisis (ASSOCC) framework [19]. This is a complex model that can simulate the spread of the virus in an artificial population. It includes many aspects of life, such as different age groups, social group, work, school, (grocery) shopping, leisure activities, and different needs [44]. There are also a number of different policies that can be tested, policies such as closing specific locations, working from home, lockdown and global lockdown, social distancing, etc. [44].

All of these aspects are managed by a need-based deliberation model. In terms of the decision context, ASSOCC contains many types of information, but this information is not explicitly defined as a decision context. The information is rather spread throughout the simulation and is interwoven within the decision making code, i.e., the need-based deliberation. This interwoven information makes it difficult to add changes or additional aspects to the model. Since the model uses need-based deliberation, the agents are only capable of deliberating on the spot, one action at a time. The agents can, for example, not perform planned behaviour, which might be relevant in some policy testing scenarios. For example, when a global lockdown is coming, people might stock up on goods or even have more leisure activities. This type of behaviour is difficult to implement using a need-based approach. However, there are other communities studying agents where planned behaviour is very common, Multi-Agent Systems, for example, which is described in the next section.

2.3 Multi-Agent Systems

An agent-based simulation is by definition a Multi-Agent System (MAS). MAS are developed with the goal of guaranteeing an outcome. The agents focus on planning (using rule-based systems) and strategic decision making, agent-based simulations generally do not have this kind of advanced deliberation. However, it has been argued that agent-based simulations could actually benefit by considering techniques from the MAS community [22, 64]. A branch of MAS agents that is capable of planning is concerned with Belief, Desire, and Intentions (BDI) [6].

2.3.1 BDI Agents

BDI or Belief, Desire, and Intentions is a formalisation for plan-based deliberation for agents [6]. The agents have some knowledge about their world that is represented by their beliefs. The agents have a couple of states they want to achieve, the desires. From these desires, they choose a desire to achieve, which will become the intention. Intention serves as the desired goal state for the agent. Knowing which state the agent wants to achieve allows it to create an action sequence to get to that state, which is called planning. A known BDI multi-agents programming language is 2APL by Dastani [14]. This platform allows for easier implementation of multiple agent systems with BDI.

The execution of a 2APL agent is carried out in a fixed order, as seen in Figure 2.2. The cycle loops and has a couple of distinct steps that make it able to select goals, make plans, execute plans, and process messages of other agents. For more details, see the paper by Dastani [14]. This platform is even used in agent-based simulations to model a pandemic [53, 52]. The 2APL framework was extended and used in a large-scale pandemic simulation platform called PanSim [3].

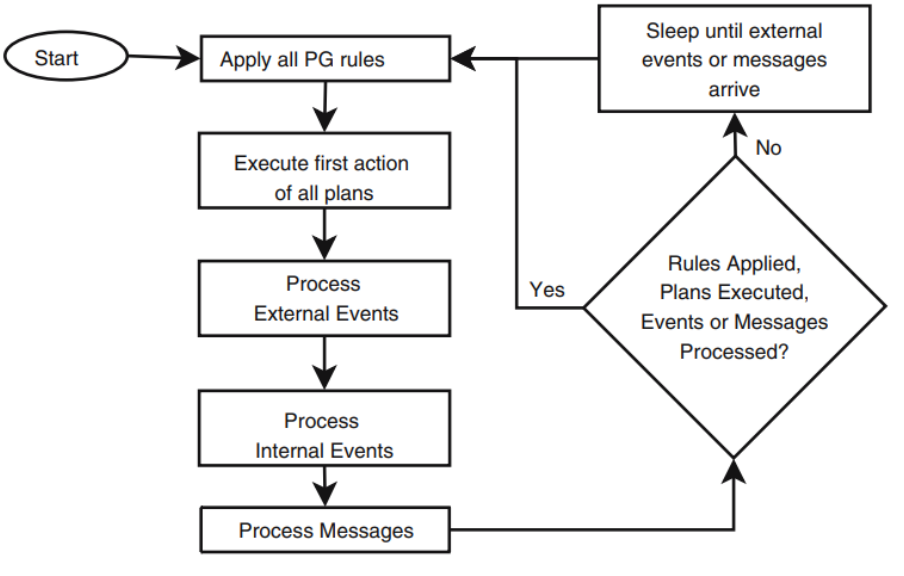


Figure 2.2: The 2APL execution cycle, from Dastani [14]

In terms of the decision context, typically 2APL agents have their internal information in the form of beliefs. They also consider the environment and the other agents to some extent. They can take the other agent's actions, location, capabilities, and intentions into account, but will only do so to achieve their own goal. They do not consider helping other agents if it does not help them in their own goals. They also do not consider norms (unless the framework is extended, see next section) or social groups.

2.3.2 BOID

The BOID framework [9, 8] extends BDI with normative aspects. BOID stands for Beliefs-Desires-Obligations-Intentions where obligation is the normative extension. The architecture allows for agents following social norms. In the goal adoption phase, the agent considers whether the goal conflicts with internalised 'social' norms. However, still as argued in Balke and Gilbert [2] this architecture lacks many advanced concepts of normative reasoning. It also does not contain explicit awareness of groups or group behaviour.

2.3.3 Problems with MAS agents

While on an abstract level both agent-based simulations and MAS are similar, as both are systems with multiple agents. Multi-Agent Systems is concerned with the agents following the plans or the protocol. The agents preferably work together, and in most MAS systems optimise their choice of action. The focus on agent-based simulations is mainly on the emergence of simulated humans. Since it is human behaviour that should be simulated, the irrational, impulse, emotional behaviour should also be taken into account. While in MAS the agent could imitate the action of another agent, it will only do so if it sees a clear benefit. The BDI agent will always solve problems in a 'rational' plan-goal-based approach. Thus, both types of deliberation are important. Fortunately, the agent-based simulation community has developed some adaptive deliberation frameworks that can choose between deliberation types.

2.4 Adaptive deliberation frameworks for Agent-Based Simulations

The previous section described agent frameworks with a rational rule-based approach. Not all agent frameworks have this fixed style of reasoning that always uses the same type of deliberation. There are also frameworks inspired by cognitive science that are capable of switching the type of deliberation depending on the situation.

2.4.1 Consumat

The Consumat framework, for example, by Jager [40]. This framework uses up to six deliberation types (see Figure 2.3. Agents using the Consumat framework select a deliberation type based on the satisfaction and uncertainty within the current decision situation. If the agent has a very high need satisfaction, it will repeat the previous action (repetition). If the agent is less satisfied, it will either try to improve its situation by choosing a similar action (if it is certain) or it will imitate other agents (if it is uncertain). When the agent has a low need satisfaction, the agent will, in case of certainty, apply satisficing and consider actions until an action meets the satisficing threshold. In the case of uncertainty, the agent will consider the actions taken by other agents that have a similar profile as the agent but are doing better right now. If the agent has a very low need satisfaction, it will instead deliberate and consider all actions within a reasonable time horizon and choose the optimal action sequence. One can see that the less happy the agent is or the more uncertain the agent is about the decision, the cognitive effort of the deliberation types increases.

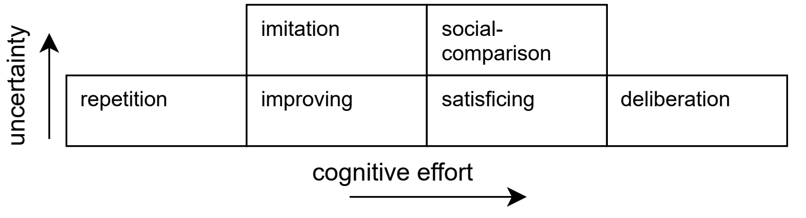


Figure 2.3: The Consumat framework, adopted from [40]

The Consumat framework combines different behavioural theories into one simple framework. It is one of the few frameworks that actually does this. Unfortunately, it lacks social concepts such as normative behaviour (e.g., wanting to be part of a group) or theory of mind (i.e., thinking about intentions, needs, goals, etc. of other agents). The deliberation selection mechanism is rational, that is, maximise utility, rather than dependent on the situation the agent is in [45]. For a more human-like decision making model, other aspects of the situation need to be taken into account. Whether the agent repeats or deliberates is not just dependent on its current level of need satisfaction, but could also depend on how familiar the situation is. With familiar situations usually repetition being performed and unfamiliar situations usually other decision processes. In the case of addictions, people could have a very low need satisfaction but still opt for the default action of repeating the addictive behaviour.

Considering the decision context, the Consumat agents consider both internal and external information. The agents consider the actions of other agents and can change their actions based on what other agents do. Due to the model being capable of using different types of deliberation, it deals with different types of information. The Consumat model could be extended to consider aspects such as norms or social groups. However, one of the downsides of the models, which would still remain when being extended, is that it only uses limited information from the context to select the deliberation type. As mentioned above, selection is based on current need satisfaction and uncertainty. These are only two aspects, while it is desired to base the deliberation type on other aspects of the context as well. In the case of interruptions from the context, for example, a friend group asking to go out for movies. This could change the deliberation type to a group-type reasoning; however, this is not currently implemented in Consumat. The principles of the Consumat model have, however, been used in a new model, the HUMAT model; this model is described in the next section.

2.4.2 HUMAT

The HUMAT framework [1], uses the same basic principles as Consumat. Both models are inspired by cognitive aspects of human decision making. However, the models are fundamentally different. While, Consumat has a mechanism that is based on selecting a specific deliberation type (from six deliberation types), the HUMAT framework is a need-based deliberation framework.

The HUMAT framework is first mentioned in the work by Antosz [1]. HUMAT is part of the SMARTEES project, which is a project to delve into social innovation dynamics [1]. The HUMAT framework uses needs to deliberate, and in principle will find the action with the highest need satisfaction. The needs, range from -1 (negative) to 1 (positive) and are split into three categories. 1) experiential: think of comfort, pleasure, safety and familiarity, 2) social needs: the desire for social approval from others, belonging, and social status, and 3) values: principles like societal goals, freedom, equality, justice, and transparency.

The framework considers all the available actions for an agent and determines the action with the highest need satisfaction. If there is a best action the agent will select that action. If there are actions with similar need satisfaction, then the framework will consider the cognitive dissonance. Cognitive dissonance happens when the potential choices leave the agent with a mix of negative and positive needs. If there is a similar cognitive dissonance, the agent will recalculate with only the experiential needs. And otherwise a random action is selected.

Social normative aspects are modelled in the framework through the social needs. Social networks play a large role within HUMAT. In Consumat, the agents were just taking in information from the environment. The HUMAT agents can actually influence the other agents as well. This is the case when there is cognitive dissonance. When there is cognitive dissonance, there can be two types of dilemmas, a social dilemma or a non-social dilemma. In a social dilemma, the agent would after performing the action of interest, positively satisfied non-social needs, but negatively satisfied social needs. The agent will then signal, which is basically trying to change the opinion/beliefs of another agent it its network. If the agent has a non-social dilemma, it will communicate with another agent in its network and asks that other agent about information; if this succeeds the initial agent will change its own opinion/beliefs.

The HUMAT framework is similar to the ASSOCC framework. Both frameworks use a need-based deliberation; however, HUMAT contains a couple of extra deliberation components for when the need-based deliberation cannot find a clear preferred action. It contains cognitive dissonance to further make a choice of an action. However, while both frameworks do consider many types of information, both internal and external, they miss the more advanced aspects seen in MAS, such as planning. Planning is something people do and would be important to have in the model.

The HUMAT framework connects many aspects of human decision-making into a socio-cognitive architecture. It contains needs, social aspects, values, normative aspects can be represented. However, compared with the MAS frameworks, BDI and BOID, it lacks that type of plan making and normative consideration. Since it is need-based, it cannot make elaborate plans when needed. By now it should be clear that there are quite some architectures for computational agents. Each architecture specialises in certain aspects of human behaviour (plan-making, normative behaviour), and some, such as the Consumat and HUMAT approaches, incorporate even multiple aspects of human behaviour. However, so far, all of the frameworks still lack some crucial aspects for social simulation agents for policy testing. As this requires an interdependent framework that combines normative and social aspects [43]. It would be good to have a categorisation on the different deliberation types available for computational agents to use as basic principles for such a framework.

2.4.3 CAFCA

The Contextual Action Framework for Computational Agents (CAFCA [30]) provides a categorisation of different decision situations (Figure 2.4). CAFCA categorises nine decision situations (cells) over two dimensions, i.e. the reasoning and the sociality dimension. The reasoning dimension contains simpler reasoning at the first row and more complex reasoning moving to the bottom. The sociality dimension includes more social aspects in columns further to the right.

		Sociality Dimension		
		Individual	**Social**	**Collective**
Reasoning Dimension	**Habitual**	Repetition	Imitation	Joining-in
	Strategic	Rational choice	Game Theory	Team reasoning
	Normative	(institutional) rules	(social) norms	(moral) values

Figure 2.4: Adopted from [30], it shows the categorisation of decision situations. In the original version of the matrix in [30] Habitual is named Automatic, the new label is introduced in [29].

The benefit of CAFCA is that it gives a general categorisation for deliberation for social agents. When modelling only a single simulation with predetermined elements, decision context is relatively easily determined. However, for a context-sensitive deliberation framework general categorisations are needed, which CAFCA happens to provide. Currently, CAFCA is the best-fitting framework for contextual deliberation in social simulation. However, it can be replaced when better frameworks are developed. Even though the CAFCA framework is actually a very abstract framework and is not implemented in any agent system, it can be used very well as a conceptual background for implementing agents. With CAFCA described, the background for the deliberation in agent-based social simulation for this thesis has been described. The next section will describe the aspects of scalability in agent-based simulations.

2.5 Scalability in Agent-Based Simulations

Scalability is an important concept in computer science [50, 36, 38, 66]. When designing a system or an algorithm, it is often important to consider scalability. If the system stops functioning after a certain amount of resources are in use, this may be detrimental to the use of the system. As explained by Bondi [5] scalability can have different forms within a network, system, or process. The article mentions four types of scalability that can be considered; load scalability, space scalability, space-time scalability, and structural scalability. In agent-based simulation, there is also not just one metric to consider in terms of scalability, but rather multiple.

From the introduction of this thesis it should have been clear that there are multiple aspects to scalability in agent-based simulation. Scalability can be related to the number of agents that the system can simulate. It can also relate to the number of sub models or other aspects that can be implemented into the Agent-Based Simulation. Lastly, it can also focus on the number of deliberative aspects that can be taken into account in the deliberation of the agents. The framework presented in this thesis is primarily concerned with the number of deliberative aspects, but this is related to the number of agents and the number of aspects as well.

2.5.1 Scaling Number of Agents in Agent-Based Simulations

Scaling the number of agents becomes increasingly relevant for studying societal phenomena. Although initial models simulate a smaller number of agents, Shelling [58] for example simulates 138 entities, some newer models simulate more than a million [52]. For example, studying the effects of measures on the pandemic could benefit from a model that can scale up to a million agents [52]. The authors achieve this using the PanSim framework [3]. This framework distributes the calculations over multiple cores using high-performance computing. The agents in PanSim are 2APL agents, these agents are in principle capable of BDI reasoning and normative reasoning. In terms of social rules, imitation among agents is happening; however, other than that these agents are relatively limited socially. Compared with the CAFCA matrix, this model still lacks social and collective columns. As discussed earlier, adding all these social concepts as extensions to BDI would be very complex and could drastically increase computational complexity.

A creative approach that does not directly scale the number of agents, but is able to represent more individuals is the work by [62]. This model starts with 2500 agents, where each agent represents one person. As soon as 100 agents are infected, the model zooms out by a factor of ten. Now each agent represents ten individuals, and the number of infected agents moves from 100 to ten. This can continue if necessary until a large number of individuals are represented. The problem with this model is that deliberation has to be abstract enough to be able to zoom out.

It would be difficult to apply this scaling approach to a model such as ASSOCC [19]. This has to do with the detailed deliberation that is specifically tailored to each agent representing a single person. The agents each have 12 needs, such as food safety, financial needs, sleep, compliance and more. If one would zoom out and abstract in such a way that each agent represents an apartment building of 100 or 1000 agents. Would those needs still make sense? Imagine the case where it is night and most agents should sleep. Most apartment agents would then have a high sleep need and would choose to sleep. However, some apartment agent could have a higher need for leisure and choose to go out. This would mean, since an apartment complex agent, represents 100 agents, literally all of them would go out on the same night, which is not that realistic. As more realistic behaviour, one would rather expect that some individuals would leave the apartment complexes, not everyone in the whole apartment at once. This effect could lead to different results in terms of spreading the disease. If each apartment has a couple of agents going out and coming back with the virus, the global spread of the virus could happen much faster. Much faster when compared to all agents in single apartment going out and getting infected, while other apartment complexes stay uninfected.

2.5.2 Scaling Sub Models in Agent-Based Simulations

As proposed in Jensen et al. [43] it is desirable to be able to scale the amount of sub models in the model for policy testing purposes. Not only should the directly relevant aspects to the phenomena be incorporated, but also some aspects that could be affected by the policies. Let us give an example on how to model the spread of a virus during the pandemic. The agents have the option to be at home, work, or perform an other activity. This model could perhaps give some insight if one wants to test just one policy. However, the model would not have sufficient alternative actions to ask a question such as *"Would the infection curve flatten, when food is brought to people's home during a lockdown?"*. To answer such a question, the other activity has to be divided up. For example, into leisure activities, grocery shopping, and luxury shopping. Now, the model could show what effect applying that model has on the infection curve. Perhaps people satisfy their most important need and stay mostly at home, flattening the curve. Or people do more luxury shopping and leisure activities and the curve becomes steeper since there are more people in less locations. These are both viable outcomes, and a model with more of these sub models that allows alternative actions could provide some insight into this. This is why having scalability in terms of sub models could be relevant.

2.5.3 Scaling Deliberative Aspects in Agent-Based Simulations

There is a wide range of deliberative models. There is the shelling model [58], where each agent considers the type (out of two types) of the neighbouring agents (up to a maximum of eight are considered) and based on this either stay or move randomly. There is also the ASSOCC [19] framework where the deliberation of the agents consider many more aspects. The agents consider their twelve needs, their disease status, the possible actions out of eight actions, the rules that apply at that moment, and the most popular action chosen by the agent's social group. ASSOCC's agents obviously consider more information in their deliberation than the Schelling agents. This makes an ASSOCC agent deliberation slower than the Schelling agent deliberation, which affects the total run time of the simulation. In the ASSOCC framework, the deliberation execution time already takes half of the total execution time of the model [42].

For realistic models for policy testing, this interdependent deliberation as seen in ASSOCC framework can be required [43]. The work also states that sufficient sub models should be incorporated in the model. If in the future the models need even more sub models than the ASSOCC framework (for example, when one wants to study the effects of tourism in ASSOCC) it can be expected that the deliberation would need to deal with more aspects, and thus be even slower. With more sub models the deliberation could scale even worse. Especially since the deliberation connects all sub models.

Most models would apply some specific optimisations in the code to decrease the execution time of deliberation. For example, pre-calculating global variables one time before the deliberation, this prevents the calculation of these variables for each agent. Or making a subset of the actions on which calculations are performed, due to some actions not being available to the agent anyway. However, at the moment, there does not seem to be a more general methodology of scaling up the deliberative aspects in deliberation of social simulation agents.

2.5.4 Focus of this Study

This thesis will focus on context-sensitive deliberation as a means to scale the deliberative aspects in an agent-based simulation. Although scaling the number of deliberative aspects does not directly aid scaling number of agents or scaling the sub models in most simulations. There are simulations in which deliberation is the bottleneck [42]. In the future with more complicated models, especially when using interdependent deliberation as seen in the ASSOCC framework [19], this bottleneck can only become more prominent. This happens when even more deliberative aspects are added. Especially when the agents need to deliberate about other agents or groups of agents. The deliberation may become computationally slow, which could hinder simulating large amount of agents. In this case, a method that scales deliberative aspects could be of use, especially when it is combined with other techniques that scale the simulation as a whole.

2.6 Conclusion

This section described the background for this thesis. It started with defining the decision context based on the existing literature on context. Then it introduced the reader to Agent-Based Simulations, Multi-Agent Systems and adaptive deliberation frameworks for Agent-Based Simulation. The concept of Consumat where different types of deliberation are used in one deliberation framework is interesting. This could allow for a more efficient deliberation, and at the same time it incorporates other types of deliberation, and as such it is more complete. The Consumat framework, however, only considers its satisfaction and uncertainty variables for deciding upon a deliberation type. Consumat does not consider other information from the decision context, thus a different approach is needed. The CAFCA matrix could be a better starting point for formalising context-sensitive deliberation. However, it still needs some work to be usable. Finally, scalability within social simulation is described. Scalability can be directed towards increasing the number of agents, the number of sub models in a simulation, but also the increasing the number of deliberative elements. The framework proposed in this thesis will focus mainly on the latter.

Chapter 3

Framework: Conceptualising Context-Sensitive Deliberation

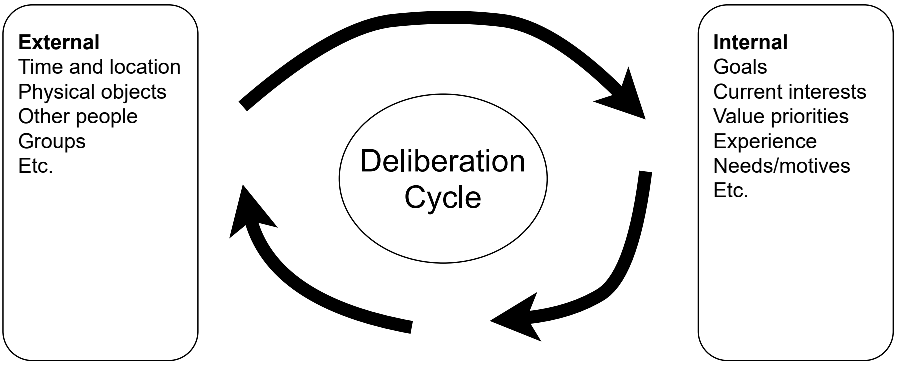
This chapter is dedicated to conceptualising context-sensitive deliberation. The first section shows what aspects context-sensitive deliberation would require, which is previously presented in Jensen et al. [45]. The second section shows the Dynamic Context-Sensitive Deliberation framework more specifically for action taking of agents. The latter has previously been presented by Jensen et al. [41]. This chapter will answer the sub research questions **RQ1.1** and **RQ1.2**, and answer research question **RQ1**.

3.1 Aspects of Context-Sensitive Deliberation

As described in Chapter 2, a new type of deliberation framework is needed. A framework that can adapt deliberation dependent on the decision context. This adaptation should not be based on a single fixed parameter, as seen in the Consumat framework [40], but rather on potentially any relevant aspect of the decision context.

3.1.1 Context-Dependent Deliberation Cycle

We propose a conceptual context-sensitive deliberation cycle that allows adaptive deliberation; see Figure 3.1. The cycle consists of three main aspects; internal information from the agent, external information from the agent's environment, and the CAFCA matrix as a guide for which type of deliberation and information to consider. The context-sensitive deliberation cycle dynamically explores the decision context as follows. It starts with less complex deliberation types and information and adds more complexity until the decision problem is solved. The context exploration should happen dynamically based on the goal or other motivation of the agent, the information from the context, and the deliberation type used, while allowing for adaptation of each of these elements during deliberation.



	Individual	Social	Collective
Habitual	1.1 Repetition	1.2 Imitation	1.3 Joining-in
Strategic	2.1 Rational choice	2.2 Game Theory	2.3 Team reasoning
Normative	3.1 (institutional) rules	3.2 (social) norms	3.3 (moral) values

Figure 3.1: Conceptual Context-Sensitive Deliberation Cycle, from [45]

Deliberation for an agent usually (that is, unless there is an important event interrupting) starts with a minimal context (external environment) and a goal or other motivation (internal state). Using this model, the agent should deliberate using the most simple deliberation type in the CAFCA matrix, i.e. repetition (1.1). If an action can be selected, the deliberation terminates. If the model fails to select an action, it should explore other types of deliberation and information based on the reason it failed. Did it fail because there was not enough information and thus it could not make a selection? Or did it fail because there was not a pre-existing plan? Does the agent need help from others? With a different deliberation type selected, the model should explore the context further, i.e., the relevant external and internal information related to the deliberation type. This deliberation process iterates, expanding the context dependent on the relevant CAFCA cell, selecting cells based on the context and decision problems encountered, or if needed, adjusting the goal or motivation of the agent. The general direction of exploration of the CAFCA matrix should be from less complex to more complex, that is, from top-left to bottom-right. The further to the bottom-right the more complex the deliberation becomes, thus if the problem could be solved by simpler methods first, the deliberation process would be more efficient.

3.1.2 Information relevance and transitions in CAFCA

To make the deliberation cycle (Figure 3.1) more concrete, we describe the information relevance and transitions for each of the cells in the CAFCA matrix. Figure 3.2 shows the relevant information per cell, while Figure 3.3 shows the transitions between the different cells.

Figure 3.2 shows the relevant information that is required from the context to make a decision using that type of deliberation. For example, in the repetition (1.1) cell, the agent only needs the accessible objects, people, and actions currently performed as this is enough to perform a default action or perform a pre-made plan. If the plan fails different information is needed, so a switch to another CAFCA cell is required. In, for example, the imitation (1.2) cell, the agent is interested in other agent's behaviour and goals, beliefs, and intentions to determine if their behaviour is relevant. By changing cells, the perspective of what is relevant to the decision context changes. The decision context is explored depending on what is relevant for the goal and the current deliberation matrix, which creates a focused decision context tailored to the deliberation problem of the agent in the given situation.

For readability purposes, we show the relevant information of previous cells (those that are directly above or to the left) in grey. The cells more to the right or bottom can always contain the relevant information from preceding (horizontally and vertically) cells. For example, in the moral values (3.3) cell, the accessible objects and people in the repetition (1.1) cell could still be relevant. Using this categorisation makes it possible to focus on relevant parts of the context and build a context specifically for the decision problem at hand.

Relevant information for each cell is also information that may hinder achieving the goal or motive even when this is not directly indicated in our matrix. For example, at the strategic level in the rational choice (1.2) cell the agent may consider stealing something. However, a conflict arises as there is a rule against stealing. To be aware of this, such rules should be part of the decision context when they become relevant, even though the rules are not explicitly mentioned in the rational choice (1.2) cell but rather in the institutionalised rules (1.3) cell. If a conflict with a rule arises, the agent moves from the strategic to the normative layer, where rules are more explicitly part of the context since now they should be evaluated (as seen in Figure 3.3).

Figure 3.3 shows potential triggers for transitioning between CAFCA cells. When the deliberation type cannot find a solution from the explored context, either the context may be explored further or a different deliberation type may be considered. Depending on the currently selected deliberation type and

	Individual	Social	Collective
Habitual	Accessible objects, Accessible people, Actions currently performed Accessible means being accessible to the DB in the current context.	Theory of Mind: G, B, I Actions performed by relevant people Accessible objects, Accessible people, Actions currently performed Relevant people are those who have a similar goal to the DB. There is a minimal theory of mind.	Theory of Group: G, B, I Expected action as team member ToM: G, B, I Actions performed by relevant people The group considered is the group that the DB wants to join. The DB need information to perform actions to belong to the group.
Strategic	Useful objects, useful people, Utility Accessible objects, Accessible people, Actions currently performed The set of objects and people is extended to include also not directly accessible objects for plan making.	ToM: Mental attitudes ToM: G, B, I Actions performed by relevant people, Utility Useful objects, useful people Relevant people are those who can aid or hinder the DB. Mental attitudes referes to the information needed to make an estimation of the actions that other agents will perform.	ToG: Mental attitudes, roles Agents in my group ToM: Mental attitudes, Theory of Group: G, B, I Expected action as team member The mental attitudes and roles are information needed for the DB to make decisions in the group. E.g. status, structure of team, mental models, roles
Normative	Related rules, Related laws, Useful objects, Useful people, Utility Rules and laws that are relevant for the current context	Related social norms People's opinion towards those norms Related rules, Related laws, ToM: Mental attitudes Social norms related to the current context. That may hinder or lead behavior of the DB.	(Moral) values of self, Theory of Mind: values, Theory of Group: values ToG: Mental attitudes, roles Agents in my group Related social norms People's opinion towards those norms Consider values of self, others, group.

Figure 3.2: CAFCA information relevance. DB = deliberating agent, G = Goals, B = Beliefs, I = Intentions, ToM = Theory of Mind, ToG = Theory of Group

context, different transitions are possible. Ending in the 'Moral' values if the decision drags on. Note that these triggers are a collection of possible triggers that is not exhaustive. Depending on the domain of the simulation and the context, other triggers could be added.

The system should generally start from the repetition cell. If this cell cannot provide a solution, other cells should be considered, but there can be different reasons. If there is information missing, perhaps new information can be found by imitating other agents (going from 1.1 to 1.2). If the goal is group-related, it could be better to consider collective information (going from 1.1 to 1.3). A

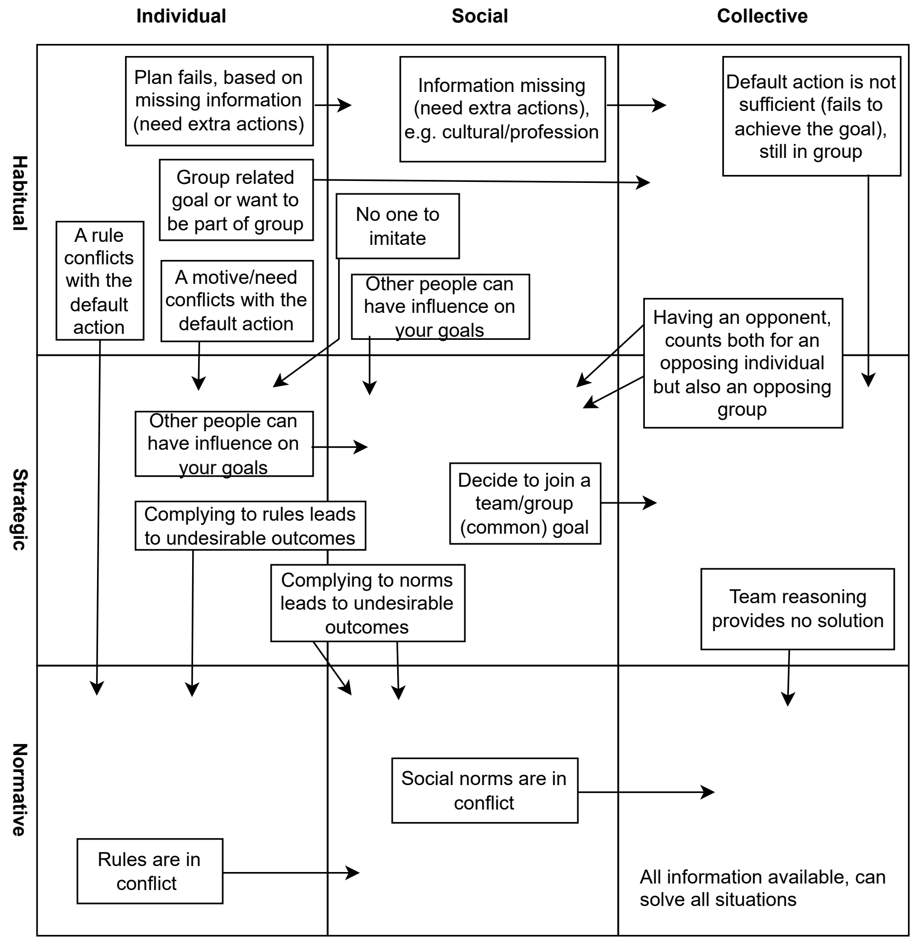


Figure 3.3: CAFCA cell transitions, adjusted figure based on the figure in [45].

goal or motive can conflict with a default action (going from 1.1 to 2.1). And lastly, it can also be a rule that conflicts with the default action the rule has to be evaluated to see if it can be broken or not (going from 1.1 to 3.1).

If the system transitions to a cell while the preconditions of that cell are not met, the system will move directly to the next or previous cell. This process is also dependent on the cell, but is related to not meeting the preconditions of the cell. For example, after moving from the imitation (1.2) cell to the joining-in (1.3) cell because the agent wants to join a group, the agent may become aware that it does not share the same goals as the group and move backwards to imitation.

In case there are conflicts or when multiple cells may be applicable for transitioning towards, one could base the decision of transitioning on the characteristics of the agent. For example, there can be agents that move faster to the social dimension to find the solution, while other agents will move down (deeper) into the matrix to do more complex but individual deliberation straight away. There could also be agents that do not even consider breaking the rules or norms, these agents would not even use the normative layer (with the exception of the moral values (3.3) cell), only in very extreme circumstances. In short, the exact reasons for switching are dependent on the agent and context.

Towards a More Concrete Framework

We refrain from explicitly stating how to implement all of the cells as this highly depends on the simulation itself. However, we can provide some typical examples of formalisations or implementations shown by the literature. Imitation can, for example, be imitating the direct neighbours or neighbours in a certain radius in a Cellular Automata implementation or imitating the agents in the same building or same network in other simulation. BDI agent theory can be used for rational choice [55] as this is problem solving that, in principle, does not consider social aspects. Game theory is a way of solving problems in the social strategic cell and there is enough literature to be found, see for example [4] for an introduction. For team reasoning typical examples are the work of Sugden [61] who explains and formalises team reasoning, or [23] which is a book formalising team work in agent systems. Institutions have been formalised by Esteva [32]. Some normative frameworks are provided by Savarimuthu [57] which gives an overview of the norms in simulation, while Castelfranchi [10] shows an architecture that uses norms. For values, one could consider the formalisation by Heidari [37].

Even with the availability of the context-sensitive deliberation cycle (Figure 3.1) and literature that describes how to implement different deliberation types, there is a step missing. The literature is too specific and does not give handles on how to connect different deliberation types. The context-sensitive deliberation cycle shows the basic aspects needed, but is not concrete enough for direct use in agent action taking. To move further a more concrete context-sensitive deliberation framework needs to be created, one that shows specifically how action taking can be performed by an agent.

3.2 Dynamic Context-Sensitive Deliberation

This section proposes how to conceptualise context-sensitive deliberation for action taking in agents. A crucial difference here with the Context-Sensitive Deliberation cycle (Figure 3.1). That is, instead of switching to a completely new context by transiting to a different cell in the CAFCA matrix. The framework will instead gradually incorporate more information from the information relevance matrix. This is the complexity by need principle, where the information considered starts out simple and slides to include more complex information until an action is decided upon. In this case, need should not be confused with needs such as a need for food or safety. Rather complexity by need should be seen as, if needed due to any kind of information from the decision context, the framework can add more complex information. The word dynamic is added to the name of the framework due to this gradual complexity by need principle.

To achieve this gradual complexity by need, rather than switching between specific cells. The framework should instead deliberate more generally and use the information and deliberation types from the matrices as a guide. It should in principle deliberate about what kind of deliberation to use, hence a meta-deliberation layer should be part of the framework. Meta-deliberation can select which deliberation and information need to be used. This section will describe the Dynamic Context-Sensitive Deliberation framework and is based on Jensen et al. [41].

3.2.1 The Framework

As discussed, meta-deliberation is needed for Dynamic Context-Sensitive Deliberation (DCSD). To deliberate on a more abstract level, Meta-deliberation should contain the most basic elements necessary for deliberation. The framework is meant for deliberation for social agents and therefore should output an action. On the meta-level, actions are thus the starting point and serve as the absolute minimal elements in our framework. Actions require plans that are sequences of actions. To create a plan a goal is needed. Goals are related

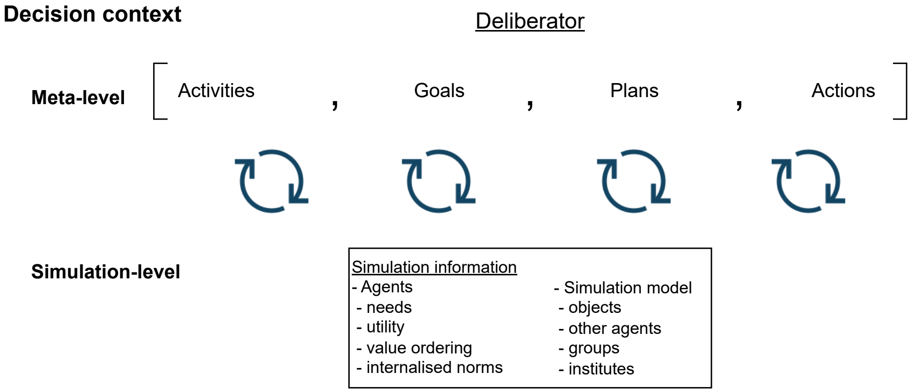


Figure 3.4: The Dynamic Context-Sensitive Deliberation Framework, adopted from Jensen et al. [42]

to activities, e.g., if the activity is *playing soccer*, goals could be *winning the game*, *playing together with friends* and *physical activity*. The *activity* is what the agent is involved in. Having activities, goals, plans, and actions as elements serve as the basis for this meta-deliberation. Meta-deliberation in the framework is thus represented by a tuple containing those four elements. Figure 3.4 shows the DCSD framework. Meta-criteria determine which element of the tuple the deliberator should manipulate. The criteria then determine how and with which information this manipulation should be performed. These concepts are explained in more detail below.

3.2.2 Elements of the Tuple

The meta-level tuple consists of four abstract deliberative elements. Those are activities, goals, plans, and actions.

Activities

An activity determines what information is relevant for the context. To give an example, when the activity is *working*, a laptop, colleagues, and tasks could be relevant information. While, for the activity *grocery shopping* information such as food, stores, and employees could be relevant. Activities serve as starting points. An activity often comes with pre-determined information such as a goal, a plan and an action. For example, the habit of grocery shopping can contain the following elements:
$Activity : grocery\_shopping$, $Goal : get\_food$
$Plan : \{drive\_to\_supermarket, buy\_food, drive\_home\}$
$Action : drive\_to\_supermarket$

Goals

Goals specify what the agent wants to achieve. A goal allows the agent to select or create plans. Selecting a plan can be done with pre-existing plans that fulfil the goal. In some situations, the agent has default plans available. The creation of plans based on goals can be performed by classic planning described in [6]. A goal makes the decision context more specific than an activity.

Plans

A plan is a sequence of actions. Plans enable achieving goals that require multiple actions. This enables agents to achieve longer-term goals. An example of a $grocery\_shopping$ plan was shown in the activities section. Habits can be represented by plans. A plan does not have to be complete to be useful. There can also be partially filled in plans such as the plan below. These plans can still be useful to the agent as the agent in this case only has to deliberate about which mode of transport to take (e.g. $TRANSPORT = bicycle, car$).
$Plan : \{drive\_to\_supermarket\_by\_TRANSPORT, buy\_food,$
$drive\_home\_by\_TRANSPORT\}$

Actions

Agents use actions to manipulate themselves and the simulated world. The goal of the framework is to determine a single action for the agent. The action

buy_food removes *food* from the store and gives the *food* to the agent. Actions can also have effect on the internal state of the agent. The action eat_food removes *food* from the simulated world and decreases the hunger need in an agent. Some examples of actions are:
$Actions : buy\_food, eat\_food, sleep, work, drive\_car\_home, play\_soccer$

3.2.3 Meta-Criteria, Criteria and Simulation Information

The framework uses meta-criteria and criteria to manipulate the tuple. Meta-criteria determine which element the deliberator should manipulate. More precisely, whether to add (expand) elements or subtract (narrow) elements. The criteria then determine how this manipulation should be performed and what type of information is required from the context.

Meta-Criteria

The meta-criteria are defined as being one of the following.
$Meta\_criteria : \{narrow\_activities$, $narrow\_goals$, $narrow\_plans$, $narrow\_actions$, $expand\_actions$, $expand\_plans$, $expand\_goals$, $expand\_activities\}$

The indicated order should be seen as default; however, it can be deviated from in certain decision contexts. By default, if there are element types that contain multiple elements, they will be narrowed first, e.g. if there are multiple activities relevant, the framework will prioritise selecting one activity. If no element type contains more than one element, then by default the element types will be expanded starting with the actions. The mechanism of expanding and narrowing will be further explained in Section 3.2.4.

Criteria

Selection between activities, goals, plans, and actions can be performed by a vast amount of methods. The criteria determine how and based on which type of information the elements in the tuple have to be manipulated. As indicated before, there is a vast amount of information (see Figure 3.2) and deliberation types (see CAFCA Figure 2.4) available. We will not describe all of them, but rather give some examples of criteria below. Criteria should be selected based on the complexity by need principle, generally starting with the least complex criteria and gradually increasing complexity. For example, in most situations, using a default action is preferred over deliberation about norms to choose an action. Also, criteria that require only information about the agent's internal state are prioritised over criteria requiring information from the other agents' mental state. As a general heuristic, one could consider Figure 3.2 by using information from the top-left first, moving gradually to the bottom-right as deliberation continues.

- **Default heuristic:** When there is a default option, take it. For example, for going to work, most people have a default mode of transportation. One could read Gigerenzer [35] for more examples of and information about heuristics.

- **Typical:** In some contexts some activities, goals, plans, or actions are typical in general or for the agent. For example, on a Friday evening it could be typical to go to a bar, go to the cinema, watch movies at home.

- **Urgency:** Some activities can be more urgent than others. E.g. sometimes an important meeting at work may make a person skip breakfast. This could be based on which need is more important at that moment.

- **Utility:** Utility can be a criteria for choosing between actions [33]. It can be determined individually, but also using game theory [4] or team reasoning [61]. The aspiration and take-the-best heuristics can be used.

- **Preference:** It is possible to make a preference ordering. There can be a default preference, but it can also come from, for example, values [37], but also rules and norms [10]. The aspiration and take-the-best heuristics [35] can be used.

Simulation Information

The information that can be relevant for the deliberation is determined by the simulation model implemented. Basically, any information in the simulation can be considered part of the decision context. Both physical such as the agents, places, affordances, but also social aspects such as social networks, norms, and agent internal state. In principle, all of this information is readily available as it is formalised and implemented. The framework can draw in any of this information to expand the decision context and manipulate the tuples. As mentioned above, this relevant information can be categorised using the information relevance matrix (Figure 3.2).

3.2.4 Manipulating the Tuple

The goal of the framework is to find a single action for the agent. The framework achieves this by adjusting the elements in the tuples using information from the simulation. Figure 3.5 gives an example of a deliberation process that includes both expanding and narrowing the tuple.

1) In this specific example, the deliberator has the activity: *Leisure* and the goal: *Hang out with friends.* 2) Since there are no plans or actions available, the framework will select the meta-criteria: *expand_plans* by default. As criteria: *Typical plans by goal* is selected, which can be a low computational cost method to find plans when a goal is known. In this specific situation, the agent has two plans that are typical and can satisfy the goal: *Hang out with friends.* Those are *Go to the pub* and *Go to the cinema.* 3) Since there are now two plans

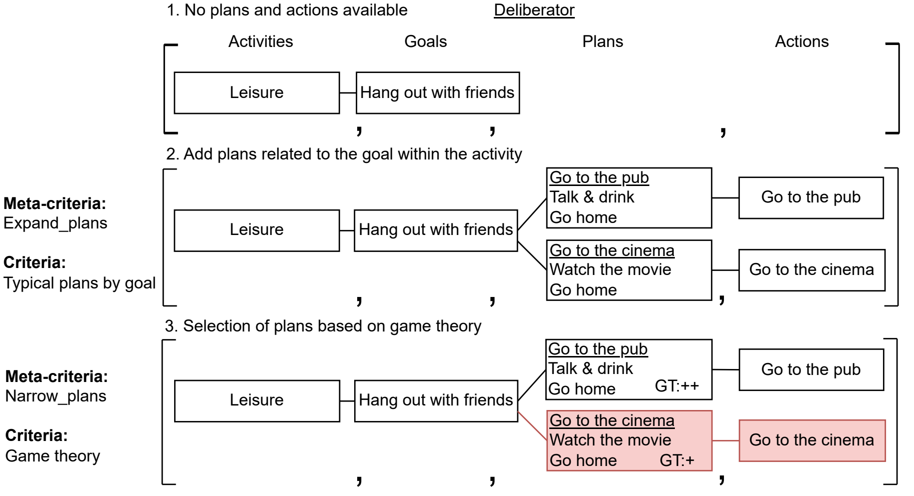


Figure 3.5: Example of manipulating the tuple. The plans are expanded based on the available goal and then narrowed based on game theory.

available, the deliberator chooses the meta-criteria: *narrow_plan*. The goal is to *hang out with friends*, to succeed, the agent needs to be with its friends. To incorporate the preferences of the friends in the decision making *game theory* is selected as criteria. The agent uses relevant information for game theoretical computation, e.g. its own preferences and the expected actions of friends. And based on these calculations, attaches a score to each plan. In this case, we assume a preference over *Go to the pub* (two pluses) compared to *Go to the cinema* (one plus). *Go to the pub* is selected as the preferred plan, thus, giving the agent one action ending the deliberation cycle.

While the tuple of this framework contains BDI [6] concepts such as goals, plans and actions. It should not be seen as a BDI framework since the deliberation is different. A typical BDI framework uses plan-based reasoning. This framework uses a variety of different deliberation methods dependent on the available information. Sometimes, a default action, sometimes imitate other agents, sometimes utility-based deliberation. The aim is to actually use other simpler deliberation methods before using more complex method such as plan-based reasoning.

As a side note, the tuple does not have to be used completely. The elements used depend on the simulation model in question. If the specific simulation does not require activities, goals, or plans, they can be left out. Actions are, however, essential as the framework is designed for action taking, so it should be able to select an action.

3.2.5 Conceptualising DCSD for a Simulation

The DCSD framework with the information and transitioning matrices gives a handle on how to conceptualise DCSD for a simulation. However, it would be difficult to specify these handles, as the exact conceptualisation of the DCSD depends on social simulation in question. Social simulation models vary greatly and often require different types of information.

In the CAFCA paper [30] a simulation model is analysed about extortion racketeering by the Italian Mafia in Sicily [59]. This model is an example of a strategic-social model of decision making. They propose that the model can be extended by adding normative or collective aspects. Adding these aspects can have different effects on the capabilities of the agents and thus on the outcome of the model.

Wijermans and Verhagen [65] take an existing simulation on the analysis of governance of common pool environments in the form of fishery and extend the agents with more deliberation types. They assess, using the CAFCA matrix, that the initial simulation only contains strategic-social components. And that a collective component needs to be added to better model the phenomena.

Another example is the previously mentioned Agent-Based Social Simulation of the Coronavirus Crisis [17], ASSOCC. It is a framework that simulates the spread of the Covid-19 virus in a population. The agents use need-based deliberation to determine their next action. According to Jensen [42] the deliberation of these agents considers the information related to the following cells, repetition (1.1), rational choice (1.2), institutionalised rules (1.3), and game theory (2.2).

These models do not only deal with different information because they have different domains. That is, governance of common pool fishery [65], extortion racketeering by the Italian Mafia, and the effects of policies on a global pandemic [19]. They also use different types of deliberation because a different phenomenon is simulated. Thus, when conceptualising DCSD for a specific simulation some intermediate steps should still be taken. Categorising the relevant information (as seen in [42] and determining the deliberation switches can be a good starting point for conceptualisation.

3.3 Conclusion

This section described context-sensitive deliberation. It first introduced aspects that are necessary for Context-Sensitive Deliberation, based on the work in [45]. Then it showed the Dynamic Context-Sensitive Deliberation framework as shown in [41].

3.3.1 Answering Sub Research Question 1.1: What are the main aspects necessary for context-sensitive deliberation?

This question is answered in the first part of this chapter. The main aspects required for context-sensitive deliberation are the following: 1) a cycle that cycles between deliberation and gathering more information for deliberation, 2) a categorisation of information from simpler to more complex information, and 3) explicit triggers to switch to different deliberation and thus different types of information. These three aspects are explained in the text and shown in the following figures, 1) Abstract Context-Sensitive Deliberation Cycle (Figure 3.1), 2) CAFCA information relevance (Figure 3.2), and 3) CAFCA cell transitions (Figure 3.3).

3.3.2 Answering Sub Research Question 1.2: How to formalise context-sensitive deliberation as framework for agent action taking in agent-based simulations?

This question is answered in the second half of this chapter. This part of the chapter makes the more abstract framework shown in the first part of this chapter more concrete. The framework presented in this part is the Dynamic Context-Sensitive Deliberation framework. At the core of the framework is the meta-deliberation which consists of actions, and possibly plans, goals, and activities (see Figure 3.4). These sets can be manipulated by the information provided by the context. The Meta-deliberation keeps the framework dynamic as it is possible to do simple deliberations which is efficient and when necessary slide to more complex deliberation. By formalising it in this way, the framework can be used to select actions for agents, as shown in the example in Figure 3.5.

3.3.3 Answering Research Question 1: How to formalise context-sensitive deliberation?

By answering the two sub questions, this research question can be answered. The first sub question provided information about the aspects necessary for context-sensitive deliberation. This provided initial handles in the form of the information relevance and state transitioning matrix. These handles can be used to categorise the deliberation of a social simulation. The second sub question described how context-sensitive deliberation can be formalised to provide an actions for an agent. This part provided the Dynamic Context-Sensitive Deliberation framework which provides a formalisation for context-sensitive deliberation for agent action taking.

Chapter 4

Methodology

This chapter is dedicated to explaining the methodology for evaluating the realism and scalability of DCSD. First, an agent-based simulation will be selected as a use case. The ASSOCC framework seems to be a good candidate since it has an extensive deliberation system that contains social and normative aspects. The ASSOCC framework suffers in terms of scalability, mainly due to complex deliberation being the bottleneck. Section 4.2 will explain the ASSOCC framework and especially its need-based deliberation. Section 4.3 explains how DCSD is located in the ASSOCC deliberation model on an abstract level. It proposes to compare the ASSOCC framework with a DCSD variant of the ASSOCC framework, DCSD ASSOCC. Section 4.4 will explain how to measure the retaining of realism in the ASSOCC framework. This is done using criteria and more detailed time series analysis. In Section 4.5, it is explained how to measure scalability when comparing deliberative models. This is done by comparing the deliberation execution time when changing the deliberative models and the number of agents.

4.1 ASSOCC as a Use Case

Based on the works of Dignum et al. [19], Dignum [18] and Jensen et al. [43] it can be argued that the ASSOCC framework is a relatively realistic model for modelling restrictions and their effects on the spread of the COVID virus. It is a rather complex model including medical, social, and economical aspects. It strikes the balance between a more detailed model that is still abstract enough to be useful for spreading of the Covid virus [18]. It can be used for many possible scenarios, from closing of specific types of buildings, to testing the effect of different cultures, or testing track and tracing apps. All these and more scenarios have been discussed in detail in the ASSOCC book [19].

To represent all these different scenarios, the ASSOCC framework contains many aspects of life. The ASSOCC deliberation model is a need-based deliberation model that can consider all these aspects of life at the same time. The need-based model serves as an abstract umbrella that connects all actions and aspects of life into an interdependent social deliberation model [17]. This already satisfies two of the three requirements needed for a realistic model for policy testing [43].

The third requirement for realism to make useful predictions is to have a large enough number of agents to represent all different kinds of groups and people in the simulation [43]. This requires scalability of the simulation. The ASSOCC framework is capable of simulating 4000 agents in 25 minutes, which is put as a practical boundary in [63]. However, simulating 10,000 or even a million agents is impractical due to the time it will take. Adding additional components to the model or increasing the number of agents will further increase the run time. All in all the ASSOCC framework cannot be further scaled up unless other techniques are applied.

If we disregard the inefficiencies of the Netlogo platform and purely consider the run time of parts of the ASSOCC framework, it becomes clear where the bottleneck lies. We perform a typical run of the ASSOCC model found on GitHub[1]. The model is run with 350 households, which is about 1000 agents and a random seed of two. A quick measurement of execution time indicates that more than half of the execution time, about 55.6%, comes from the deliberation. The other, non-deliberation processes, such as spreading the disease, performing the actions of agents, updating the beliefs, and other functions all take less time. However, it makes sense that deliberation is the largest bottleneck, as it takes into account all information for each agent every time step, so it can be expected to take up most execution time.

Thus, ASSOCC can be argued for to be a realistic simulation however, it is not very scalable. The deliberation is the main bottleneck; more specifically, the need-based deliberation is slow since it considers all the information all the time for all the agents. Dynamic Context-Sensitive Deliberation uses only relevant information to deliberate which is expected to increase scalability while retaining realism. Thus, the ASSOCC framework is a good fit for evaluating DCSD. The next section will present the relevant aspects of the ASSOCC framework.

4.2 The ASSOCC Framework

The ASSOCC framework [19] is an agent-based simulation framework that supports decision makers on the Covid crisis. As stated in [44], "the purpose of this framework lies in providing support for stakeholders for making informed decisions regarding the management of the Covid-19 disease." The framework achieves this by modelling on one hand the psychological and social aspects relevant in a crisis situation (e.g. needs, norms, habits, social network) and on the other hand modelling features relevant for policy makers (e.g. global

[1] `https://github.com/lvanhee/COVID-sim`, commit: 3ba4d3f

lockdown, quarantine, closing schools). The model represents a city that contains a variety of locations that its population can visit (e.g., schools, shops, workplaces, leisure places). The actions the agents can perform are tied to the locations. If the agent wants to shop it should go to the shop, or if the agent wants to work it should usually go to the workplace. The agents use a need-based deliberation system to determine their daily life behaviour. There are in total twelve needs, e.g. health, risk-avoidance, sleep, autonomy, etc. This deliberation system evaluates from all the available actions which actions give the highest potential need satisfaction. The agents are essentially optimising their own need satisfactions. This will be explained in more detail later in this chapter. Agents can become infected with Covid-19 disease when they are in the same location as other infected agents. Based on an epidemiological model, over time, this will produce an infection curve which represents how many agents are infected. By analysing the infection curve and the behaviour of the agents, the model can be used to understand the effectiveness of non-pharmaceutical interventions. Figure 4.1 shows an overview of the elements in the model. For our validation purposes we will adjust the deliberation function (surrounded by thick dotted lines on the right) of the ASSOCC model, in principle the rest of the model stays the same. The model is written in Netlogo and the Original implementation can be found at GitHub[2].

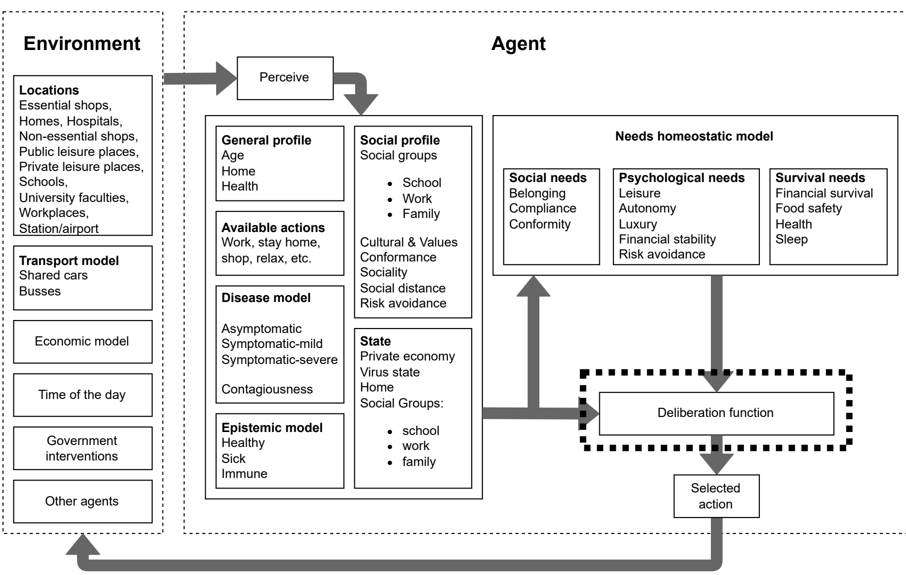


Figure 4.1: High-level overview of the framework's components from [44]

[2]https://github.com/lvanhee/COVID-sim

4.2.1 Locations and Actions

First we describe the relevant aspects of the ASSOCC model (which is completely described in [44]). The agents can visit the following locations: essential shops, homes, non-essential shops, private leisure places, public leisure places, schools, university faculties, workplaces and hospitals. Since we do not study the effect of the spread of the virus in this paper, we excluded the hospitals and migration. The locations determine rigidly what action the agents can perform. E.g. at a workplace agents can only work, or go to a different location. Note that some agents work at for example an essential shop, during working hours the action is still *work* for those agents. The following actions are defined.

Actions = {Rest at home (*RH*), Work (*W*), Study at school (*SS*), Study at university (*SU*), Leisure at public leisure (*LPU*), Leisure at private leisure (*LPR*), Buy at Essential Shop (*BE*), Buy at Non-Essential Shop (*BNE*), Work at Home (WH), Getting treatment at Hospital (TR)}.

4.2.2 Time and Age Groups

The time is represented through four slices of the day: *morning*, *afternoon*, *evening* and *night*. Each of them have different implications for the agents. For example, in the night the agents sleep, while in the other parts of the day they go to their jobs or other places. The days of the week are explicitly modelled and there is a difference between weekdays (when agents study and work) and weekends. The agents are represented with four different age groups. The *young* representing the age group 0-19, the *students* representing the age group 20-29, the *workers* representing the age group 30-69 and the *retired* representing the age group 70+. The children have limited actions, only rest at home, study at school and have leisure time. The students study at the university.

4.2.3 The Agents' Needs

The twelve needs are represented by the following set $\mathcal{N}$={*food_safety*, *fin_survival*, *sleep*, *health*, *conformity*, *compliance*, *risk_avoid*, *fin_stability*, *belonging*, *autonomy*, *luxury*, *leisure*}. The needs are modelled as a tank whose content, although dependent on the specific need, usually diminishes over time. Agents need to perform specific actions to fulfil the needs. A fully satisfied need has a value of 1. A depleted need has a value of 0. The lower the value compared to the other need values, the more *salient* the need is. Certain actions can have a positive or negative effect on the needs. Table 4.4, which is shown later in this chapter, shows a simplified mapping of actions and their effect on relevant needs. The Original ASSOCC deliberation calculates for every action the expected need satisfaction and will choose the action that has the highest overall expected need satisfaction.

4.2.4 Covid-19 Virus in the Model

The disease model is modelled after the Susceptible, Exposed, Infectious, Resistant (SEIR) model [47]. In a typical run, the agents start out as susceptible, with the exception of three agents that start out exposed. The exposed agents will become infectious and can infect susceptible agents that consequently become exposed. Most agents will become resistant; however, some agents may die. The actual disease model contains more stages and is described in detail in [44]. Generally, agents will seek treatment when they are in either the *hospital-to-death* or the *hospital-to-rec* state, however it also depends on the needs. The disease model in the ASSOCC model that will be used for our experiments is the Oxford model. This is based on the simulation by Hinch et al. [39]. This model has the following disease states available for the agents.

- **Infected, but not hospitalised:** just-contaminated, asymptomatic-to-recovery, pre-symptomatic-to-mild, symptomatic-mild, pre-symptomatic-to-severe, symptomatic-severe, infected-critical, severe-to-hospital, severe-to-rec, mild-to-rec.
- **Infected and needs to be hospitalised:** hospital-to-death, hospital-to-rec.
- **Other:** healthy, immune, dead.

In the list one can observe a difference between asymptomatic and symptomatic disease states. Sometimes agents are infected; however, they are not aware that they are infected. The agents have a *believe infected variable* to account for whether they believe they are infected which does not have to align 100% with their actual infected status. An infected agent can be asymptomatic and not believe it is infected. But it can also happen that an agent is not infected and beliefs to be infected due to fake symptoms. It is actually the believe-infected variable that will influence the agent's decision making. Making e.g. the agent is more likely to stay at home when it believes it is infected, through the effect on the needs.

The spread of the virus is calculated for agents at the same location, where infectious agents have a probability of infecting susceptible agents. The exact calculations are described in Jensen et al. [44]. The spread of the virus is affected by the number of infectious and susceptible agents at a location, the infectiousness of the infected agents, the density factor of the location, and whether the agents apply social distancing.

The agents can choose to apply social distancing at a location. This decreases the probability of getting infected. This also has effect on the needs such as risk-avoidance, compliance and conformity. This is explained in more detail in the Section 4.2.6 and the full model is presented in Jensen et al. [44].

4.2.5 The Relevant Policies

The ASSOCC model contains many government policies that can be used to affect the spread of the virus. Table 4.1 shows the interventions. The full table of interventions can be found in Jensen et al. [44]

Policy	Description
Work from home	The agents should work from home. This is only possible for agents who work at some workplaces.
Global lockdown	Every citizen is supposed to stay at home for a certain time period. This is to slow the spread of the virus, however agents can still break quarantine.
Self-isolation and self-quarantine	Agents that have COVID-19 stay at home (including home-office), but are still allowed to go shopping.
Social distancing	Agents are required to keep a certain distance to other agents, e.g. 2 meters. If this is enabled there is a smaller risk of agents infecting each other.

Table 4.1: Description of policy interventions, from [44]

4.2.6 ASSOCC Need-Based Deliberation

The agents deliberate using their needs. Figure 4.2 shows the deliberation cycle. The agents start deliberation by considering the available actions. They are based on specific information as indicated in the figure: time, age, interventions. Secondly, the agents will determine for each available action the expected total need satisfaction. This is done by using all information from the needs, and many other aspects of the simulation, e.g. active interventions, network actions, and agent status. Finally, the action with the highest overall need satisfaction is chosen and performed by the agent. As typical in social simulations the agents actions update the world. After all agents performed their action, other aspects of the model are updated. Then time moves forward, and the agents will start deliberating again.

The Default Action Schedule

In the ASSOCC model the agents generally follow a schedule dependent on their age group and the time. Table 4.2 shows this schedule. Workdays and weekend days are differentiated for all the age groups except for the retired. If there is no specific action desired at the specific time then it is specified with free choice. The deliberation system is designed in such a way that usually agents will choose the default action, but can deviate when there is a high

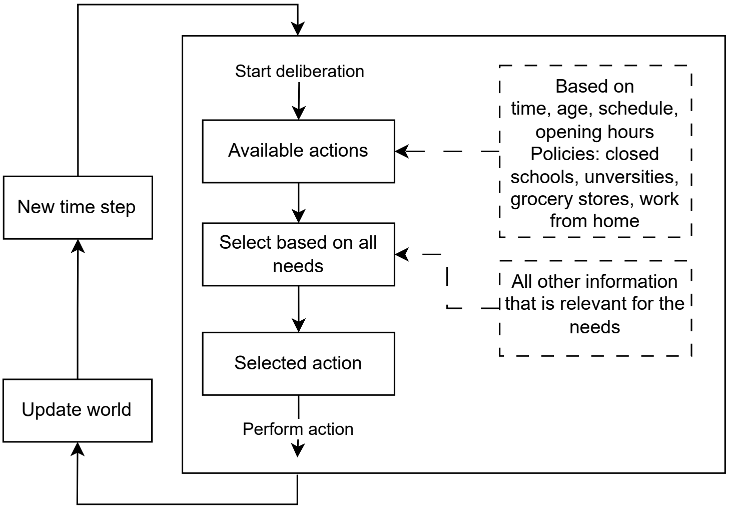


Figure 4.2: ASSOCC Need-Based Deliberation

motivation to do so. E.g. very salient sleeping need, might make the agent rest at home instead of going to work. However, not all actions are always available, when the actions are available is described in the next section.

Table 4.2: The daily schedule of the agents dependent on the age of the agent, adopted from [44]. C: Child, S: Student, W: Worker, and R: Retired. WD: workdays, and WE: weekend days.

Age Group	Morning	Afternoon	Evening	Night
C (WD)	School	School	Free choice	Home
C (WE)	Free choice	Free choice	Free choice	Home
S (WD)	University	University	Free choice	Home
S (WE)	Free choice	Free choice	Free choice	Home
W (WD)	Work	Work	Free choice	Home
W (WE)	Free choice	Free choice	Free choice	Home
R (always)	Free choice	Free choice	Free choice	Home

The Available Actions

Table 4.3 describes the action codes and their availability. The code corresponds to the actions mentioned in Section 4.2.1. The availability column shows that the rest at home (RH) action is always available. This means that even when an agent is supposed to work it does have the availability to rest at home. The study at school (SS), study at university (SU) and work (W) actions are tied specifically to working hours and their respective age groups. Some workers can work from home (WH). The leisure and shopping actions, LPU, PLR, BE, and BNE are available to agents during times when they have free choice. Although the shop is open, the worker, student, and children agents cannot visit the shop for shopping during working hours. This is where the ASSOCC deliberation model is simplified. It cuts of certain actions for specific situations, for simplicity and efficiency sake. The getting treatment at hospital (TR) action is available when an agent has visible symptoms or believes to be sick.

Code	**Availability**
RH	Always
SS	Children only, during working hours.
SU	Students only, during working hours.
W	Workers only, during working hours.
WH	Workers that can work from home, during working hours.
LPU	Only at free choice times (Table 4.2).
LPR	Only at free choice times (Table 4.2).
BE	Only at free choice times (Table 4.2), closed on Sundays.
BNE	Only at free choice times (Table 4.2), closed on Sundays.
TR	Agent who believes it has symptoms.

Table 4.3: Availability of actions: Working hours are Mo-Fr, the morning and afternoon.

Social Distancing

In addition to determining the action, the agent will also determine whether to social distance or not. Social distancing will reduce the disease transmission to about 8% of the normal value. Social distancing is related to the risk avoidance, conformity, and compliance needs. The agent will satisfy the risk-avoidance need if it choses to social distance, it will de-satisfy the risk-avoidance need if it choses to not social distance. The conformity need is satisfied if the agent applies the same social distancing choice as its social network. The compliance need is affected by the social distance choice and whether social distancing is required by the government.

Calculating The Optimal Action

The need-based deliberation calculates for all available actions the summed expected need satisfaction. The available actions are determined based on the type of the agent and the time as shown in Figure 4.2 and Figure 4.3. The summed expected need satisfaction is calculated by evaluating how the action will influence the needs if the action were to be taken. The detailed formalisation of need-based deliberation can be found in detail in Jensen [44]. Rather than explaining the formalisation, the effect of actions on the needs is represented here in a more general way.

Need	**RH**	**SS** **SU**	**W**	**WH**	**LPU** **LPR**	**BE**	**BNE**	**TR**
Risk-avoid.	+		**-**	+	**-**	**-**	**-**	+**S**
Compliance	**-**	+	++**NL**	+**L**	**-**	**-**	**-**	**-**
Fin. Stab.			+	+		**-**	**-**	
Belonging	+	+		+	+		+	
Leisure	+				++			
Luxury							+	
Autonomy	**-**	+	++	+	**-**	**-**	**-**	**-**
Food Safety						+		
Fin. Surv.			+	+		**-**	**-**	
Health	+**S**	**-S**	**-S**	**-S**	**-S**	**-S**	**-S**	++**S**
Sleep	+							+
Conformity	*The preferred action is dependent on the network*							

Table 4.4: Need satisfaction of each relevant need for each action. Based on Appendix C in The ASSOCC Book [19] The needs in red only apply during working hours for non-retired agents. NL: No lockdown/quarantine, L: Lockdown/quarantine, S: Agent is sick.

Table 4.4 shows roughly the effect of the actions on all the needs. Some needs are abbreviated for table size purpose, Risk-avoid. is the risk-avoidance need, Fin. Stab. is the financial stability need, Fin. Surv. is the financial survival need. In the table a plus signifies a positive effect on a need, i.e. increasing the level which decreases the salience of the need. A minus signifies a negative effect on a need, i.e. decreasing the need level which increases the salience of the need. Two plusses indicate that there is a larger positive effect than one plus. For example, the leisure need is better satisfied by a leisure action (two plusses) than the resting at home action (one plus).

Generally, the need levels will decrease over time if no action is performed to satisfy them. This ensures that the agents keep satisfying their needs by choosing actions. If an agent cannot perform a specific action for a long time, the corresponding need becomes more salient over time and the agent will eventually perform an action that satisfies that need. This deliberation system will always consider all information to be able to make sure needs are not forgotten. However, this also slows the whole system down. Dynamic Context-Sensitive Deliberation can be expected to better scale the deliberation. To evaluate this a proper experiment has to be conducted. The next sections will explain how the DCSD will fit in the ASSOCC framework's deliberation and how to empirically compare the two models.

4.3 Evaluation: Comparison of the Models

This section briefly explains the difference between the Original ASSOCC model and the DCSD version. As shown in Figure 4.1 only the deliberation function of the ASSOCC model will be adjusted in the comparison experiment. Figure 4.3 shows the deliberation of the Original ASSOCC model on the left and on the right shows the DCSD version of ASSOCC, which will be called DCSD ASSOCC from now on.

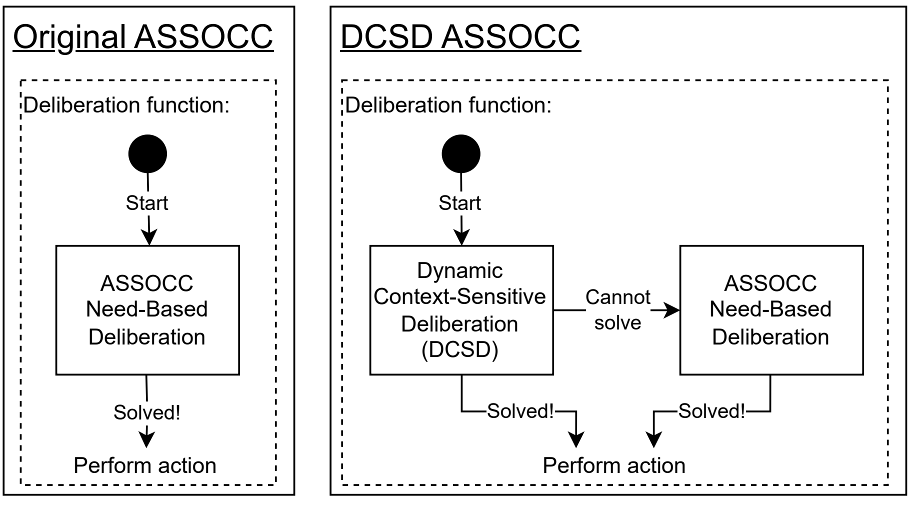


Figure 4.3: The abstract deliberation functions of Original ASSOCC (left) and the DCSD ASSOCC (right).

The Original ASSOCC deliberation function starts and uses need-based deliberation to find the optimal action based on the needs. This was extensively described in the previous section. After an action is selected the action can be performed. The DCSD ASSOCC deliberation function will initially use DCSD, a faster deliberation inspired by the framework, to find an appropriate action for the situation. If the DCSD succeeds the agent will perform the action; if it does not succeed, the slower need-based deliberation will be used as a backup.

4.3.1 DCSD Versions

The DCSD will be modelled after the information relevance and cell transitioning matrix, respectively Figure 3.2 and Figure 3.3. As described in Jensen et al. [42] deliberation in the ASSOCC framework uses information from the following cells: repetition, rational choice, institutional rules, and game theory. This creates four DCSD versions, where the most simple is the Habitual DCSD, only using information from the repetition cell. Secondly, the Strategic DCSD, using information from the repetition and rational choice cell. Third, the Normative DCSD, using information from the repetition, rational choice, institutional rules cell. And fourth, the Social DCSD, using information from all the four mentioned cells.

After creating the Social DCSD it can be analysed to see if further improvements have to be made. Perhaps some specific decision situations can be solved by expanding the DCSD model even more. This will prevent deliberation to use slower need-based deliberation, making the deliberation as a whole quicker. This fifth optimised version of the DCSD is the Full DCSD. This version of the ASSOCC model will also be called DCSD ASSOCC later in this thesis. These DCSD versions will be explained extensively in the Chapter 5.

4.3.2 Retaining Realism and Increasing Scalability

The dynamic as shown in Figure 4.3 is inspired by Kahneman's [46] thinking fast and thinking slow principle. In theory, this kind of model should be capable of fast, efficient deliberation most of the times and slower, complex deliberation sometimes. Leading to a system that is both quick, thus scalable, and retains the behavioural output of a more complex model, thus realistic. However, to test whether this works in practice, should be assessed by empirically comparing the models.

The following two sections in this chapter will explain how the Original ASSOCC and DCSD ASSOCC will be compared. Section 4.4 will explain how the models are compared in terms of whether they retain realism. This comparison is based on determining criteria based on the behaviour of agents and the infection curve in the Original ASSOCC models. To state that the DCSD ASSOCC retains realism it should provide output that fits within the defined criteria. Section 4.5 will explain how to measure an increase in scalability. To do this, the deliberation execution time of both models is compared. It can be expected that the DCSD ASSOCC deliberation is faster than the Original ASSOCC deliberation.

4.4 Validation: Realism

As indicated by Jensen et al. [43] three requirements are relevant for realism in social simulations for policy testing. Those are multiple aspects of life, interdependent social behavioural system, and scalability. Not all of these aspects must be explicitly considered for determining whether the DCSD retains realism. Given that the DCSD only changes the behaviour of the agents and not the elements in the simulation, the aspects of life in the simulation remains unchanged. The DCSD is then only required to make use of these multiple aspects of life. Measuring the scalability is covered in the next section. Thus, the main requirement that will be analysed is the behavioural system, more specifically the output of the agents actions and the effects of those actions.

As argued at the beginning of this chapter and also in Dignum [19], the ASSOCC framework contains agents that behave realistic enough to make useful predictions in the COVID crisis. If we can implement DCSD into the ASSOCC framework, and show that this DCSD ASSOCC model portrays similar enough behaviour, then the DCSD ASSOCC is also realistic enough. However, this comparison of behaviour is not trivial as the ASSOCC framework is a large simulation with many potentially interesting parameters. How this behavioural comparison will be performed will be explained in the next section.

As of now, there does not seem to be much literature on comparing the realism of behaviour between two models. There is some literature where models are compared, for example, the work by Edmonds [27] which compares an older model of cod fishing with an improved model and shows where the older model made mistakes. However, this is not a comparison that assesses the similarity of behaviour between the agents in two models.

Often when thinking about measuring the similarity of two models or experiments, a statistical comparison comes to mind first. However, due to the stochastic nature of social simulations, especially more complex simulations such as ASSOCC, this may not be the right tool for the job. Even if the settings are all the same and the only change in the model is a different random seed. The results between one run and another run would not be statistically similar. There can be some variation in those runs even though the behaviour and patterns in general are the same.

Instead, a less strict measurement is required that allows for some variety between individual runs. This can be achieved by defining a couple of criteria that define a range of a certain output parameter. This output parameter can be a specific action frequency at a specific time. For example, the amount of children that chooses the study action during working hours. Or, the time in ticks at which the infection curve peaks. These criteria can be fit specifically to the behaviour of agents in the ASSOCC model. This of course needs to be complemented with a more in-depth analysis of the behaviour and other output over time, but criteria could already give a good initial indication of similarity between the models.

4.4.1 Defining Criteria For Realistic Behaviour

The criteria are divided into roughly three subgroups. 1) The criteria for daily life activities, such as leisure and shopping, excluding obligatory activities. 2) The criteria for obligations such as measuring how often agents actually work and study when they are supposed to. 3) Criteria related to the virus, such as being at home when being sick, the amount of agents staying in quarantine and when the peak of infections happens. For the implemented criteria, see the *context_ criteria_ measurements.nls* Netlogo file in GitHub [3].

Criteria For Daily Life Activities

The first criterion assesses whether the agents perform their daily life activities. These criteria are based on the ASSOCC agents' schedule as seen in Table 4.2. One of the more simple criteria to start with that applies to all the agents is based on that agents are supposed to be resting at home when it is night. This also applies to agents who are sick as these agents are supposed to be resting at home anyway. The agents have the option of doing leisure activities in the night, however, in the ASSOCC model it is rather the exception that agents do leisure actions during the night. Since it is very rare for the agents to not rest at home during the night, the criteria is set to 99%. The agents that are excluded are agents that are in treatment. The criteria is the following:

C1: On average 99% or more of the agents should be at home at night.

We acknowledge that this specific criteria would not work in all models. Perhaps in a model about studying the effects of the pandemic among students, it is not a realistic criteria to have. Students are quite likely to go out during the night and probably rest at home the morning after. However, this ASSOCC simulation is simplified on this matter and within the ASSOCC framework most of the agents resting at home at night is realistic enough behaviour.

In the ASSOCC model, there are some optional actions that can satisfy specific needs. These are essential shopping (grocery shopping), non-essential shopping (luxury shopping), and leisure. Leisure can be performed at two locations, but since it does not matter for the leisure need where the action is performed, no distinction will be made between private or public leisure for the criteria. Each of those three types of actions should be performed at least once every three weeks. This might seem like a low amount, but remember that this is the absolute minimum. Also, here this criteria would not be applicable to all social simulations that have these actions. Some may have different frequencies, but in ASSOCC the agents use these actions to satisfy their needs and therefore, have a certain regularity.

C2: On average 99% or more agents should have performed a leisure action in the past three weeks.

[3] `https://github.com/maartenjensen/ASSOCC-context`

C3: On average 99% or more agents should have performed an essential shopping action in the past three weeks.

C4: On average 99% or more agents should have performed a non-essential shopping action in the past three weeks.

For each of these criteria, there is a separate counter in each agent. This counter will increase at the start of each day by one. If the counter is smaller than or equal to 21 the agent still fits within the accepted boundaries. The counter is not increased when the agents know they are sick or are in quarantine. These agents are also not taken into account in the measurement.

Criteria For Obligations

The worker, student, and children agents have obligated activities during the working hours. The workers have to work, the students and children have to study (Table 4.2). Whenever an agent is not sick or in quarantine it is expected that the agent does this obligated activity. It could happen that an agent deviates from this in special circumstances, for example, an agent could have a critical need like food shopping which makes the agent go out to go shopping. Generally though, the agents perform the activity that they are obliged to do during working hours.

Let us first define the criterion for workers. Workers should hardly ever skip work when they are obliged to work. There are in principle two types of workers, those who can only work at their workplace and those who can work from their workplace or from home. For the workers who can only work at their workplace, they are only taken into account if they are not in quarantine or not believing they are sick. The workers that can also work from home have a slightly different measure. They are still measured when they are in quarantine but do not believe to be sick, since they can work from home. The criterion that measures this is the following:

C5: On average 98% or more workers should not skip work when they are able to work.

For workers who can work from home, it should not be the case that they always work from home. To check this, another criterion has been added that specifically checks this. It specifically checks which proportion of workers work from the workplace, while they have the possibility of working from home. This criterion is not as strict as the other criteria, since working from home sometimes is technically still allowed and happens in the real world rather frequently as well. This criterion is defined as the following:

C6: On average more than 85% off workers should work at the workplace *when possible instead of working from home.*

The student agents have a criterion similar to criterion 5. As explained before, when students should study, they should not frequently skip studying at university. For students, this criterion is relaxed a bit since there are less consequences on skipping a study day than skipping a working day for workers. This criterion is defined as the following:

C7: On average 95% or more students should not skip studying at the university when they are able to study.

The children agents have a criterion similar to the previous criterion. As explained before, when children should study, they should not frequently skip studying at school. For children, this criterion is relaxed a bit since there are less consequences on skipping a study day than skipping a working day for workers. This criterion is defined as the following:

C8: On average 95% or more children should not skip studying at the school when they are able to study.

Criteria Related To The Virus

Since the ASSOCC framework is meant to study the Covid-19 pandemic it is essential to include criteria related to the Covid-19 virus. The agents are influenced by the virus in a number of ways. The agents can become sick and when they know they are sick they should generally stay at home. The agents should stay in quarantine when they know they are sick or when there is a global lockdown. In addition, an important aspect of measuring the progression of the virus is the infection curve. The criteria defined in this section will all relate in some way to the virus.

When agents are sick they should rest at home, or when they are severely sick they should get treatment at the hospital. They can deviate from this occasionally, but generally should rest at home or get treatment as this is better for their health.

C9: On average 90% or more agents should rest when they know they are sick.

The agents are supposed to stay in quarantine when they know they are sick or when there is a global lockdown. Being in quarantine means that the agent is either resting at home or is having treatment at the hospital. The number of agents quarantining when they should, should not be too low as actually most people would stay in quarantine when they have to. However, having 100% of the agents stay in quarantine would also not be realistic as in reality some people occasionally break quarantine. The more realistic number would be somewhere in between. This leads us to criteria 10:

C10: Agents should rest at home when in quarantine between 90% and 100% on average.

During a global lockdown, we can expect that there are many agents who are not sick but still have to be at home. This could lead to more agents breaking the quarantine than in a standard run. Thus for a global lockdown experiment the range for C10 are decreased to be between 80% and 98%

The agents have different age groups who respond differently to the virus. For example children will be less likely to be severely sick, while elderly are more likely to be sick. To test whether the different age groups adhere to the criteria, criteria C10 is also applied to each separate age group. This gives us the criteria C11, C12, C13 and C14 for respectively children, students, workers, and retired:

C11: Children should rest at home when in quarantine between 90% and 100% on average.

C12: Students should rest at home when in quarantine between 90% and 100% on average.

C13: Workers should rest at home when in quarantine between 90% and 100% on average.

C14: Retired should rest at home when in quarantine between 90% and 100% on average.

For the global lockdown scenario, the same applies for C11 to C14. The range is reduced to be between 80% and 98%

The last criteria, and perhaps one of the most important, is the criteria on when the peak of infections occurs. If the peak of infections occurs too soon, it could indicate that the disease was spreading too fast. Or that the disease died down too quickly. If the peak of infections happens too late it could indicate that the spread of the virus is going too slow. For example, when all the agents are not leaving their houses. In this criteria there is also a difference between a standard infected run and a global lockdown run. The standard infected run tends to peak rather quickly and reaches its peak around 100 ticks. Therefore, the ranges have been set to 75 and 125.

C15: The infection peak is happening between tick 75 and tick 125.

For the global lockdown run the peak of infections should happen at a different moment as well. In principle during a global lockdown the spread of the virus slows down, causing the infection curve to flatten. As soon as the lockdown is lifted the number of infected starts to peak, at least in the default

ASSOCC model run. This means that the peak of infections should happen later than the global lockdown. Thus, for global lockdown runs the peak of infections will be adjusted to be between 250 and 400 ticks.

Of course, the time of the peak does not give enough information on its own. However, it is a good starting indicator of whether or not there may be something wrong. The criteria that will later be assessed in Chapter 6 will also be accompanied by plots of the behaviour of the agents and the population status. First, the criteria will now be used on the Original ASSOCC simulation as an example of how they function.

4.4.2 Assessing Original ASSOCC With Criteria

With the criteria defined, they can be evaluated using the ASSOCC framework. In principle, the ASSOCC model should pass all the criteria. In this section, we quickly assess whether the ASSOCC model actually passes the defined criteria. The random seed for the individual runs in this section, and in further sections, is set to two. This is purposely set to two, as this provides a relatively representative run for all runs. The DCSD ASSOCC will not only be evaluated by this single run, but in the end of Chapter 6 will be evaluated by multiple runs to not be dependent on a single lucky run that passes the criteria. First, we start with assessing the Original ASSOCC without infected, then testing a run with infected and lastly a run with global lockdown. The ASSOCC framework contains many parameters, the most relevant are described directly in the text. Other parameters that are of influence on the model but would distract the reader are described in the Appendix 8.1.

Original ASSOCC - No Infected

Firstly to get the baseline the ASSOCC model is run without infected enabled, i.e. preset *0.0 Original ASSOCC-no-infections* and as described before, random seed 2. The settings related to that preset are described in more detail in Appendix 8.1.3. Table 4.5 shows the criteria, the value and whether the run passed the criteria. Criteria C9 to C15 are irrelevant as they relate to the virus and are therefore excluded from the table for this run. As expected, all the relevant criteria pass. It can be seen that C3 passes, but is less than 100%, this means that sometimes agents do not do essential shopping for more than three weeks. This could be explained by that the agent in question has a housemate who does more of the essential shopping and brings enough food to the house. This could then satisfy the food safety need of the agent in question without it actually performing the essential shopping action. Since the value of C6 Work at workplace is less than 100%, this indicates that sometimes agents worked from home. Although it is preferred that the agents work from the workplace, it could happen that the agent has a certain need that is very salient that makes it prefer to work from home. For example, when the belonging need is very salient.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	100	TRUE
C3	Recently Ess Shopping, mean > 98%	98.91	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	100	TRUE
C6	Work at Workplace when possible, 85% < mean	99.94	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	100	TRUE

Table 4.5: Criteria Results for 0.0 Original ASSOCC-no-infections

Original ASSOCC - With Infected

The virus criteria, C9 to C15 can be checked when the simulation is run with infected enabled, i.e. preset *0.1 Original ASSOCC* and random seed is 2. The settings related to that preset are described in more detail in Appendix 8.1.3. Table 4.6 shows the results of measuring the criteria. All measurements of the run pass the criteria. It can also be seen that the quarantine values (C10 to C14) are between 90% and 100%. This is very important as otherwise all the agents would perfectly follow the rules, which is not the case in real life. People sometimes deviate from the rules. That the agents sometimes break quarantine is also reflected by the agents not always resting when they are sick (C9). If the agents are sick, they should be in quarantine, since this is happening less than 100% it implies that agents sometimes break quarantine (for a run without global lockdown). The infection peak occurs at tick 118, which is roughly in the middle of the boundaries set by the criteria (C15).

Original ASSOCC - Global Lockdown

The final scenario includes the infected and a global lockdown, the *0.2 Original ASSOCC-lockdown* preset. The settings are described in more detail in Appendix 8.1.3. Table 4.7 shows the results of the criteria. Most values are similar, except for some of the values relating to the virus (C10 to C15). However, as the criteria have different ranges for the global lockdown scenario the model still meets those criteria. With global lockdown it is expected that people break quarantine more. This can be expected since during the global lockdown most agents are not sick and need to stay at home. This effect can be seen by generally lower values for the quarantine criteria (C10 to C14), when compared to the values in Table 4.6. Especially younger agents, such as children and students break quarantine more frequently, potentially because they feel less sick when they are sick. Thus, if they get sick after the global lockdown they will be motivated to actually go out of their home, since during the global lockdown their needs have become more salient. The peak of infections is happening after the global lockdown (C15), so this criterion passes as well.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	100	TRUE
C3	Recently Ess Shopping, mean > 98%	99.26	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	99.62	TRUE
C6	Work at Workplace when possible, 85% < mean	94.49	TRUE
C7	Not Skip School, mean > 95%	99.99	TRUE
C8	Not Skip University, mean > 95%	100	TRUE
C9	Rest When Know Sick, mean > 90%	96.01	TRUE
C10	People in Quarantine, 90% < mean < 100%	97.29	TRUE
C11	Children in Quarantine, 90% < mean < 100%	99.12	TRUE
C12	Students in Quarantine, 90% < mean < 100%	97.7	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	96.52	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	95.58	TRUE
C15	Infection Peak Tick, 75 < value < 150	118	TRUE

Table 4.6: Criteria Results for 0.1 Original ASSOCC

It should be noted that the criteria are only approximations. To solidify analysis of realistic behaviour it is also required to analyse the behaviour of the agents in more depth. For example analysing the chosen actions over time and analysing the infection curve. How this will be measured will be shown in the next section.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	100	TRUE
C3	Recently Ess Shopping, mean > 98%	99.17	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.6	TRUE
C5	Not Skip Work, mean > 98%	99.98	TRUE
C6	Work at Workplace when possible, 85% < mean	94.45	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	100	TRUE
C9	Rest When Know Sick, mean > 90%	96.78	TRUE
C10	People in Quarantine, 80% < mean < 90%	92.46	TRUE
C11	Children in Quarantine, 80% < mean < 98%	95.05	TRUE
C12	Students in Quarantine, 80% < mean < 98%	85.93	TRUE
C13	Workers in Quarantine, 80% < mean < 98%	91.64	TRUE
C14	Retireds in Quarantine, 80% < mean < 98%	92.64	TRUE
C15	Infection Peak Tick, 250 < value < 400	318	TRUE

Table 4.7: Criteria Results for 0.2 Original ASSOCC-lockdown

4.4.3 Experimental setup and plots

The criteria alone are not enough to measure the realism of the model. A more detailed analysis of behaviour over time is also required. For example, the actions of the agents can be plotted over time, see Figure 4.4. This plot shows per tick which activities are chosen by the agents for the Original ASSOCC model. The horizontal axis represents the time and the vertical axis the percentage of agents. However, as can be seen immediately, this graph is very difficult to read. Perhaps a general trend can be seen, but its difficult to distinguish specific actions, such as how often and when agents do grocery shopping, since these lines are hidden behind other lines.

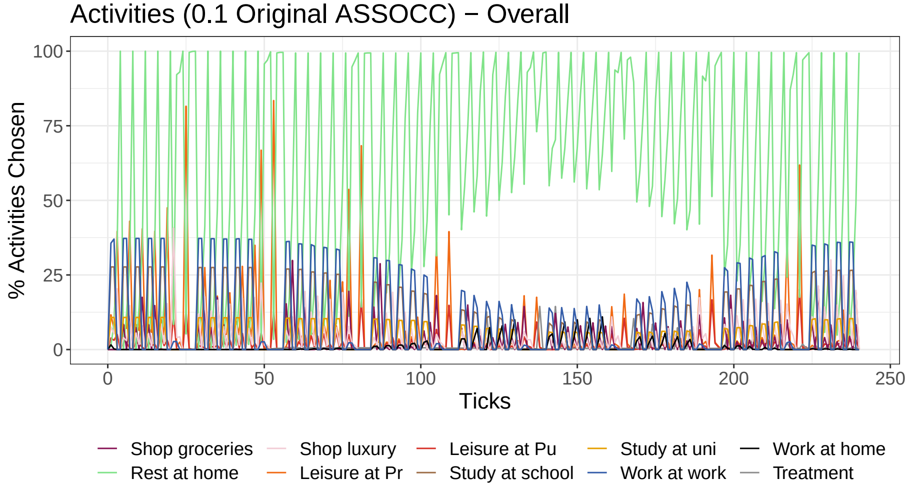


Figure 4.4: Agents' actions for Original ASSOCC

Smoothed Time Series Graphs

To illustrate the trends more clearly, we will instead simplify the graphs for Chapter 6. We start by decreasing the number of lines by combining some of the activities. Rest or working at home is taken together, this groups all the actions that an agent can do at home (Rest or Work at home). The other obligatory actions are grouped together, i.e. work at work, study at school, and study at the university (Work or Study out). Grocery shopping and luxury shopping is taken together (Shopping). Private leisure and public leisure is taken together (Leisure). Finally, getting treatment is removed since it is not relevant in showing the behavioural patterns, since treatment is done by so few agents. Figure 4.5 shows this with smoothed lines using the ggplot smooth function in R. The smoothed graphs will be useful to illustrate certain trends more clearly. In the graph it should become clear the agents' behaviour get most affected around tick 150. The agents are most at home, while the other activities such as working, studying, shopping and leisure decrease in frequency.

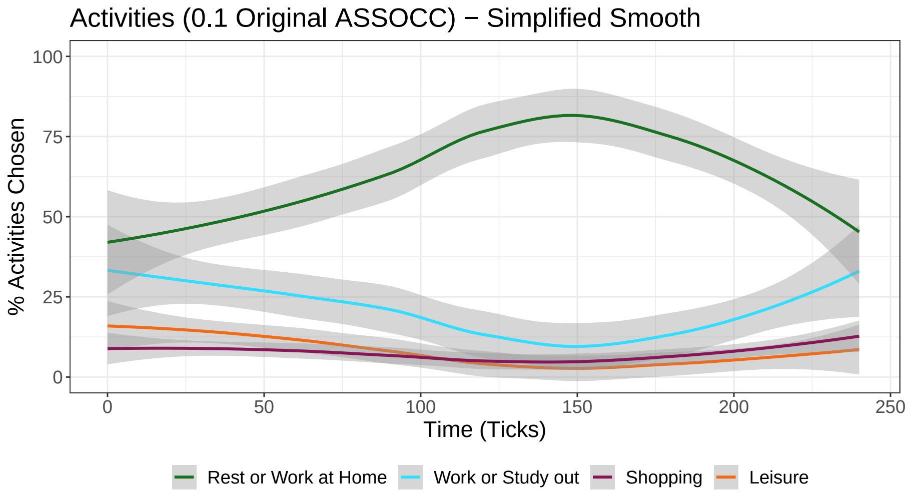


Figure 4.5: Activities Simplified Smoothed - Original ASSOCC

However, the peak of infections occurs slightly before tick 150. This is shown in Figure 4.6 where the amount of infected, believe infected and healthy agents are plotted. The agents will only adapt their behaviour when they believe to be infected (dark red), not when they are unaware that they are

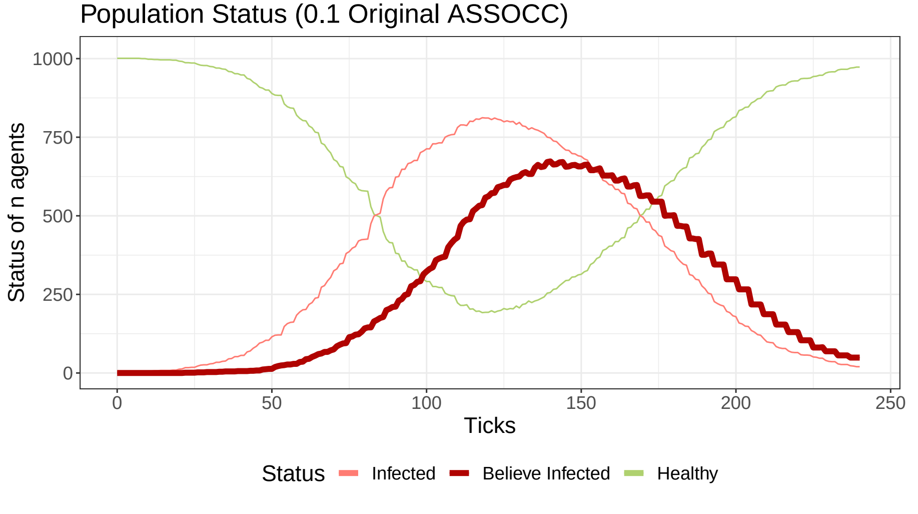


Figure 4.6: Population Status - Original ASSOCC

infected. This is why the behaviour matches the believe infected peak and not the infected peak. Remember it is the believe infected that counts for changing the behaviour. If the agent is infected but not aware of being infected then the agent's behaviour is not influenced by the disease. Only when the agent believes its infected (Believe infected = true) will it adapt its behaviour.

Daily Average Behaviour

Figure 4.7 shows the daily average with more details than the smoothed graph. It shows the distinction between weekends and working days. This type of graph can show the details of specific behaviour more easily. During weekends, e.g., at days 20 and 21, agents stay at home more, they do not work or study (except for a small amount of agents who work in the shops) and instead do more leisure activities. It should be noted that on day zero, which is the initialisation tick, all the agents are at home. This is the reason why day zero has 100% of agents resting at home during that day.

Figure 4.7: Activities Simplified Daily Average - Original ASSOCC

Other Types Of Plots

The analysis in Chapter 6 does not have to be limited to just the mentioned activities of the agents and the infection curve. It can also be expanded by other interesting plots if needed. Plots such as the average needs satisfaction level of all the agents or an individual agent. It is possible to zoom in more specifically on a specific type of behaviour, to distinguish again between luxury shopping and grocery shopping. These types of plots will where needed be brought forward and explained in Chapter 6.

4.5 Validation: Scalability

In the previous section, we explained how the realism of ASSOCC is measured and how we will compare the DCSD model using these criteria. In this section, we will discuss what aspects to measure for determining the scalability of DCSD. The scalability is related to the deliberation execution time of the models, and this will be the main variable of interest.

There will be two types of experiments that have two distinct variables that are changed. The first variable is the version of DCSD. Starting from the most simple DCSD going to a more complete DCSD to evaluate the effects on the deliberation time. Based on these results, it can be determined whether DCSD can solve the deliberation bottleneck in ASSOCC. The second parameter is the number of agents. This number will be increased from 1000 to 10,000 agents. That is, to evaluate whether the previous result is retained with higher agent numbers.

4.5.1 Experiment 1: DCSD Versions

The first experiment will measure deliberation execution time when incrementing the deliberative aspects of the DCSD and compare this with the baseline. The baseline measure will be made using Original ASSOCC. The model will be run with the default of 350 households, that is 1004 agents, and for 240 time ticks. The deliberation execution time will be measured using the Netlogo profiler. The inclusive time of the *select-activity* function in the profiler provides the deliberation execution time in milliseconds.

After measuring the baseline, the execution time of the different versions of DCSD will be measured. The setting again will be 350 households, that is, 1004 agents, and 240 time ticks. Each DCSD version can then be compared with the baseline to determine the speed-up and percentage increase. This will give a good indication of the speed-up benefits of incrementing deliberative aspects in the DCSD.

It can be expected that the more complete the DCSD, the more often it will be able to select an action. Consequently, by selecting more actions in the DCSD it can be expected that deliberation execution time decreases as the slower need-based deliberation is used less often. In the end, this experiment will determine whether the deliberation bottleneck in ASSOCC can be solved using DCSD.

4.5.2 Experiment 2: Increasing Number of Agents

The second experiment will measure deliberation with increasing number of agents. This is to evaluate whether the deliberation speed-up is retained over larger agent numbers. If the DCSD allows for a large speed-up that becomes lower, the higher the agent number, it cannot be claimed that DCSD successfully scales up social simulations. To state that the speed-up is retained over larger agent numbers, the speed-up with 10,000 agents should be similar or larger than the speed-up with 1000 agents. For this experiment, the number of agents will start at 1000 and increment to 10,000. More specifically, the number of households was set to 350, 700, 1400, 2100, 2800 and 3500. This gives respectively the following agent numbers 1004, 2008, 4016, 6016, 8024, 10,028.

4.5.3 Experimental Setup

The experiments will be run using the behaviour space of Netlogo. Since the aim of the experiment is to measure the execution time, only one core should be used in the behaviour space. If the simulations were to run in parallel the true execution time of a single run will get lost as the runs will interfere with each other. Thus, rather than in parallel the simulations for scalability have to be run serially. Both experiments should be run at least five times to account for some variability in individual runs. Adding many more runs is not expected to change the outcome by much. Since the difference in execution time between the Original ASSOCC and DCSD ASSOCC should be so great that adding more runs should not change the final point being made.

The type of hardware should not matter as long as the ram can support the model. That is, that it would not crash during a run due to memory issues. It is okay for the experiments to take some more time, since they are run to prove an academic point, they are not run during a crisis situation where actual policy makers are dependent on them. Therefore, a high performance computer is not needed and a normal desktop computer can be used if this is more convenient.

The simulations were run on a Windows 10 desktop with the following specifications. The desktop contains 32 GB of Ram memory and an Intel Core i7-8700 CPU with 3.20GHz (12 CPUs). However, as discussed before, only one CPU is used at a time to run the simulation in series rather than in parallel.

4.5.4 Preparing ASSOCC - Optimising

When evaluating the baseline, the Original ASSOCC framework, it became clear that its deliberation scaled quadratically. The code used for the experiment was from the ASSOCC framework[4] where the random seed was set to two. Note that the get-tested activity was removed for this run, as this action will not be used by the agents in the experiments for this thesis. If the get-tested action was left in the deliberation, it would slow down ASSOCC deliberation which would make an unfair comparison with the DCSD model. Figure 4.8 shows the results of plotting the deliberation execution time with increasing number of agents. The number of agents increases on the x axis. The execution time in milliseconds is shown on the y axis.

Figure 4.8 shows that the line does not scale linearly but quadratically. When 1004 agents are simulated, the time is about 222 seconds. The time

[4]See GitHub`https://github.com/lvanhee/COVID-sim`, commit: 3ba4d3f

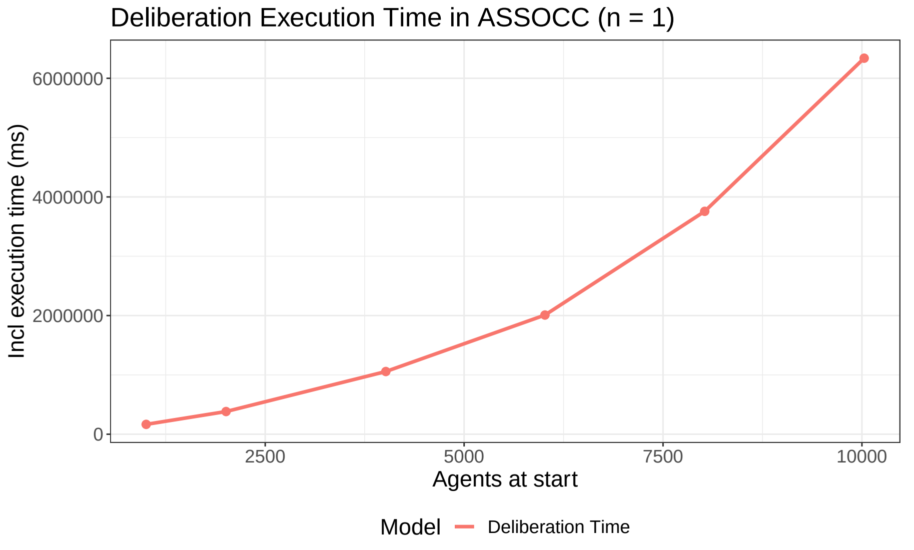


Figure 4.8: Deliberation Execution Time - Unoptimised Original ASSOCC

with 10,028 agents is 7881 seconds. The latter is not 10 times as much, as the increase of agent numbers, but rather more than 35 times as much. The Original ASSOCC deliberation, i.e. need-based deliberation, does not scale well with increasing number of agents.

Since the need-based deliberation will be part of the DCSD model, it should be optimised for a proper comparison. The deliberation should at least be linear when increasing the number of agents. Fortunately, it is relatively easy to make the ASSOCC need-based deliberation scale linearly.

In the ASSOCC framework, the agent deliberation is separated from the action taking of the agents. No changes are made to the state of the world during the deliberation, except for the action the agent will take. This means that information about other agents can be pre-calculated before deliberation in most cases. Doing this for all superlinear functions in the deliberation would in principle make the deliberation linear. The loop that would otherwise be in the deliberation loop for all agents is placed before the all agent deliberation loop. This type of optimisation has been performed in two types of functions. The first is based on pre-calculating the number of agents at a location which is necessary. The second is based on pre-calculating the available hospital beds.

The risk-avoidance and belonging needs require the number of agents for calculating their need satisfaction. In Original ASSOCC, to calculate the number of agents at a location, all agents are asked if they are at that location. This was not efficient and lead to a quadratic computational complexity. This function was made linear by adding a counter of the number of agents at each location. The calculation of number of agents at each location is then performed before the deliberation of all agents. Since it is pre-calculated the calculation has a linear computational complexity. This change is performed in the *need_management.nls* file.

The available hospital beds is relevant for the risk-avoidance and health needs. Calculating the number of hospital beds depends on the number of beds in use and the number of hospital personnel that is available to work, i.e., those who do not have visible symptoms. The number of hospital personnel that is available determines the capacity of the hospitals. If this number is lower, there are less beds available. This is determined by checking for each worker whether they work at a hospital and are sick or not. The latter causes a quadratic computational complexity when performing deliberation for all agents. This was solved by pre-calculating the available hospital beds once, before all agents deliberate. The adjusted code can be found in the *hospital.nls* file.

Optimised Original ASSOCC

Figure 4.9 shows the optimised Original ASSOCC, from now on called Original ASSOCC. This version of the ASSOCC framework, which can be found on GitHub [5] with preset *0.1 Original ASSOCC* (see Appendix 8.1 for more info). This is then not to be confused with the unoptimised ASSOCC version[6]. As seen in the figure the deliberation scales linearly over the number of agents. This allows for a proper comparison of the scalability of the deliberation.

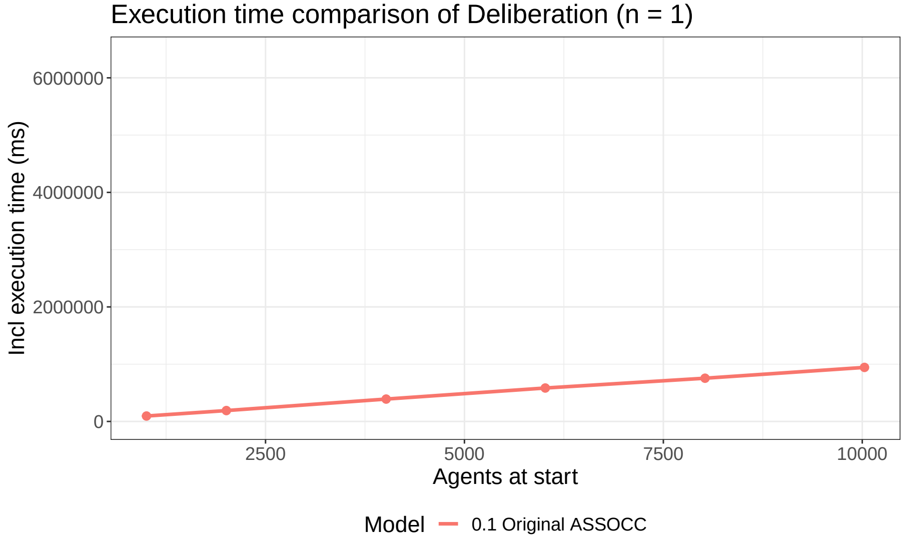


Figure 4.9: Deliberation Execution Time - Unoptimised Original ASSOCC

[5] `https://github.com/maartenjensen/ASSOCC-context`
[6] See GitHub `https://github.com/lvanhee/COVID-sim`, commit: 3ba4d3f

Despite optimising the deliberation, with the default setting of 1004 agents, the deliberation is still the main bottleneck. Deliberation still takes 42.6% of the total execution time while other components take less (see Figure 4.10). The non-deliberation takes 57.4% of the time. The spread of the disease takes 29.2% of the execution time, agents performing the actions only take about 16.1%, the tick and other smaller functions take 9.5% of the time, while updating the agents minds takes 2.5% of the time. All the other aspects take a lower proportion of execution time, hence the deliberation is the main bottleneck in the ASSOCC framework. It makes sense that deliberation is the largest bottleneck sense it takes into account all information for each agent every time step so can be expected to take up most execution time. Whether DCSD can solve the bottleneck will be shown in Chapter 6, but first the DCSD needs to be fully conceptualised and implemented.

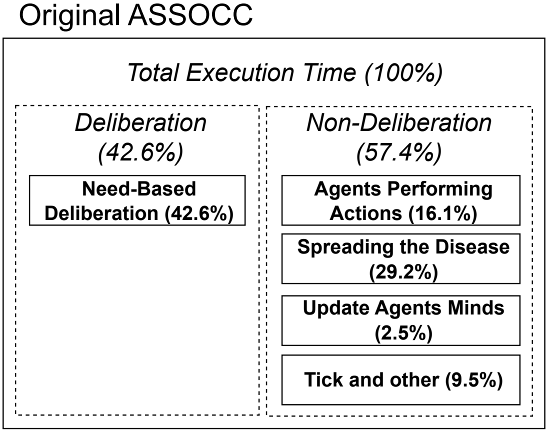


Figure 4.10: Execution Time Percentages Per Component - Original ASSOCC

4.6 Conclusion

In this chapter, we proposed the use of the ASSOCC framework as a use case to validate the DCSD. The ASSOCC framework has been explained in sufficient depth in this chapter for the reader to understand the following chapters. Then the DCSD ASSOCC was introduced in an abstract sense. DCSD is added in front of the need-based deliberation, rather than replacing the complete need-based deliberation model. This was followed by an explanation of how to measure retaining realism in the ASSOCC model. The last section described how the increase of scalability will be measured. Before, the results can be shown, the DCSD should be conceptualised and implemented. This will be shown in the following Chapter 5.

Chapter 5

Implementation: Applying DCSD to ASSOCC

This chapter will explain how the DCSD framework from Chapter 3 will be applied to the ASSOCC simulation [19] described in Chapter 4. The DCSD framework does not contain an explicit description of how it should be implemented. Some intermediate steps are required to implement the DCSD. First, it is necessary to specify the elements of the decision context of ASSOCC agents, i.e. create an information relevance matrix and a cell transitioning matrix. Secondly, from these matrices a conceptual deliberation model will be made following the DCSD framework and the complexity by need principle. Finally, the DCSD model is implemented in the ASSOCC framework, see also the GitHub code[1]. This chapter will investigate **RQ2**.

5.1 Defining the ASSOCC Decision Context

The decision context has been mentioned before and is defined by Jensen et al. [41] as the following:

"The decision context is any information that can be used in the decision making of an agent in a social simulation. Any information is information internal to the agent, external to the agent (i.e. the simulation environment), and also includes other agents' internal states."

The information relevance matrix [45, 41] and CAFCA cell transition matrix [45] can be used to determine this information. The information relevance matrix (Figure 3.2) describes information, both external and internal, that could be relevant in an agent's decision making. The CAFCA cell transition matrix (Figure 3.3) describes which information could specifically be used to

[1] `https://github.com/maartenjensen/ASSOCC-context`

change to a different deliberation type. By filling in these matrices with the deliberation of ASSOCC agents, the ASSOCC decision context elements can be defined.

5.1.1 Information Relevance Matrix

In the ASSOCC model agents use many different types of information to deliberate [44]. Some information is related to the agent's internal status like the needs, some information is related to norms, e.g. whether the agent should be in global lockdown, some information is related to social groups, e.g. the previous action of the network of the agent. In Figure 5.1 this information has been categorised using the information relevance figure from Chapter 3 (Figure 3.2).

	Individual	**Social**	**Collective**
Habitual	*Repetition* **Information used** Time: 1. Morning, afternoon, evening, night 2. Workday, weekendday Agent: age, sick status, can work from home? Available locations/actions Conflicting needs	*Imitation* Not represented	*Joining-in* Not represented
Strategic	*Rational choice* **Information used** All the needs and their satisfaction level.	*Game theory* **Information used** The past actions of people in the social network (family/friends) at a similar time. E.g. what the network did last working day in the evening.	*Team reasoning* Not represented
Normative	*Institutionalised rules* **Information used** The policies that are enabled such as global lockdown, quarantine and recommended to work from home.	*Social norms* Not represented	*Moral values* Not represented during deliberation, only during initialisation to determine need priority

Figure 5.1: Information relevance for ASSOCC, first seen in Jensen et al. [42].

Repetition

Starting from the *repetition* deliberation type, this cell captures the relevant information for determining a default action. This can be the time in the model, e.g. on a Monday morning. Also, the agent's age, whether the agent is sick, and whether the agent can work from home. These variables are internal knowledge of the agent and can be relevant for determining a default action. For example, the age will determine whether the agent regularly should go to school, university, work or none of these for retired. If the agent is sick it should by default rest at home. Agents who can work from home will do so during global lockdown.

As mentioned above, the locations that are available determine for the most part the available actions for the agent. The agents can only perform the essential shopping action if the essential shops are open. Information about locations and actions that are available is included, as this would directly determine whether the default action is available or not.

Sometimes needs can conflict with the scheduled action (Table 4.2). For example, when a worker agent has a salient food-safety need on Monday morning. On Monday morning the action work at workplace is scheduled, however, this action does not satisfy the food-safety need. Instead, if the agent would want to satisfy the food-safety need the grocery shopping action should be performed. In the case of working, the food-safety need can be considered the conflicting need. To be able to get an agent out of the scheduled behaviour, it needs awareness of conflicting needs. The conflicting needs are therefore included in the repetition cell.

The agent does not need to know about non-conflicting needs in the repetition cell. In the example of Monday morning where the worker is supposed to work at the workplace. Some of the non-conflicting needs are compliance and belonging (Table 4.4). Even if one or more of these needs is salient it will not stop the agent performing the scheduled action, work at workplace. Only conflicting needs can hinder taking a scheduled action.

Rational Choice

In some cases, the agent cannot follow the default action as there is a salient conflicting need. This would require a comparison of needs to see which need preference is stronger. For example, if the agent has a salient leisure need, but the sleep need is more salient than the leisure need. The agent will still prefer to rest at home. To do these kinds of comparisons, all the needs and their satisfaction levels could be relevant.

Institutionalised Rules

The institutionalised rules are the policies that can be imposed by the government. In Jensen [44] there is an extensive table, i.e. Table 3.9 Policy description [44], that mentions all the policies in the ASSOCC model. Specifically for our purpose, we will only consider the global lockdown policy, whether agents need to be in quarantine, and whether they are recommended to work from home (see Table 4.1). Deviating from any of these active policies requires the agent to weigh its need levels (as included in the rational choice deliberation type).

Conformity

Sometimes agents want to satisfy the conformity need, this requires information on the actions of other agents. This information fits best in the *game theory* cell as the agent does not just simply imitate but rather uses information from the past to create an expectation of where the agents of its social network will be.

Values and Other Deliberation Types

Values are incorporated in the ASSOCC model, however only to determine the needs priority of agents during initialisation. Values are not explicitly considered in the need-based deliberation itself. The agents do not explicitly use *imitation*, *joining-in*, *team-reasoning* and *social norms* in their deliberation. Thus to summarise Figure 5.1 the relevant information determining the decision context can be found in the repetition, rational choice, institutionalised rules and game theory deliberation types.

5.1.2 Cell Transitioning Matrix

Determining the information relevance was the first step, the next step is to determine which information is used to transition between deliberation types. For this purpose, the cell transition matrix (Figure 3.3) from Chapter 3 is used as inspiration. This matrix is not exhaustive, but it serves as inspiration. For the ASSOCC model only four deliberation types have to be considered for cell transitioning, i.e. repetition, rational choice, game theory, and institutionalised rules. Figure 5.1 shows these deliberation types and their transitions.

Repetition

In the ASSOCC simulation, needs can conflict with the default action. For example if it is night the default action (when considering the agent's schedule) is rest at home. However, if the leisure need is salient it will conflict with the rest at home action, since leisure at private/public leisure would be the preferred action. In this case the agent needs more information to determine whether the leisure need is more salient than other needs that support the rest at home action. Thus, the deliberation moves to the rational choice cell (box 1.). If the sleep need is more salient than the leisure need the agent will still rest at home. However, if the leisure need is the most salient need, then the agent may pick the leisure action (after more checks have been performed).

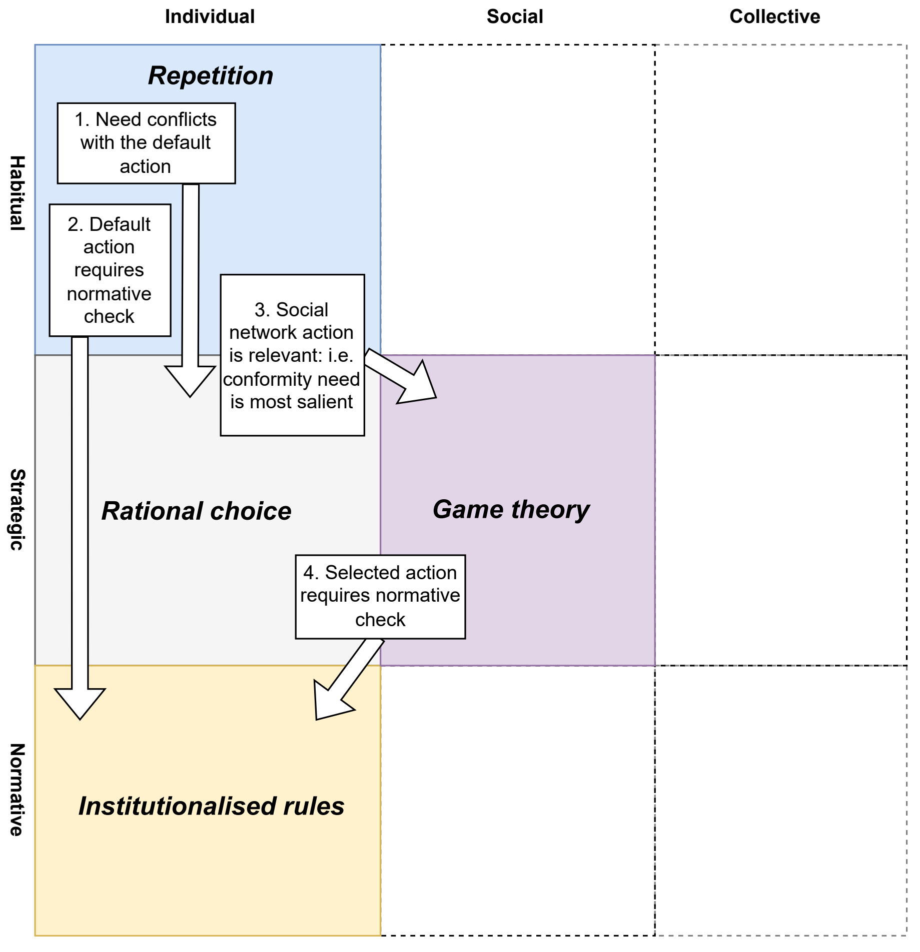


Figure 5.2: Cell transitions for ASSOCC

Rational Choice

In the rational choice box the need levels can be compared. This may sometimes lead to action selection when no normative considerations have to be taken. However, if normative considerations have to be taken, more information needs to be taken into account. In the running example, if it is night and the agent has leisure as the most salient need. Leisure in private/public leisure becomes the preferred action in that decision context. This action, in the pandemic simulation, requires a normative consideration before it can be selected. Thus, if a preferred action requires a normative check, there is a transition to institutionalised rules (see box 4. in Figure 5.2).

Institutionalised Rules

The institutionalised rules cell adds normative check such as quarantine check and breaking quarantine. If the agent prefers to have leisure in a private/public leisure place, the agent will check if it should be in quarantine. If the agent does not have to be in quarantine, it is free to choose a leisure action outside of home. If the agent must be in quarantine, it will consider whether the leisure need is more salient than the compliance need (rational choice information). If leisure is more salient then the agent can break the quarantine norm and take the leisure action. If leisure is less salient than compliance, the agent will most likely stay at home. This is happening in box 2. in the repetition cell and box 4. in both the rational choice and game theory cells.

Game Theory

In some situations, the agent values other agents' decisions more. This could happen when the conformity need is salient. See box 3. which is relevant both in the repetition and rational choice cell. If it is night and the agent's default action is rest at home, but the conformity need is salient, there could be a conflict. The agent needs to investigate what its network is going to do or at least create an expectation of what its network is going to do. This is where the deliberation requires information from the social column, more specifically the game theoretical cell. Depending on what the agent expects other agents to do it could change its chosen action. In ASSOCC the agents safe the action of other agents for different times of the day. This information will be used to determine whether the agent needs to rest at home at night or perform a leisure action. If the network action is rest at home, the agent will rest at home since it does not conflict with the default action. If the network action is leisure, the agent first needs to determine a couple of things before it can select the leisure action. It should determine whether conformity is important enough to not choose the strongly preferred rest at home action. And it needs to determine whether the agent is in quarantine or can leave quarantine. Thus, also incorporating information from institutionalised rules. In this case, the information from all four of the cells is considered.

5.2 Meta-Deliberation of Context-Sensitive Deliberation in ASSOCC

Now that the information relevance and the transition matrix have been defined, we can define the DCSD further. Starting from the DCSD framework as described in Figure 5.3 (also mentioned in Chapter 3). Most of the information described in the information relevance matrix is simulation-level information. Only the available actions are meta-level information. The idea in the framework is that the meta-level elements get updated by the simulation-level information and vice versa until an appropriate action is found. First we will consider which elements of the meta-level will be relevant for the ASSOCC DCSD.

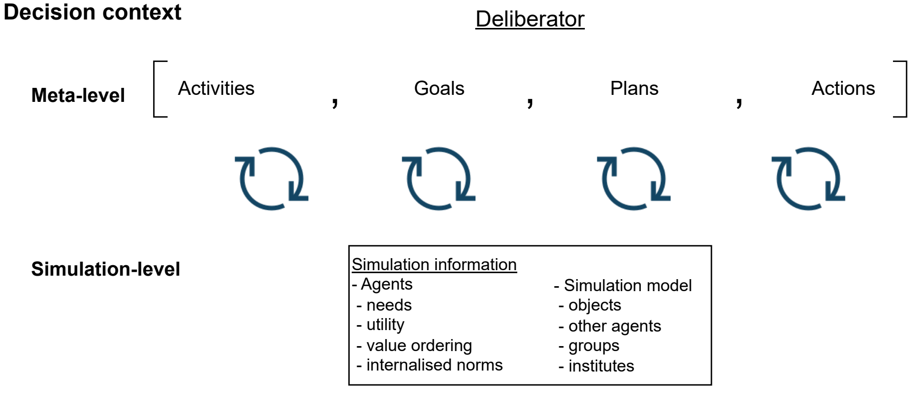


Figure 5.3: The Dynamic Context-Sensitive Deliberation Framework, adopted from Jensen et al. [42]

5.2.1 Meta-Level Tuple

The meta-level tuple, as seen in Figure 5.3 consists of activities, goals, plans, actions, and meta-criteria. This section describes which of these elements are relevant or irrelevant in the ASSOCC simulation context.

Activities

Activities are not needed, as there are only few actions in the ASSOCC model. There is no instance where there are many actions relating to a single activity. There is the leisure action that can be done at two different locations, i.e. private and public leisure. However, leisure is still only one action that has similar effects in those locations. Shopping is divided into two actions, i.e. essential shopping and non-essential shopping. These are two distinct actions, where one action is for shopping groceries, and the other is for shopping luxury items. They also do not justify adding the activities set to the tuple as there are only two actions.

Adding activities in the tuple would rather be expected in a simulation that has more actions related to a single activity. For example, a simulation with a detailed shopping action set, and at least one other detailed action set of an unrelated topic. For example, where agents could either move to specific isles within a store to shop specific products. And where they can pay at the cashier or use the self-scanning devices. If the simulation is this detailed, it makes sense to include activities in the tuple. Since then, as soon as the agent enters the store it could be in the shopping activity context. This would filter out actions such that it only shows the shopping related actions. This makes action selection more efficient, but this is not the case in ASSOCC.

Plans and Goals

The ASSOCC agents do not make long-term decisions. They decide in each step what their next action should be. Thus, they do not require plans. One could say that the goals are implicitly included in the model. For example, the agents have the goal of satisfying their needs, working when it is working time, generally following the policies. However, since they are not explicitly represented in the model, the DCSD ASSOCC will not need to contain them.

Actions

The ASSOCC model has in total ten actions, see Table 4.3. The DCSD should consider the available actions when deliberating. Using the information from the simulation, the action set can be expanded or narrowed down. This principle has been illustrated in Section 3.2.4.

Meta-Criteria

Since there are only actions in the tuple, most meta-criteria become irrelevant. Only action expansion or narrowing becomes relevant. By using relevant information from the context the action set can be adjusted. The cell transition matrix determines when to include additional information from other cells. Now that we have lined out the information and components for the DCSD ASSOCC version, the next section will discuss the conceptual architecture.

5.2.2 The ASSOCC DCSD Conceptual Architecture

As described in Chapter 4, the DCSD does not completely replace the ASSOCC's need-based deliberation. Rather, DCSD is used first, and when it cannot find an action, the slower need-based deliberation is used. Figure 5.4, which has been shown before, illustrates this principle. The abstract deliberation function in the figure is, of course, too abstract to implement. In the next section, we will explain how the DCSD ASSOCC model is conceptualised and how it should be implemented.

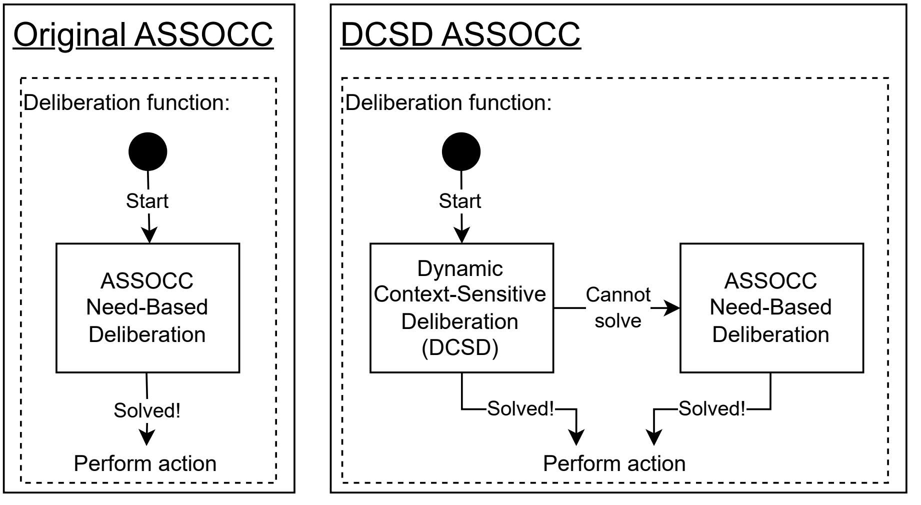


Figure 5.4: The Original ASSOCC deliberation and DCSD ASSOCC deliberation represented in an abstract manner.

5.3 Decision Trees for DCSD ASSOCC

In earlier work, a prototype DCSD model was developed [42]. This type of system was closer in terms of implementation to the DCSD model seen in Figure 5.3. It explicitly modelled the action set tuple that was manipulated by deliberation. The advantage of this system was mainly its flexibility. If another component is added to the simulation, the parameters of the prototype DCSD could be adjusted relatively easily. The down side of this flexibility is that there is more overhead, making execution time lower. Since the ASSOCC framework will not be changed, we can opt for a more efficient approach to modelling the DCSD. That is, the decision trees.

Figure 5.5 shows an example decision tree. The tree is made for a worker agent who needs to determine which action to take. The main variable used to determine which action is appropriate is the time variable. The time can be one of three states $\{night, freetime, worktime\}$. The tree starts at the black circle at the top. It will first determine whether the time is night (the blue box), if the time is night the choice is easy and the agent rests at home (the green box at the top right). If the time is not night the decision tree will distinguish whether it is working time or not. If it is working time the agent will choose the work at workplace action. If it is not working time it is freetime which gives the agent more possible actions. There are three available actions for freetime

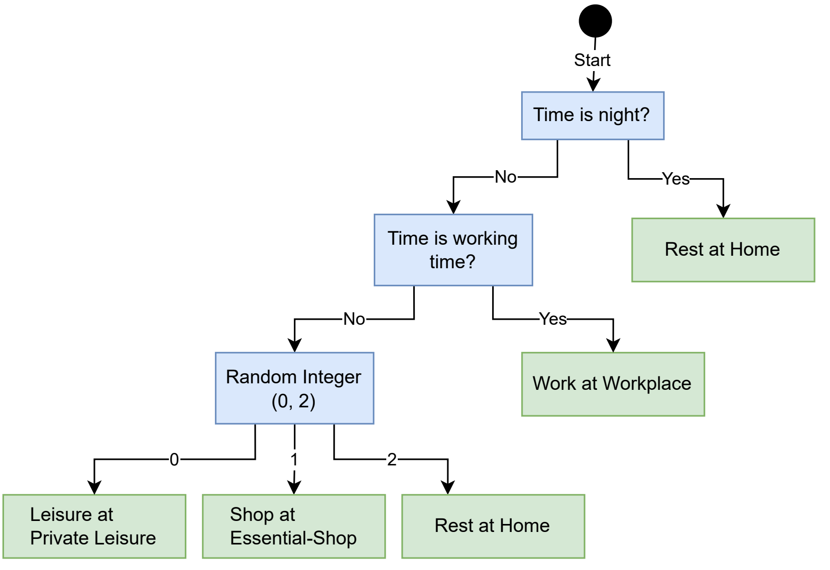


Figure 5.5: An example decision tree for a worker agent

in this case, i.e. {*Leisure at Private Leisure*, *Shop at Essential-Shop*, *Rest at Home*}. In this simple example, the agent choses one of the actions at random.

This is a very simple example that does not capture the complexity of the ASSOCC deliberation. However, it illustrates the basics of using a decision tree. It might seem complex to build a useful decision tree for the ASSOCC model, but if the important elements such as the relevant information, the transitioning mechanism, and the meta-deliberation are defined, the process of building the decision trees becomes quite straightforward.

5.3.1 Complexity by Need in Decision Trees

As shown in Figure 5.5 in some cases the decision tree only needs one check to select an action. If it is night, the agent will Rest at Home and will not consider the other aspects of the decision tree. Exactly this concept ties in very well with the complexity by need concept. It is possible to create a tree that starts with simple information and deliberation types and gradually increases complexity as we go deeper into the tree.

Figure 5.6 illustrates in an abstract sense how such a decision tree could look. In this tree habitual information (blue box) is always used initially. In some cases, this will be enough information to select an action. This first available action should be the default action for this specific context. If no action could be selected, the model moves deeper into the tree and starts to use information of the needs of the agents (grey box). Again, either an action can be selected or the model moves deeper into the tree. Now normative information

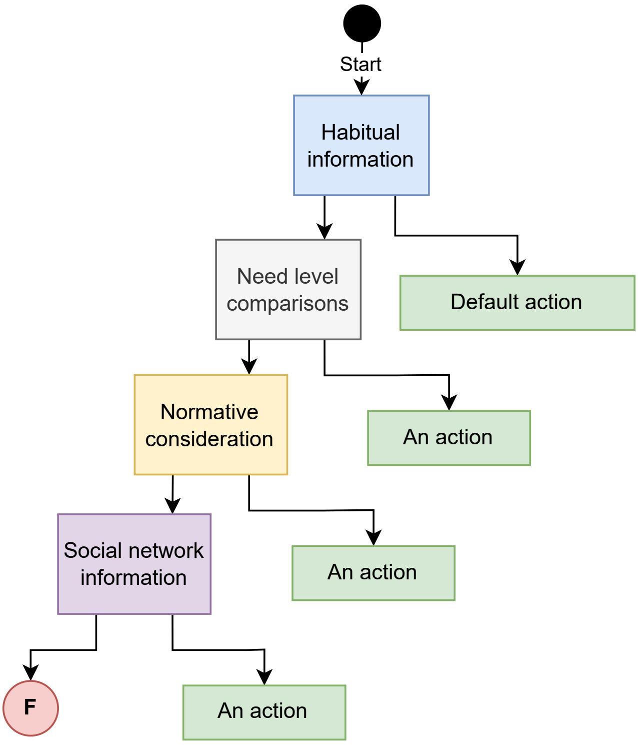


Figure 5.6: The different deliberation types throughout the decision tree.

will be considered (yellow box) and finally social information (purple box) can be considered. If all else fails ASSOCC need-based deliberation can still be used, see the encircled F at the bottom left (red circle).

It should become clear from this example that when using a decision tree it is relatively straightforward to incorporate the complexity by need principle. By making sure the most frequent deliberation patterns are more on the top of the tree, while the more exceptional complex patterns could be at the bottom of the tree. Of course, the order does not have to be fixed and in some contexts it could be possible that normative considerations are considered initially. However, most importantly, there should be very little overhead in selecting a default action.

5.3.2 Determining the Initial Context in Decision Trees

As mentioned above, to retain the complexity by need principle, the overhead in the DCSD model should be minimal. This means that the context determination and the most frequently chosen actions should be easily accessible. Easily accessible is related to both the depth of the action in the tree and the type of information that is relevant. Thus, preferably the action should be available at a more shallow position in the tree and preferably only use information from the repetition deliberation type.

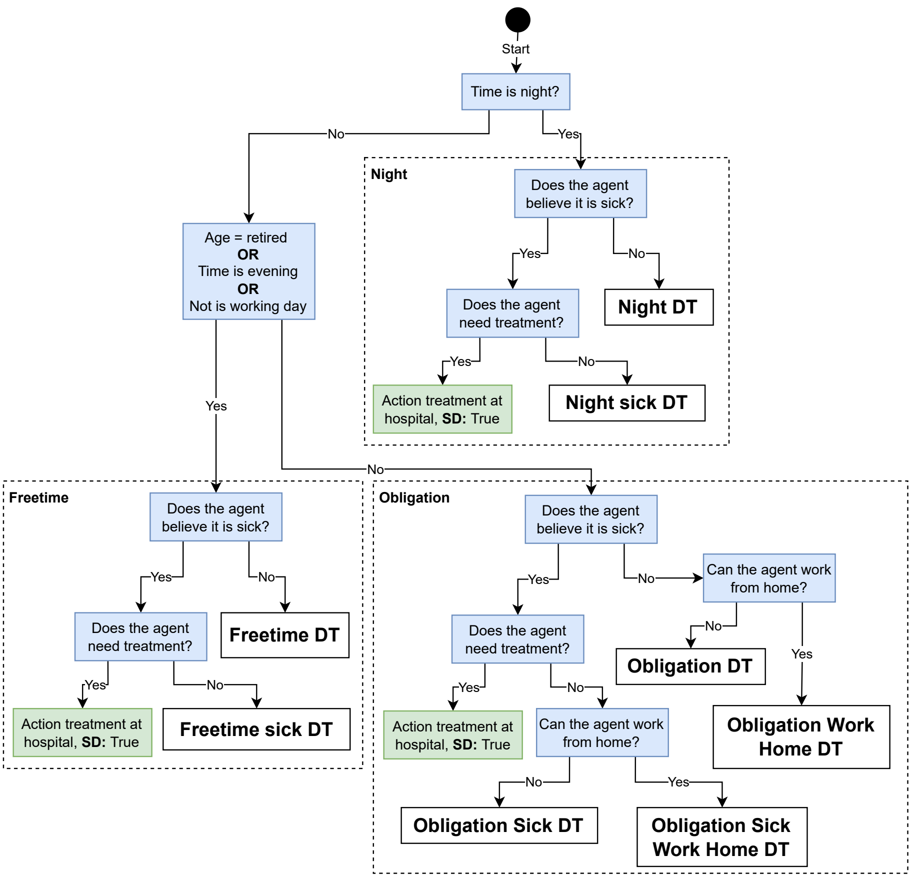


Figure 5.7: The deliberation tree for determining the initial context

Determining which information is relevant for the initial context contains a scalability and realism trade-off. There is no clear answer on how much information is enough or too little as this highly depends on the simulation and the agent's situation. If all information is considered all the time, there is no risk of leaving out crucial information; however, it will be computationally speaking slower. If only a single aspect of information is given, e.g. the time, then crucial aspects such as the agent being sick or having a salient need could be missing.

Considering all this information from the repetition box in Figure 5.1 is a good starting point. As Table 4.3 shows, the time variable and the age variable have a large impact on the available actions. Being sick enables the treatment at the hospital action and being able to work from home enables the work at home action. Using these types of information, it is possible to distinguish a couple of specific contexts.

Figure 5.7 shows the deliberation tree to determine the initial context. It starts by splitting the context using the time, distinguishing three distinct times $\{night, freetime, obligation\}$, see also Jensen et al. [42]. This is further divided by whether the agent believes that it is sick. If the agent is not sick the deliberation can already enter some of the decision trees, i.e. Freetime DT and Night DT. If the agent is sick, it will first determine if it needs treatment or not. If it needs treatment it will directly perform the treatment at hospital action. If it does not need treatment, it can enter the Freetime Sick DT or Night Sick DT depending on the time. If the time is obligation and the agent is not retired it will also determine whether it can work from home or not to select the correct decision tree from the remainder four decision trees, i.e. Obligation DT, Obligation Work Home DT, Obligation Sick DT and Obligation Sick Work Home DT. These trees will be described in more detail below.

Treatment for the Agent

The agent might need treatment if it believes that it is sick. This depends on the infection status of the agent. If the agent has an infection status of *hospital-to-death* or *hospital-to-rec*, the agent will choose to have treatment at the hospital. This is a good example of why sometimes making the decision tree can be so straightforward. The infection status explicitly indicates that the agent should be hospitalised. This provides enough information to select the get treatment action such that the agent gets hospitalised. The agent thus does not require to explicitly check all its needs and other information, since the infection status variable encompasses enough information. When the agent gets hospitalised for the disease, it should be isolated, therefore, social distancing is always set to true when getting treatment. The latter is indicated in the green action box by SD: true.

Choosing Social Distancing

When the agent needs treatment, it is clear that the agent applies social distancing, this is less straightforward in other situations. In these situations, choosing social distancing is dependent on the needs. These relevant needs are risk-avoidance, compliance and conformity. In most situations, risk-avoidance and compliance are considered together to determine social distancing. In the case where conformity is the most salient need the social distancing of the network is considered.

Social distancing determined by the risk-avoidance and compliance needs is dependent on the quarantine status of the agent. If the agent is not in quarantine, the agent will social distance when the risk-avoidance need is below a threshold. Technically speaking this means that if *risk-avoidance-level* $<$ *ce-risk-avoidance-threshold-for-sd* the agent will apply social distancing; otherwise the agent will not apply social distancing.

If the agent should be in quarantine the previous rule still applies, but another is added. In that case, when the compliance need is below a specific threshold, the agent will perform social distancing. Technically speaking, this would be implemented as, if *should-I-stay-home?* and *compliance-level* $<$ *ce-compliance-quarantine-threshold-for-sd* the agent will apply social distancing. If both the risk-avoidance and compliance conditions are not met, the agent will not apply social distancing.

The rationale behind the compliance need is the following. During the global lockdown, many agents are in quarantine but are not sick. These agents will usually have a satisfied risk-avoidance need since they are healthy themselves. This would mean that the agents that do break quarantine will just act like nothing ordinary is happening, i.e., they will perform their action without social distancing. However, in reality it can be expected that at least some people will social distance during a global lockdown. Therefore, the social distancing has now also been linked to the compliance need. If the agent does out of quarantine and it has a low compliance, it will at least social distance to partially follow the law. Determining social distancing by using risk-avoidance and compliance (if necessary), is indicated in the green action selection boxes by SD: risk avoidance.

The social distancing can also depend on the conformity need. When conformity is most salient, the social distancing depends on whether the social network chose to social distance. In the green action selection boxes, this is indicated by SD: social network SD. For simplicity sake, social distancing is not influenced by the compliance need. This would require more considerations, such as normatively checking what rule the government posed.

Given the *ce-risk-avoidance-threshold-for-sd* and *ce-compliance-quarantine-threshold-for-sd* parameters it is now relatively easy to control the infection curve of the runs. By adjusting either or both of these variables, different infection curves can form. By putting these variables low, the agents will social distance more, which results in a flatter curve. By putting these variables higher, the agents will social distance less, which results in a steeper curve. By adjusting the compliance parameter, the infection curve can be adjusted more specifically during global lockdown. All these adjustments are useful when the DCSD behaviour is slightly different from that of the Original ASSOCC. Then these variables can be used to tweak the infection curve in a more realistic pattern.

5.3.3 Explaining the Decision Trees

The night decision tree is activated when the minimal context is equal to $\{time = night, status = not\_sick\}$. Table 5.1 shows which actions are available given that the time is night. ASSOCC's need-based deliberation only allows rest at home at night, or additional getting treatment at the hospital if the agent is sick. Since technically the leisure at private leisure and leisure at public leisure are available, they are added as possible actions in the DCSD. In the ASSOCC model the resting at home action is strongly preferred during the night. This is reflected by the rest at home action satisfying many different needs (see Table 4.4). The leisure activities should be seen as deviations from the default and will only be performed if the agent has a salient enough leisure or conformity (with the network performing leisure as well) need.

Actions	**Description**
Rest at home	Default, **strongly** preferred
Leisure at private leisure	
Leisure at public leisure	

Table 5.1: Available actions - Minimal context night

Knowing the available actions, a list can be created of needs that are related to those actions (see Table 5.2). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. All the red plusses and minuses can be ignored since the time is night. In Table 5.2 below, the four needs at the top positively support the rest at home action. The leisure need is supported more by leisure activities than by the rest at home activity. Therefore, if the leisure need is salient enough, the leisure activity will be preferred. Since having a salient leisure need conflicts with the default action of resting at home, it is noted as conflicting (accentuated with the red colour). The same applies to the conformity need.

The need	**Relation to action**
Sleep	Supports default
Risk avoidance	Supports default
Belonging	Supports default
Autonomy	Supports default
Leisure	**Conflict: Leisure activity**
Conformity	**Conflict: Network action**
Compliance	Normative consideration

Table 5.2: Relevant needs - Minimal context night

If the conformity need is salient enough, the agent prefers the action that the social network has performed. This network action could be rest at home, but could also be one of the leisure activities. Other needs such as financial stability/survival, luxury and food safety are not influenced by the available actions. The health need is only influenced by the actions if the agent is sick, but in this minimal context the agent is not sick. The compliance need is only relevant if the agent needs to make a normative consideration, i.e. if the agent wants to perform one of the leisure activities it needs to check whether it is not in quarantine. This will be explained in more detail later. The complete decision tree can be quite overwhelming at first and therefore we decided to gradually introduce parts. Starting with the habitual behaviour decision tree.

Habitual Behaviour

Figure 5.8 shows the initial deliberation model. This model is much like the Kahnemann thinking fast, thinking slow principle. The Habitual DCSD will be used to make a decision first. It uses only information from the repetition cell, and could be seen as the fast thinking part. If it cannot select an action the need-based deliberation from the ASSOCC model will be used. This need-based deliberation will always find an action as it will choose the action that satisfies the needs the most. The ASSOCC need-based deliberation can be seen as the slow deliberation part.

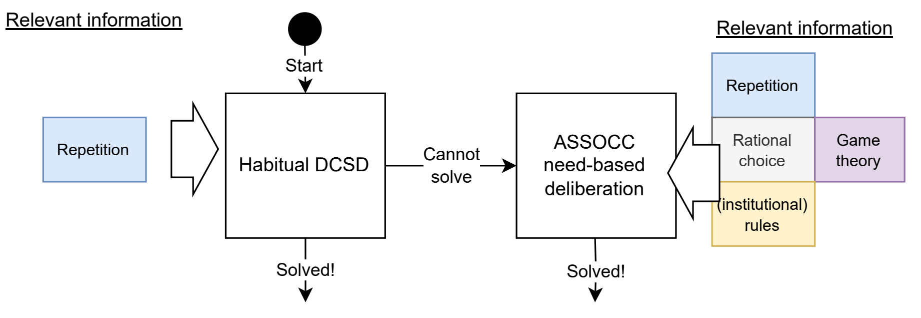


Figure 5.8: DCSD in ASSOCC - Habitual

The available information for the DCSD based on the repetition cell is the following. The minimal context information, consisting of the time: night and the status: not sick. The available actions: rest at home (default action), leisure at public leisure, and leisure at private leisure. And the conflicting needs and whether they are salient or not. The need is salient when it is below *ce-need-salient-threshold* (0.5). This information is already enough to setup a simple habitual system. Note that unlike actual human habits, this system does not learn. Instead habitual refers to selecting a default action related to the situation. It either selects the default action or enables more complex deliberation when there are conflicting needs.

Figure 5.9 shows this simplified version of the night decision tree. If there is no conflict the default action rest at home will be selected. If there is a conflicting need salient (leisure or conformity), then the decision tree will use need-based deliberation (indicated by the red encircled F).

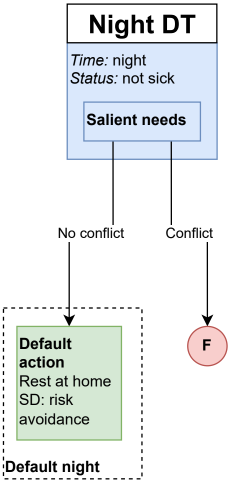


Figure 5.9: Decision tree - Night Habitual

Strategic behaviour

This section will explain the decision tree that is expanded with strategic information and deliberation. Figure 5.10 shows that both the habitual and strategic information are considered in the DCSD. The complex need-based deliberation is still the same and will stay the same in the upcoming expansions of the DCSD. It is only the DCSD that is extended to increase its capability of solving decision making problems. The information that is added in addition to the

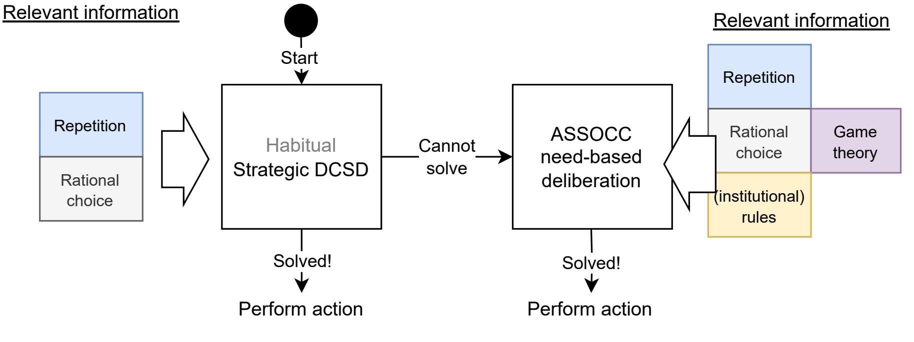


Figure 5.10: DCSD in ASSOCC - Strategic

information that was already available are the levels of all the relevant needs (not only the conflicting needs).

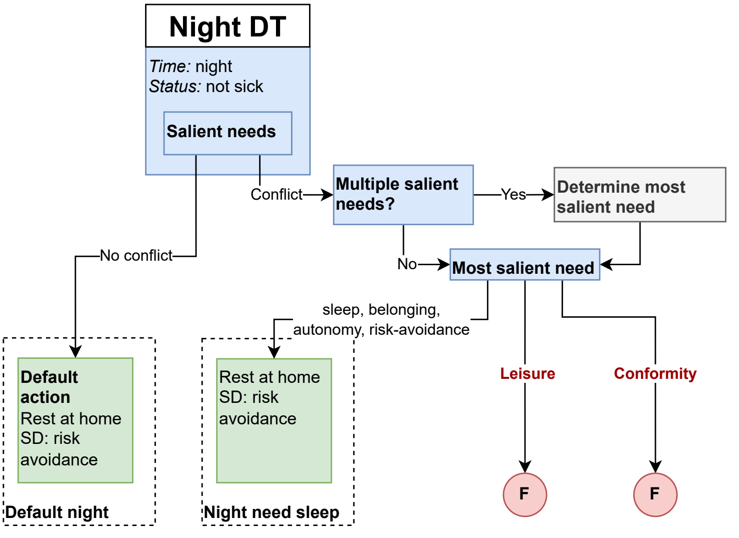


Figure 5.11: Decision tree - Night Strategic

As shown in Figure 5.11 the initial part is the same. If there is no conflicting need the default action is selected. However if there is a conflicting need it is further specified which need this is. It can happen that there are for example two needs salient, one conflicting and one supporting the default action. If the leisure need level is 0.3 and the sleep need level is 0.2. Both needs are below the salient_need_threshold. However the sleep need is the most salient need. The sleep need supports the default action, meaning that the decision tree selects rest at home. This will happen for the sleep, belonging, autonomy and risk-avoidance needs. If the leisure need would be the most salient need, the decision tree does not have all the information to solve (as it lacks normative information), thus need-based deliberation will be activated (F). The same applies to a most salient conformity need, which requires normative and social information.

Normative Behaviour

Figure 5.12 shows that normative aspects have been added to the DCSD. The information added is that the agent can now check whether it should stay at home or not. The agent should stay home when it is in quarantine (due to global lockdown) or when it is sick (which is not the case in this decision tree).

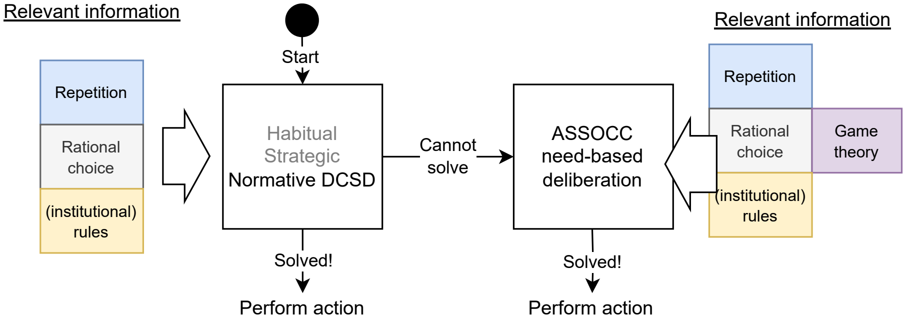


Figure 5.12: DCSD in ASSOCC - Normative

Enabling the normative information allows the decision tree to select actions outside of the home, such as one of the leisure actions. The rest of the decision tree is similar to the previous, except for the leisure need. Since its night, the leisure need has to be critical for the DCSD to choose one of the leisure actions (*leisure-level* $<$ *ce-need-critical-threshold*). If the leisure need is not critical the DCSD gives up and the need-based deliberation will deliberate further. If the leisure need is critical, it is checked whether the agent does not have to stay at home (normative) or the leisure need is more salient than the compliance need. If any of these conditions is true, the agent can perform a leisure activity, otherwise the agent will rest at home. If the agent wants to do a leisure activity, the risk avoidance level is checked. If *risk avoidance level* $<$ *#risk-avoidance-private-leisure-preference* then the agent prefers to have leisure at a private leisure place to decrease the chances of getting infected. Otherwise, the agent will choose randomly whether it will do the leisure at public leisure or leisure at private leisure action.

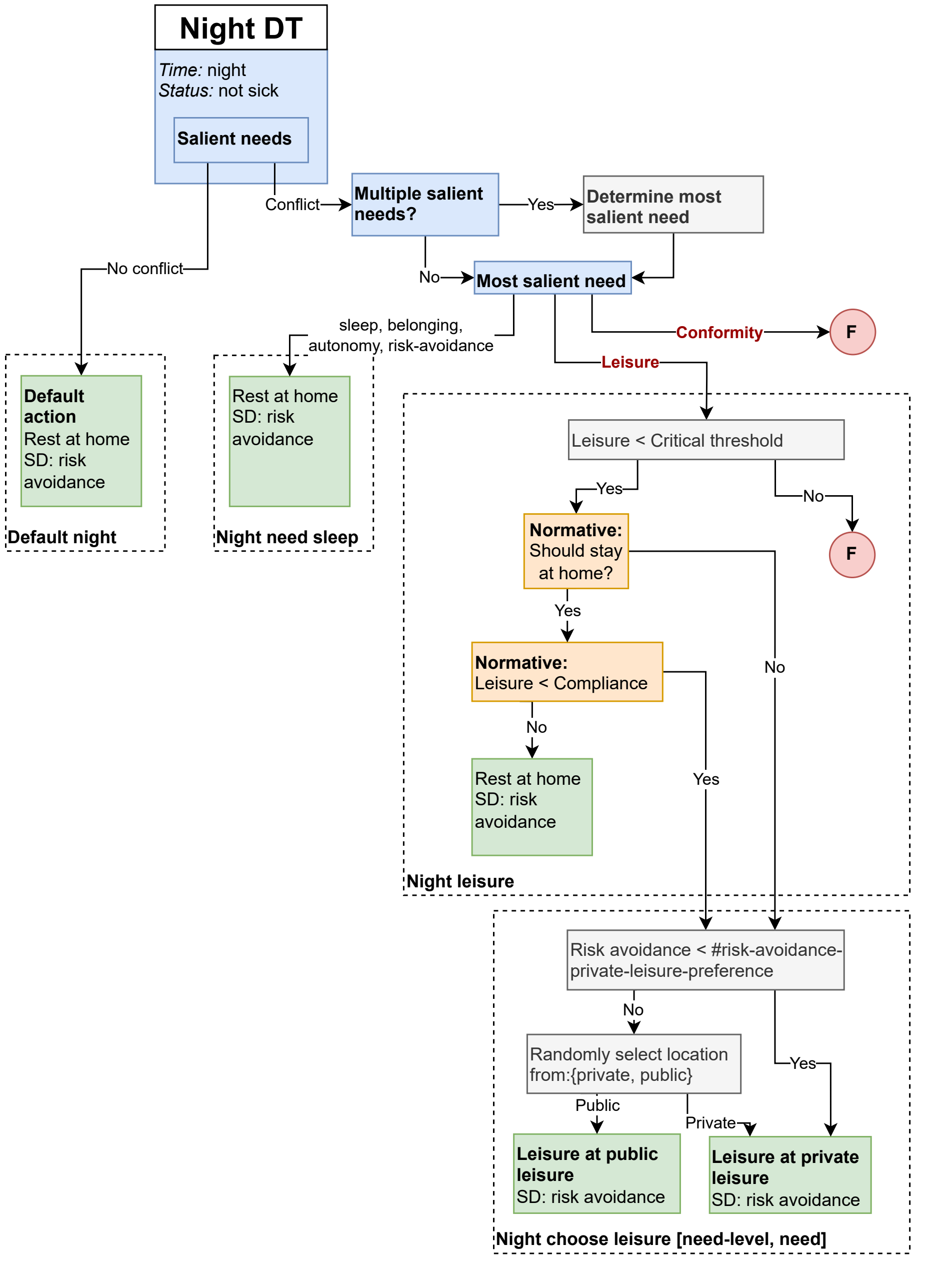


Figure 5.13: Decision tree - Night Normative

Social Behaviour

Figure 5.14 shows that social aspects have been added to the DCSD. The information added is that the decision tree now has access to the action performed by the social network of the agent. Despite this being precomputed according to the information relevance matrix this fits in the social dimension.

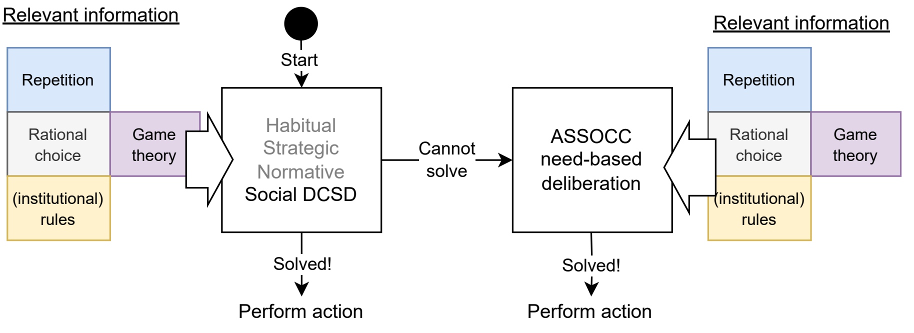


Figure 5.14: DCSD in ASSOCC - Social

Figure 5.15 shows the decision tree with the social aspects added. The leisure decision tree is abstracted to the dashed box called, night leisure. If the conformity need is the most salient the agent will first check the network action. If the network action is rest at home, the agent will also rest at home. The social distancing is determined by the social distancing behaviour of the network. If the network-action deviates from either rest at home or a leisure action the agent will enter need-based deliberation (F). This is very unlikely to happen but could happen when most of the agent's network performed the getting treatment at hospital (TR) action. If the network-action is leisure first there is a check whether the *conformity need* $<$ *critical threshold*. If this is not the case the need-based deliberation is activated. Otherwise there will be normative checks similar to the one described in Section 5.3.3. If the agent does not have to stay home or when the *conformity need* $<$ *compliance need* the agent will perform the same leisure action as the network, either LPU or LPR. If this is not the case the agent will rest at home. In both cases the agent performs social distancing (SD) according the the network SD.

5.3.4 The Complete Night State Decision Tree

Figure 5.16 shows the final night decision tree. This tree is fully implemented in the code. There are in total five other decision trees to describe, seven if we differentiate the decision trees for obligation and obligation for agents that can work from home. The remainder of the decision trees and the corresponding tables for available actions and relevant needs will be discussed in the following sections.

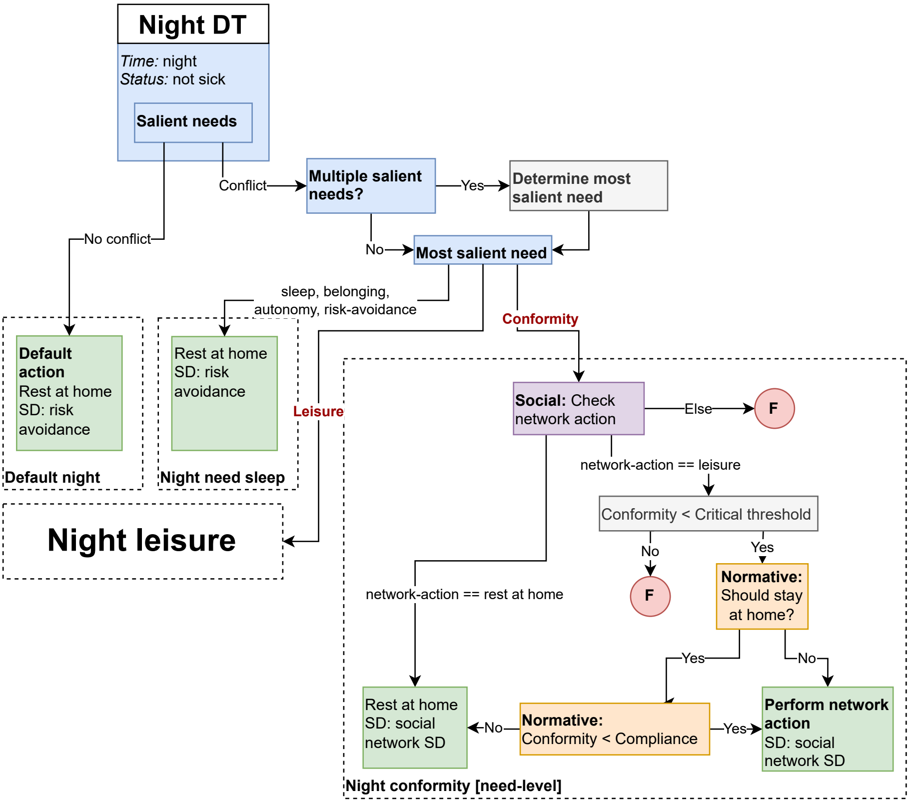


Figure 5.15: Decision tree - Night Social

Some Extra Notes on Decision Trees

Before moving on to the other decision trees. It should be noted that the decision trees and Netlogo code[2] could deviate slightly. The decision trees and the implementation follow the same logic; however, there might be a different order of the if statements in the implementation. This could have been chosen by the author if, for example, it would be more computationally efficient.

Some parts of the decision tree are not implemented in the code. This is the case for some of the needs in the obligation work home and obligation sick work home decision trees. In these sections, it is indicated which parts are not implemented by making that part of the decision tree opaque and adding a red cross. Implementing these might make the execution time of the DCSD ASSOCC slightly lower, however, it would have been a lot of effort for minimal gains. This minimal is expected to not affect the answer on the main research question; therefore, it is chosen to not implement those parts.

[2] `https://github.com/maartenjensen/ASSOCC-context`

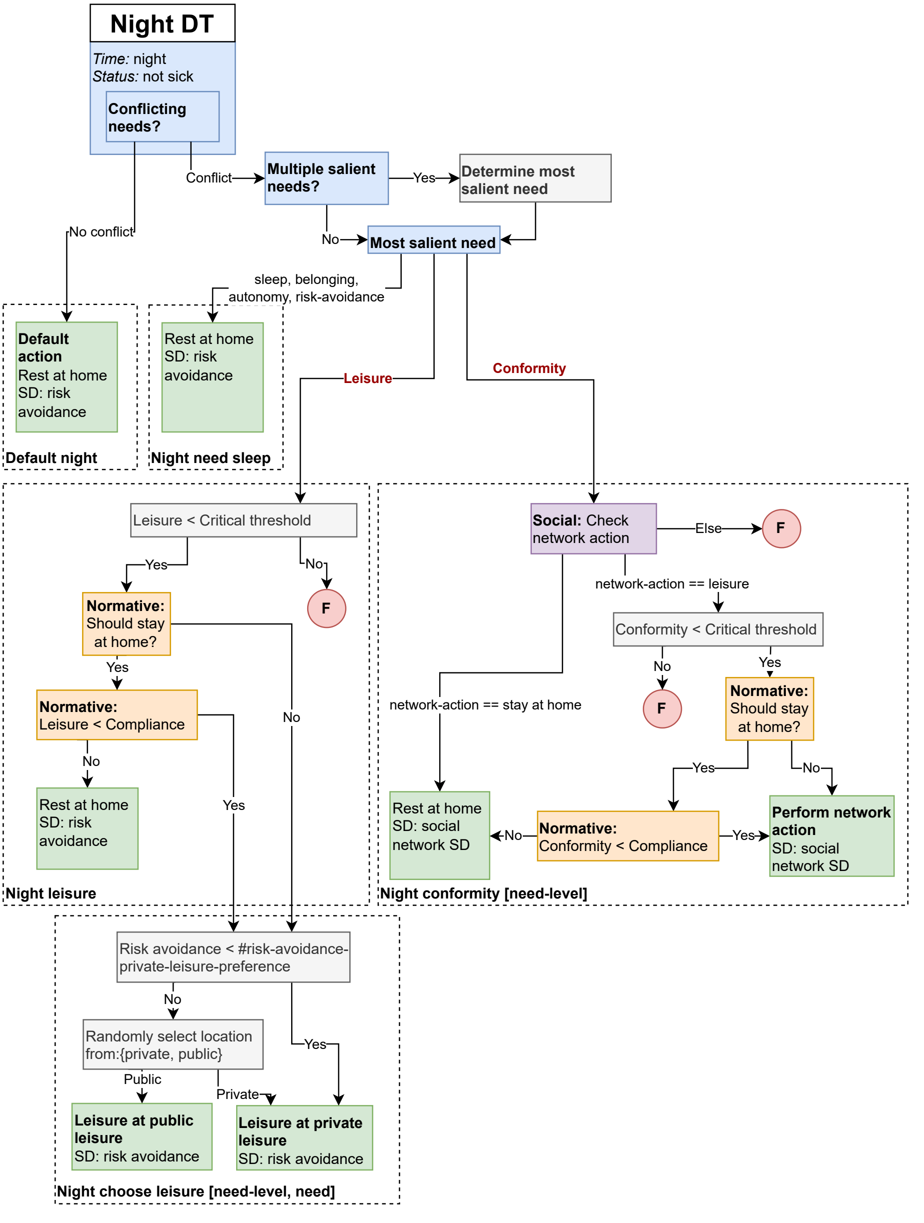


Figure 5.16: Decision tree - Night Full

5.4 The Complete DCSD ASSOCC in Decision Trees

The night decision tree has been fully described in the previous section. This section will describe the remaining decision trees. Starting with the night sick decision tree. Then the freetime and freetime sick decision trees. And ending with all the obligation decision trees that are not relevant for the retired.

5.4.1 Night Sick Decision Tree

The night sick decision tree is selected when the minimal context is equal to $\{time = night, status = sick\}$ and the agent does not need treatment. Table 5.3 shows the actions available in this context. Again, leisure is possible if the agent really wants to, even though the ASSOCC need-based deliberation will not have this available.

Actions	**Description**
Rest at home	Default, **strongly** preferred
Leisure at private leisure	
Leisure at public leisure	

Table 5.3: Available actions - Minimal context night sick

Based on the available actions, a list can be created of needs that are related to those actions and the sick status of the agent (see Table 5.4). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. Table 5.2 below, the four needs at the top positively support the rest at home action. The leisure need is supported more by leisure activities than by the rest at home activity. Therefore, if the leisure need is salient enough, the leisure activity will be preferred. Since having a salient leisure need conflicts with the default action of resting at home, it is noted as conflicting (bold font). The same applies to the conformity need.

The need	**Relation to action**
Sleep	Supports default
Risk avoidance	Supports default
Belonging	Supports default
Autonomy	Supports default
Compliance	Supports default
Leisure	**Conflict: Leisure activity**
Conformity	**Conflict: Network action**

Table 5.4: Relevant needs - Minimal context night sick

The Decision Tree

Figure 5.17 shows the decision tree. There are a couple of aspects that deviate from the described decision trees before. They are highlighted in the text below. Other than those roughly the same principles and colour coding apply.

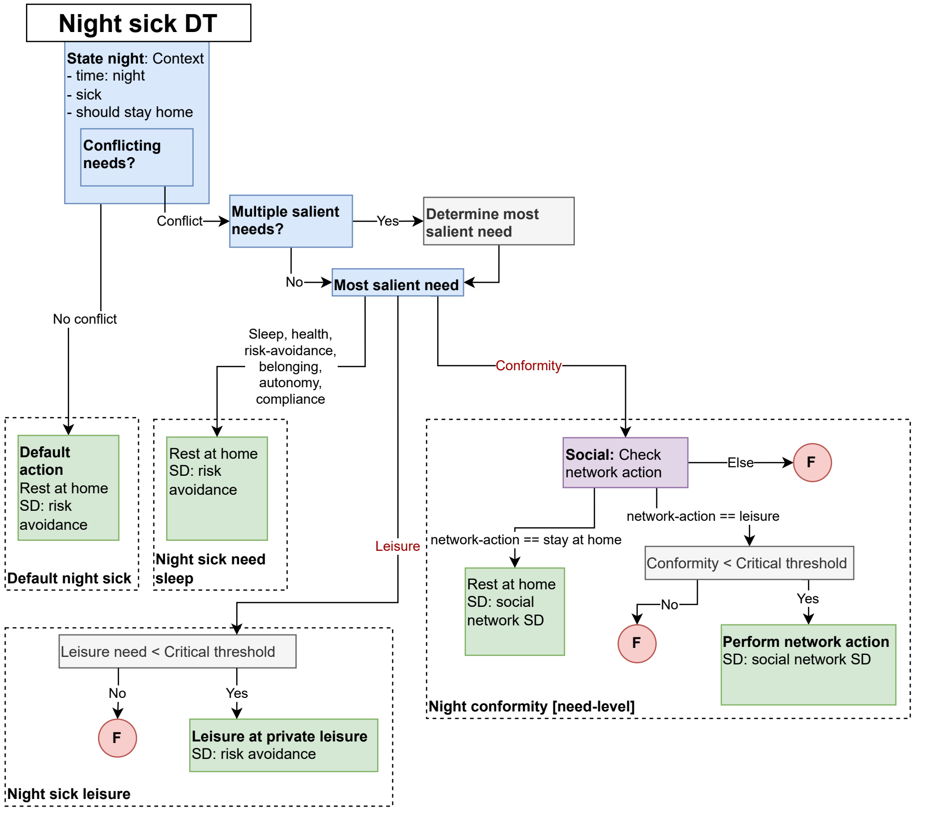


Figure 5.17: Decision tree - Night Sick Full

Normative Considerations

The agent does not explicitly have to check if it is required to stay at home in the decision tree. This is the case since if the agent is sick it is already implied that the agent is required to stay at home given the pandemic context. If the most salient need is compliance, the default action of resting at home is supported. In this decision context, the agent believes it is sick and in the context of a pandemic this means the agent needs to stay at home. The agent is not allowed to leave the house, and since this knowledge is already given a priori based on the disease status, it does not have to be retrieved somewhere in the decision tree (as a normative check).

Also, since the compliance need is already in the needs that are being checked. This gives a bit of implied knowledge. If there are no conflicting needs, the default action is selected without further deliberation. If there are salient needs, the default action is selected if the compliance need is the most salient. If another need is the most salient need, it will imply that that need is more salient than the compliance need. When this is the case, there is no need for an actual comparison between the most salient need and the compliance need (which has been shown, for example, in the night state decision tree Figure 5.16).

Since both the agent's check whether to stay at home and the check with the compliance need level are not necessary, both are implied by the state. No normative checks have to be performed on this decision tree. This has simplified the decision process.

Private Leisure

If the agent wants to satisfy the leisure need. It will choose to have leisure at the private leisure place. This choice was made because the agent believes itself to be sick and, therefore, still wants to minimise the risk of spreading to others. Since in terms of leisure satisfaction, there is no difference in leisure satisfaction between leisure at a public leisure place or leisure at a private leisure place.

Conformity

The bottom right side of Figure 5.17 shows the decision tree for conformity. If the network action is rest at home, the agent will perform this action. If the network action is leisure, the agent will perform the leisure action if the conformity need is below the critical threshold. If this is not the case, the need-based deliberation is used. In all other cases, need-based deliberation is used.

5.4.2 Freetime Decision Tree

The freetime decision tree is activated when the minimal context is equal to $\{time = freetime, status = not\_sick\}$. Table 5.5 shows which actions are available in this decision context. It should be noted that rest at home is only **weakly** preferred in this decision tree. This is in contrast with the decision

Actions	Description
Rest at home	Default, **weakly** preferred
Leisure at private leisure	
Leisure at public leisure	
Shopping at essential shop	Students, Workers, and Elderly only
Shopping at non-essential shop	Students, Workers, and Elderly only

Table 5.5: Available actions - Minimal context freetime

trees we have seen before, where the default was strongly preferred. If the default is weakly preferred, the most salient need does not have to be below the critical threshold for the agent to choose a different action from the default. This will be explained in more detail below in the decision tree.

The need	**Relation to action**
Sleep	Supports default
Risk avoidance	Supports default
Belonging	**Conflict: Rest or Leisure action**
Conformity	**Conflict: Network action**
Leisure	**Conflict: Leisure action**
Luxury	**Conflict: Non-essential shopping**
Food safety	**Conflict: Essential shopping**

Table 5.6: Relevant needs - Minimal context freetime

Based on the available actions, a list can be created of needs that are related to those actions and the sick status of the agent (see Table 5.6). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4.

The Decision Tree

Figure 5.18 and Figure 5.19 together show the complete freetime decision tree. For readability purposes the decision tree has been split. By default the agent will rest at home. This default action is also supported by the sleep and risk-avoidance needs. The belonging, leisure, conformity, food safety and luxury needs can make the agent deviate from this. Belonging can be satisfied by rest at home, however also by leisure activities. Therefore the decision tree is extended and in the case that the agent does not have to stay in quarantine it can choose a leisure action. Since the default action is weakly preferred (see Table 5.5). The belonging need does not have to be critical for the agent to select an action that deviates from the default action. The same holds for all other conflicting needs. Leisure can also be satisfied by rest at home. However only slightly and therefore the leisure activities are preferred. If the agent does not have to stay at home or if leisure and compliance are below the critical threshold, the agent can select a leisure action.

Figure 5.19 shows the decision tree for conformity, food safety and luxury. Conformity has a couple of extra checks. While rest at home does not require extra checks the other actions require the normative check. If the network action is a shopping action the agent will also check its financial needs. Shopping requires a financial need check, where the food safety level or the luxury level have to be lower than both the financial stability and financial survivability.

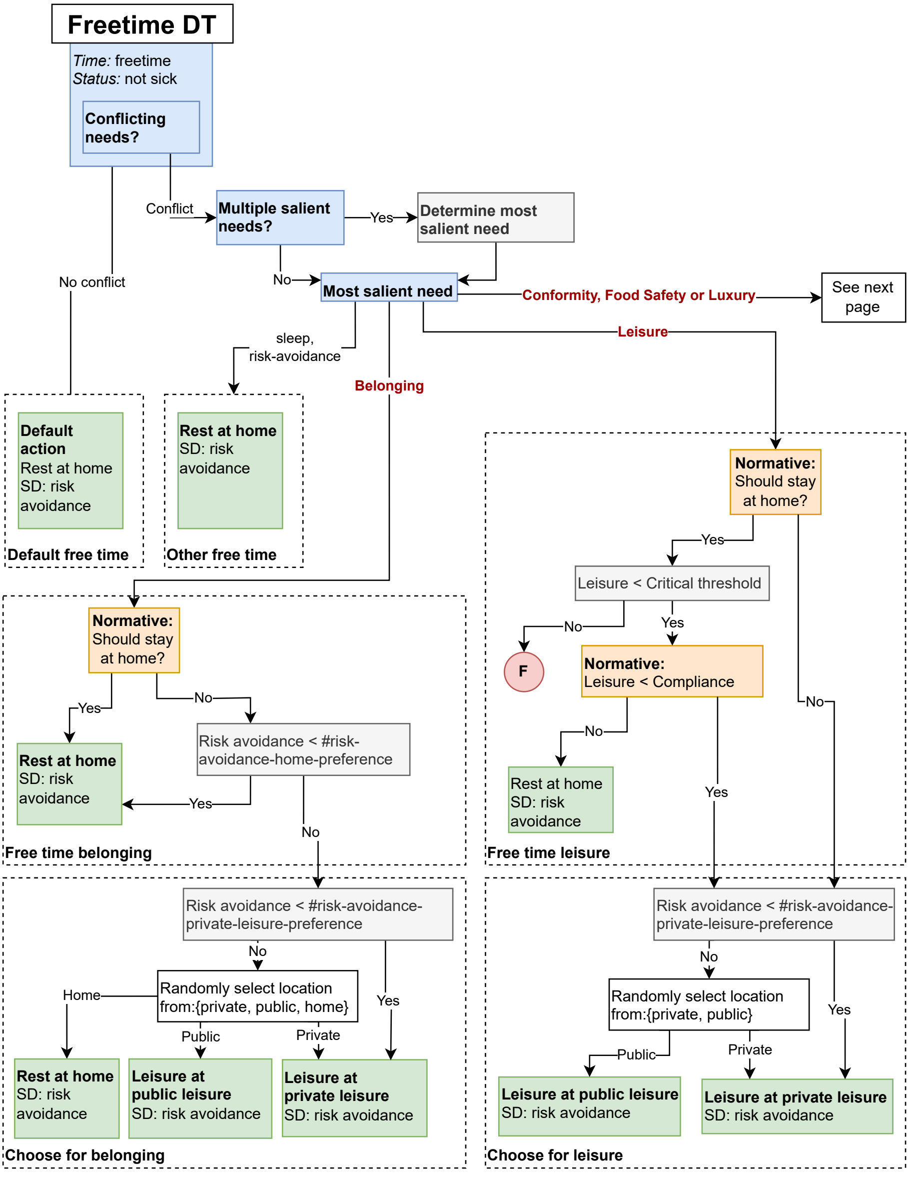


Figure 5.18: Decision tree - Freetime 1

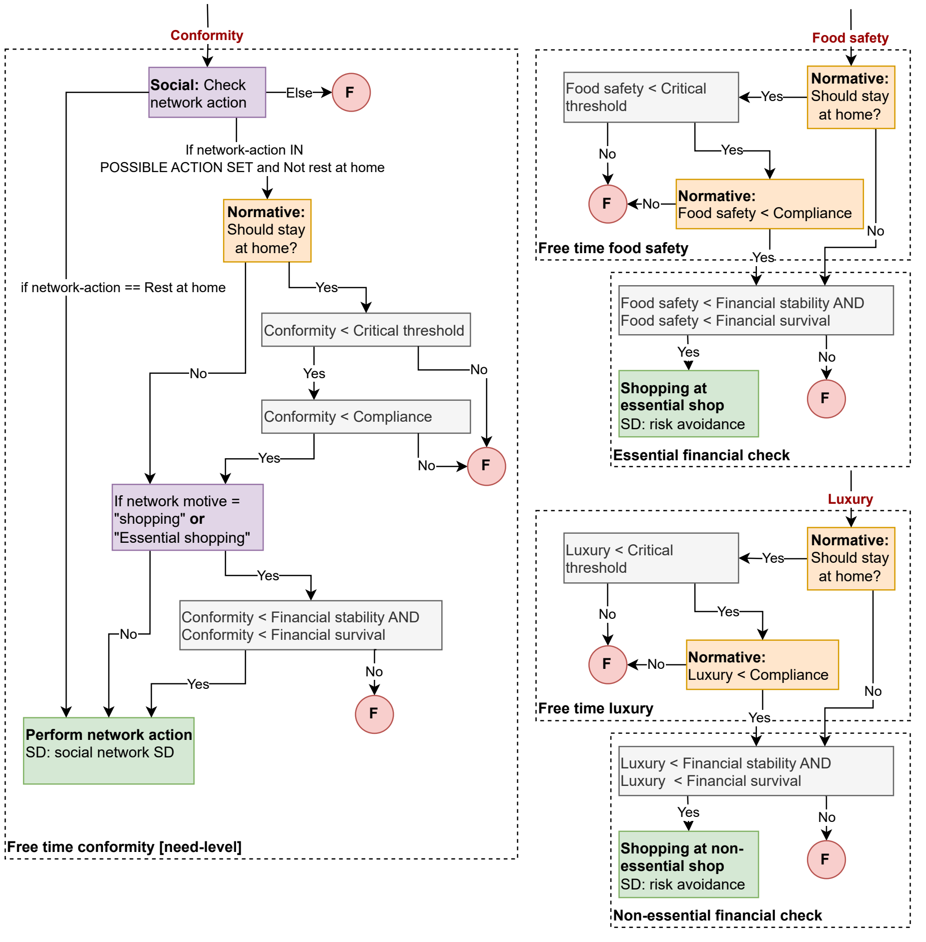


Figure 5.19: Decision tree - Freetime 2

Leisure Habits in Freetime

During experimentation with the DCSD ASSOCC model it became clear that leisure actions were not frequently taken by the agents. Rather than doing extensive rebalancing of the needs of the whole ASSOCC model, it was chosen to incorporate leisure habits. The leisure habits are preplanned moments during the agent's freetime where the default action is not resting at home, but the default action is a leisure action.

Leisure habits are implemented in the following way. They are enabled by setting the *ce-leisure-habits* variable to true. Normally, when it is free time the agent will have *rest at home* as default action that is weakly preferred. The leisure habits, change the default action at specific preplanned moments to performing a leisure activity at either public or private. This default action is strongly preferred. The decision tree for this specific situation is the following. See below.

The preplanned moments on which agents can have habits are stochastically determined at initialisation of the agents. First, it is determined based on whether the agent is retired on which moments the agent has freetime, and thus on which moment a leisure habit can be planned. The moments in which a leisure habit can be planned are indicated by free choice in Table 4.2, seen in the previous chapter. The difference is that retired agents are available on all the days in the morning, afternoon and evening, while the agents of the other age groups are not available on Monday to Friday in the morning and afternoon. It can happen that a leisure habit is attributed on a Saturday morning or afternoon to a worker agent that work in a shop. In this case the obligation decision tree and the leisure habit is not used. Although since there are only about 15 worker agents that work in the shops, this will not have a noticeable impact on the output of the simulation.

For all agents a leisure action is planned during the evening with a probability of 0.7. Then for all agents a leisure action is planned during the available time slots for the morning and afternoon. For retired agents this could be at any day, for other agents this would only be on Saturday or Sunday. Lastly another leisure habit is planned during the available time slots for the morning or afternoon. This last leisure habit is planned certainly for retired agents,

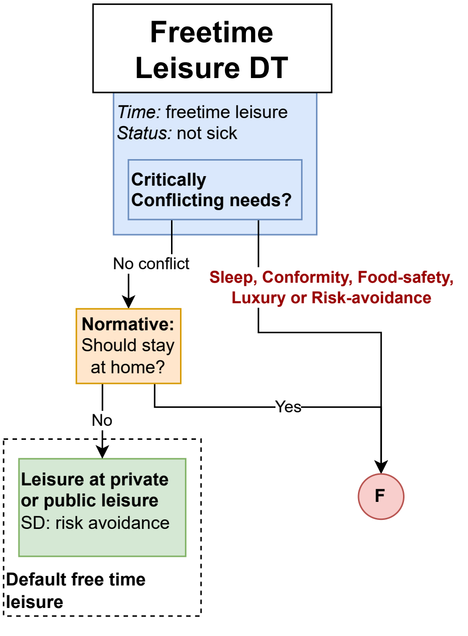


Figure 5.20: Decision tree - Freetime Leisure Habits

and with a probability of 0.5 for all the other agents. The leisure action that is planned has a probability of 0.2 to be *leisure at public leisure* action and otherwise is a *leisure at private leisure* action.

Figure 5.20 shows the decision tree when a leisure habit applies. It will even be checked before the Freetime decision tree. If there is a leisure habit available the freetime leisure decision tree becomes active, otherwise the standard free-time decision tree becomes active. In the free time leisure decision tree there are five conflicting needs that are checked. The needs are sleep, conformity and risk-avoidance and for the non-child agents food-safety and luxury. The needs however are checked if they are below the critical threshold. If they are only salient and not critical the agent will still prefer to perform the leisure activity as this is strongly preferred. This is similar to the night decision tree where rest at home is strongly preferred. If there are one or more critically conflicting needs the agent will use need-based deliberation. If there is no conflict with critically conflicting needs it is checked whether the agent should stay at home or not. If the agent does not have to stay at home it will perform the preplanned leisure activity, otherwise it will use need-based deliberation. For simplicity reasons compliance is not checked in this decision tree. The need-based deliberation can handle the conflict that may arise when the agent is in quarantine and the compliance need is very salient. Since this specific case is not expected to be frequently happening, we chose to keep this particular decision tree is simple as possible.

5.4.3 Freetime Sick Decision Tree

The freetime sick decision tree is activated when the minimal context is equal to $\{time = freetime, status = sick\}$. Table 5.7 shows which actions are available given the context. This is similar Table 5.5 with the only difference being that the default action is **strongly** preferred instead of **weakly**. This chose has been made since the agent is sick and therefore **strongly** prefers to rest at home.

Actions	**Description**
Rest at home	Default, **strongly** preferred
Leisure at private leisure	
Leisure at public leisure	
Shopping at essential shop	Students, Workers, and Elderly only
Shopping at non-essential shop	Students, Workers, and Elderly only

Table 5.7: Available actions - Minimal context freetime sick

Based on the available actions, a list can be created of needs that are related to those actions and the sick status of the agent (see Table 5.8). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. Compared with Table 5.6 in the previous section, the same needs are relevant, however there are more needs supporting the default action.

The need	Relation to action
Sleep	Supports default
Health	Supports default
Risk avoidance	Supports default
Belonging	Supports default
Compliance	Supports default
Leisure	**Conflict: Leisure action**
Conformity	**Conflict: Network action**
Luxury	**Conflict: Non-essential shopping**
Food safety	**Conflict: Essential shopping**

Table 5.8: Relevant needs - Minimal context freetime sick

The Decision Tree

The complete decision tree is shown by Figure 5.21 and Figure 5.22. By default the agent will rest at home. This is also supported by many of the needs, i.e. sleep, health, risk-avoidance, belonging, and compliance. The leisure, conformity, luxury and food safety needs can make the agent deviate from this default. When the leisure need is critical the agent will select the private leisure action. The public leisure action is not considered since the agent is sick and

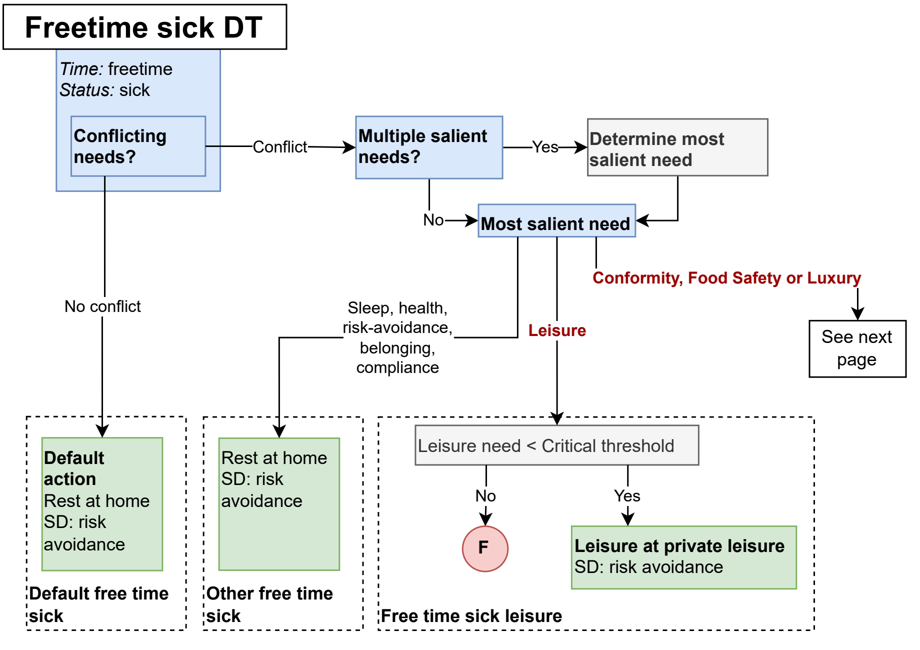


Figure 5.21: Decision tree - Freetime sick 1

wants to avoid places where many other agents come. The private leisure action gives the agent an equivalent amount of leisure and is therefore sufficient.

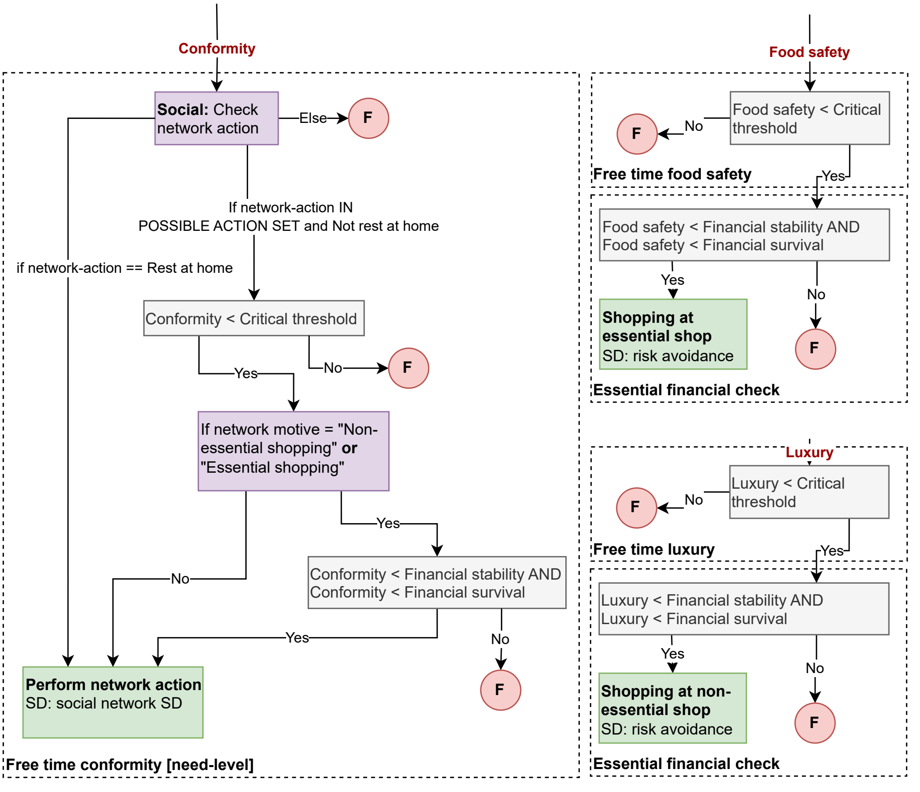


Figure 5.22: Decision tree - Freetime sick 2

Figure 5.22 shows the decision tree for conformity, food safety and luxury. Shopping requires a financial need check, where the food safety level or the luxury level have to be lower than both financial stability and financial survivability. Conformity checks the network action and if there is a clear preference selects the network action. If there is no clear preference ASSOCC need-based deliberation will be used.

5.4.4 Obligation Decision Tree

The obligation decision tree is activated when the minimal context is equal to $\{time = obligation, status = not\_sick\}$. Table 5.9 shows the actions available for agents. Although it is strongly preferred that the perform their study or work action, the other actions are available. If the agents can rest at home as an alternative if they really want to, they should also be able to shop or have leisure. The workers have by default the work at workplace action, the students the study at university action and the children the study at school action

Actions	**Description**
Work or Study	Default, **strongly** preferred
Rest at home	
Leisure at private leisure	
Leisure at public leisure	
Shopping at essential shop	Students or Workers
Shopping at non-essential shop	Students or Workers

Table 5.9: Available actions - Minimal context obligation

Based on the available actions, a list can be created of needs that are related to those actions and the sick status of the agent (see Table 5.10). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. The compliance and belonging needs support the action Work or Study action. Then there are some needs that support the Study or Work action even more such as autonomy, and financial survival and financial stability for workers. If these needs are salient then agents may even break quarantine to go to study or work. This is not default behaviour and that is why they are considered conflicting. The other needs give a preference for other actions that the default, as explained in previous sections.

The need	**Relation to action**
Compliance	Supports default
Belonging	Supports default
Autonomy	**Conflict: More likely to work or study**
Financial survival	**Conflict: More likely to work (worker only)**
Financial stability	**Conflict: More likely to work (worker only)**
Risk-avoidance	**Conflict: Rest action**
Sleep	**Conflict: Rest action**
Leisure	**Conflict: Leisure action**
Luxury	**Conflict: Non-essential shopping**
Food safety	**Conflict: Essential shopping**
Conformity	**Conflict: Network action**

Table 5.10: Relevant needs - Minimal context obligation

The Decision Tree

Figure 5.23 shows the first part of the obligation decision tree. By default the agent will check whether it should be in quarantine. If not then it will perform its obligation activity, study or work. If it should stay in quarantine it will rest at home. This same decision tree is activated when compliance and belonging are the most salient needs. It is a slightly different decision tree when the autonomy need is salient, and for workers the financial survival and financial stability needs. If one of these needs is critical, then the agent will even break quarantine to go to study or work. This deviates from the default action/case where the agent would rest at home if it is in quarantine. A most salient risk-avoidance or sleep make the agent rest at home if it should or if one of these needs is critical. Otherwise it will use full assocc deliberation.

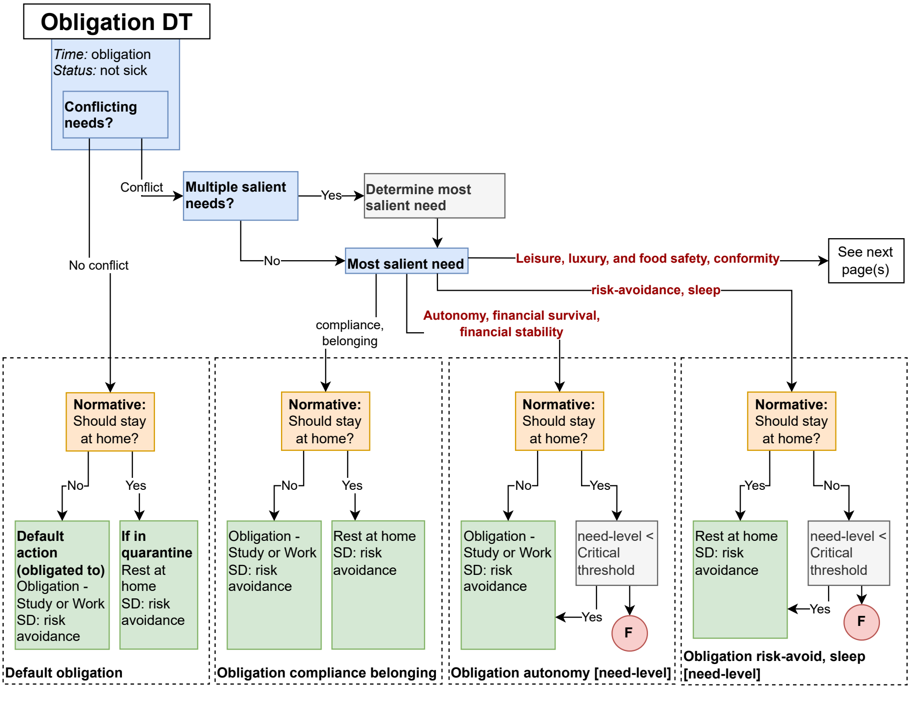


Figure 5.23: Decision tree - Obligation 1

Figure 5.24 shows the decision tree for when the food safety or luxury need is the most salient. Actually, it is very similar to the free time sick 2 decision tree (see Figure 5.22 before). There is only an extra check on whether the agent is a student. If the agent is a student then the food safety need is compared with the financial needs to determine whether the agent can shop. If the agent is a worker this checking of the financial needs is already performed when determining the most salient need.

Figure 5.25 shows the leisure need has to be critical, otherwise need-based assocc deliberation is activated. The tree is similar to the one in free time sick 2 (see Figure 5.22). The decision tree for the most salient conformity need, first determines the network action and then performs the normative considerations. In the implementation this is slightly different as it first performs the normative consideration and then checks the network action. For readabil-

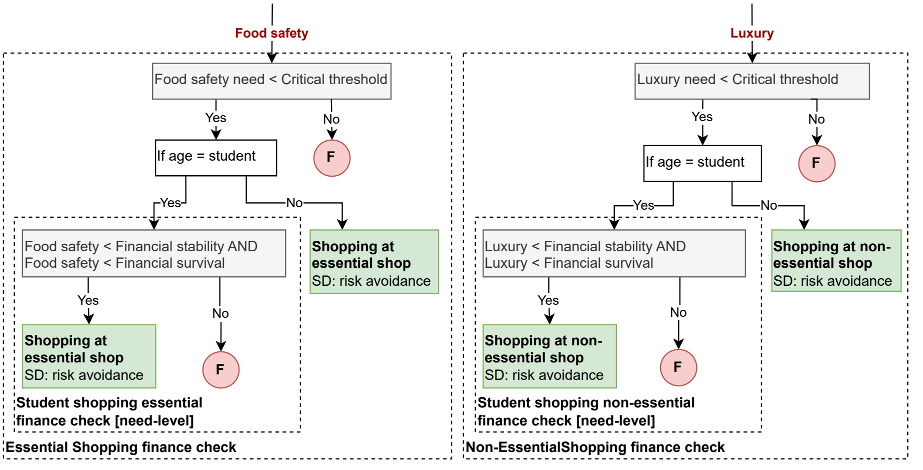


Figure 5.24: Decision tree - Obligation 2

ity purposes this decision tree first considers the network action, but logically speaking the figure and the implementation are equivalent. This can be seen in the *context_state_obligation.nls* file. Table 5.9 shows that the possible action set depends on the age of the agent, as children cannot choose a shopping action.

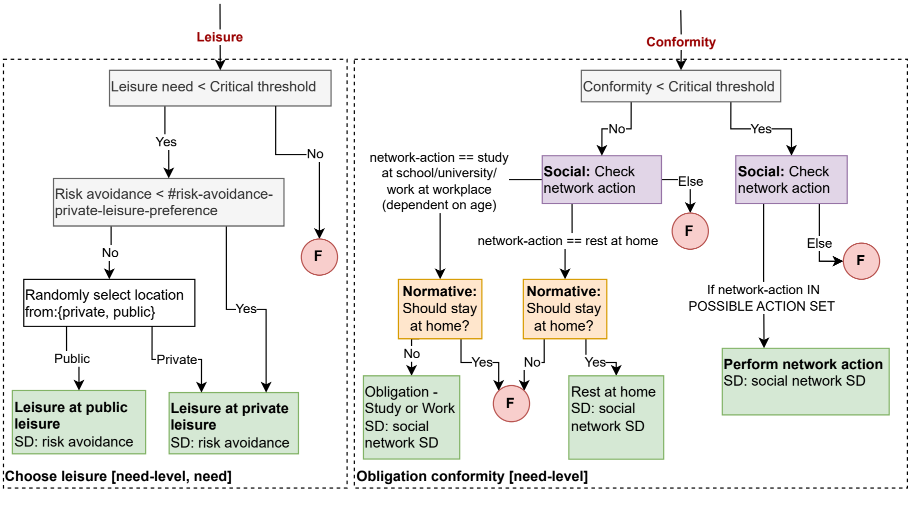


Figure 5.25: Decision tree - Obligation 3

5.4.5 Obligation Sick Decision Tree

The obligation sick decision tree is activated when the minimal context is equal to $\{time = obligation, status = sick\}$. Table 5.11 shows the actions available for the agents in this context. The actions are equivalent to the obligation decision context. However, rather than work or study as default action the default action is rest at home.

Actions	**Description**
Rest at home	Default, **strongly** preferred
Work or Study	
Leisure at private leisure	
Leisure at public leisure	
Shopping at essential shop	Students or Workers
Shopping at non-essential shop	Students or Workers

Table 5.11: Available actions - Minimal context obligation sick

Based on the available actions, a list can be created of the needs related to those actions and the sick status of the agent (see Table 5.12). In this figure, worker and student only is abbreviated to W/S only. The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. There are five needs supporting the default action (decision tree). The other needs all conflict with that decision tree. The financial stability and financial survival need can influence a worker agent to take the work action. The other needs can influence the agent to take other actions.

The need	**Relation to action**
Sleep	Supports default
Health	Supports default
Risk avoidance	Supports default
Belonging	Supports default
Compliance	Supports default
Autonomy	**Conflict: more likely to work**
Conformity	**Conflict: Network action**
Leisure	**Conflict: Leisure action**
Luxury	**Conflict: Non-essential shopping (W/S only)**
Food safety	**Conflict: Essential shopping (W/S only)**
Financial stability	**Conflict: more likely to work (Worker only)**
Financial survival	**Conflict: more likely to work (Worker only)**

Table 5.12: Relevant needs - minimal context obligation

The Decision Tree

Figure 5.23 shows the main part of the obligation sick decision tree. The default action when there is no conflict is to rest at home. Rest at home is also chosen when one of the following needs is the most salient: sleep, health, risk-avoidance, belonging or compliance. If autonomy or for workers the financial survival and financial stability is most salient, the agent will be more likely to study or work. There are two settings in the model based on the *ce-only-obligation-when-health-riskfree-enough* parameter. The initial model would only check the need-level of the most salient need to see if it is below the critical threshold. If this is the case the agent will study or work. With this decision tree many of the children would go to school to study while they were sick. Therefore the tree is changed and when *ce-only-obligation-when-health-riskfree-enough* is set to true, the decision tree on the bottom right is selected. This decision tree will in addition to checking whether need-level $<$

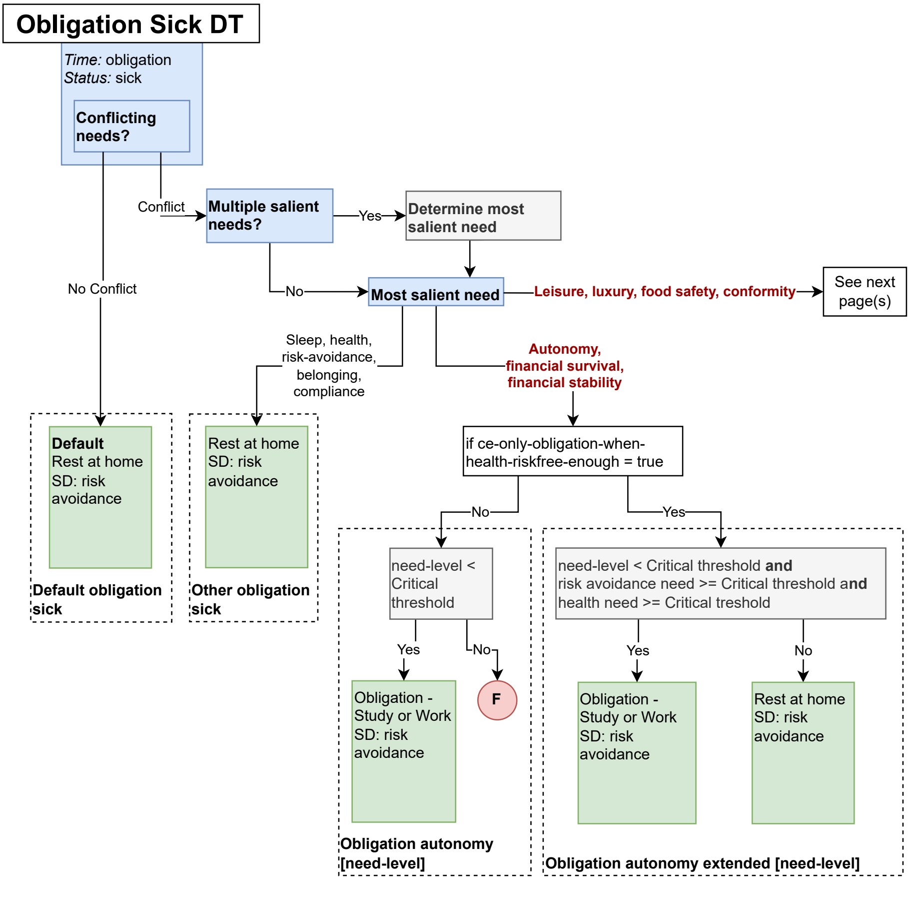


Figure 5.26: Decision Tree - Obligation Sick 1

critical threshold. Also check whether the risk-avoidance need and the health need are at least higher than the critical threshold. The rationale behind this is that agent should at least be somewhat healthy and risk taking to be able to go to work, university or school. The other conflicting needs are described in the diagrams below.

Figure 5.27 shows the decision trees for when food safety and luxury are most salient. These trees are identical to Figure 5.24 in the Section 5.4.4 and will therefore not be explained further.

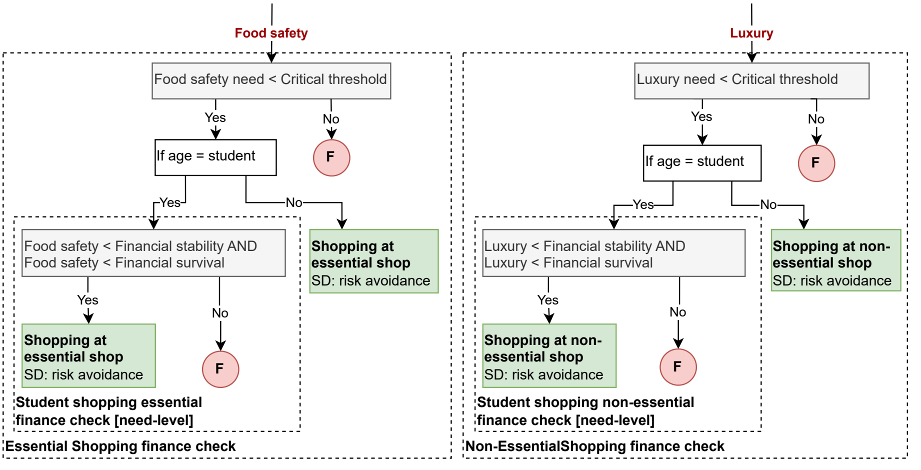


Figure 5.27: Decision Tree - Obligation Sick 2

Figure 5.28 shows the decision trees for leisure and conformity. The leisure decision tree has been simplified; the agent prefers to take the leisure at private leisure action if the leisure need is critical. Otherwise, ASSOCC need-based deliberation is used. If the conformity need level is not smaller than the critical threshold, the decision tree is simple. That is, if the network action is rest at home, the agent will also rest at home, otherwise ASSOCC need-based deliberation is used. If conformity is smaller than the critical threshold, it becomes slightly more complex. The social network action should be in the possible action set. Then it is checked if the network action is shopping or essential shopping. If this is not the case the network action will be performed with social distancing the same as the social network. If the network action is shopping or essential shopping, there is first a check whether the agent is a student and if so the financial needs are considered.

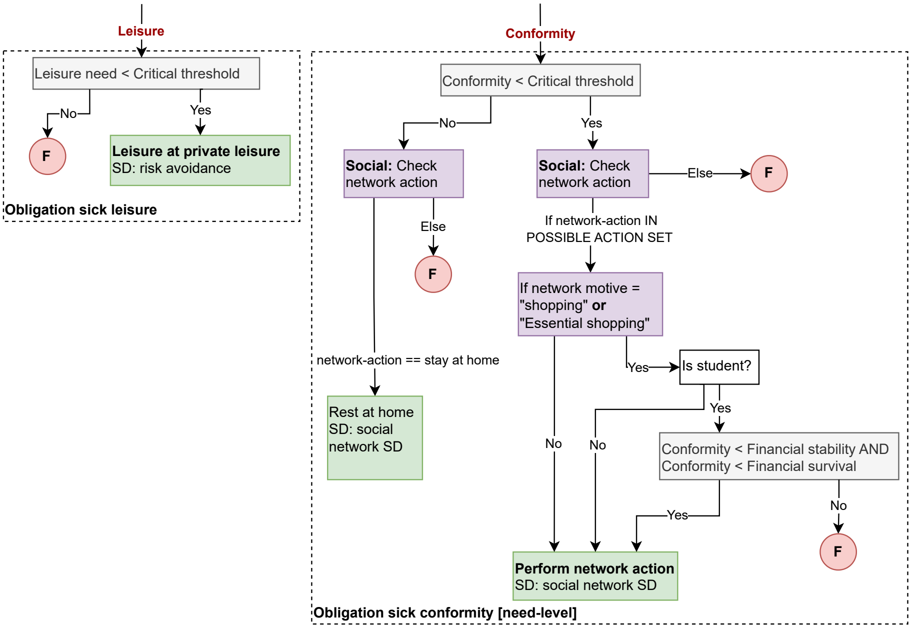


Figure 5.28: Decision Tree - Obligation Sick 3

5.4.6 Obligation Work Home Decision Tree

The obligation decision tree is activated when the minimal context is equal to $\{time = obligation, status = not\_sick, worker, can\ work\ from\ home\}$. This decision tree is only available for workers who can work from home. Table 5.13 shows the actions available for agents. Although it is strongly preferred that the agents work at the workplace, if the agent is in quarantine it will often just choose to work from home. The other actions are also available.

Actions	**Description**
Work at workplace	Default, **strongly** preferred
Work at home	
Rest at home	
Leisure at private leisure	
Leisure at public leisure	
Shopping at essential shop	Students or Workers
Shopping at non-essential shop	Students or Workers

Table 5.13: Available actions - Minimal context obligation WH

Based on the available actions, a list can be created of needs that are related to those actions and the sick status of the agent (see Table 5.14). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. The compliance, financial survival and financial stability support the default action of working from home. The autonomy need supports work from workplace more strongly as it will even allow for disregarding normative checks if the autonomy need is critical. The other needs can be fulfilled by other actions and that is why they are conflicting.

The need	**Relation to action**
Compliance	Supports default
Financial survival	Supports default
Financial stability	Supports default
Autonomy	**Conflict: Support default more strongly**
Sleep	**Conflict: Rest action**
Belonging	**Conflict: Prefer to work from home**
Risk-avoidance	**Conflict: Rest action**
Conformity	**Conflict: Network action**
Leisure	**Conflict: Leisure action**
Luxury	**Conflict: Non-essential shopping**
Food safety	**Conflict: Essential shopping**

Table 5.14: Relevant needs - Minimal context obligation WH

The Decision Tree

Figure 5.29 shows the first part of the obligation work home decision tree. By default, the agent will check whether it should be in quarantine. If not, then it will perform its work at workplace action. If it should stay in quarantine, it will instead work at home. This same decision tree is activated when compliance, financial survival, and financial stability are the most salient needs. It is a slightly different decision tree when the autonomy need is salient. The leisure, luxury and food-safety decision trees are equivalent to the trees described in the normal obligation decision tree (see Figure 5.24 and Figure 5.25). The decision trees for other salient needs are not implemented, which is indicated by that part of the decision tree being opaque and containing a red cross. Implementing those aspects of the decision trees is expected to have a low impact on the execution time of the DCSD as a whole. That is, since the obligation work home decision tree and obligation work home sick decision tree are only used for a relatively small portion of the agents. And especially for those specific needs that are not the default, we do not expect many function calls.

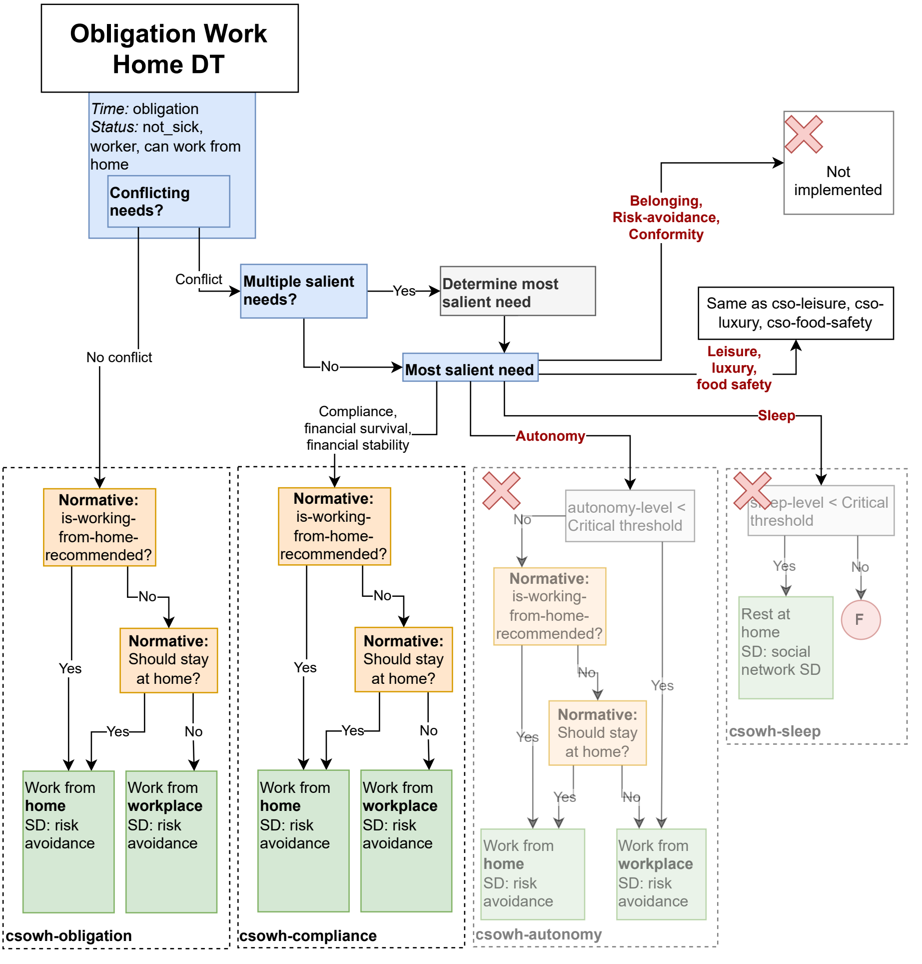


Figure 5.29: Decision tree - Obligation Work Home 1

5.4.7 Obligation Work Home Sick Decision Tree

The Obligation Work Home Sick decision tree is activated when the minimal context is equal to $\{time = obligation$, $status = sick$, $worker$, $can\ work\ from\ home\}$. This decision tree is only available for workers that can work from home. Table 5.11 shows the actions available for the worker agents in this context.

Actions	**Description**
Rest at home	Default, **strongly** preferred
Work or Study	
Leisure at private leisure	
Leisure at public leisure	
Shopping at essential shop	
Shopping at non-essential shop	

Table 5.15: Available actions - Minimal context obligation WH sick

Based on the available actions, a list can be created of the needs related to those actions and the sick status of the agent (see Table 5.12). The relevant needs are based on the previously mentioned need satisfaction and actions table, Table 4.4. There are five needs that support the default action (decision tree). The other needs all conflict with that decision tree. The financial stability and financial survival need can influence a worker agent to take the work action. The other needs can influence the agent to take other actions.

The need	**Relation to action**
Sleep	Supports default
Health	Supports default
Risk avoidance	Supports default
Belonging	Supports default
Compliance	Supports default
Autonomy	**Conflict: more likely to work**
Conformity	**Conflict: Network action**
Leisure	**Conflict: Leisure action**
Luxury	**Conflict: Non-essential shopping**
Food safety	**Conflict: Essential shopping**
Financial stability	**Conflict: more likely to work**
Financial survival	**Conflict: more likely to work**

Table 5.16: Relevant needs - Minimal context obligation WH sick

The Decision Tree

Figure 5.30 shows the main part of the obligation work home sick decision tree. The default action when there is no conflict is to rest at home. Rest at home is also chosen when one of the following needs is the most salient: sleep, health, risk-avoidance, belonging, or compliance. The leisure, luxury and food safety needs are the same for obligation work home sick as they are for obligation sick. Therefore, they are not explicitly mentioned here. If autonomy, financial survival, or financial stability is critical, the agent will work from home instead. The opaque decision tree part with the red cross, that is conformity, is not implemented. The effects on the DCSD ASSOCC as a whole by not implementing this function are negligible.

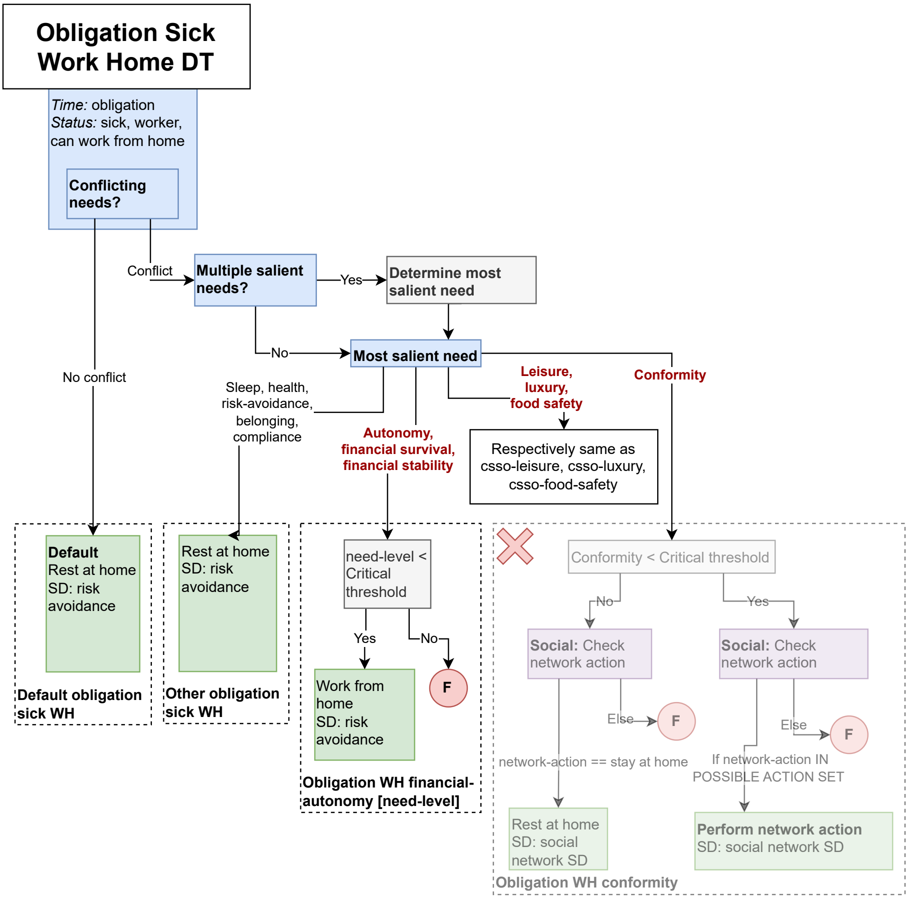


Figure 5.30: Decision Tree - Obligation Work Home Sick

5.5 Optimising DCSD ASSOCC - Obligation DT

In principle, one can extend the DCSD decision trees until DCSD can make a decision in every situation. This is a lot of work, and at some points the speed-up benefits will not be worth the extra effort. The decision trees could easily become twice as large by expanding for all decision situations. Instead of expanding the decision trees as a whole, a more targeted optimisation is desired. By analysing which decision situations the ASSOCC's need-based deliberation uses more frequently, specific places of performance loss can be found.

5.5.1 A Small Optimisation Analysis

The frequency can be measured by running the simulation and keeping track of which decision situations require ASSOCC's need-based deliberation. To do this, in the code each red encircled F gets a counter. Every time a need-based deliberation is used since DCSD could not find an action, the number relating to that decision situation is incremented by one. At the end of the simulation run it becomes clear which unsolved decision situations are more frequently visited. Then, a simulation run was performed, and we measured all need-based deliberation calls per decision situation. The settings for the run were, 350 households, context-setting is four, the simulation was run for 240 ticks and the random seed was zero. As it is only an approximation and we do not need very precise numbers, one run is sufficient.

Table 8.17 in the Appendix 8.2 shows the complete results for each of the decision situations. To summarise those results, most of the need-based deliberation calls (more than 80%) happen in four specific functions. These functions are the food safety and luxury functions in the obligation and obligation work home decision trees. The other 17 functions account for only 20% of the number of need-based deliberation calls. Based on these results, it was clear that an optimisation should be performed in the obligation and obligation work at home decision trees.

5.5.2 Optimising the Obligation Decision Tree

The decision trees for food safety and luxury are equivalent in the decision trees for obligations and obligations at home. Figure 5.31, which has also been shown earlier, shows the part of the decision trees that should be optimised. There are two situations in which the decision tree chooses the full ASSOCC deliberation. 1) When the most salient need in question (food safety or luxury) is not critical. 2) This case is only for students: when the most salient need in question is not less than both financial stability and financial survival.

Rather than just need-based deliberation, we could decide that agents will go to work if the food safety or luxury need is most salient but not critical. Figure 5.32 shows this change. It checks if the agent has to stay at home. If the agent does not have to stay home, it will instead study or work as this is strongly

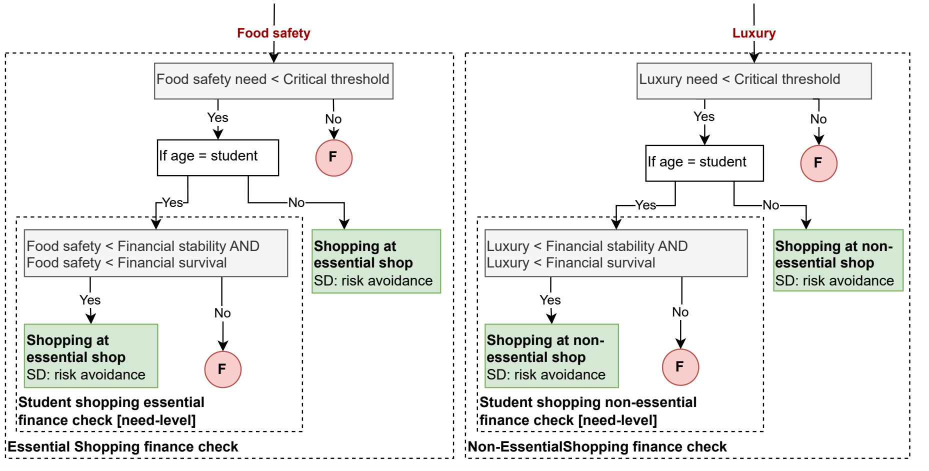


Figure 5.31: Decision tree - Obligation 2

preferred. In the exceptional case where food safety or luxury is salient, but not critical, and the agent needs to stay home, need-based deliberation is used again. With this optimisation, the DCSD model should be efficient enough.

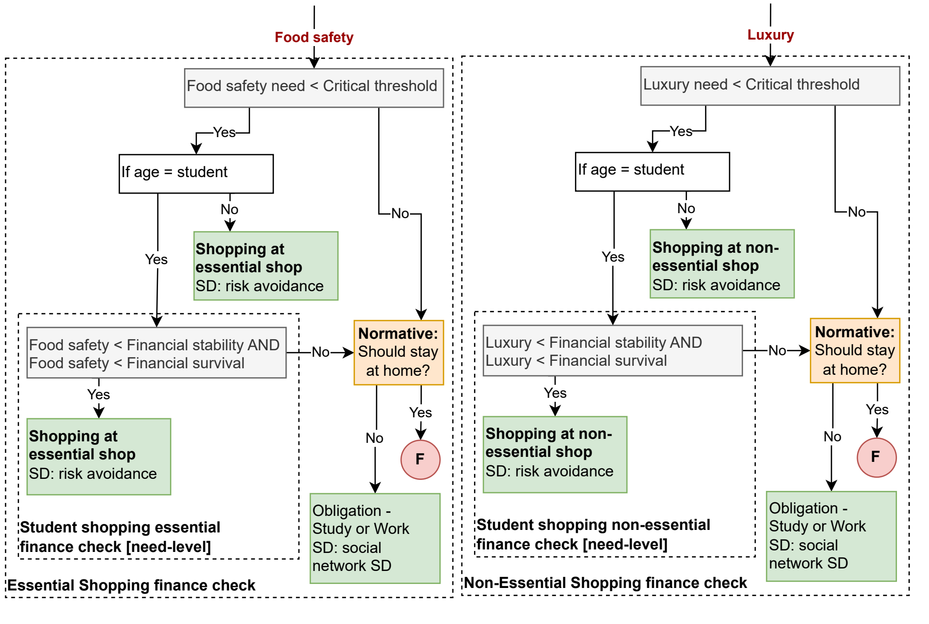


Figure 5.32: Optimised food safety and luxury in obligation decision tree

5.5.3 The Final DCSD Model

Figure 5.33 shows the abstract representation of the DCSD ASSOCC model. Now, it includes all the relevant information and the optimisation mentioned above. This version of the DCSD has been named the Full DCSD. Later in the thesis, DCSD ASSOCC will be used to refer to ASSOCC model with Full DCSD enabled. By using Full DCSD, the need-based deliberation calls for food safety and luxury in obligation and obligation WH, have been greatly reduced.

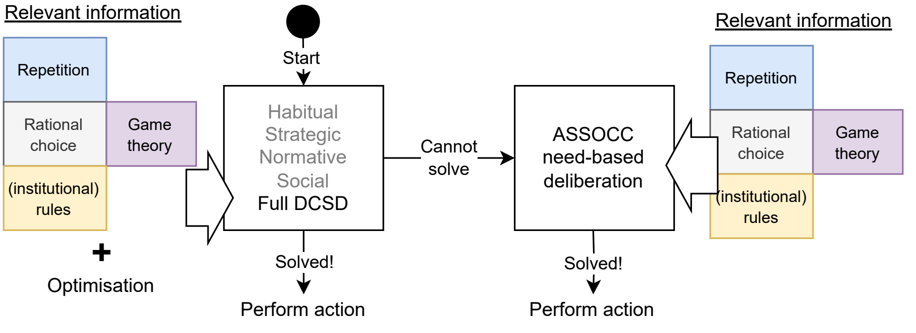


Figure 5.33: DCSD in ASSOCC - Full

5.6 Conclusion

This chapter showed how the DCSD framework, Chapter 3, can be used to create a concrete deliberation model for the ASSOCC framework [19] (described in Chapter 4). First, the general aspects such as the information relevance matrix and the cell transitioning matrix are determined. Secondly, the DCSD ASSOCC is conceptualised as a decision tree.

The first step was to determine the building blocks of DCSD using the ASSOCC framework. The information relevance matrix was filled in. The ASSOCC agents use the information from the following cells to deliberate: repetition, rational choice, institutional rules, and game theory. Then it was determined which elements of the meta-deliberation are usable in a DCSD ASSOCC model. This were only actions, as the other meta-deliberation aspects were not explicitly used in the ASSOCC framework.

The second step was to determine the conceptual architecture that can then be implemented in the ASSOCC framework. As a conceptual architecture, decision trees were chosen for their efficiency and flexibility. On the one hand, decision trees allow for quick decisions when the deliberation can stay in the shallow part of the tree. On the other hand, they allow for more complex deliberation by deliberating deeper into the trees. Finally, the DCSD model is implemented in the ASSOCC framework.

5.6.1 Answering Research Question 2: How can context-sensitive deliberation be implemented taking both efficiency and realism into account?

This chapter answers the research question by showing an implementation of context-sensitive deliberation. It uses the Dynamic Context-Sensitive Deliberation framework to implement Context-Sensitive Deliberation in the ASSOCC model. To achieve this, first, the relevant deliberation elements in the ASSOCC simulation are determined, using the information relevance matrix, the transition matrix, and the meta-deliberation as blueprint. The second step was to develop an implementable conceptual model using the DCSD framework. This conceptual model of context-sensitive deliberation was modelled using a decision tree as an algorithm. Since frequently chosen actions, default actions, have been implemented at a shallow level in the decision tree, the decision tree remains efficient. The more complex deliberations that require more information are accessible further into the decision tree. The actual implementation is available on GitHub[3]. In the following chapter, Chapter 6, DCSD ASSOCC model will be extensively compared with the Original ASSOCC model to evaluate context-sensitive deliberation in terms of realism and scalability.

[3]`https://github.com/maartenjensen/ASSOCC-context`

Chapter 6

Evaluation

This chapter will show the comparison between Original ASSOCC and DCSD ASSOCC as initially described in Chapter 4. First, it will be determined whether the DCSD ASSOCC model is realistic enough and then it will be determined whether DCSD ASSOCC can scale deliberation.

The chapter is structured in the following manner. The first five sections are related to determining the realism of the DCSD. Through these five sections it is extensively shown how changes in deliberation can affect the simulation outcome. The sections are named after the versions of the DCSD as shown in Figure 6.47. Section 6.1 describes the Habitual DCSD. Section 6.2 describes the Strategic DCSD. Section 6.3 the Normative DCSD. Section 6.4 the Social DCSD. Section 6.5 describes the Full DCSD or DCSD ASSOCC. DCSD ASSOCC is compared with Original ASSOCC and it is determined whether DCSD is realistic enough.

The section after the realism sections, Section 6.6, will show and discuss the results in terms of scalability in deliberation. In the first experiment it will be determines whether DCSD can solve the deliberation bottleneck. In the second experiment it will be determined whether this result is maintained with higher agent numbers. The last section contains the argument for why DCSD can scale deliberation in agent-based simulations.

Section 6.7 shows the conclusion in which both the argument for retaining realism and the argument for increasing scalability are brought together. Here, the trade-offs that have been made for the DCSD will be evaluated. This will answer **RQ3**.

6.1 Realism: Habitual Behaviour

This section is concerned with the results of habitual deliberation in the ASSOCC model. It explains how habitual information can be used in deliberation and what possible pitfalls can be. Habitual deliberation relates to following default behaviour, but could also relate to learning new default behaviour. Learning new behaviour is out of scope for this thesis, thus the deliberation in this section is related to default behaviour and when to not perform this default behaviour only. The section starts with a naive model of habits and explains why this is not realistic enough and how it can be improved to the Habitual DCSD.

6.1.1 Rigid Habitual Deliberation - Without Infected

When one wants to decrease the computational complexity of deliberation, one of the first things that comes to mind is to incorporate some kind of default behaviour into the model. Very naively one could create a deliberation model that uses only habitual behaviour to determine the action. Figure 6.1 shows how this purely habitual model would look. It would only consider habits and will always find a solution using just the habits. The information considered is only information from the repetition cell. For example, the time and type of day. If its night, the agents should rest at home. If its a working day, the agents that have obligations need to work or study. And on some specific days and time, e.g., Saturday afternoon, the agents have planned behaviour such as shopping and leisure activities.

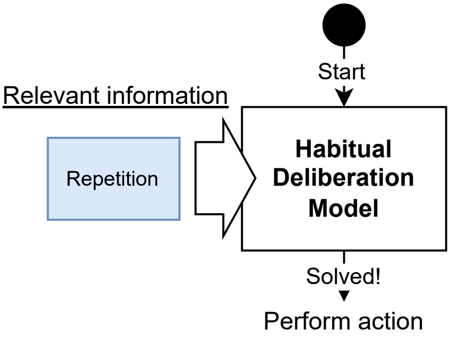


Figure 6.1: Rigid habitual deliberation

The experimental preset is *1.1 rigid-habits-no-infected* and the random seed is set to 2 (this is the case for all upcoming individual runs), the more detailed preset is described in Appendix 8.1.3. This preset will enable the rigid habitual deliberation as indicated in Figure 6.1. The model is also run without infected initially to determine whether the daily life behaviour of the agents is realistic enough. To check whether this provides realistic enough behaviour, first the criteria are analysed, and consequently the time series are analysed for more detail.

Rigid Habits No Infected - Criteria

Since the model is run without infected the only relevant criteria are C1 to C8. Table 6.1 shows the results of the experiment, all the criteria pass. The habits are rigid, i.e. there is no mechanism that makes the agent deviate from the habit. This causes all the values to be at 100% and not be slightly lower as the agents have no means of deviating their behaviour.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	100	TRUE
C3	Recently Ess Shopping, mean > 98%	100	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	100	TRUE
C6	Work at Workplace when possible, 85% < mean	100	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	100	TRUE

Table 6.1: Criteria Values for 1.1 rigid-habits-no-infected

Rigid Habits No Infected - Behaviour and Population Status

Figure 6.2 shows the activities that are chosen by the agents over time. It is simplified to only show rest at home, working or studying activities, shopping (essential and non-essential combined) and leisure activities (public and private leisure combined). It should become clear from the plots that during weekends the agents perform the rest at home activity the most. During working days the agents work or study. Leisure is done mostly during the weekend and only a bit during working days. And shopping is done regularly throughout the week. Based on the criteria and the behavioural time series, the model seems realistic enough.

Figure 6.3 shows that everyone is healthy. This is expected as the model was run without infected enabled.

Rigid Habits No Infected - Scalability

The rigid habits model performs well with the daily life behaviour. However will it perform well in terms of scalability? The deliberation execution time of the Original ASSOCC model for 240 ticks without infected is 86 seconds, a bit more than one and a half minutes. The rigid habits model only requires 0.7 seconds. Which is roughly 123.5 times quicker! This is promising result,

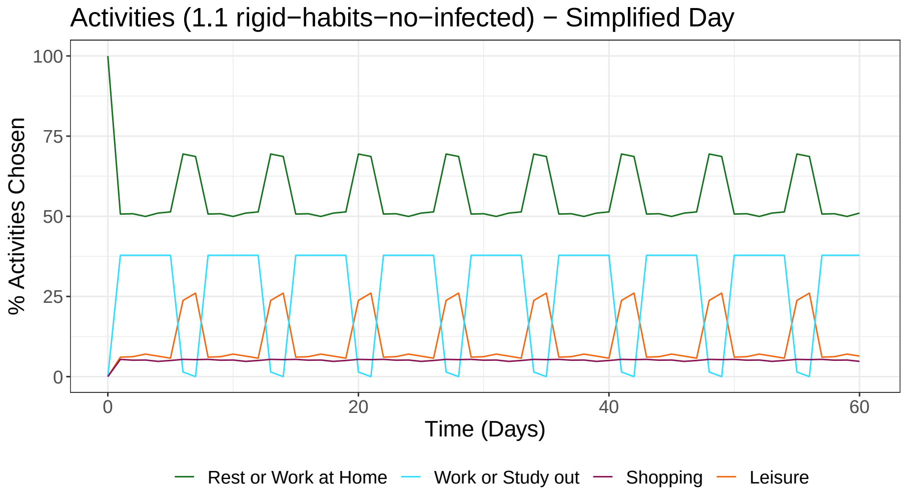


Figure 6.2: Activities in rigid habits without infected

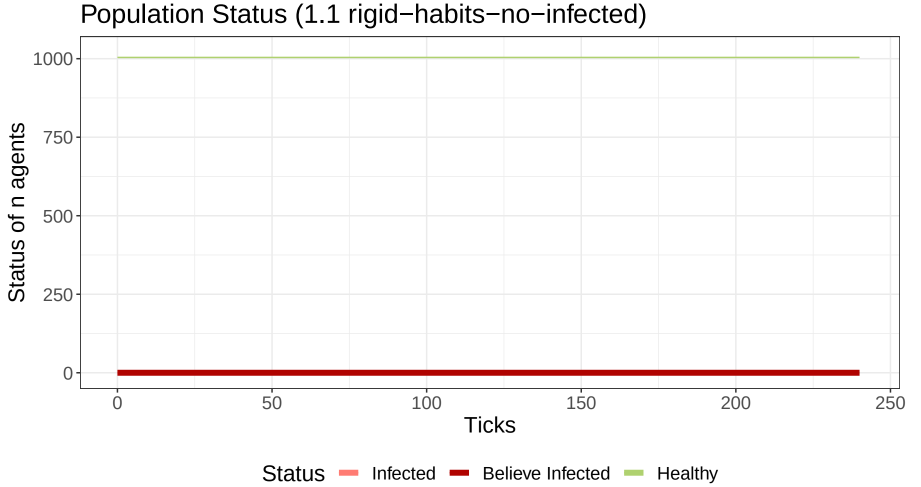


Figure 6.3: Population status in rigid habits without infected

however the model should still be tested with infected enabled to determine whether it is realistic enough. This is done in the next section.

6.1.2 Rigid Habitual Deliberation - With Infected

This time the simulation with the rigid habits deliberation model is run with infected enabled. The experiment preset is *1.2 rigid-habits-infected* which retains all the other settings from the previous experiment with the exception of enabling infected. Appendix 8.1.3 shows the detailed preset settings.

Rigid Habits Infected - Criteria

Table 6.2 shows the criteria measured in this experiment, this time all the criteria are measured. Like the previous experiment, criteria C1 to C8 all pass and have the same values. For criteria C9 to C14 it is a different story and

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	100	TRUE
C3	Recently Ess Shopping, mean > 98%	100	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	100	TRUE
C6	Work at Workplace when possible, 85% < mean	100	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	100	TRUE
C9	Rest When Know Sick, mean > 90%	55.08	FALSE
C10	People in Quarantine, 90% < mean < 100%	55.08	FALSE
C11	Children in Quarantine, 90% < mean < 100%	50.72	FALSE
C12	Students in Quarantine, 90% < mean < 100%	46.01	FALSE
C13	Workers in Quarantine, 90% < mean < 100%	46.09	FALSE
C14	Retireds in Quarantine, 90% < mean < 100%	82.51	FALSE
C15	Infection Peak Tick, 75 < value < 150	110	TRUE

Table 6.2: Criteria Values for 1.2 rigid-habits-infected

they do not pass. The agents do not rest at home enough when they are sick (C9), only half of the agents does, which is too low. The agents also break quarantine too frequently as shown in C10 to C14. Only the infection peak (C15) is within reasonable ranges. We need to analyse the behaviour over time to understand completely why agents do not rest enough when sick and break quarantine.

Rigid Habits Infected - Behaviour and Population Status

When plotting the behaviour of the agents (Figure 6.4) it is the same as the behaviour in the previous section (Figure 6.2). Initially, this seems to be fine; however, when considering the infection curve (Figure 6.5) something strange is happening. The infection curve is different from the one in the previous section where everyone stayed healthy (Figure 6.3), however the behaviour of the agents did not change at all! Just after the actual infected peak, at about 125 ticks (about 31 days), the believe infected peaks. About 75% of the agents at that moment are aware that they are sick and yet the behaviour is not impacted by the slightest. One would expect the agents to be at home more

Figure 6.4: Activities in rigid habits with infected

and doing less activities that are out of the home. However the pattern in the activity plot are exactly the same as before. This also became clear from the criteria that half of the agents is not home when they are sick, but these results confirm. Given these results, the current rigid habits deliberation framework cannot be considered realistic enough for use in the ASSOCC framework.

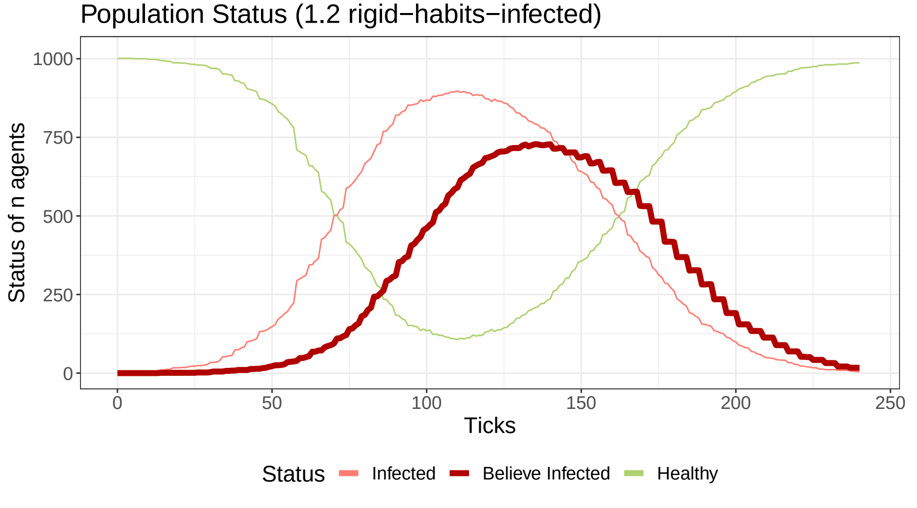


Figure 6.5: Population status in rigid habits with infected

Can Rigid Habits Be Improved?

The rigid habits deliberation could be extended by adding exceptions for being sick by adding additional if statements. This would make the agent able to adapt more. However there are many more situations the agents should be able to adapt too. For example when the schools get closed, children should not go to school. When the food safety need is very salient, the agents should go to an essential shop as soon as possible, this counts for all 12 needs! Agents should also social distance and respond to other measures such as global lockdown. If all of these exceptions were to be added to the deliberation system, it would become very computationally complex. This would lead to a high execution time and very cumbersome code to work with. In the end using rigid habits would practically not be beneficial. However, inspired by Kahnemann [46] Thinking fast and slow, there is a way that habitual deliberation can be of use when used in the deliberation model properly. This will be explained in the next section.

6.1.3 Habitual DCSD - Initial Model

From the previous section, it should become clear that a deliberation model that uses habits should at least be able to deviate from the default behaviour. The Habitual DCSD is capable of doing just this. Inspired by Kahnemann [46] Thinking fast and slow, the Habitual DCSD and Original ASSOCC's need-based deliberation cooperate (see Figure 6.6). The Habitual DCSD uses information from the repetition cell to determine which and whether it can take the default action. If this is not possible due to a conflicting need it will instead of need-based deliberation. An example is, an agent registers that its afternoon and has as habitual action Rest at home. However, the food safety need is very salient, thus the agent cannot follow the habit and needs to use a more complex deliberation, need-based deliberation. This system should be able to deal with changes in the environment in contrast with the rigid habits deliberation.

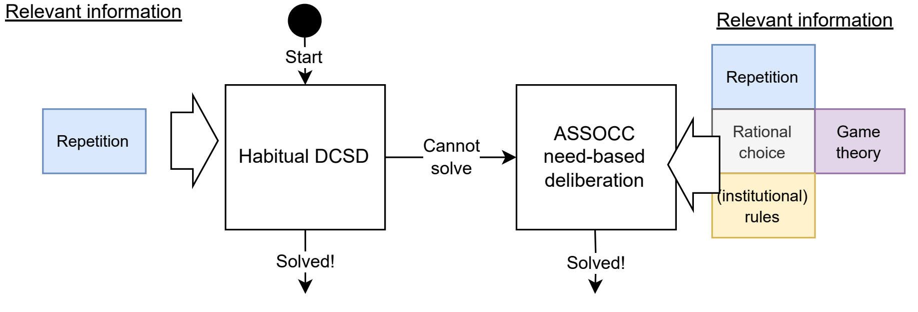


Figure 6.6: Habitual deliberation in DCSD

The experimental preset is *1.3 DCSD-1*, which runs the simulation with infected. This activates the Habitual DCSD model instead of the rigid habits deliberation model. The more detailed preset settings are shown in Appendix 8.1.3. The purpose of this experiment is to show whether the Habitual DCSD provides adapted behaviour and whether this actually gives scalability benefits.

Habitual DCSD - Criteria

Table 6.3 shows the full list of criteria. It can be seen that the model passes almost all the criteria. In contrast with the rigid habits model, this Habitual DCSD model also passes the quarantine criteria (C10 to C14) and the criteria where agents have to rest at home when sick. This is caused by the model being actually adaptable to the situation, which will be extensively shown in the next section. The model does not pass C2 as the value of the percentage of people who have been doing leisure activities the last three weeks is much lower than 98%. Why this exactly happens will be investigated below.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	44.49	FALSE
C3	Recently Ess Shopping, mean > 98%	99.58	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.99	TRUE
C5	Not Skip Work, mean > 98%	98.75	TRUE
C6	Work at Workplace when possible, 85% < mean	92.1	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	100	TRUE
C9	Rest When Know Sick, mean > 90%	96.24	TRUE
C10	People in Quarantine, 90% < mean < 100%	97.14	TRUE
C11	Children in Quarantine, 90% < mean < 100%	98.44	TRUE
C12	Students in Quarantine, 90% < mean < 100%	96.93	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	95.63	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	95.97	TRUE
C15	Infection Peak Tick, 75 < value < 150	116	TRUE

Table 6.3: Criteria Values for Habitual DCSD

Habitual DCSD - Behaviour and Population Status

Figure 6.7 show that also in this run there is an infection curve. Figure 6.8 shows that the agents actually adapt their behaviour. When believe infected is high (roughly from tick 75 to 225), the agents tend to Rest at home more while working less. It can be seen that the shopping behaviour stays roughly similar, however one should not forget that shopping is in a sense essential, for satisfying the luxury and food safety needs. There are no other actions agents can do to satisfy those needs therefore these actions still persist.

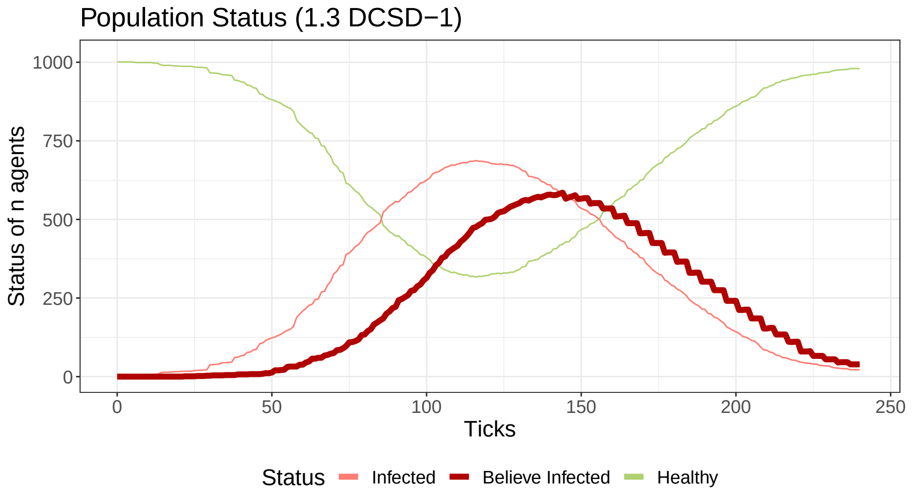


Figure 6.7: Population status for Habitual DCSD

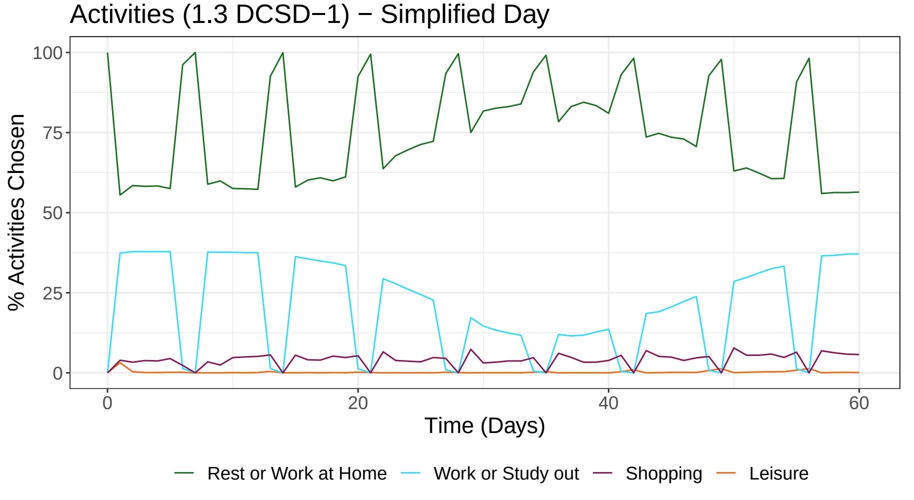


Figure 6.8: Activities for Habitual DCSD

The agents behaviour became more realistic in the sense that they adapt to the infectious state. However it became less realistic in the sense that they are not performing their full range of activities. The leisure activities are hardly ever chosen, while the agents should portray quite some leisure activities in the weekends and even some at working days.

Habitual DCSD - Investigating The Lack of Leisure Activities

As indicated by C2 the agents take leisure actions too infrequently. In principle, the Habitual DCSD should perform leisure activities, since it does take into account the agents' needs. When zooming in on some specific needs over time it becomes more clear (see Figure 6.9). The Habitual DCSD has the salient need threshold set to 0.5. Both the luxury need and the food safety need get low enough to frequently get below this threshold of 0.5. However, the leisure need is on average much higher and for many agents may not reach the salience threshold of 0.5.

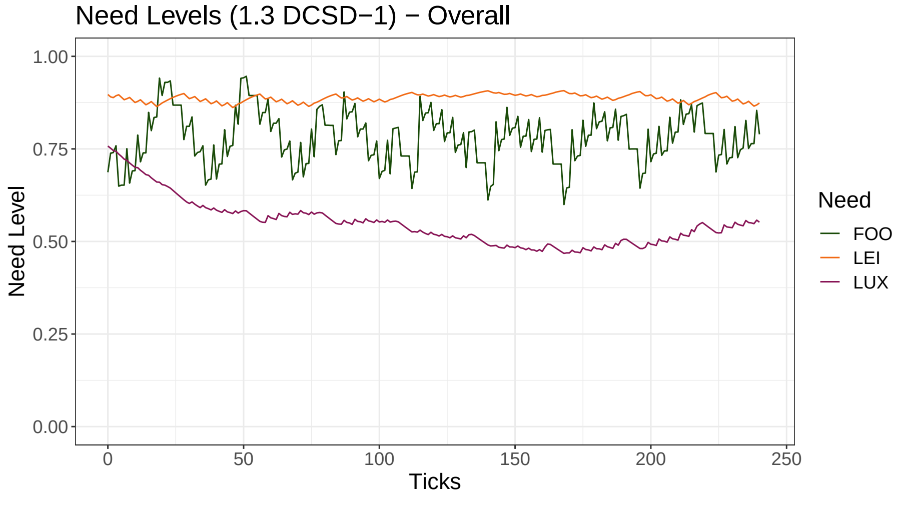


Figure 6.9: Average needs for the population in Habitual DCSD

Figure 6.10 shows the needs over time for a specific agent, a worker agent with id 596. For illustrative purposes, we added a dotted horizontal line (red) at y equals 0.5 that represents the salience threshold. As expected, the food safety need and luxury need get satisfied quickly when they get below the threshold. Between tick 125 and 180 it takes more time to satisfy those needs. This is due to the agent being sick which makes the agent less likely to go out for shopping or leisure activities as it is more likely for the agent to rest at home.

The leisure need never falls below the threshold and thus never pushes the Habitual DCSD to break out of the habit of resting at home. One may wonder how the agents satisfy leisure while not going to leisure activities; however, in the ASSOCC implementation, the agents get a slight leisure satisfaction (0.1) by resting at home. This is sufficient to keep the leisure need at such a high level since the leisure need has a low decay rate, i.e. 0.01 per tick. To solve this and make the agents perform leisure activities, the DCSD model needs to be tweaked.

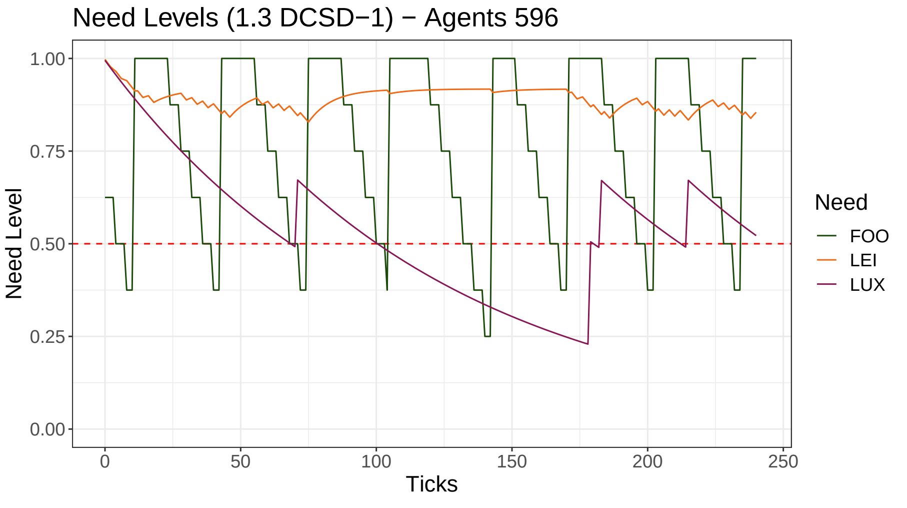


Figure 6.10: Individual needs for agent 596 in Habitual DCSD

6.1.4 Habitual DCSD - Leisure Habits

A couple of things have been tried to adjust the Habitual DCSD model to select leisure actions more frequently. Changing the salience threshold specifically for the leisure need, initially gave more leisure activities in the evening and the weekends. However, when enabling global lockdown agents were too frequently going for leisure activities. The problem with this is that the agents were breaking lockdown too frequently. The ASSOCC need-based deliberation was calibrated with high precision, so small changes can have large effects on the behaviour of the agents.

Instead of balancing the needs, we added leisure activities as a default action at specific time points. There is now a leisure decision tree for these specific time points, where the agent will choose the leisure action if other needs are not salient. This leads to the agents performing leisure activities. The preset *1.4 DCSD-1-leisure-habits* enables these leisure habits in the Habitual DCSD model. See again Appendix 8.1.3 shows the detailed preset settings.

Habitual DCSD Leisure Habits - Criteria

Table 6.4 show the criteria. All the criteria other than C2 pass. For criteria C2 the value is actually much higher, instead of 43% of the population it became almost 84%. However, the value is still not high enough to meet the criteria of 98%. This is happening due to the DCSD model not having unlocked its full potential by enabling all the other layers. This criteria will be met when the normative layer is enabled, as well be explained in Section 6.3.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	83.79	FALSE
C3	Recently Ess Shopping, mean > 98%	98.79	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	99.31	TRUE
C6	Work at Workplace when possible, 85% < mean	90.3	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	99.98	TRUE
C9	Rest When Know Sick, mean > 90%	96.92	TRUE
C10	People in Quarantine, 90% < mean < 100%	97.6	TRUE
C11	Children in Quarantine, 90% < mean < 100%	98.92	TRUE
C12	Students in Quarantine, 90% < mean < 100%	97.2	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	96.35	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	96.99	TRUE
C15	Infection Peak Tick, 75 < value < 150	107	TRUE

Table 6.4: Criteria Values for 1.4 DCSD-1-leisure-habits

Habitual DCSD Leisure Habits - Behaviour and Population Status

In Figure 6.11, the population status graph again shows a curve. The agents activities in Figure 6.12 reflect that the behaviour is changed due to many agents believing they are infected. On closer inspection, it can be seen that the leisure activities are present in the activities graph. There are some small peaks during the weekends.

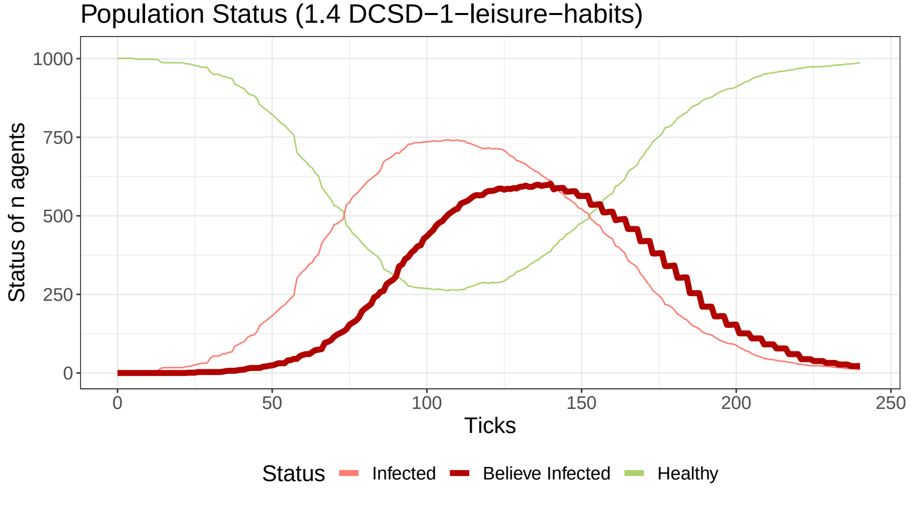


Figure 6.11: Population status in Habitual DCSD

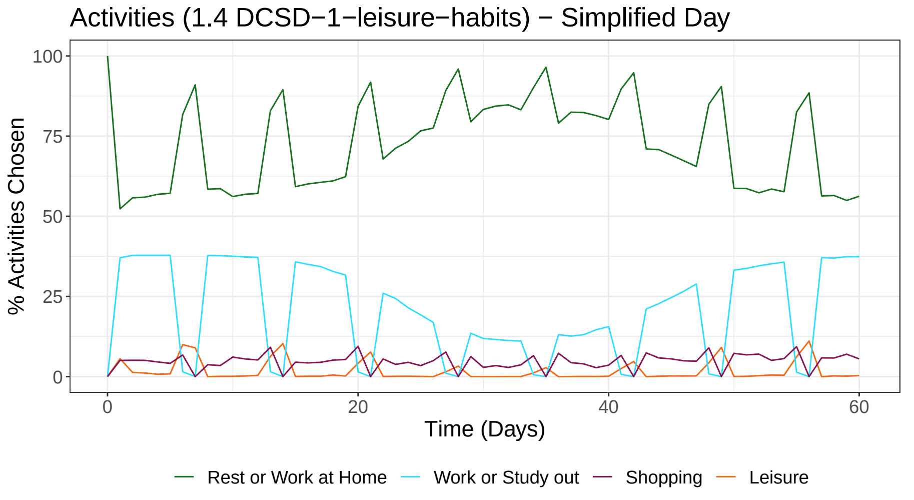


Figure 6.12: Activities in Habitual DCSD

To illustrate more clearly, Figure 6.13 shows in detail (per tick) when leisure activities are chosen. From this figure it becomes clear that especially during the working days there are hardly any leisure activities performed. As discussed in the criteria, the agents do not perform leisure activities frequently enough. This can be explained by the fact that the DCSD still misses the normative

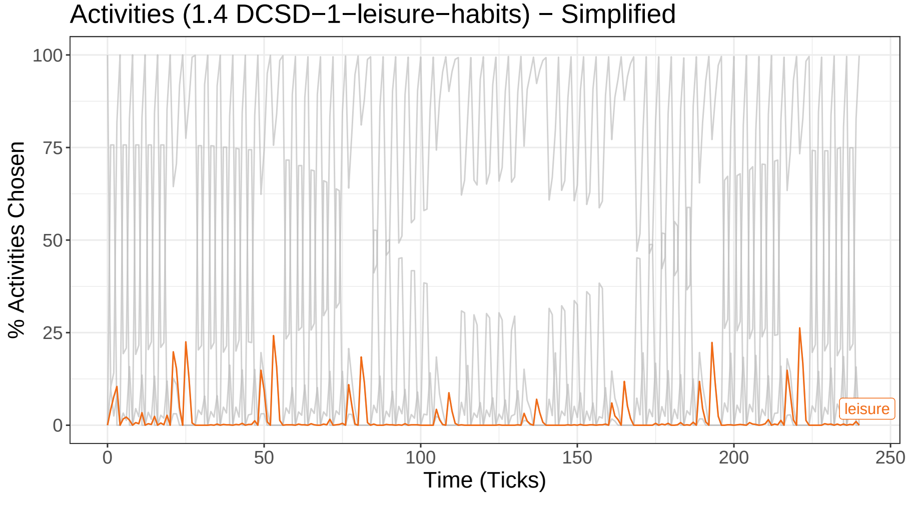


Figure 6.13: Leisure highlight in Habitual DCSD

information. The Habitual DCSD determines that at specific time points the agents have a leisure action as default. However, this leisure action cannot be selected by the DCSD since a normative check is required before an agent chooses an action that is not resting at home. Instead, the DCSD will not be used and need-based deliberation determines the action. The need-based deliberation is used a bit more during free time which causes the agent to take slightly more leisure activities. However, to have the full benefit of the leisure habits the normative layer needs to be activated. Thus for now criteria C2 can be ignored as the model is expected to pass the criteria when its normative layers are enabled.

6.1.5 Scalability Aspects of Habitual DCSD

The Original ASSOCC deliberation takes 96 seconds for full deliberation. The Habitual DCSD deliberation takes 44.1 seconds, which is more than twice as fast. This is a promising result, however, it cannot be called a break-through yet. One might wonder why it is not 10 times faster? Should it not be more than 90% of the activities chosen by habits? In the Habitual DCSD, there are 240,695 deliberations performed. Of these 99201 were deliberation using need-based deliberation. So about 59% is using the fast DCSD, while 41% is the slow need-based deliberation. Since slow deliberation is still used relatively often, the model only got about two times as fast.

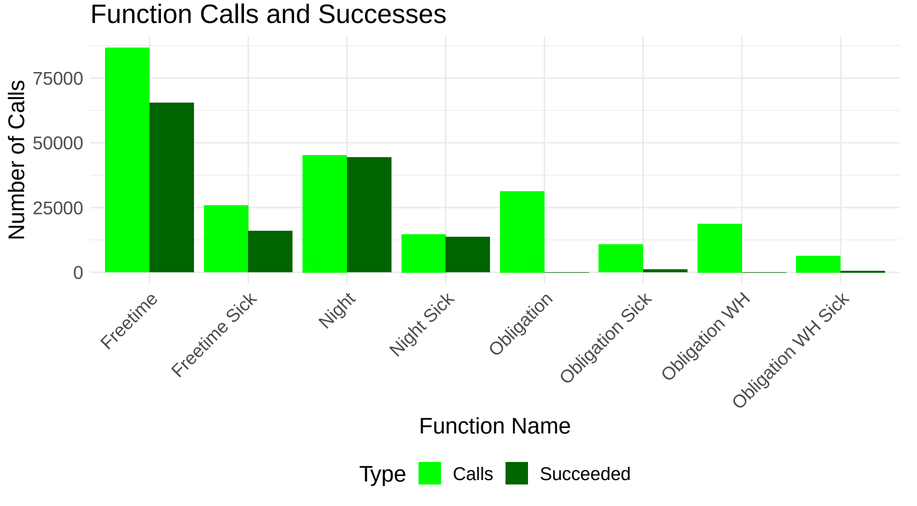


Figure 6.14: Context State Success rates Habitual DCSD

Figure 6.14 helps us understand where this bottleneck comes from in the Habitual DCSD. Based on the eight decision trees the agents can use. The figure shows the amount of deliberations performed in these states and the success rate, i.e. when the Habitual DCSD was able to decide upon an action. It turns out that in some situations the habitual DCSD is very successful, such as in the night and its quite effective during freetime. However, during obligation time, the Habitual DCSD is hardly ever successful.

For the obligation states where the agent is not sick, the Habitual DCSD never provides an action. This could work in non-pandemic simulations; however, since it is a pandemic simulation, the agents need a normative check before leaving the house. This normative check is not yet available in the Habitual DCSD, since normative decisions come in the Normative DCSD. The same goes for work from home, since the agent will first check whether working at the workplace is possible (this is already a normative check). For being sick during working hours, there are often multiple needs conflicting. This is also something the Habitual DCSD cannot deal with and thus uses need-based deliberation instead. This does not directly lead to problems in the behaviour of the agents; however, the DCSD at this stage does not yet provide many scalability benefits.

6.1.6 Summary

To summarise, purely using habits makes execution times extremely fast but is either to simple to be realistic or too complicated to give the scalability benefits. The habitual DCSD which uses a habit when possible and otherwise uses need-based deliberation shows roughly a two times speed-up while retaining most of the behavioural patterns of Original ASSOCC. The model passes almost all the criteria, as agents perform a variety of daily activities (rest at home, work/study, leisure and shopping). The agents adapt, and when agents are sick they are more likely to stay at home. The only criteria that does not pass is C2, the leisure criteria. The agents do not perform leisure activities enough however, this will be solved once the normative layer of the DCSD is activated. Before this is done, the strategic layer will be expanded in the next section.

6.2 Realism: Strategic Behaviour

The Habitual DCSD is expanded with information from the strategic layer, this will be called Strategic DCSD (see Figure 6.15). The Strategic DCSD is capable of everything the Habitual DCSD is capable of but adds comparing the need levels to determine the most salient need. This is relevant when multiple needs are salient. The section starts with an initial model of the Strategic DCSD; this works, however, could function better. The section then ends with an improved Strategic DCSD.

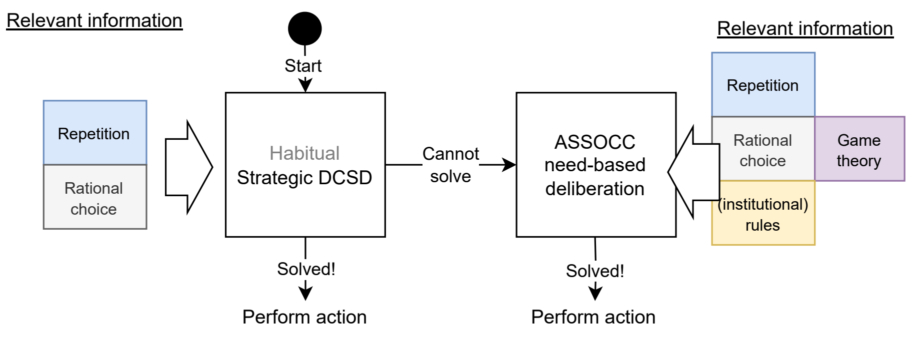


Figure 6.15: Strategic deliberation in DCSD

6.2.1 Strategic DCSD - Initial Model

The ASSOCC framework has been run with the Strategic DCSD enabled, therefore the experimental preset is *2.1 DCSD-2*. The more detailed preset settings can be found in the Appendix 8.1.3. The purpose of this experiment is to show whether the Strategic DCSD gives more scalability benefits while providing realistic enough behaviour. First the criteria will be measured as a quick general check, this is followed by a section with more detailed analysis on the behaviour of the agents.

Strategic DCSD - Criteria

Table 6.5 shows that most criteria pass. As expected and discussed in the previous section, criteria C2 does not pass due to the lack of normative information in the DCSD. The other criteria that do not pass are C11 and C12. The values in C11 and C12 indicate that children and students break quarantine too frequently. Why this exactly happens is investigated in more detail in the following section.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean $>$ 99%	100	TRUE
C2	Recently Leisure, mean $>$ 98%	82.81	FALSE
C3	Recently Ess Shopping, mean $>$ 98%	99.31	TRUE
C4	Recently Non-Ess Shopping, mean $>$ 98%	100	TRUE
C5	Not Skip Work, mean $>$ 98%	99.73	TRUE
C6	Work at Workplace when possible, 85% $<$ mean	89.32	TRUE
C7	Not Skip School, mean $>$ 95%	100	TRUE
C8	Not Skip University, mean $>$ 95%	99.91	TRUE
C9	Rest When Know Sick, mean $>$ 90%	90.67	TRUE
C10	People in Quarantine, 90% $<$ mean $<$ 100%	94.03	TRUE
C11	Children in Quarantine, 90% $<$ mean $<$ 100%	88.2	FALSE
C12	Students in Quarantine, 90% $<$ mean $<$ 100%	87.49	FALSE
C13	Workers in Quarantine, 90% $<$ mean $<$ 100%	96.84	TRUE
C14	Retireds in Quarantine, 90% $<$ mean $<$ 100%	98.74	TRUE
C15	Infection Peak Tick, 75 $<$ value $<$ 150	128	TRUE

Table 6.5: Criteria Values for 2.1 DCSD-2

Strategic DCSD - Behaviour and Population Status

Figure 6.16 shows the activities and Figure 6.17 shows the population status. Both graphs look similar to the graphs for the Habitual DCSD, albeit the peak of the infection curve happens a bit later, but this can be explained by the variety between single runs. The agents respond to believing they are infected

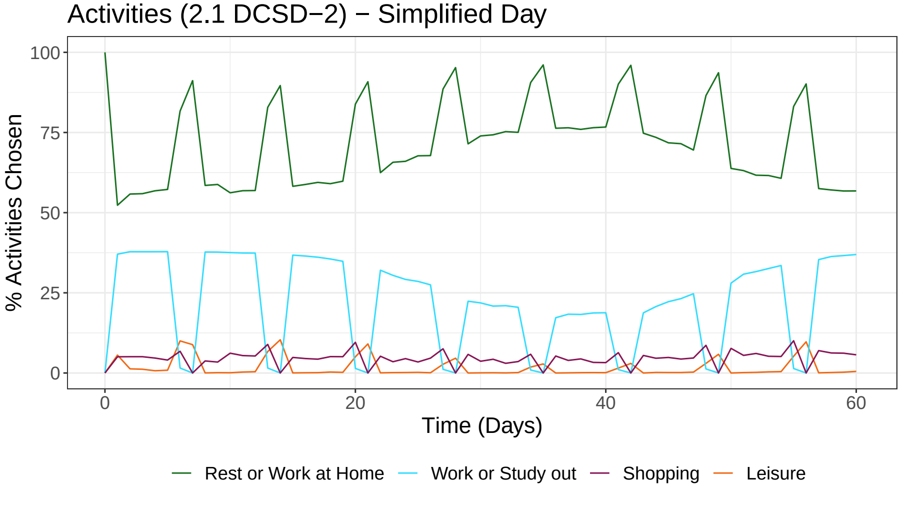


Figure 6.16: Activities for Strategic DCSD

by staying home more frequently. The agents perform their other activities regularly such as working, shopping and leisure.

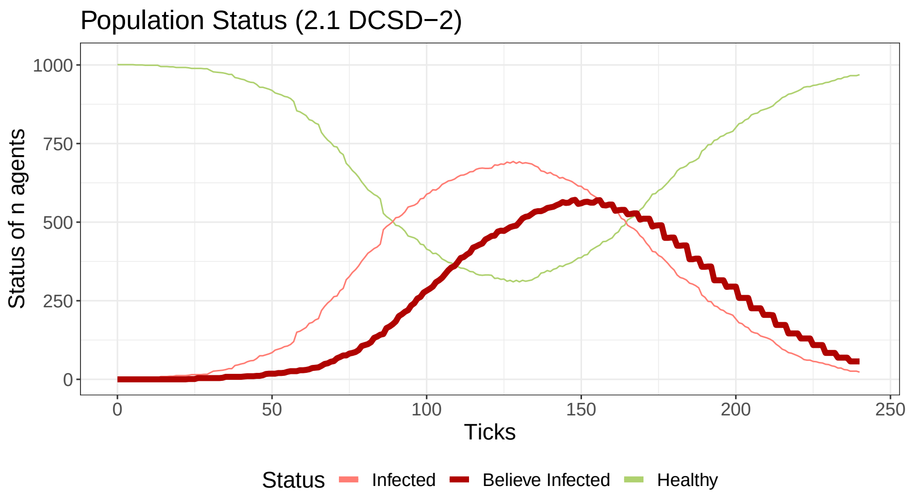


Figure 6.17: Infections for Strategic DCSD

Inspecting Working Day Activities

However something interested can be observed when comparing working-day activities between the Strategic DCSD and the Habitual DCSD. Figure 6.18 shows the work and study activities over time, the top figure is for the Habitual DCSD and the bottom for the Strategic DCSD. From the graph it becomes clear that workers and children are heavily impacted during the peak of believe infected. The behaviour of students is slightly affected. Comparing the top graph with the bottom we observe that the children are not as heavily impacted as before. Before the amount of children going to school would be as low as 3%. While in the bottom graph its about 8%, which is more than twice as much. Students are also more frequently going to the university in the Strategic DCSD model. It is not desired that children and students study when they are sick which is probably what is happening.

Figure 6.19 shows this difference in behaviour even more clearly. It shows over time the children that should be in quarantine (blue line) and the children that break quarantine (red line). In the top figure (Habitual DCSD) children hardly ever break quarantine. While in the bottom figure (Strategic DCSD) children break quarantine very frequently. The same applies to the students, as seen in Figure 6.20.

Since this is quite a change in behaviour, it is at least worth investigating why this happens. What does the strategic DCSD do different from habitual DCSD to make children more likely to skip quarantine to go to school?

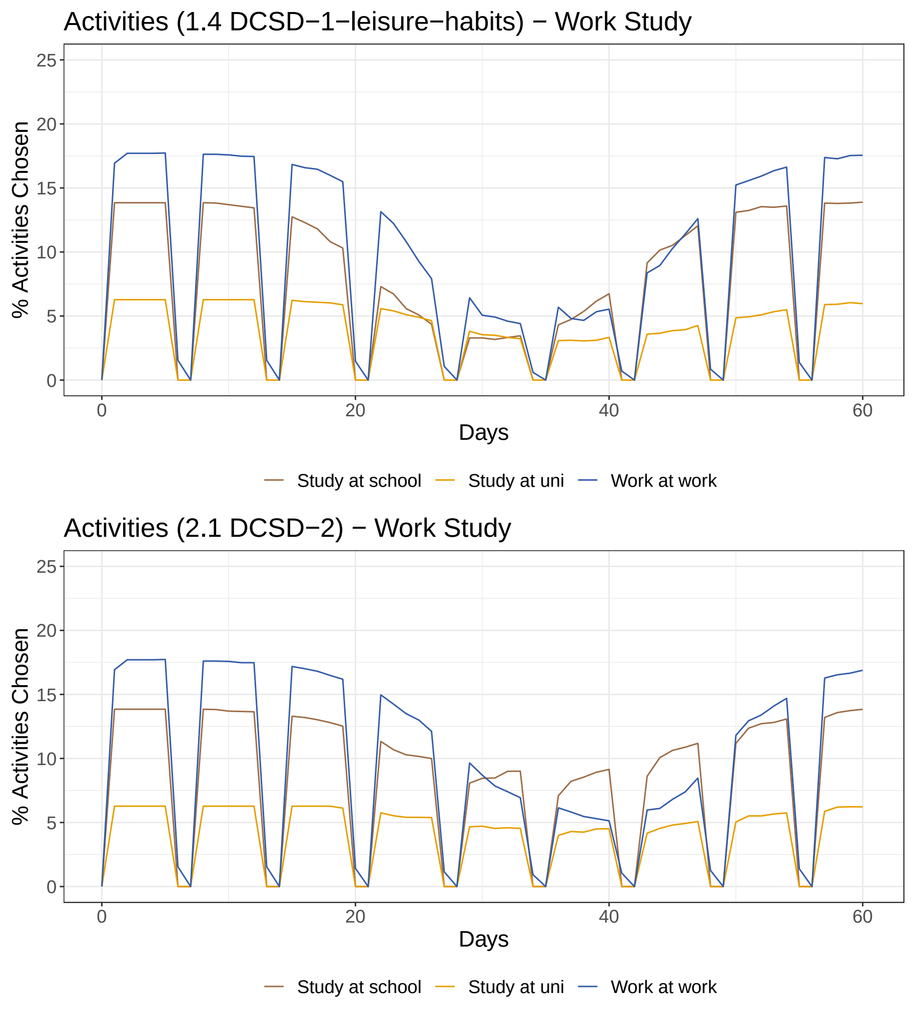
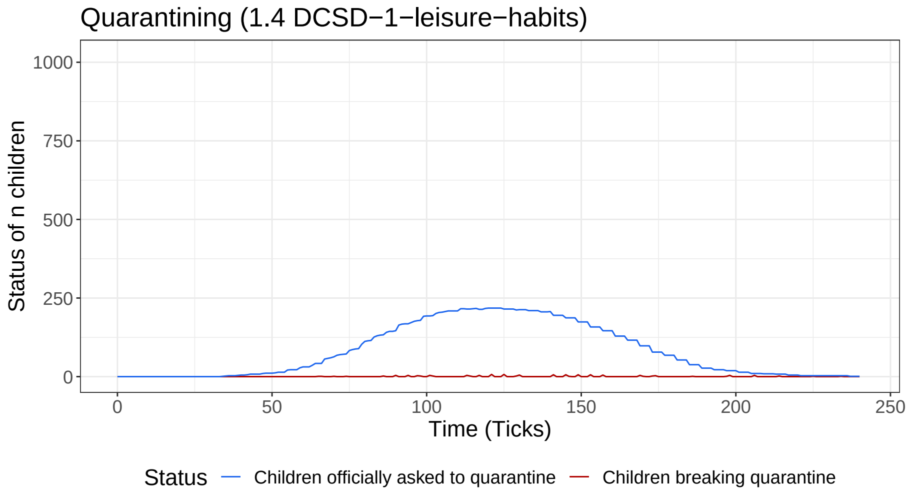


Figure 6.18: Work and study activities for Habitual and Strategic DCSD respectively

Understanding this helps to understand using the DCSD properly so that the right information is considered. The reason for this change of behaviour is that the need for autonomy in children is the lowest of all needs. This means that even when their health and risk-avoidance is low, if autonomy is the lowest, the children go to school. This is part of how the initial Strategic DCSD is conceptualised, as it will only consider the most salient need. It will usually not consider the other needs for deciding upon an action.



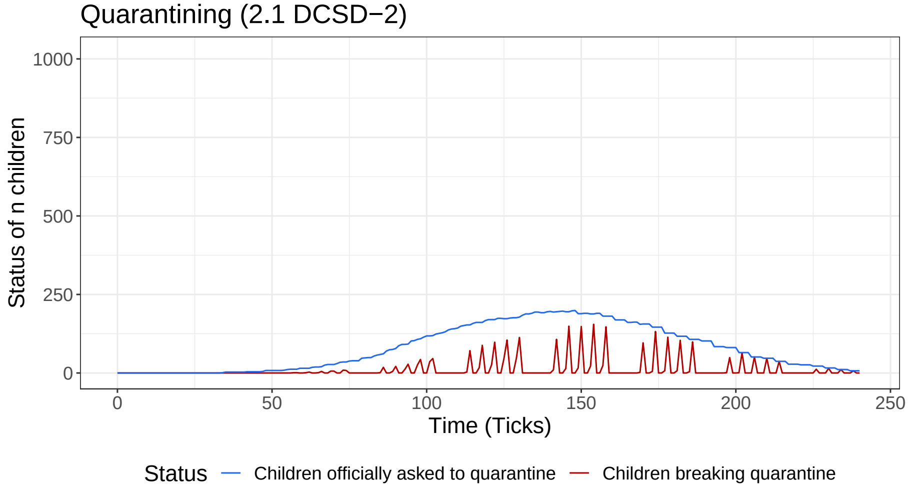


Figure 6.19: Quarantine for children for Habitual and Strategic DCSD respectively

This may be less realistic since it is not desirable children go to school when they are feeling very unhealthy (e.g. health < 0.1) or when they feel very much at risk (e.g. risk-avoidance < 0.1). Even if autonomy is lower than the other needs, if the other needs are critical the child should still stay at home. However, when the other needs are not critical then perhaps the child could go to school.

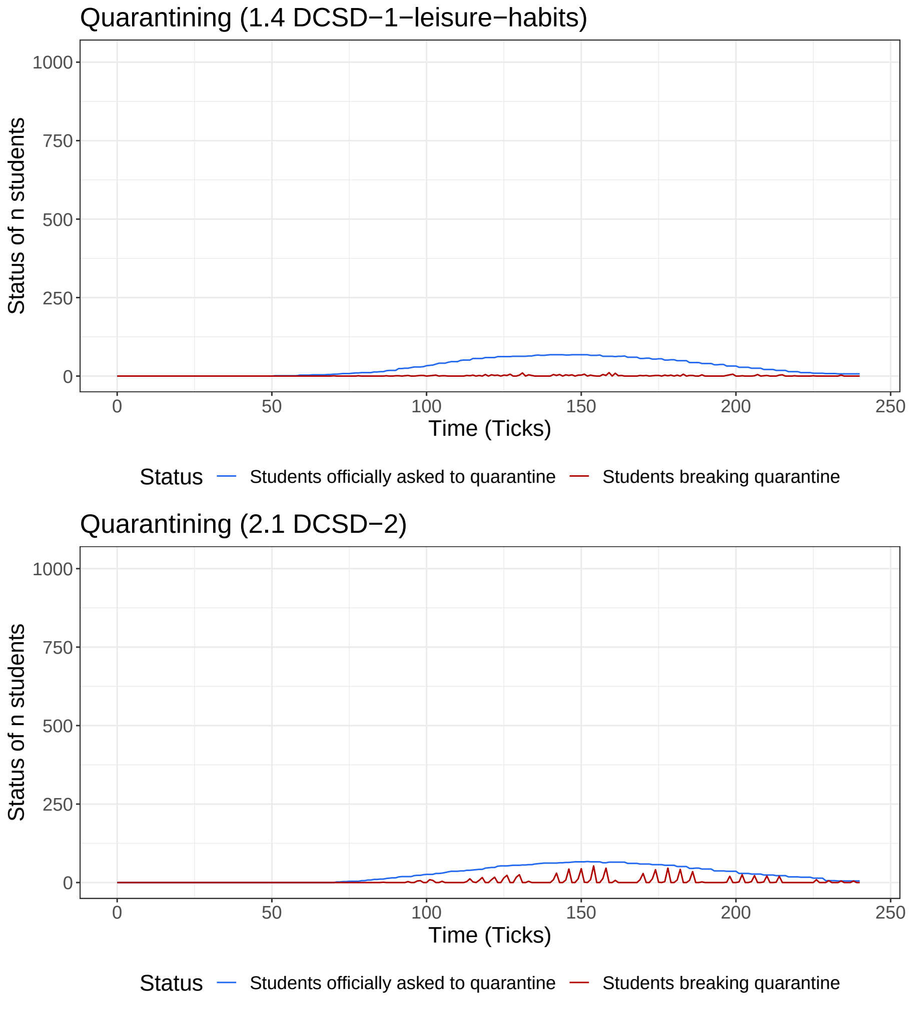


Figure 6.20: Quarantine for students for Habitual and Strategic DCSD respectively

6.2.2 Strategic DCSD - Upgraded Model

The DCSD contains a setting that solves the problem of children breaking quarantine to go to school too frequently. The upgraded Strategic DCSD is active with the preset *2.2 DCSD-2-obligation-constraint*, see for more details Appendix 8.1.3. Now the Strategic DCSD explicitly checks the health need and risk avoidance need for agents that are 1) are deliberating for when its working time, 2) sick, and 3) have a critical autonomy, financial survival or financial stability need.

Upgraded Strategic DCSD - Criteria

Considering the criteria shown in Table 6.6 it becomes clear that C11 and C12 pass. The values of C11 and C12 now fall between the required values. C2 is unchanged but this was expected and will be dealt with in the Normative DCSD. The next section will investigate the behaviour of the agents in detail.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	83.45	FALSE
C3	Recently Ess Shopping, mean > 98%	99.02	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.99	TRUE
C5	Not Skip Work, mean > 98%	99.57	TRUE
C6	Work at Workplace when possible, 85% < mean	88.21	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	99.87	TRUE
C9	Rest When Know Sick, mean > 90%	95.35	TRUE
C10	People in Quarantine, 90% < mean < 100%	98.67	TRUE
C11	Children in Quarantine, 90% < mean < 100%	99.24	TRUE
C12	Students in Quarantine, 90% < mean < 100%	97.85	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	98.43	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	98.73	TRUE
C15	Infection Peak Tick, 75 < value < 150	126	TRUE

Table 6.6: Criteria Values for 2.2 DCSD-2-obligation-constraint

Upgraded Strategic DCSD - Behaviour and Population Status

Running the simulation with this setting retains similar activity patterns (see Figure 6.21). The slight difference that can be observed is that during the peak of people knowing they are infected (see Figure 6.22), around tick 150 or day 38, the amount of agents working or studying out decreased more. In Figure 6.21 this dropped to about 12.5% while in Figure 6.16 this drops to about 18.5%. This decrease is expected since now more agents will not study when they are sick.

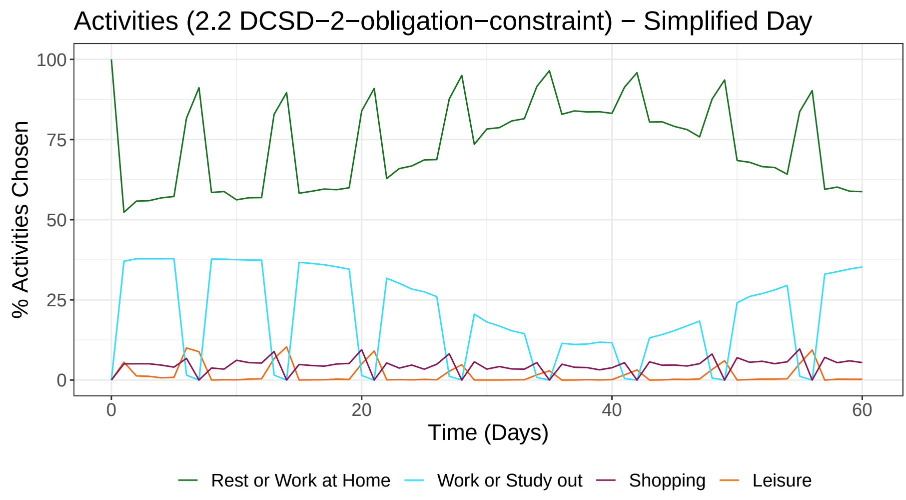


Figure 6.21: Activities for Upgraded Strategic DCSD

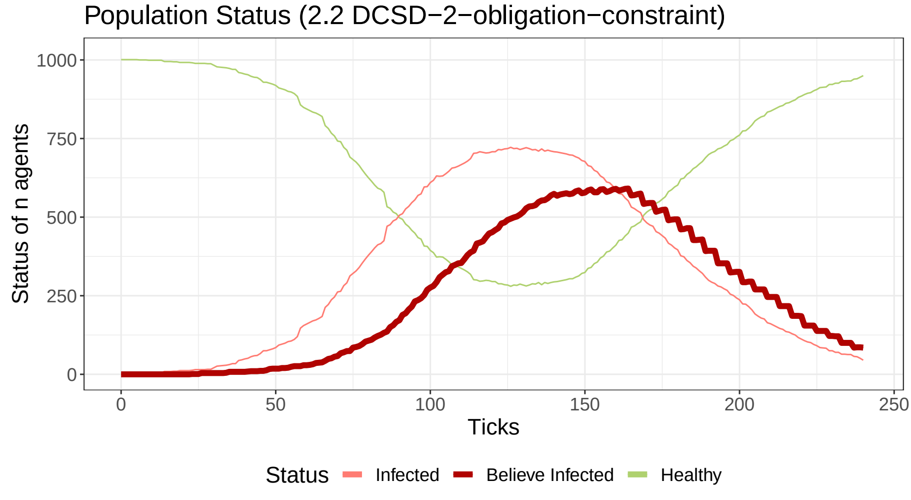


Figure 6.22: Infections for Upgraded Strategic DCSD

Figure 6.23 shows the work and study activities. On closer inspection of the behaviour we can see that children study less frequently at school, the line drops below 5%. Plotting the children in quarantine, Figure 6.24, shows that, similar to Habitual DCSD, only a few children break quarantine now.

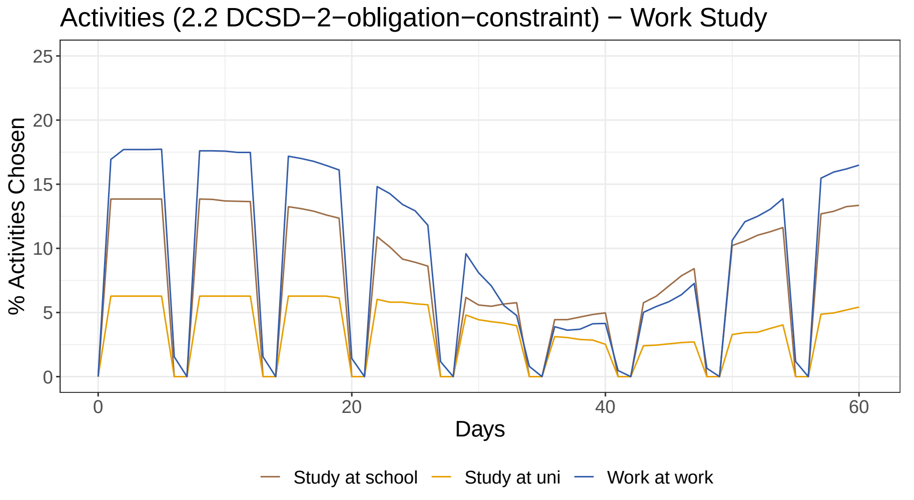


Figure 6.23: Work and study for Upgraded Strategic DCSD

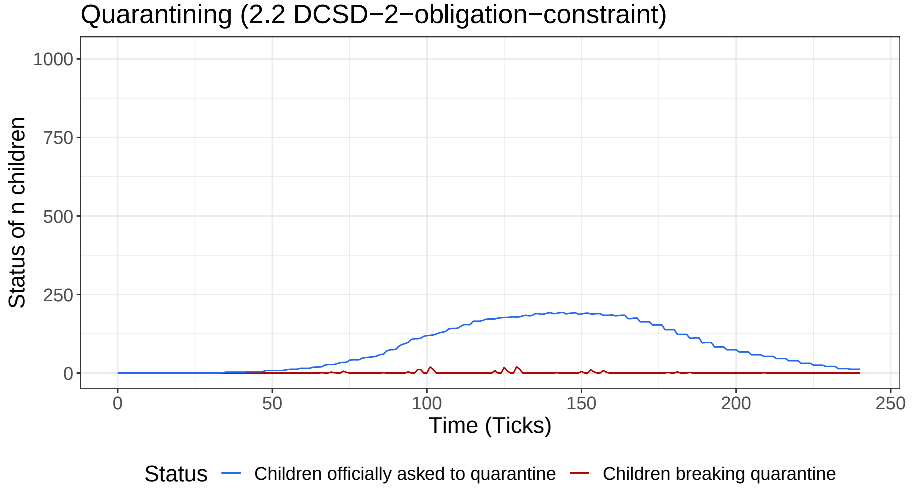
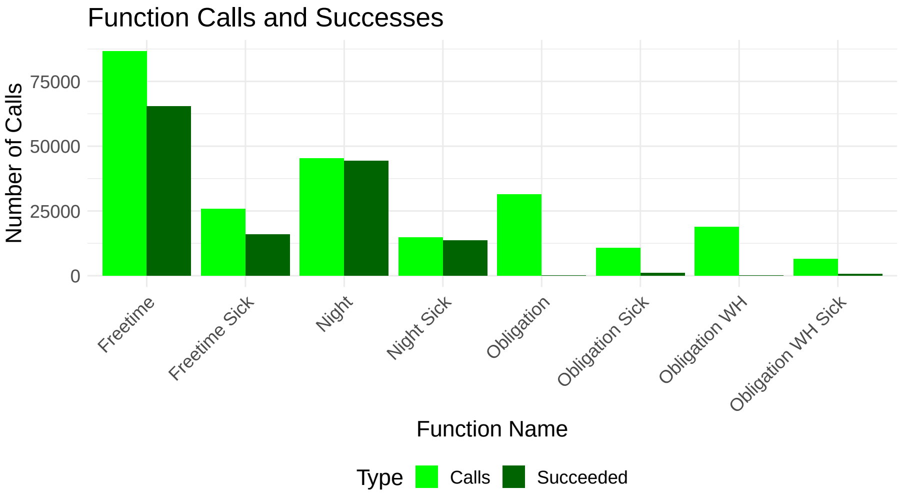


Figure 6.24: Quarantine for children for Upgraded Strategic DCSD

6.2.3 Scalability aspects of Strategic DCSD

The Habitual DCSD takes 44.1 seconds for deliberation. The Upgraded Strategic DCSD takes 30.4 seconds for deliberation. Compared with the Original ASSOCC deliberation which takes 96 seconds, the Upgraded Strategic DCSD is already 3.2 times as fast. Now we are investigating in which states the Strategic DCSD is effective.



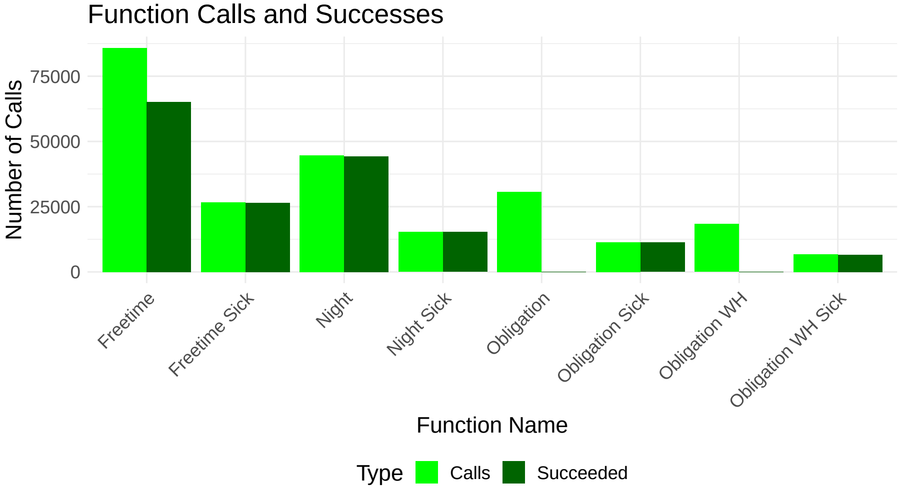


Figure 6.25: Successfulness of Decision Trees - Habitual vs Strategic DCSD

Figure 6.25 shows the successfulness of the Habitual DCSD (top) and the Strategic DCSD (bottom). The freetime, night, night sick, obligation, obligation wh are relatively similar. In freetime sick, obligation sick and obligation WH sick a large difference can be seen. This makes sense since when the agents are sick there are many needs that are salient. The Habitual DCSD cannot deal with multiple salient needs and reverts to need-based deliberation. The Strategic DCSD can select the most salient need. When the agent is sick, quite frequently the most salient needs are health or risk-avoidance which is enough information for the Strategic DCSD to select the rest at home action. However in obligation and obligation WH the Strategic DCSD is still not effective. The reason for this is that when an agent is healthy any action that involves leaving the home requires a normative check. The Strategic DCSD does not have access to the normative information and will not be able to select an action. This problem will be tackled in the next section, where normative information is included in the DCSD.

6.2.4 Summary

To summarise, adding strategic information to the DCSD makes the deliberation even faster than the Habitual DCSD. The Upgraded Strategic DCSD has a speed-up of 3.2 times compared with Original ASSOCC deliberation. The Strategic DCSD did have some initial problems, since only selecting based on the most salient need can sometimes create problems as illustrated in Section 6.2.1. This problem arose since only the most salient need (autonomy) was considered while some other salient (health and risk-avoidance) needs should have been considered as well for choosing an action. This was easily solved by adding exceptions to the deliberation model where those needs are considered in specific cases. The next section will discuss incorporating normative information into the deliberation model.

6.3 Realism: Normative Behaviour

As discussed in the previous section, the Strategic DCSD cannot select actions that are outside of the home. It requires normative information to do this. The Normative DCSD includes this information, as illustrated in Figure 6.26. Here we would like to highlight that now the deliberation system becomes less similar to Kahneman's Thinking fast and slow [46]. Kahneman proposed a two system approach of thinking fast and thinking slow. The Habitual DCSD (Figure 6.1) could be seen as a fast system, the DCSD part and a slow system the need-based deliberation. However the Normative DCSD and following DCSD's contain multiple aspects of deliberation and should not be seen as simply the fast system as Kahneman intended. Instead its sliding system that gradually moves from fast to slow deliberation passing through some intermediate states, as indicates in Figure 5.6 there are multiple layers of complexity within the DCSD.

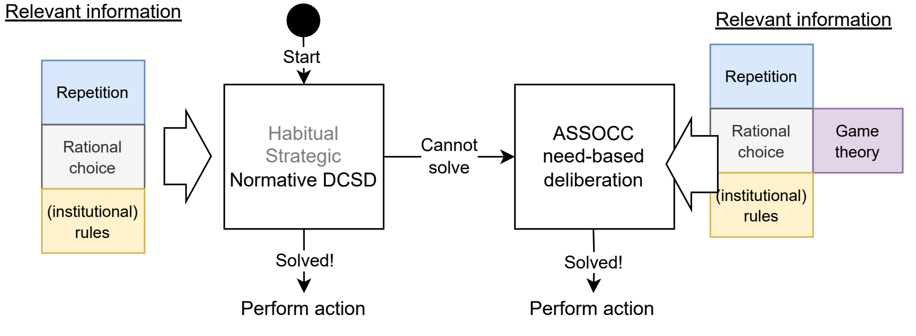


Figure 6.26: Normative Deliberation in DCSD

The Normative DCSD still retains the habitual and strategic information. This section starts with a normative component that rigidly follows the norms. It explains why doing this is not realistic enough. It follows up by showing the Normative DCSD where agents can break norms.

6.3.1 Rigid Normative DCSD

There exist different normative frameworks to formalise agent behaviour [2]. The most naïve and simplistic way could be to let the agents always follow the norms that apply. For example, if the agent needs to stay in quarantine the agent never leaves home, otherwise the agent is free to pick any other action. This type of rigid normative model is enabled when the preset is *3.1 DCSD-3-rigid-norms* (see Appendix 8.1.3). The Rigid Normative DCSD is the Normative DCSD, however, agents will always rest at home or get treatment at the hospital when quarantining applies to them. The runtime is about 8.8 quicker than Original ASSOCC, however, is it realistic enough? In the upcoming subsection the criteria will be measured, in the section that followed the details will be analysed.

Rigid Normative DCSD - Criteria

Table 6.7 shows that some criteria pass. All the criteria related to quarantining (C10 to C14) however, do not pass. They do not pass because agents are breaking quarantine too often, rather they do not pass because the agents never break quarantine. On the positive note criteria C2 passes since leisure can be performed frequently enough by agents. The next section will go into more details on the effect on the simulation as a whole.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.7	TRUE
C3	Recently Ess Shopping, mean > 98%	99.14	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.98	TRUE
C5	Not Skip Work, mean > 98%	99.16	TRUE
C6	Work at Workplace when possible, 85% < mean	93.69	TRUE
C7	Not Skip School, mean > 95%	98.82	TRUE
C8	Not Skip University, mean > 95%	98.89	TRUE
C9	Rest When Know Sick, mean > 90%	100	TRUE
C10	People in Quarantine, 90% < mean < 100%	100	FALSE
C11	Children in Quarantine, 90% < mean < 100%	100	FALSE
C12	Students in Quarantine, 90% < mean < 100%	100	FALSE
C13	Workers in Quarantine, 90% < mean < 100%	100	FALSE
C14	Retireds in Quarantine, 90% < mean < 100%	100	FALSE
C15	Infection Peak Tick, 75 < value < 150	118	TRUE

Table 6.7: Criteria Values for 3.1 DCSD-3-rigid-norms

Rigid Normative DCSD - Behaviour and Population Status

The Rigid Normative DCSD seems to give realistic results initially. Figure 6.27 for the activities and Figure 6.28 for the infections. It is difficult to see from these activity plots alone that agents never break quarantine. The agents stay more frequently home when more agents believe they are infected. The agents do their daily activities as they work or study, they go shopping and perform leisure activities. They even perform more leisure activities than in the Strategic DCSD. This is due to the added normative layer to the DCSD which allows the leisure habits to take full effect. All in all the results look relatively realistic.

The quarantine graph shows, as expected, that agents never break quarantine (Figure 6.29). This does not seem like a big problem directly, as most agent should follow the rules anyway. However, 100% of the agents following the rules does not reflect society well as there are always at least some people

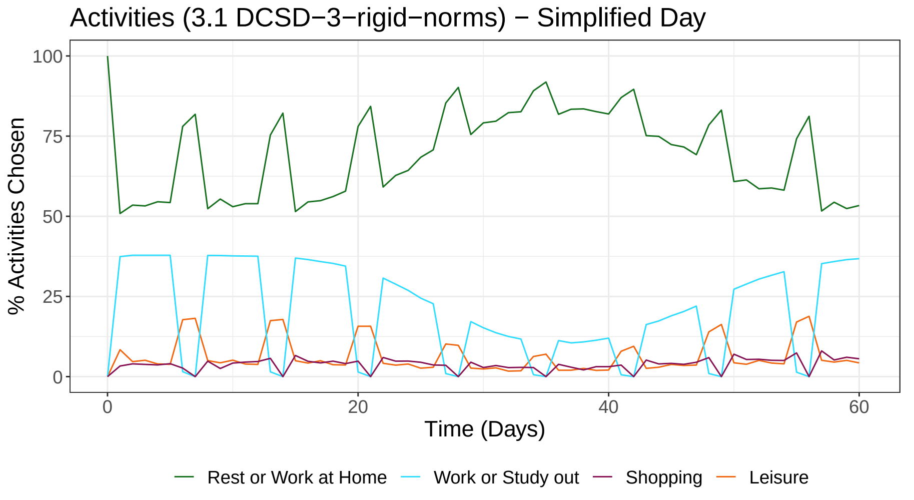


Figure 6.27: Activities for Rigid Normative DCSD

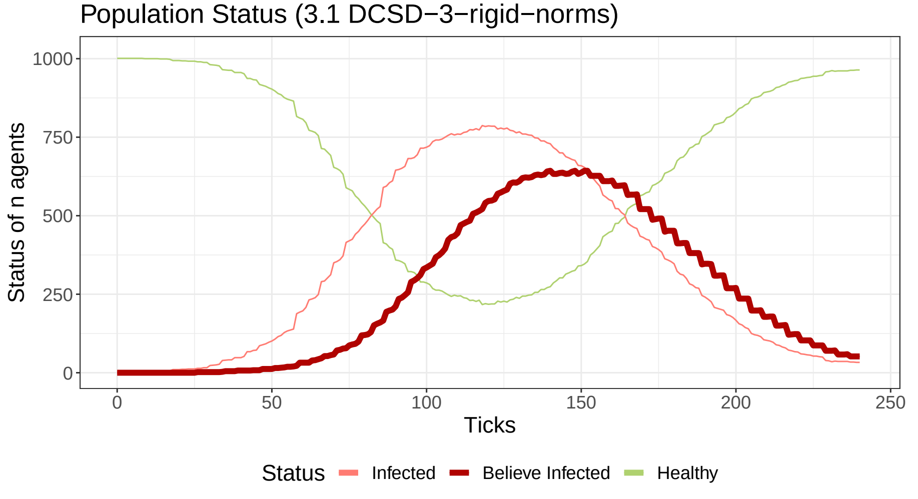


Figure 6.28: Population Status for Rigid Normative DCSD

breaking the rules. Why rigid habitual behaviour is a problem is not so clear from this experiment, but will become more clear in the following section where we simulate global lockdown.

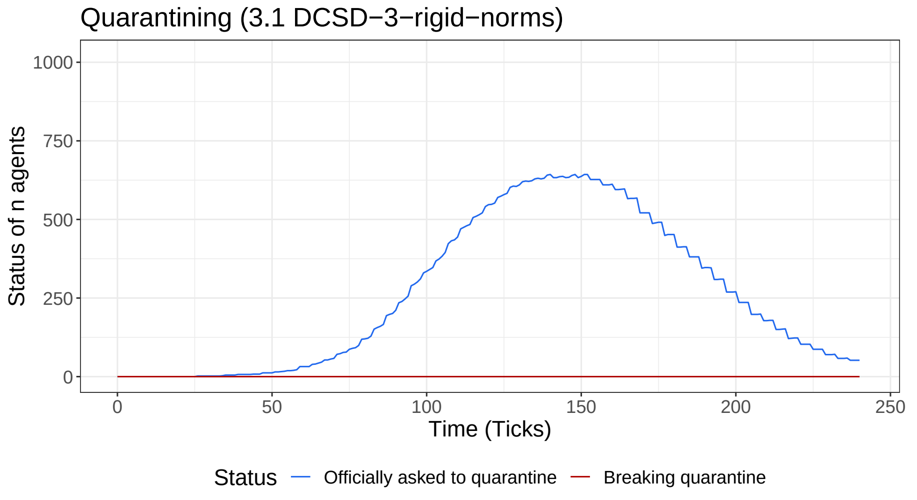


Figure 6.29: Quarantine for Rigid Normative DCSD

6.3.2 Rigid Normative DCSD - Global Lockdown

This section analyses the Rigid Normative DCSD enabled during a global lockdown, the preset is *3.2 DCSD-3-rigid-norms-lockdown*. This enables global lockdown and lengthens the number of simulated ticks from 240 to 480. This allows for analysing the global lockdown period and the post global lockdown period. The more detailed preset settings can be found in the Appendix 8.1.3. Again, the criteria are first analysed, which is followed up by the detailed behavioural analysis.

Rigid Normative DCSD Global Lockdown - Criteria

Table 6.8 shows the criteria. Again the criteria C10 to C14 do not pass as is expected. In fact, in this case, criteria C15 also does not pass. The peak of infections occurs very early, after 54 ticks or about 13 days. The next section will explain why this happens.

Rigid Normative DCSD Global Lockdown - Behaviour and Population Status

Figure 6.30) shows the actions of the agents. It should be very clear that from about day nine, when the global lockdown starts, the agents always stay at home. The global lockdown lasts roughly 60 days after which agents start with shopping a lot and afterwards continue their normal behaviour. However, this lockdown period is not realistic, as 100% of the agents stay home. This lack of realism is reflected in the infection curve, Figure 6.31, where during the global

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.78	TRUE
C3	Recently Ess Shopping, mean > 98%	98.13	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.9	TRUE
C5	Not Skip Work, mean > 98%	99.18	TRUE
C6	Work at Workplace when possible, 85% < mean	98.07	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	99.9	TRUE
C9	Rest When Know Sick, mean > 90%	100	TRUE
C10	People in Quarantine, 80% < mean < 90%	100	FALSE
C11	Children in Quarantine, 80% < mean < 98%	100	FALSE
C12	Students in Quarantine, 80% < mean < 98%	100	FALSE
C13	Workers in Quarantine, 80% < mean < 98%	100	FALSE
C14	Retireds in Quarantine, 80% < mean < 98%	100	FALSE
C15	Infection Peak Tick, 250 < value < 400	54	FALSE

Table 6.8: Criteria Values for 3.2 DCSD-3-rigid-norms-lockdown

lockdown the number of infected agents goes to zero. This does not match reality as global lockdown has hardly ever succeeded in eradicating the virus completely due to the people breaking the quarantining rules.

Figure 6.30: Activities for Rigid Normative DCSD with lockdown

Figure 6.32 shows clearly that all agents should be in quarantine in the lockdown period (the blue line). No agent breaks out of lockdown. If one wants to simulate more realistically, the deliberation should contain not only norm following but also norm breaking capabilities.

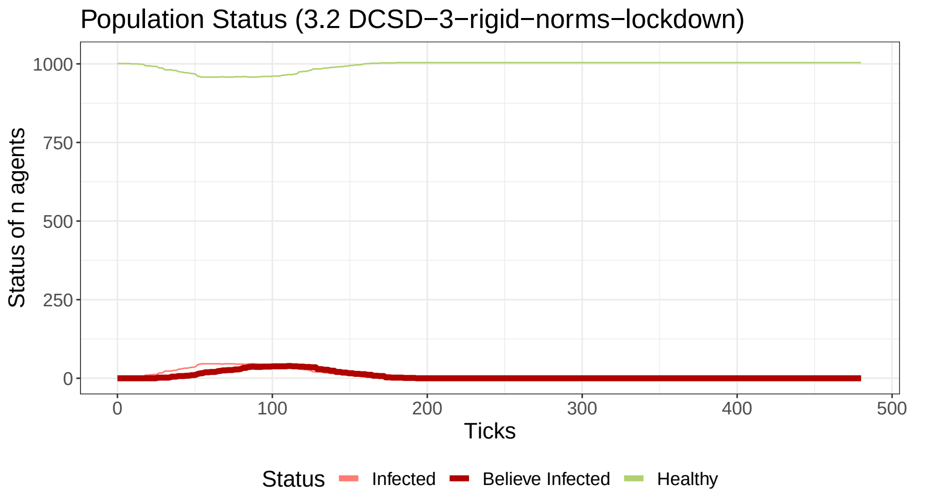


Figure 6.31: Infection graphs for Rigid Normative DCSD with lockdown

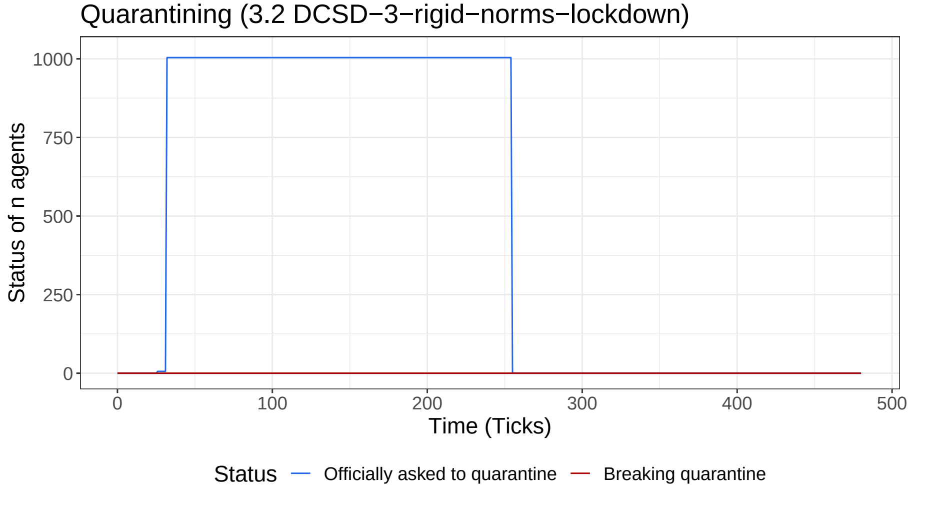


Figure 6.32: Quarantine for Rigid Normative DCSD with lockdown

6.3.3 Normative DCSD - Breaking Norms

As indicated in the previous section, adding rigid norms is perhaps easy to do but comes at the cost of realistic behaviour. The agents always follow the norms which does not make it possible to realistically simulate a global lockdown situation. The DCSD should thus have breakable norms. The preset *3.3 DCSD-3* enables the actual Normative DCSD with breakable norms, see Appendix 8.1.3 for more detailed settings. The next section will show whether this DCSD version passes the criteria, the section afterwards analyses the behaviour in more detail.

Normative DCSD - Criteria

Table 6.9 shows the criteria, all the criteria pass! This is in contrast with the Rigid Normative DCSD where the quarantine criteria did not pass. In this case for criteria C10 to C14, the agents sometimes break quarantine but not too often. The next section will analyse if the behaviour is still realistic.

Cr	**Criteria Description**	**Value**	**Pass**
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.58	TRUE
C3	Recently Ess Shopping, mean > 98%	99.13	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	99.6	TRUE
C6	Work at Workplace when possible, 85% < mean	94.49	TRUE
C7	Not Skip School, mean > 95%	98.66	TRUE
C8	Not Skip University, mean > 95%	99.22	TRUE
C9	Rest When Know Sick, mean > 90%	95.46	TRUE
C10	People in Quarantine, 90% < mean < 100%	98.06	TRUE
C11	Children in Quarantine, 90% < mean < 100%	98.75	TRUE
C12	Students in Quarantine, 90% < mean < 100%	97.7	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	97.05	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	98.71	TRUE
C15	Infection Peak Tick, 75 < value < 150	122	TRUE

Table 6.9: Criteria Values for 3.3 DCSD-3

Normative DCSD - Behaviour and Population Status

Figure 6.33 shows the activities. The behaviour is actually relatively comparable with the Rigid Normative DCSD. The agents perform all types of daily life activities, they work and study, they do shopping and perform leisure activities. The agents also respond to the pandemic as they stay at home more when the amount of agents that believe they are infected is at its heighest point, see Figure 6.34.

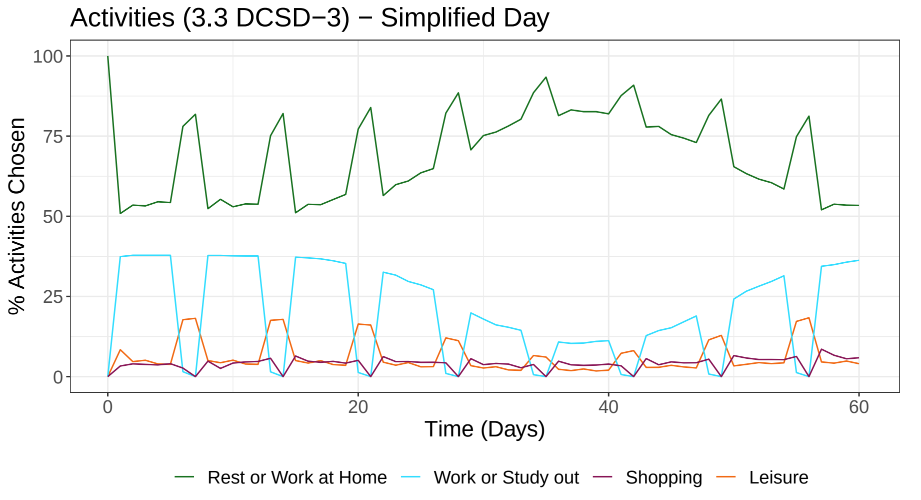


Figure 6.33: Activities for Normative DCSD

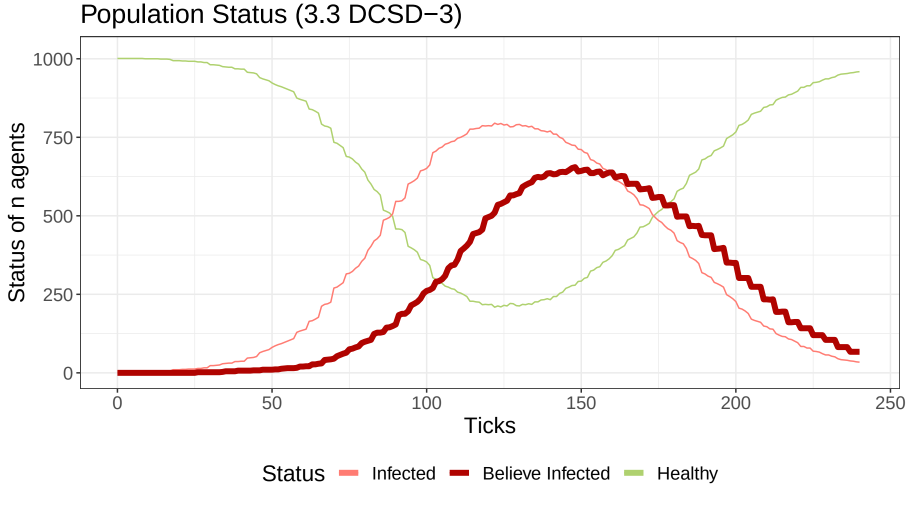


Figure 6.34: Infection graphs Normative DCSD

When consider the quarantining behaviour, Figure 6.35, it becomes clear that the agents can actually break quarantine. The red spikes at the bottom of the graph indicate the number of agents breaking quarantine. Breaking quarantine becomes slightly more frequent between tick 100 and tick 200 as between these ticks more agents in total are asked to stay in quarantine.

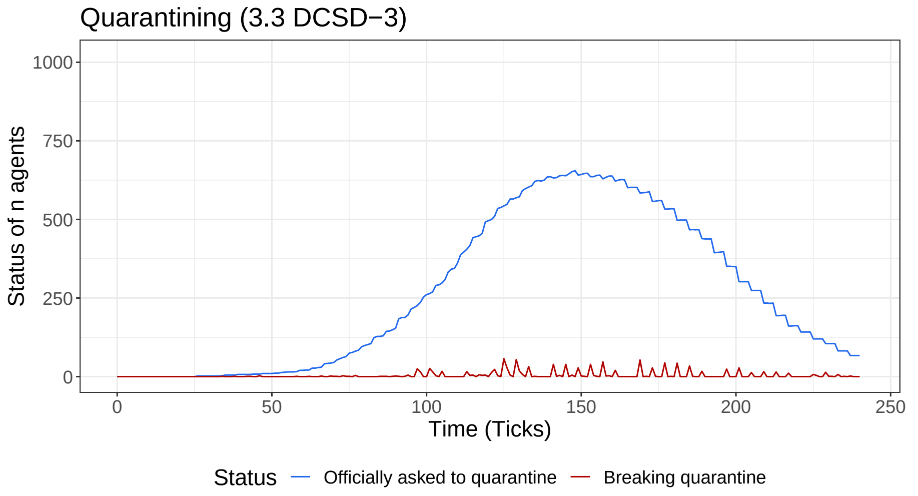


Figure 6.35: Quarantine for Normative DCSD

6.3.4 Normative DCSD - Global lockdown

This section describes the Normative DCSD with global lockdown enabled experiment, i.e. the experiment with preset *3.4 DCSD-3-lockdown*. This enables the global lockdown and changes the length of the simulated time to 480 ticks, see Appendix 8.1.3 for more detailed settings.

Normative DCSD Global Lockdown - Criteria

Table 6.10 shows the criteria for the Normative DCSD with global lockdown. All of the criteria pass again. The next section will analyse the model in more detail.

Normative DCSD Global Lockdown - Behaviour and Population Status

Figure 6.36 shows the activities. The agents perform their regular behaviour the first few days, however at around day nine the global lockdown start and the agents change their behaviour. Rather than staying at home 100% the time, some agents still break the quarantine and perform other activities. This pattern of lower frequency of activities outside of the house continues until the end of the global lockdown, after which the activities performed ramps up. Relatively quickly after the end of the global lockdown there is a dip in activities outside of the house due to the peak of infections (see Figure 6.37). It should be noted that agents rarely perform leisure activities during lockdown. However, this will be discussed in more detail in the Section 6.5 where the

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.8	TRUE
C3	Recently Ess Shopping, mean > 98%	99.11	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.98	TRUE
C5	Not Skip Work, mean > 98%	99.85	TRUE
C6	Work at Workplace when possible, 85% < mean	96.17	TRUE
C7	Not Skip School, mean > 95%	99.61	TRUE
C8	Not Skip University, mean > 95%	99.56	TRUE
C9	Rest When Know Sick, mean > 90%	94.69	TRUE
C10	People in Quarantine, 80% < mean < 90%	91.54	TRUE
C11	Children in Quarantine, 80% < mean < 98%	87.51	TRUE
C12	Students in Quarantine, 80% < mean < 98%	83.1	TRUE
C13	Workers in Quarantine, 80% < mean < 98%	94.94	TRUE
C14	Retireds in Quarantine, 80% < mean < 98%	96.02	TRUE
C15	Infection Peak Tick, 250 < value < 400	336	TRUE

Table 6.10: Criteria Values for 3.4 DCSD-3-lockdown

DCSD ASSOCC is compared with Original ASSOCC.

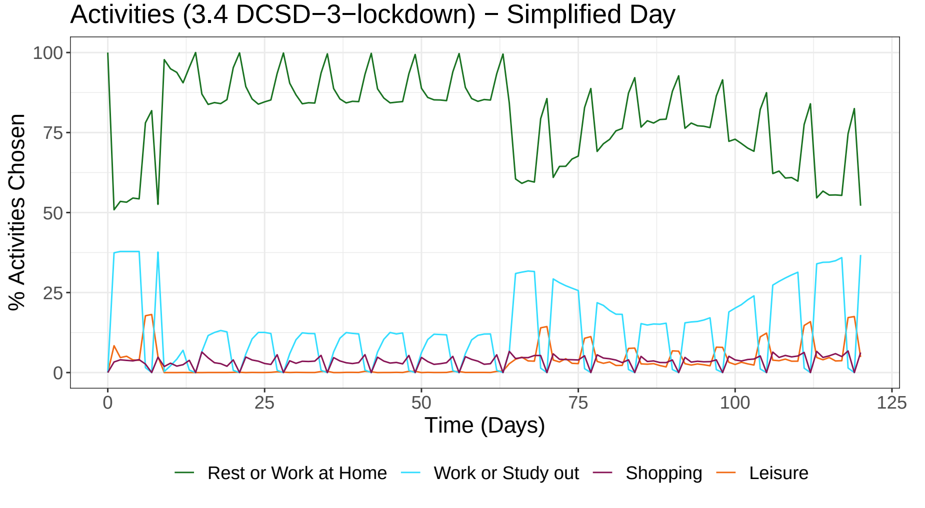


Figure 6.36: Activities for Normative DCSD with lockdown

Figure 6.38 shows when the agents quarantine. From this figure it becomes clear that during the global lockdown actually relatively a lot of agents break the quarantine. They probably are not sick or do not know they are sick and prefer to go out of the house to work and study, shop or do leisure. Since the agents break quarantine the virus manages the spread, albeit slowly. At the

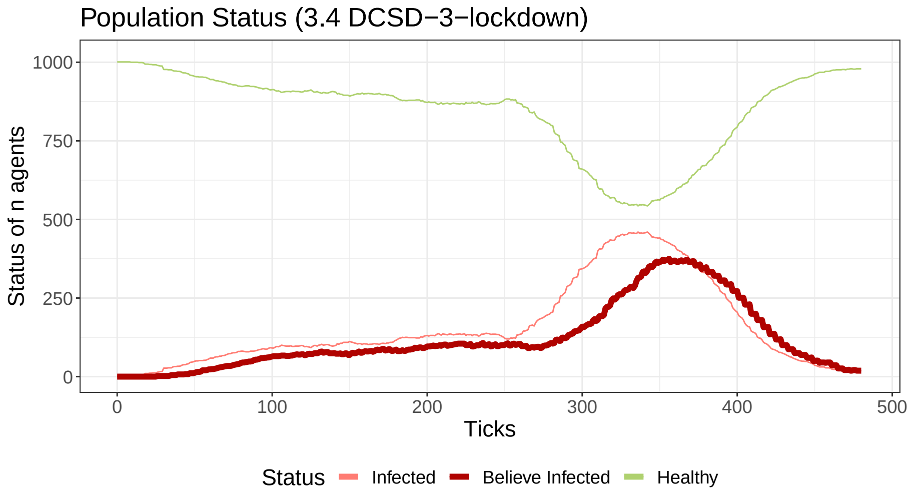


Figure 6.37: Infection graphs Normative DCSD with lockdown

end of the global lockdown there are still enough agents infected to start a higher peak of infections. This patterns, as seen in Figure 6.37 is more similar to what happened in the real world.

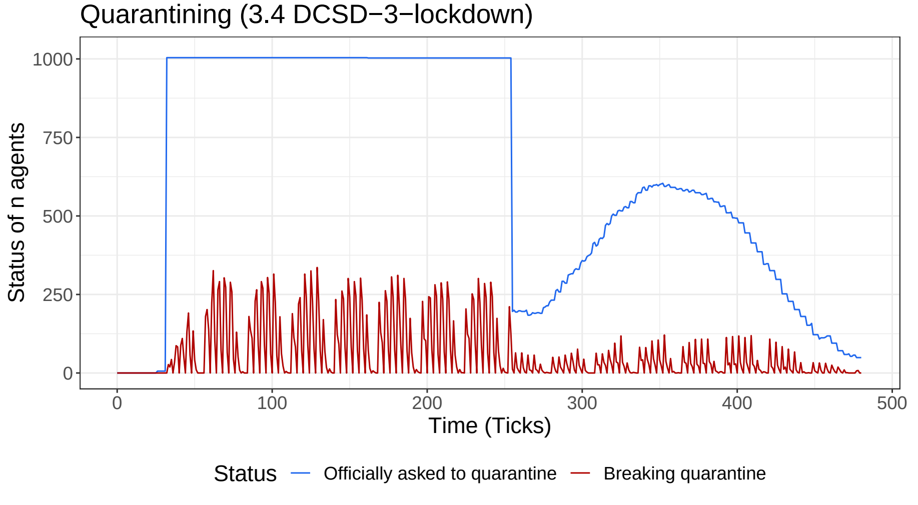


Figure 6.38: Quarantine for Normative DCSD with lockdown

6.3.5 Scalability Aspects of Normative DCSD

The Strategic DCSD takes 30.4 seconds for deliberating. The Normative DCSD takes 10.6 seconds, which is about 9.1 times quicker than the Original ASSOCC deliberation. Now we are actually getting to quite a large speed-up. As discussed in Section 6.2, using the Normative DCSD should help especially well in the obligation states. This is proven by Figure 6.39 which shows that the Normative DCSD gives high successes in also the obligation and obligation WH states. While for the Strategic DCSD these states were highly unsuccessful.

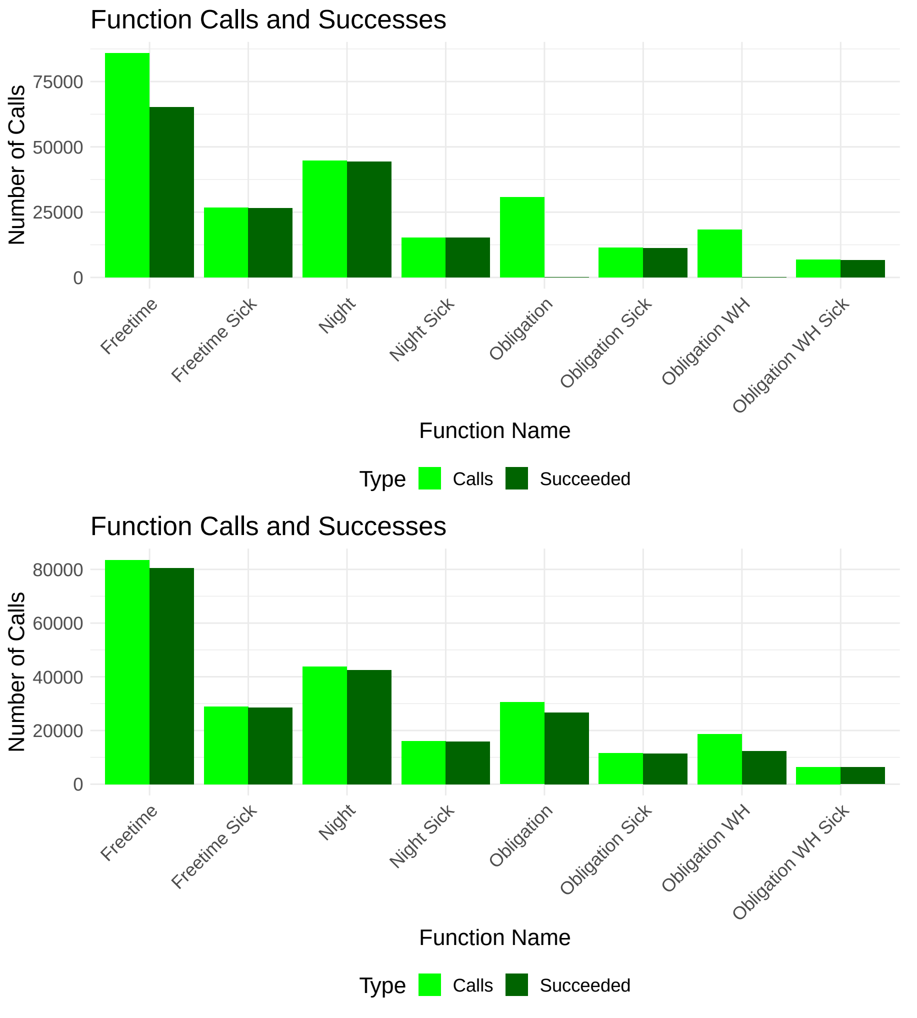


Figure 6.39: Successfulness of Decision Trees - Strategic vs Normative DCSD

The freetime state is more successful when compared with the Strategic DCSD. It may look like only small changes but remember that if we move from 90% coverage by the DCSD to 95%. The deliberation time is already halved. This is exactly why going further and trying to make the DCSD even more successful can be beneficial. The next section will discuss the benefits of adding the social behaviour.

6.3.6 Summary

To summarise, using the Normative DCSD the deliberation becomes about 9.1 times quicker than the Original ASSOCC deliberation. This is a good improvement. When conceptualising and implementing norms into a deliberation system one should be careful. If the norms are not breakable then there can be scalability benefits, however at the cost of realism. This was illustrated by the agents not breaking the global lockdown at all, which prematurely ended the pandemic, which was never seen in reality. Having breakable norms made the deliberation quicker without loosing the realism.

6.4 Realism: Social Behaviour

The ASSOCC agents use four types of information from the CAFCA matrix as illustrated in Figure 5.1. The social information is the last relevant decision information to be included in the DCSD. The Social DCSD contains all information of Normative DCSD and includes the social information. Figure 6.40 shows the abstract representation of the Social DCSD. This section will analyse the criteria and behaviour of the agents when using the Social DCSD. It ends with a quick check on the scalability performance of the Social DCSD.

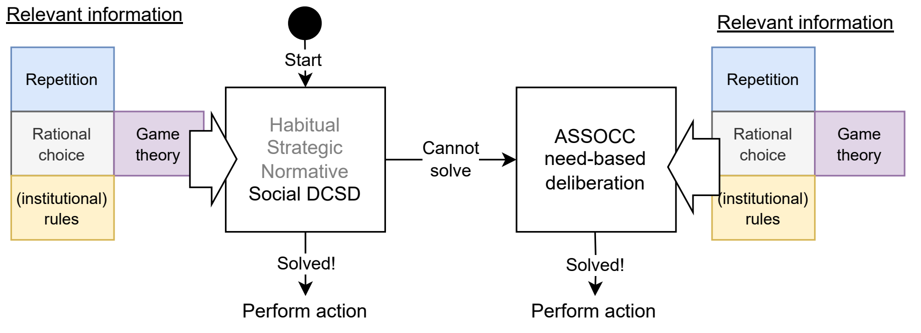


Figure 6.40: Social Deliberation in DCSD

6.4.1 Social DCSD

The Social DCSD is enabled through the preset *4.1 DCSD-4*. The Social DCSD is capable of using all deliberation information. Basically when the conformity need is the most salient, the Social DCSD uses the action of the social network as information to deliberate. The more detailed preset settings can be found in the Appendix 8.1.3. The next section will show the criteria which is followed by a more in depth analysis of the behaviour.

DCSD Social - Criteria

Figure 6.11 shows the criteria for this experiment. All the criteria pass, so not much new can be said. The next section will analyse the role of conformity within the deliberation.

DCSD Social - Behaviour and Population Status

When considering the general behaviour in the activity graph (Figure 6.41), the population status graph (Figure 6.42) and the quarantine graph (Figure 6.43), no notable differences can be seen compared to the Normative DCSD behaviour

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.81	TRUE
C3	Recently Ess Shopping, mean > 98%	99.19	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	99.29	TRUE
C6	Work at Workplace when possible, 85% < mean	93.71	TRUE
C7	Not Skip School, mean > 95%	98.86	TRUE
C8	Not Skip University, mean > 95%	98.89	TRUE
C9	Rest When Know Sick, mean > 90%	95.28	TRUE
C10	People in Quarantine, 90% < mean < 100%	98.17	TRUE
C11	Children in Quarantine, 90% < mean < 100%	98.74	TRUE
C12	Students in Quarantine, 90% < mean < 100%	97.42	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	98.06	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	98.52	TRUE
C15	Infection Peak Tick, 75 < value < 150	126	TRUE

Table 6.11: Criteria Values for 4.1 DCSD-4

output. Thus both the criteria and the plots look realistic enough. Since the behaviour did not change, we can immediately analyse the scalability aspects.

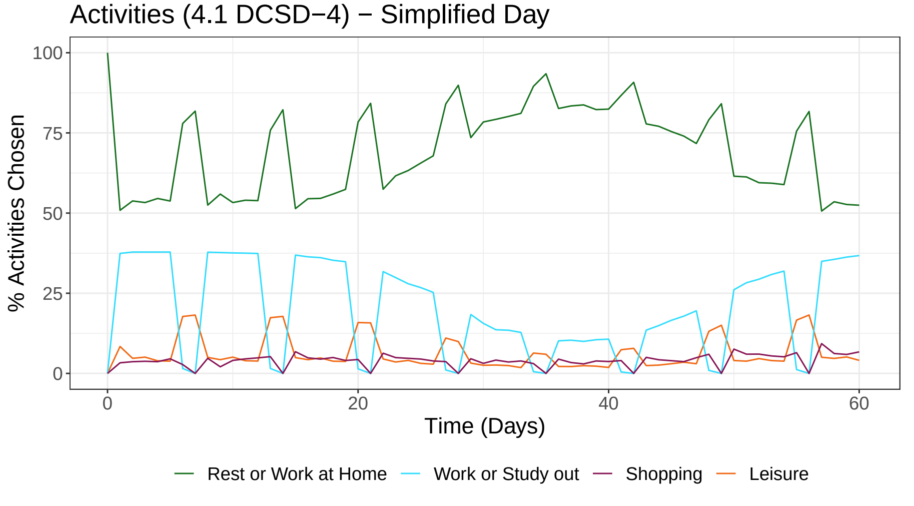


Figure 6.41: Activities for Social DCSD

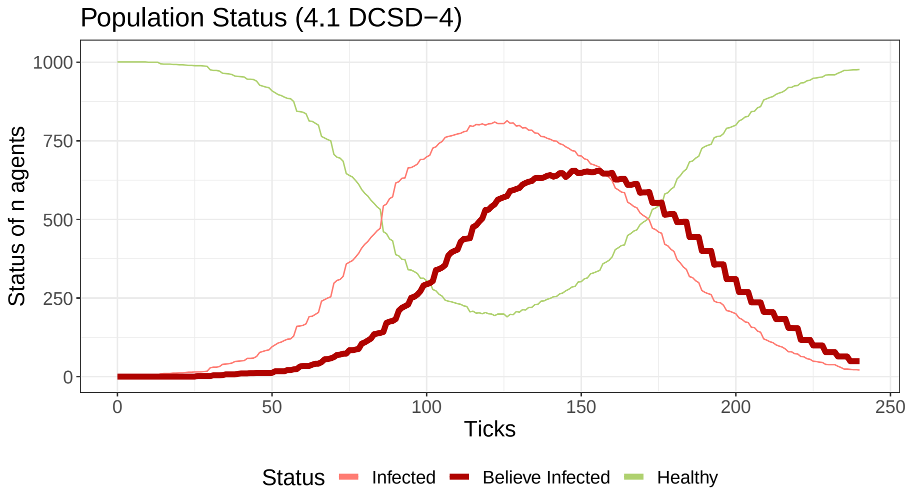


Figure 6.42: Infection graphs Social DCSD

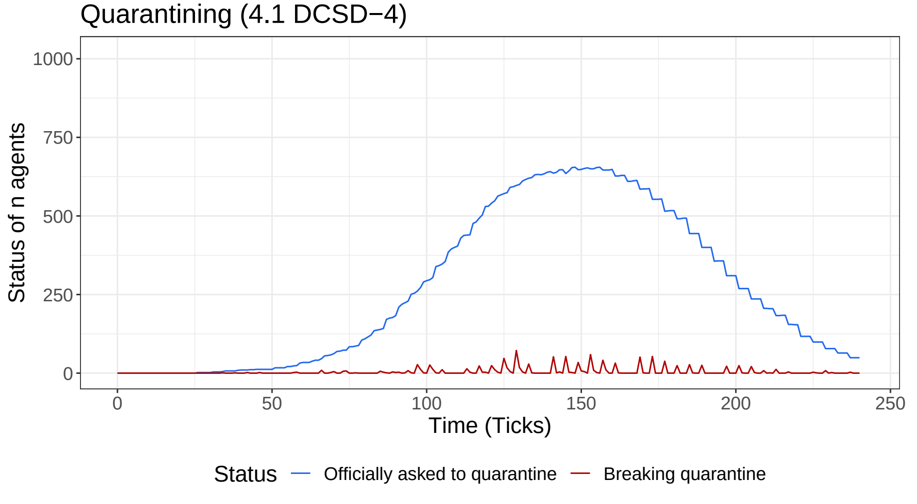
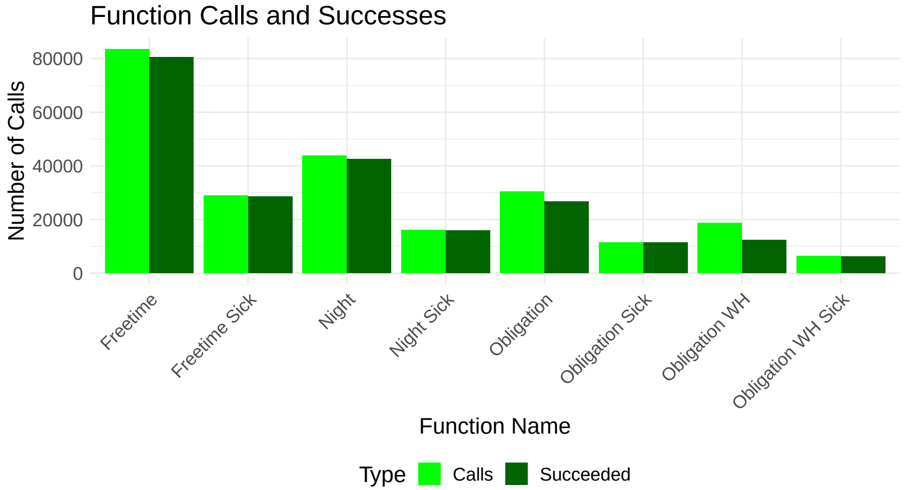


Figure 6.43: Quarantine for Social DCSD

6.4.2 Scalability Aspects of Social DCSD

The Normative DCSD takes 10.6 seconds while the Social DCSD 9.1 seconds. This is a smaller improvement than the previous section showed but still it is a notable improvement. While the Normative DCSD was 9.1 times as quick as the Original ASSOCC deliberation, the Social DCSD is slightly quicker with 10.5 times as quick. One might have expected more improvement from adding the social behaviour, but keep in mind that this behaviour is only relevant when the need of conformity is salient. Figure 6.44 compares the effect on the number of successes in the decision trees between the Normative DCSD and the Social DCSD. It shows that the freetime and night decision trees are successful slightly more often. Other differences might be to small to note.



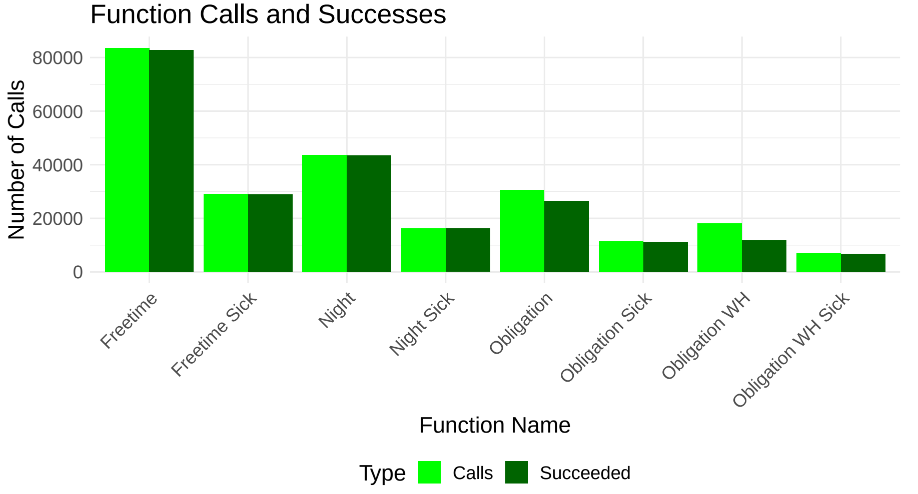


Figure 6.44: Successfulness of Decision Trees - Normative vs Social DCSD

It might be more insightful to show how often decisions are made based on the conformity need. Figure 6.45 shows the deliberation types over time. The agents always use the minimal context to determine in which state they are. This is not always enough and then other methods have to be used like the most salient need (green) or normative deliberation (orange). The details of this graph do not have to be fully understood, just remember that many different deliberation types are used very frequently.

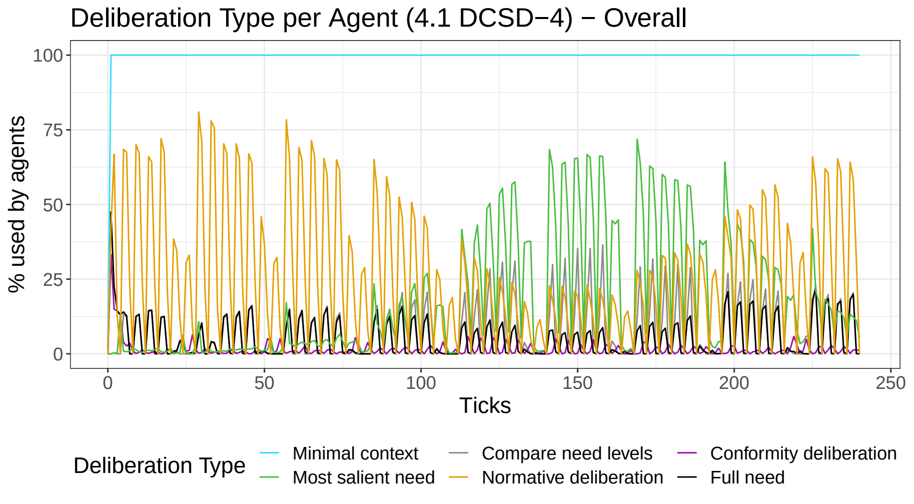


Figure 6.45: Deliberation Type for Social DCSD

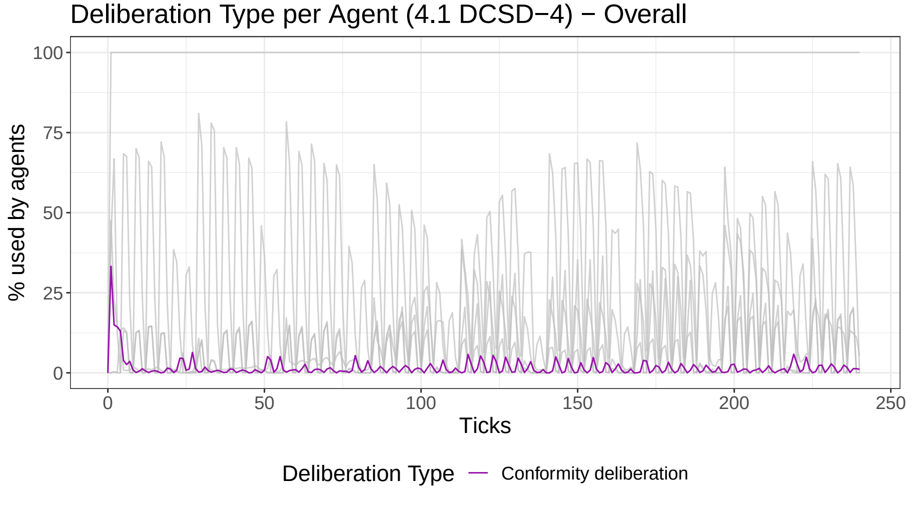


Figure 6.46: Conformity Deliberation Type for Social DCSD

Now consider Figure 6.46 where only the use of social deliberation (conformity deliberation) is highlighted. It should become clear that conformity deliberation is used less frequently, averaging perhaps just 1%. This is why there is less of an improvement than when going from the Strategic DCSD to the Normative DCSD. Nevertheless, there is an improvement in the execution time!

6.4.3 Summary

The social DCSD gave a 10.5 times speed-up compared to Original ASSOCC. This is a bit more than the Normative DCSD speed-up of 9.1 times. A question one could ask is if it would be worth it to implement the Social DCSD. However, as the percentage of solved decision situations by the DCSD gets closer to 100%, a larger speed-up can be expected. Therefore, not missing a type of information can make a big difference in the end. In the next section, the final version of the DCSD, the Full DCSD, will be evaluated.

6.5 Realism: DCSD ASSOCC

The Full DCSD is the final version of the DCSD ASSOCC model. Figure 6.47 shows how the Full DCSD is used in the ASSOCC deliberation. In short, the optimisation made is happening in the obligation and obligation work from home decision trees. There is an extra check where the agent will prefer select a working action if the food-safety or luxury needs are salient but not critical. This prevents the deliberation going to the slow need-based deliberation to solve the decision making. This was fully explained in Section 5.5. For the remainder of this thesis DCSD ASSOCC will mean, the ASSOCC model with Full DCSD activated.

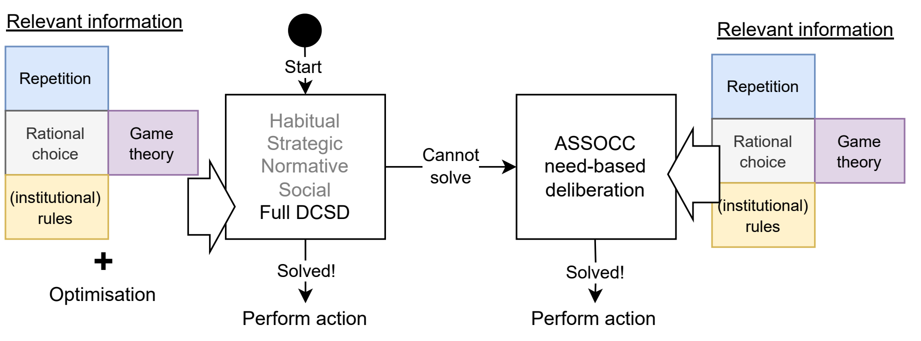


Figure 6.47: Optimisation included in the DCSD

In this section, a more detailed comparison between DCSD ASSOCC and Original ASSOCC will be made. First, it is determined whether DCSD ASSOCC passes the criteria and gives realistic enough infection curves in multiple simulation runs. To solidify the results the simulation will be ran 25 times with different random seeds. The second step is to dive into more detail and compare the behaviour, infection curves and needs over time between DCSD ASSOCC and Original ASSOCC model. This more detailed analyses is divided into 1) analysing the daily life patterns, 2) a standard run with infected and 3) the global lockdown scenario. This section ends with the verdict on whether DCSD ASSOCC can be considered as realistic enough.

6.5.1 Multiple Runs Comparison - DCSD vs Original

This section compares the DCSD ASSOCC model and the Original ASSOCC model in more detail. There are two types of experiments, the first is a default run with infected, the second is a run with global lockdown. In terms of results, first it will be determined whether DCSD ASSOCC passes the criteria for multiple runs, secondly the infections curves will be compared for similarity.

Comparison With Infected - Experiment Settings

Table 6.12 shows the experimental settings for the comparison with infected. The presets are *0.1 Original ASSOCC* for Original ASSOCC and *5.1 DCSD-5-optimisation* for DCSD ASSOCC. The number of households is set to the default of 350, infected are enabled, but global lockdown is not enabled. The random-seed is set as a range from 0 to 24, giving 25 unique runs. It has been chosen to do 25 runs because with fewer runs the infection curves keep changing, after 25 runs there is little change to the infection curves when adding more samples. The more detailed preset settings can be found in the Appendix 8.1.3.

Name	**Value**
ce-context-experiment-presets	"0.1 Original ASSOCC" "5.1 DCSD-5-optimisation"
ce-households-for-context-scenario	350
ce-enable-global-lockdown	false
with-infected?	true
stop-before-tick	241
random-seed	0, 1, 2, ..., 23, 24 (n = 25)

Table 6.12: Experimental setup realism comparison with infected.

Comparison With Infected - Criteria

Table 6.13 shows the results of measuring the criteria for all 25 runs for the *5.1 DCSD-5-optimisation* preset. In this table it is only shown how often the model passes the criteria. Since the model passes all criteria in all runs no further analysis is needed of specific runs that do not pass. The next section will compare the infection curves of DCSD ASSOCC and Original ASSOCC for multiple runs. And further into this section individual runs will be analysed thoroughly.

Cr	**Passed**	**Cr**	**Passed**	**Cr**	**Passed**
C1	25/25	C6	25/25	C11	25/25
C2	25/25	C7	25/25	C12	25/25
C3	25/25	C8	25/25	C13	25/25
C4	25/25	C9	25/25	C14	25/25
C5	25/25	C10	25/25	C15	25/25

Table 6.13: Criteria Passed in 25 runs for DCSD ASSOCC 5.1

Comparison With Infected - Infection Curve

The presets for the first experiments are *0.1 Original ASSOCC* and *5.1 DCSD-5-optimisation*. Figure 6.48 shows the total agents infected over time, note that

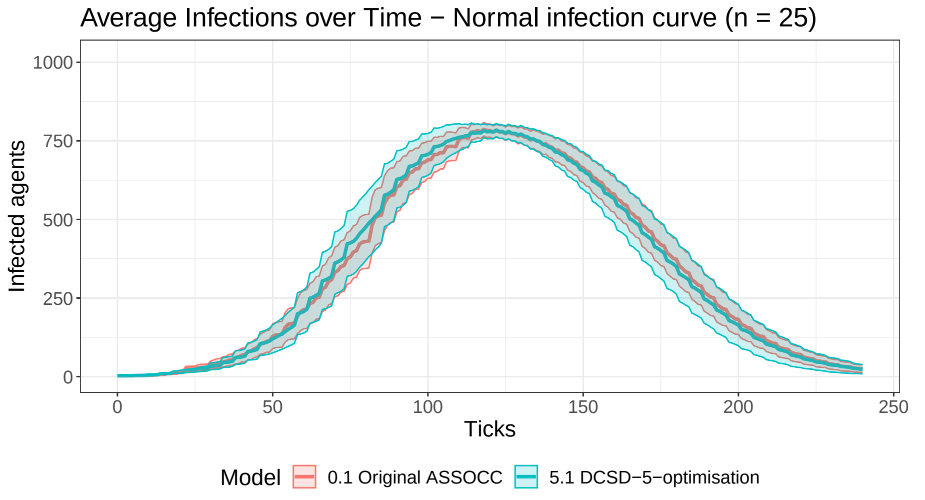


Figure 6.48: Comparing the spread of the virus - With infections

this figure is showing the actual infected agents and not the agents that believe they are infected. The infection curves of the two different presets are grouped together and averaged. The shaded area around the lines indicate the upper and lower bound of the standard deviation of the average. This gives an indication of how much variation there is between individual runs. It can be seen that the lines look very similar, they overlap almost completely. Purely looking at these graphs one could state that both models are very similar. If one were to take two times 25 runs with the Original ASSOCC model where those 25 runs use different random seeds, the result would probably show as much difference as shown in Figure 6.48. Therefore, we can confidently say that the DCSD in this preset produces a realistic enough infection curve.

Comparison Global Lockdown - Experiment Settings

To be more complete, the models should also be compared with a different setting. Table 6.14 shows the experimental settings for the comparison with infected and global lockdown. The presets are *0.2 Original ASSOCC-lockdown* for Original ASSOCC with global lockdown and *5.2 DCSD-5-optimisation-lockdown* for DCSD ASSOCC with global lockdown. The experimental settings are similar to the previous settings, with the exception of enabling global lockdown and increasing the *stop-before-tick* setting to 481. The more detailed preset settings can be found in the Appendix 8.1.3.

Name	**Value**
ce-context-experiment-presets	"0.1 Original ASSOCC-lockdown" "5.1 DCSD-5-optimisation-lockdown"
ce-households-for-context-scenario	350
ce-enable-global-lockdown	true
with-infected?	true
stop-before-tick	481
random-seed	0, 1, 2, ..., 23, 24 (n = 25)

Table 6.14: Experimental setup realism comparison with global lockdown.

Comparison Global Lockdown - Criteria

Table 6.15 shows the results of measuring the criteria for all 25 runs for the *5.2 DCSD-5-optimisation-lockdown* preset. In this table it is only shown how often the model passes the criteria. Since the model passes all criteria in all runs no further analysis is needed of specific runs that do not pass. The next section will compare the infection curves of DCSD ASSOCC and Original ASSOCC for multiple runs with global lockdown. And further into this section individual runs will be analysed thoroughly.

Cr	**Passed**	**Cr**	**Passed**	**Cr**	**Passed**
C1	25/25	C6	25/25	C11	25/25
C2	25/25	C7	25/25	C12	25/25
C3	25/25	C8	25/25	C13	25/25
C4	25/25	C9	25/25	C14	25/25
C5	25/25	C10	25/25	C15	25/25

Table 6.15: Criteria Passed in 25 runs for DCSD ASSOCC 5.2 Global Lockdown

Comparison Global Lockdown - Infection Curve

Figure 6.49 shows the total number of agents infected over time. The shaded area around the lines indicates the upper and lower bounds of the standard deviation of the average. This gives an indication of how much variation there is between individual runs. The trend that can be observed is that the number of infected increases slowly at the beginning of the simulation. During the global lockdown, on average starting at tick 25 and ending at tick 249 (224 ticks later), the curve flattens. Roughly after tick 250 the second wave starts and there is a peak of infections after which the number of infections goes towards zero. This pattern is similar between both the Original ASSOCC and the DCSD ASSOCC.

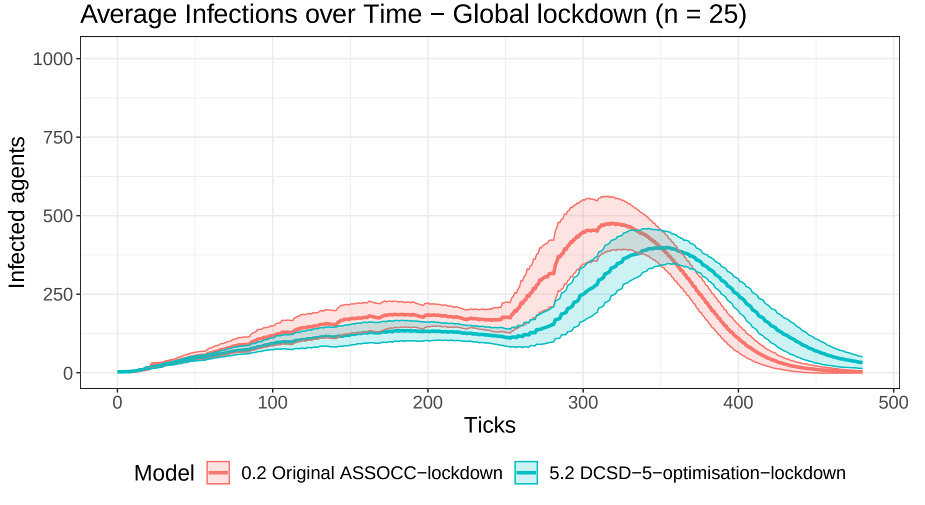


Figure 6.49: Comparing the spread of the virus - With global lockdown

The lines differ on some points, for example the Original ASSOCC line is slightly higher on average, meaning the virus spreads a bit faster. The infection curve has a higher peak that occurs a bit earlier. However, despite these differences the same general pattern remains, i.e. a relatively flat infection rate during the global lockdown and a peak after the global lockdown. This is exactly the pattern that the model should show to be realistic enough. Despite some differences in the specific behaviour of the agents (as illustrated in the earlier sections), both models provide realistic infection curves that match what happened in reality.

What should be noted here is that the shape of the infection curve can still be tweaked. For example, the rate of social distancing could be adjusted. It could be adjusted to specific situations; for example, when a global lockdown is active agents would social distance more frequently. In addition, by adapting the needs of the agents and consequently adjusting the behaviour, the infection curves will change shape.

Summary

For both the preset with infected and with global lockdown DCSD ASSOCC performs well. In 25 out of 25 runs all the criteria are passed for both presets. In addition, when comparing both infection curves, the infection curve of DCSD ASSOCC follows the same pattern as the Original ASSOCC infection curve. However, comparing only criteria and infection curve for multiple runs is not enough to have a completed validation of the realism of the model. Still, a more detailed analysis is required where the behaviour of the agents is analysed to make sure that this is realistic enough as well. This complimentary more-detailed analysis will be shown in the upcoming section.

6.5.2 DCSD ASSOCC - Patterns of daily life

This section is dedicated to a more detailed comparison between the two models. The experimental presets for Original ASSOCC and DCSD ASSOCC are respectively *0.0 Original ASSOCC-no-infections* and *5.0 DCSD-5-optimisation-no-infections*. The random seed is set to two. First, for good measure, the criteria are shown including the values. Secondly, the patterns of daily life are compared between the models. More specifically, the patterns of shopping and choosing leisure activities.

Patterns of Daily Life - Criteria

Table 6.16 shows the criteria for DCSD ASSOCC experiment without infected. The table shows that all the criteria pass since all the values are within the required boundaries. The next section will compare the two models and analyse the similarity and differences between the models that are not shown by purely considering the criteria.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.68	TRUE
C3	Recently Ess Shopping, mean > 98%	98.03	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.93	TRUE
C5	Not Skip Work, mean > 98%	99.96	TRUE
C6	Work at Workplace when possible, 85% < mean	98.59	TRUE
C7	Not Skip School, mean > 95%	100	TRUE
C8	Not Skip University, mean > 95%	99.84	TRUE

Table 6.16: Criteria Values for 5.0 DCSD-5-optimisation-no-infections

Patterns of Daily Life - Behaviour And Population Status

Figure 6.50 shows the agents' behaviour over time (averaged by day). The graphs show the run with no infections for both the Original ASSOCC model and the DCSD ASSOCC. It can be seen that the agents follow a pattern of working, studying, shopping, leisure activities, and being at home. The biggest difference between the model seems to be the frequency of agents being at home. The agents in the Original ASSOCC model are less frequently at home, while the DCSD ASSOCC agents are more regularly at home.

Working or studying out of home
Figure 6.50 shows that the working or studying out of home is very similar between the two models. There is a similar pattern throughout the days. Work or studying not at home is frequent throughout the working week (Monday to Friday). On Saturdays (see days 6, 13, 20, etc.) both models show a small

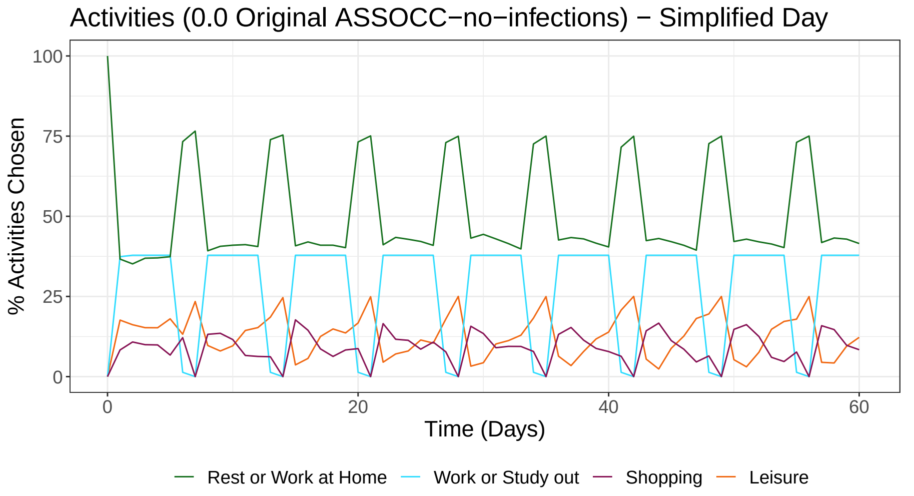


Figure 6.50: Activities comparison day - No infected

amount of activity. This is due to agents that work in the shops, who are required to work on Saturdays as well.

Shopping
Figure 6.50 showed that shopping is more prevalent throughout the week and is not happening on Sunday (see data points 7, 14, 21, etc.). In Original ASSOCC the agents shop more frequently than in DCSD ASSOCC. This difference can be explained by analysing the grocery shopping and luxury shopping behaviour separately.

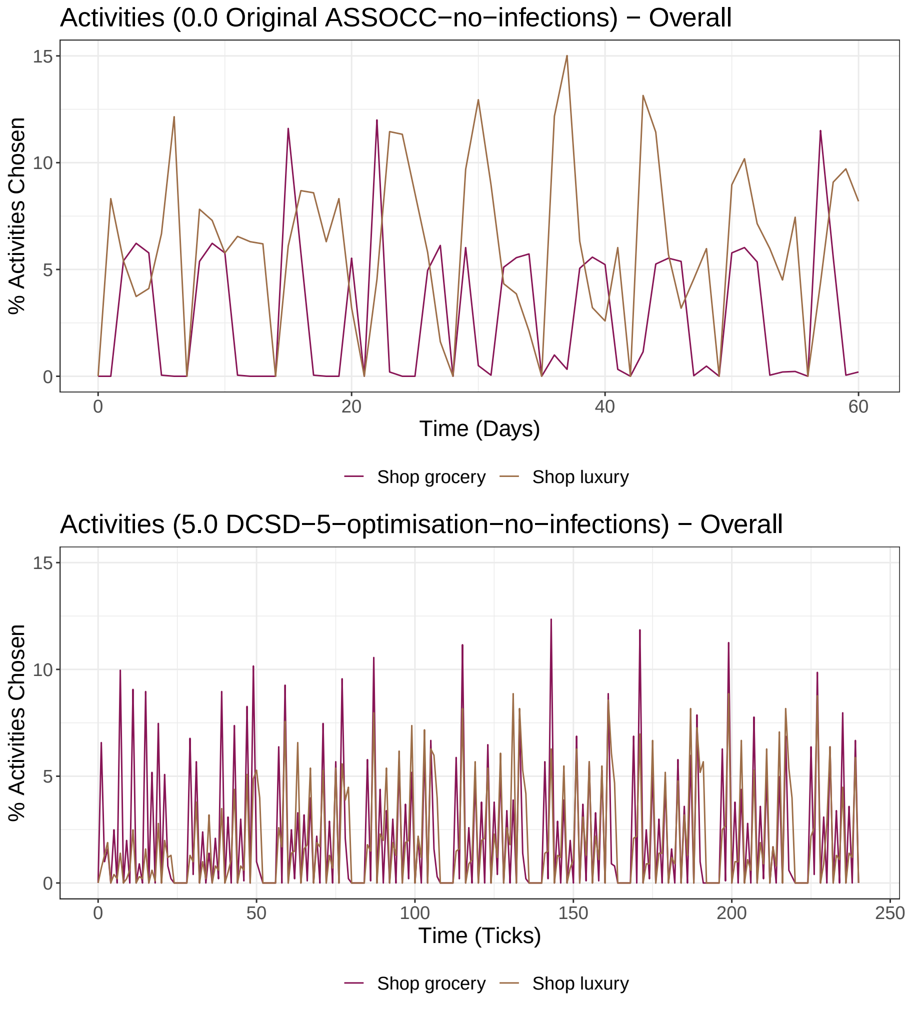


Figure 6.51: Activities comparison day - No infected

Figure 6.51 shows just the grocery and luxury shopping behaviour over time. It should be clear that luxury shopping has much higher peaks in Original ASSOCC than DCSD ASSOCC. Luxury shopping is probably more frequent in Original ASSOCC since it depends on the calculation of multiple needs, for example also belonging which could get a large boost especially in the evenings when there are many agents. There are five needs relevant for luxury shopping. Financial survival and financial stability which are negatively impacted by luxury shopping, however they are generally not salient since agents are working normally. The belonging need and luxury need are positively effected by luxury shopping while risk avoidance is negatively affected. Probably due to two needs being in favour and only one need being against the Original ASSOCC deliberation chooses luxury shopping quite frequently. For DCSD ASSOCC luxury shopping is only performed when the luxury need level is below 0.5.

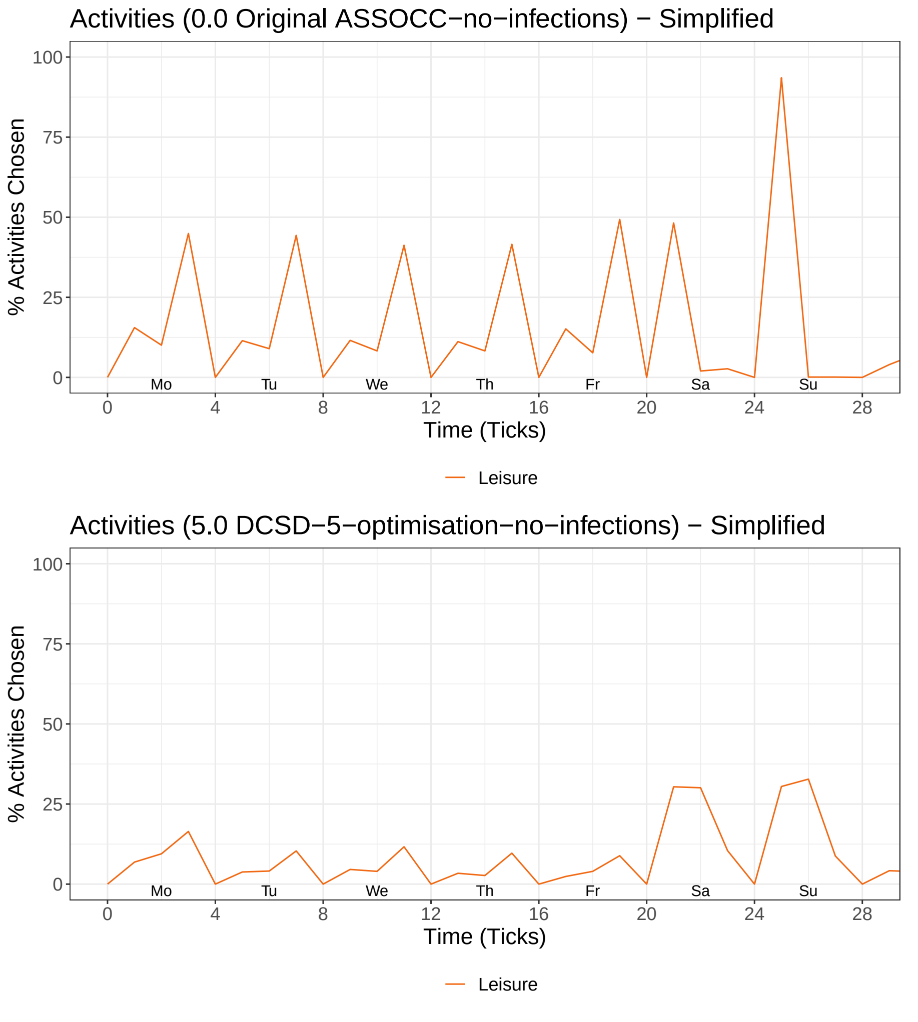


Figure 6.52: Leisure highlight comparison - No infected

Leisure
Leisure is performed throughout the week and becomes more frequent in the weekends. Figure 6.52 analysing leisure in more detail. It can be seen where leisure is more frequently performed and where not. Leisure happens mainly

in the morning on Saturday and Sunday for both simulations. However in detail leisure is done slightly differently. The Original ASSOCC agents mainly do leisure activities in the Saturday and Sunday morning. While the DCSD ASSOCC has a more spread out performance of leisure activities. The agents do leisure activities not only in the morning on Saturday/Sunday but also do plenty during the afternoon and evening. One could argue that the latter structure is a bit more realistic since leisure is not just a Saturday/Sunday morning thing, but happens in the weekend in the morning, afternoon and evening. This more specific selection of when leisure is performed was possible because of the conceptually more advanced DCSD.

One could change the frequency by adding more leisure habits, but could also argue that the DCSD leisure behaviour is more realistic. It is more spread out, and not everything just happens on the morning of the weekend which is the case in Original ASSOCC.

Rest or Work at Home

Resting at home is often the default action, chosen when other actions are not preferred. Since in DCSD ASSOCC there are on average less leisure and shopping activities it makes sense that the agents on average are home more frequently. Work at home is included in the graphs, however hardly any agent will perform this action since the simulation is without infected. Working from home is mainly chosen in a simulation run with infected and with a global lockdown, as working from home is mainly for worker agents who are not sick but have to stay at home.

6.5.3 DCSD ASSOCC - Standard Pandemic Curve

This section compares the Original ASSOCC model and the DCSD ASSOCC model with infected enabled. The simulation presets are *0.1 Original ASSOCC* and respectively *5.1 DCSD-5-optimisation*. First the criteria are measured of the DCSD ASSOCC model, secondly the more detailed behaviour is analysed.

Pandemic Curve - Criteria

The criteria for the Original ASSOCC model are already shown in Table 4.6. Table 6.17 shows those for the DCSD ASSOCC model. All of the criteria pass, thus we can move on to the next section and analyse the model in more detail.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.86	TRUE
C3	Recently Ess Shopping, mean > 98%	99.16	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	100	TRUE
C5	Not Skip Work, mean > 98%	99.57	TRUE
C6	Work at Workplace when possible, 85% < mean	99.51	TRUE
C7	Not Skip School, mean > 95%	98.87	TRUE
C8	Not Skip University, mean > 95%	99	TRUE
C9	Rest When Know Sick, mean > 90%	94.71	TRUE
C10	People in Quarantine, 90% < mean < 100%	97.58	TRUE
C11	Children in Quarantine, 90% < mean < 100%	98.24	TRUE
C12	Students in Quarantine, 90% < mean < 100%	96.73	TRUE
C13	Workers in Quarantine, 90% < mean < 100%	97.24	TRUE
C14	Retireds in Quarantine, 90% < mean < 100%	98.94	TRUE
C15	Infection Peak Tick, 75 < value < 150	118	TRUE

Table 6.17: Criteria Values for 5.1 DCSD-5-optimisation

Comparison With Infected - Time Series

Figure 6.53 shows the frequency of the activities smoothed over time. The activity patterns are relatively similar, as the agents are more frequently at home, while they work/study less frequently during the peak and also have less leisure activities. Shopping is only slightly affected in both models. The peak slightly differs, 150 for Original ASSOCC and 140 for DCSD ASSOCC, however, this can be attributed to variations in individual runs. The working or study out line looks very similar between the two graphs. There is a difference between the leisure and shopping frequency, but that has also been explained in the previous section. Leisure activities and shopping are performed less frequently in the DCSD ASSOCC. However, the agents do perform those free time actions enough to pass the criteria (see Table 6.17). Despite the rather small differences between the behaviour of the models, both behavioural patterns shown in the figures can be considered realistic enough.

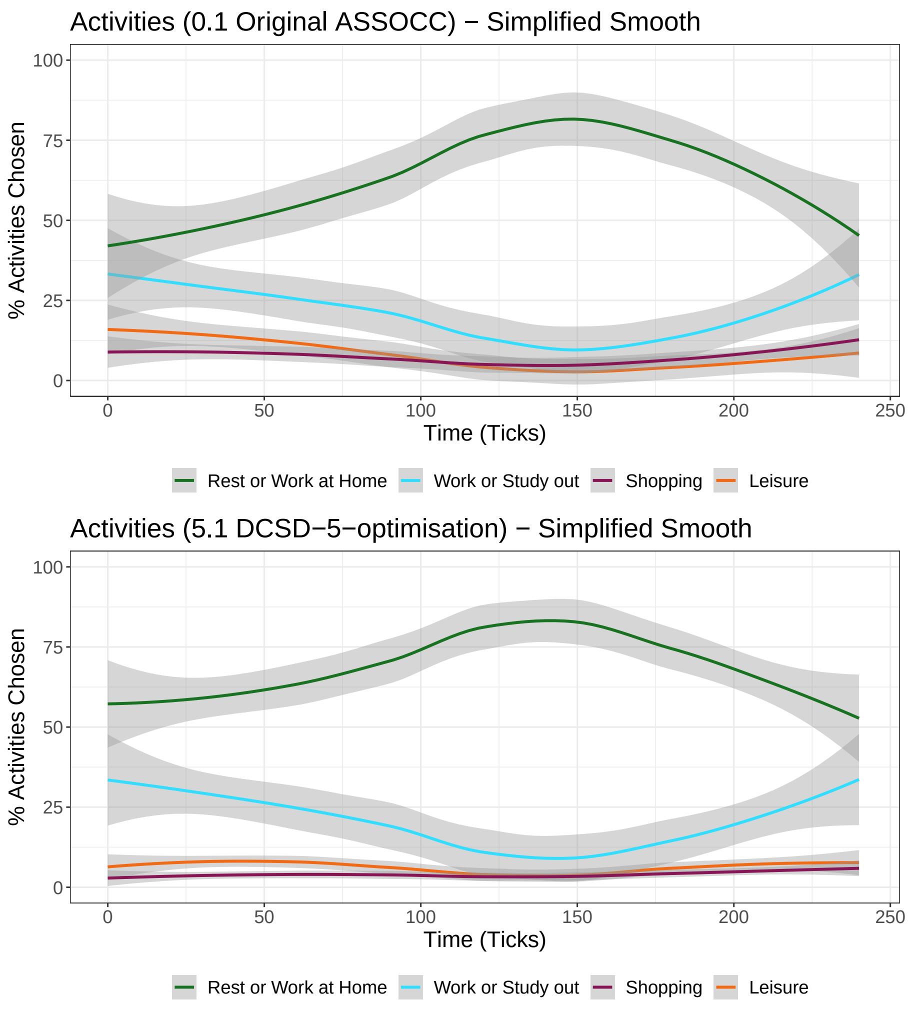
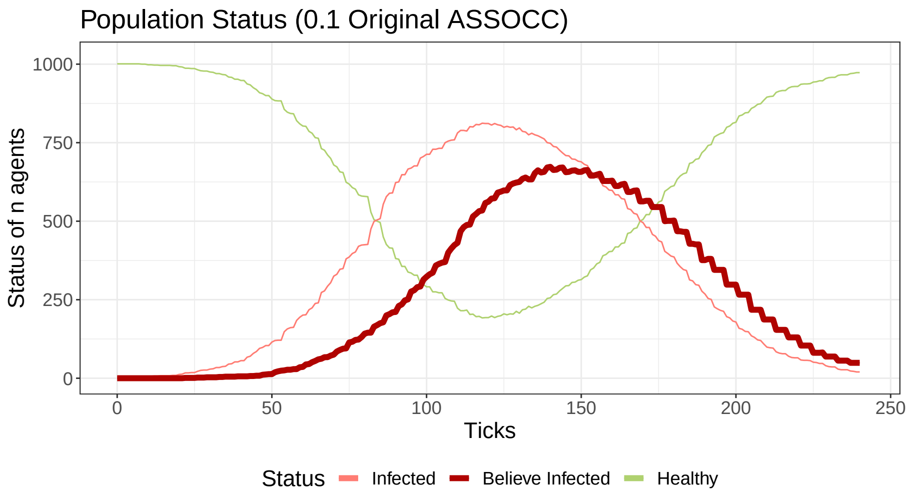


Figure 6.53: Activities comparison smoothed - With infected

Figure 6.54 shows the infected agents, healthy agents and most clearly visible the number of agents that believe they are infected. It shows that the believe infected curve is similar to the curve and peaks in the activities of the agents (see Figure 6.53). That the curves look similar was to be expected as it has already been determined before in the infection curve plots of multiple runs, see Figure 6.48. The healthy line is basically the inverse of the infected line. The believe infected line is related to the agents that get infected.



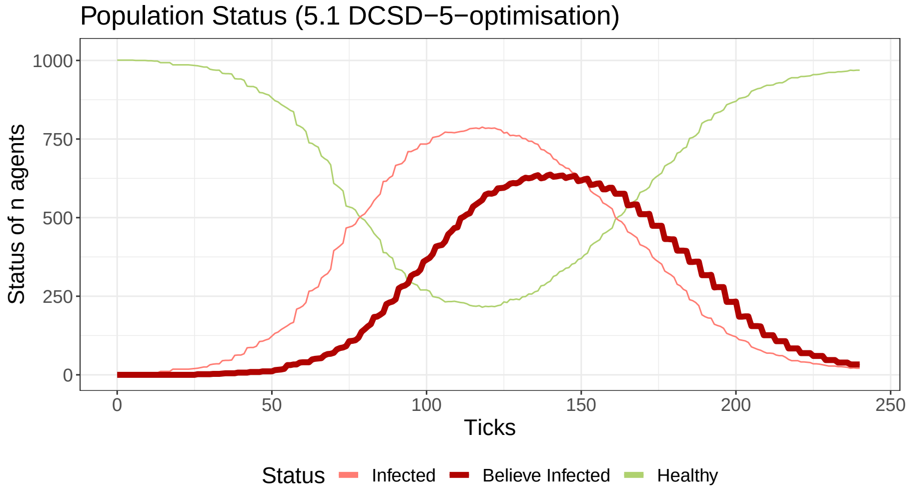


Figure 6.54: Population Status comparison - With infected

Figure 6.55 shows agents that have to be in quarantine and those that are breaking quarantine. In both runs, a similar pattern can be seen that corresponds to the believe infected curve, in Figure 6.54. Most agents stay in quarantine, but sometimes some agents break quarantine. Breaking quarantine is most frequent when there are more agents asked to quarantine, i.e. roughly between tick 100 and 200. The exact numbers differ slightly while the general pattern is similar. As shown in the criteria, Table 6.17, the DCSD ASSOCC model stays within the boundaries. All in all the behaviour of the agents in this experiment looks similar enough, and it produces very similar infection curves for both runs.

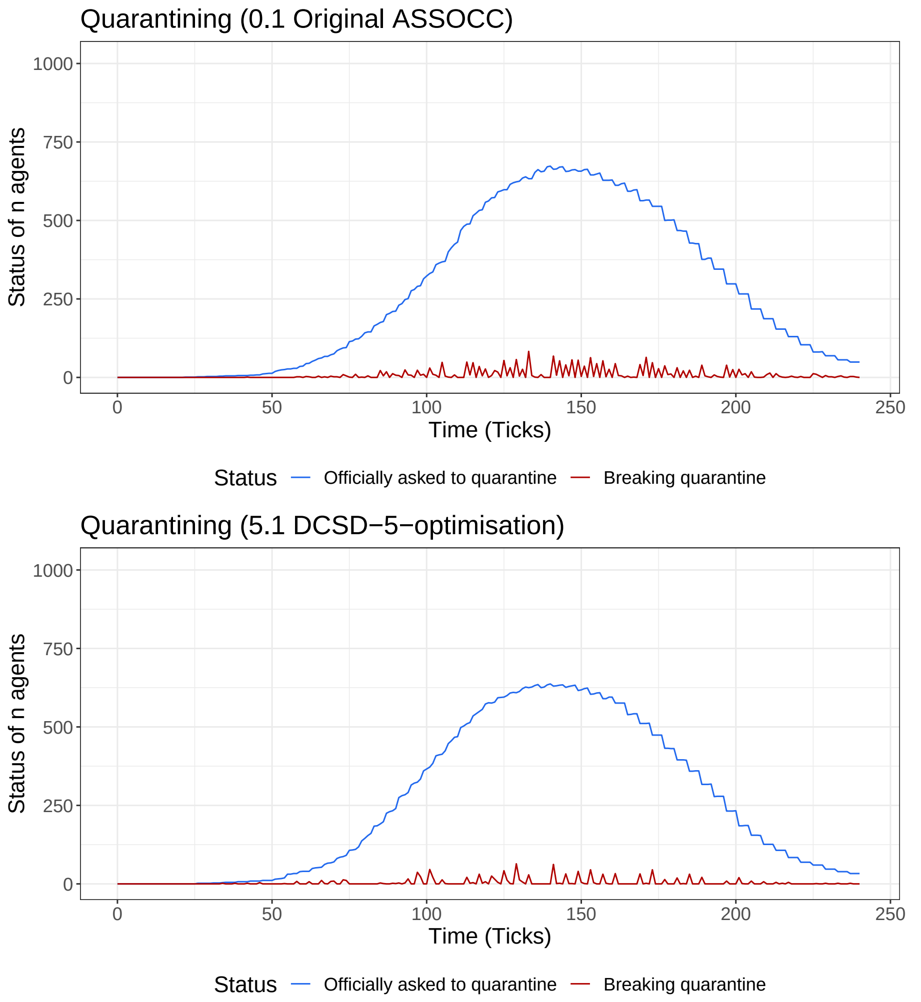


Figure 6.55: Quarantine comparison - With infected

6.5.4 DCSD ASSOCC - Global lockdown

This section compares the Original ASSOCC model and the DCSD ASSOCC model with infected and global lockdown enabled. The simulation presets are *0.2 Original ASSOCC-lockdown* and respectively *5.2 DCSD-5-optimisation-lockdown*. First, the criteria are measured of the DCSD ASSOCC model, secondly the more detailed behaviour is analysed.

Comparison Global Lockdown - Criteria

The criteria for the Original ASSOCC model with global lockdown enabled are already shown in Table 4.7. Table 6.18 shows those for the DCSD ASSOCC model. All of them pass, but let us do a more detailed analysis using the time series.

Cr	Criteria Description	Value	Pass
C1	Night Home, mean > 99%	100	TRUE
C2	Recently Leisure, mean > 98%	99.71	TRUE
C3	Recently Ess Shopping, mean > 98%	98.86	TRUE
C4	Recently Non-Ess Shopping, mean > 98%	99.88	TRUE
C5	Not Skip Work, mean > 98%	99.91	TRUE
C6	Work at Workplace when possible, 85% < mean	99.39	TRUE
C7	Not Skip School, mean > 95%	99.67	TRUE
C8	Not Skip University, mean > 95%	99.72	TRUE
C9	Rest When Know Sick, mean > 90%	94.61	TRUE
C10	People in Quarantine, 80% < mean < 90%	91.76	TRUE
C11	Children in Quarantine, 80% < mean < 98%	87.58	TRUE
C12	Students in Quarantine, 80% < mean < 98%	84.46	TRUE
C13	Workers in Quarantine, 80% < mean < 98%	94.9	TRUE
C14	Retireds in Quarantine, 80% < mean < 98%	96.05	TRUE
C15	Infection Peak Tick, 250 < value < 400	356	TRUE

Table 6.18: Criteria Values for 5.2 DCSD-5-optimisation-lockdown

Comparison Global Lockdown - Behaviour and Population Status

Figure 6.56 shows the activities averaged by day. The global lockdown starts at tick 7 and stops at tick 63 (the global lockdown has a fixed time of 56 days or 8 weeks). In Figure 6.56 it should become clear that sometimes agents break global lockdown for various reasons: e.g. work or study out of home, shopping, leisure. If one looks carefully there are some differences between the models during global lockdown. In the DCSD ASSOCC model work or study out of home is performed more frequently, while in the Original ASSOCC model leisure is performed slightly more frequently. This does not mean that DCSD ASSOCC model is less realistic. As long as the behaviour is within certain boundaries and the infection curve is similar to the Original ASSOCC model the model is realistic enough. The shopping behaviour in Original ASSOCC is much higher

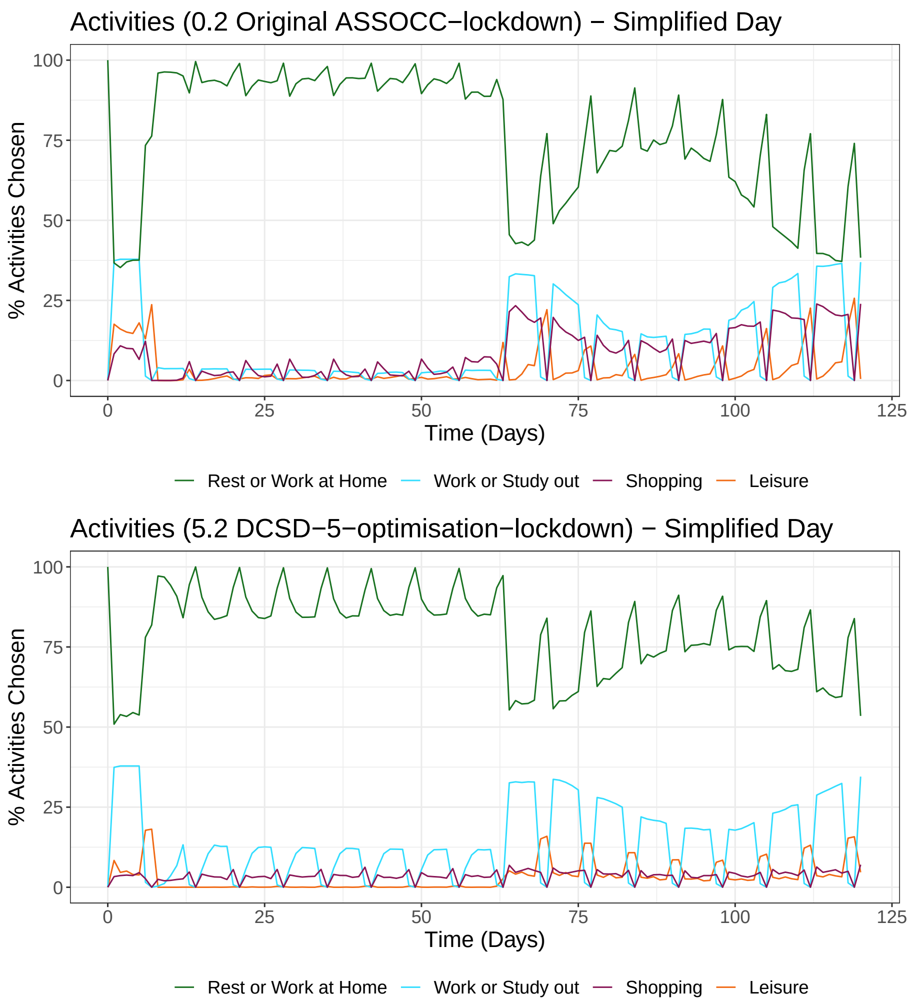
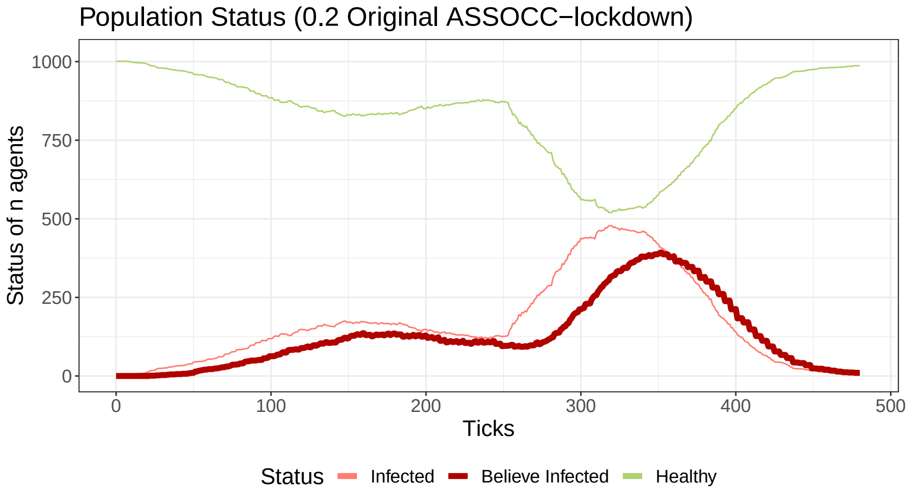


Figure 6.56: Activities comparison day - With global lockdown

Population status and quarantining

Figure 6.57 shows the population status. It can be seen that in both simulation runs the global lockdown has the effect of slowing down the spread of the virus since the curve gets flattened. About halfway during the global lockdown the number of infected (pink line) starts to go down. However, when the global lockdown ends, both graphs show an increase in the number of infected, forming the second higher wave.



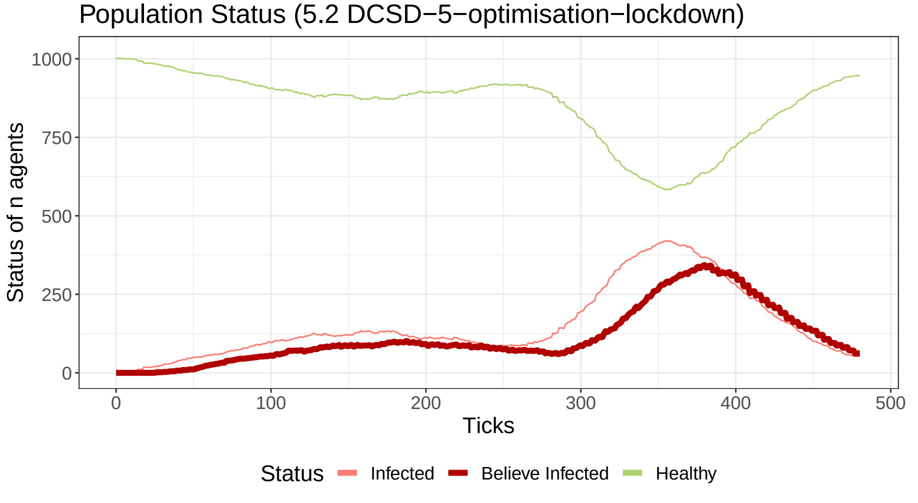


Figure 6.57: Population Status comparison - With global lockdown

Figure 6.58 shows the quarantining behaviour. The blue line shows the agents that should be in quarantine, which is all the agents during global lockdown. After global lockdown, only the agents that believe they are infected are represented by the blue line. In both models some agents break quarantine. This happens more frequently during global lockdown, this has to do with the fact that more agents are in quarantine. If more agents are in quarantine, especially when they are not sick, a higher number of agents will be breaking quarantine since there are more potential quarantine breakers.

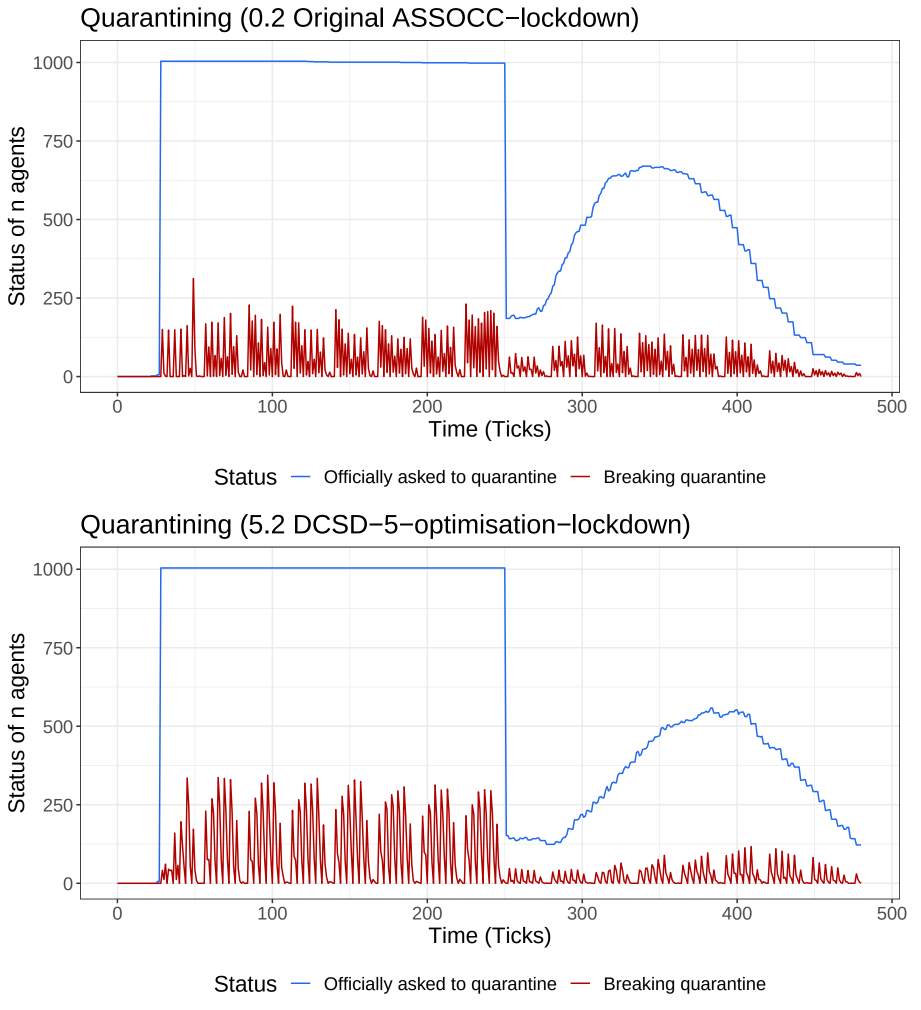


Figure 6.58: Quarantining comparison - With global lockdown

Shopping behaviour
Analysing the specific behaviour during global lockdown we can observe quite some differences. Figure 6.59 shows the grocery and luxury shopping activities. The general frequencies of shopping differ between the models, as discussed in the daily patterns section. In this section, we are more interested in the changes during and after the global lockdown.

The figure shows that in the Original ASSOCC model the agents do not

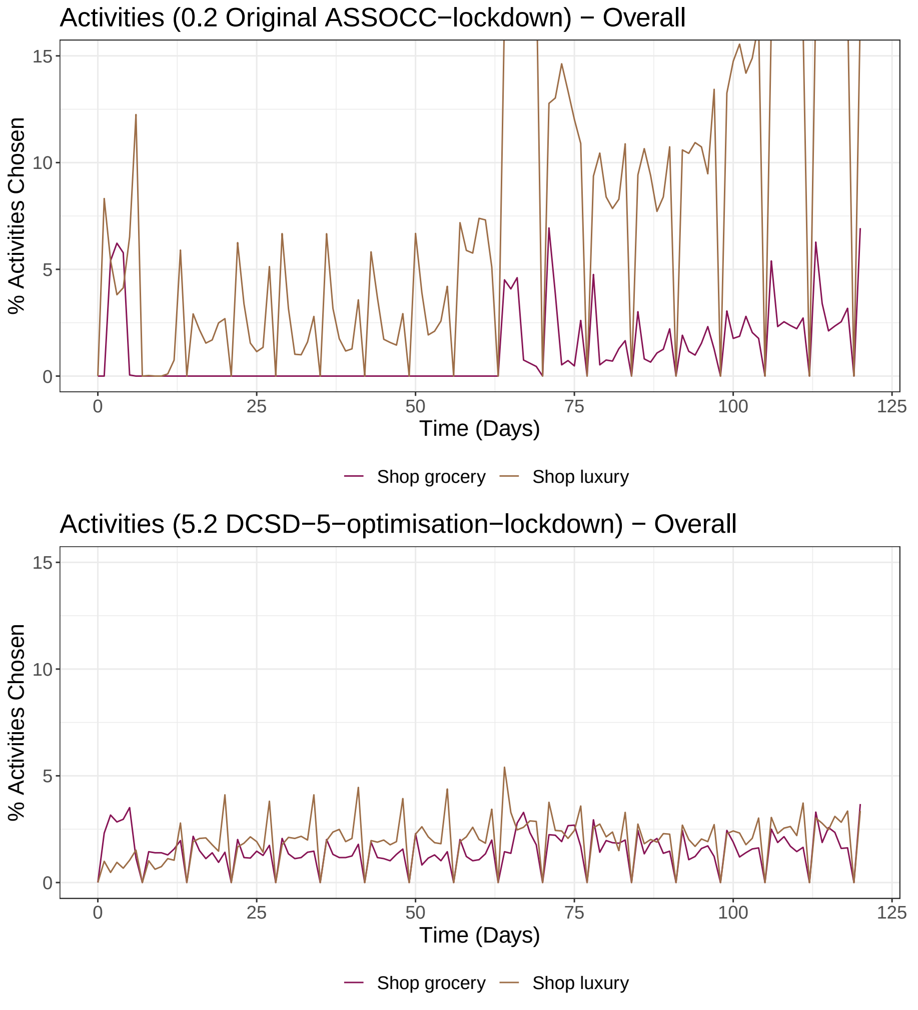


Figure 6.59: Shopping comparison day - With global lockdown

do grocery shopping during the lockdown. In the DCSD ASSOCC model the agents do perform grocery shopping during lockdown. This could be explained by the difference in food-safety need calculations between the two models. Consider Figure 6.60, the green line indicates the average food-safety need level. This level is highly satisfied for Original ASSOCC agents. However, it is less satisfied (around 0.75) for DCSD ASSOCC agents. This is probably due to the Original ASSOCC model being perfectly calculated that if food is delivered during the lockdown, the agents never have to do grocery shopping. While the DCSD ASSOCC has a setting to make it more likely for agents to shop, which specifically affects the food-safety need. With more time this could be balanced out in such a way that in the DCSD ASSOCC there is also such a balance. However, for this use case the DCSD ASSOCC model portrays realistic enough behaviour as the agents do adapt to the global lockdown by decreasing leisure and work/study actions.

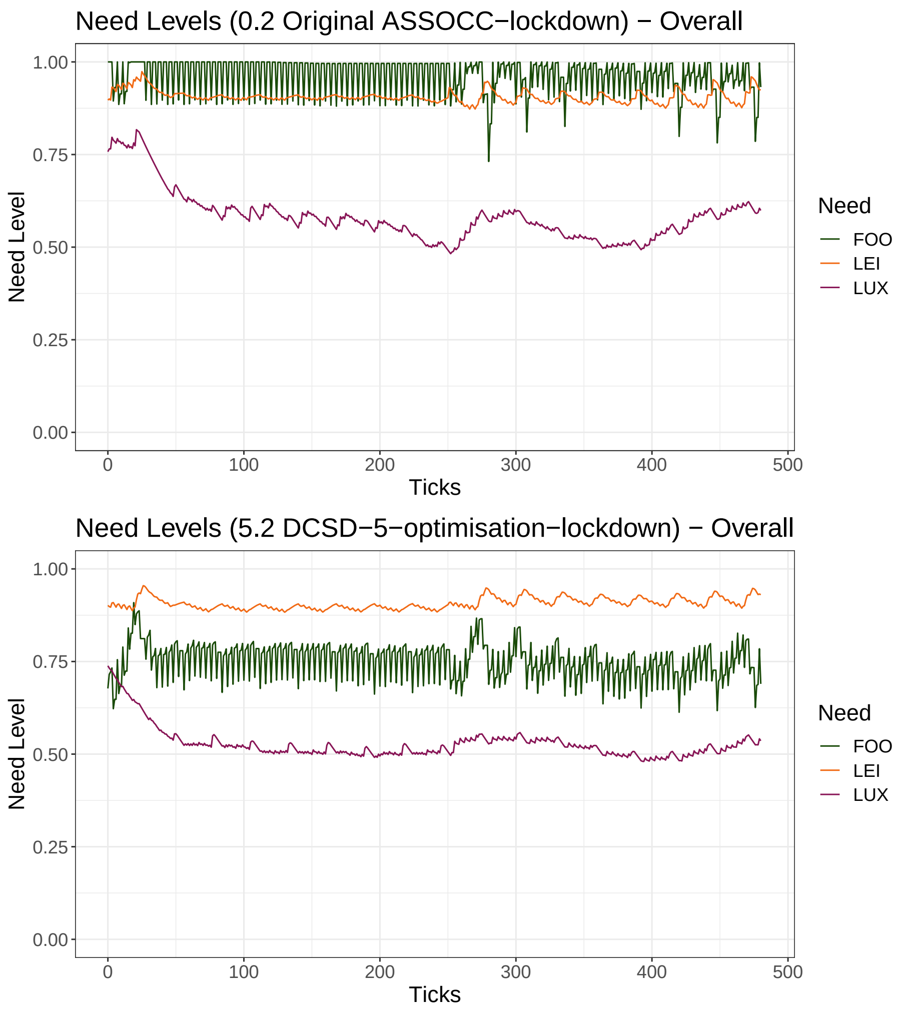


Figure 6.60: Needs comparison day - With global lockdown

Leisure activities
Figure 6.61 shows the leisure activities over time. In the top figure during the global lockdown the amount of leisure activities sharply decreases. After

global lockdown the leisure activities are performed frequently again. The bottom figure, DCSD ASSOCC, shows that during global lockdown no leisure activities are performed. This can be explained by the DCSD's implementation. As explained in Section 6.1 the leisure need level is generally quite high, around

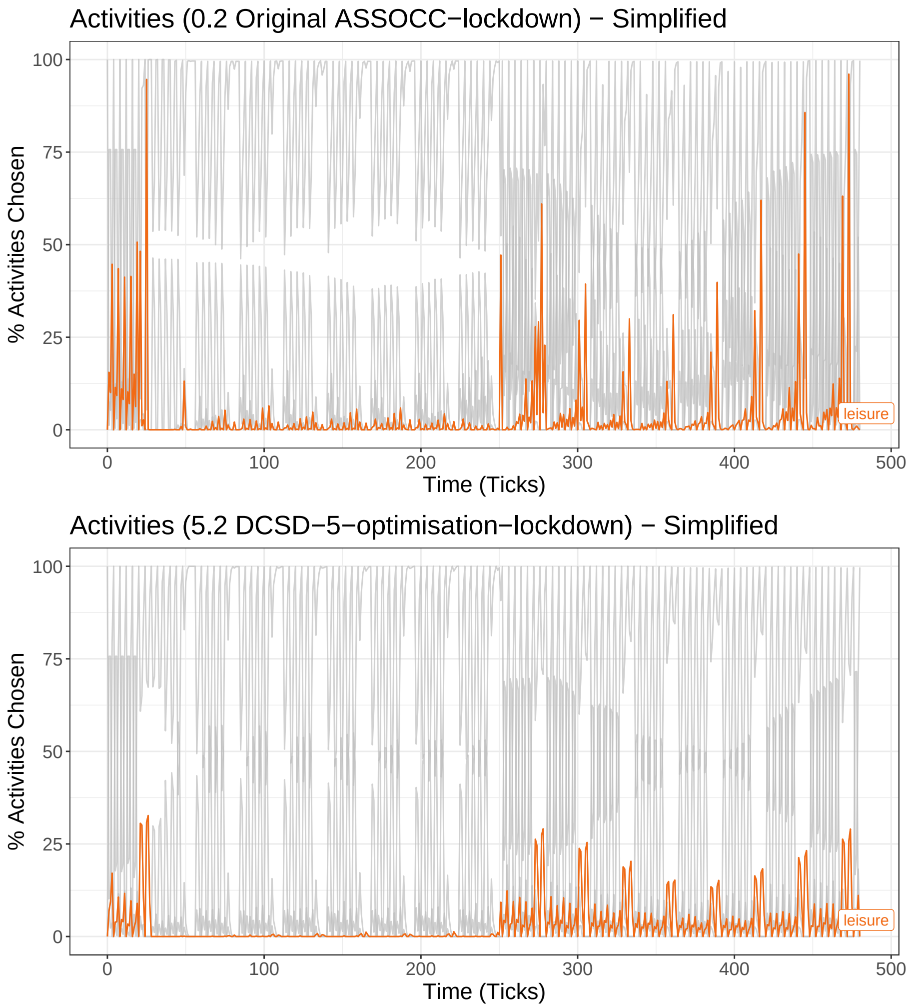


Figure 6.61: Leisure activities comparison - With global lockdown

0.9 (see also 6.60). This meant that agents would hardly ever perform leisure activities, since the level of leisure need hardly ever drops below the threshold of 0.5. The model was adapted to include some specific moments of leisure activities. Meaning, a leisure activity would be performed during free time if no other needs were salient and the agent is not in quarantine, otherwise need-based deliberation is used. During the global lockdown all the agents are in quarantine, thus the DCSD will never choose the leisure activity. The only way the agents can do leisure activities during global lockdown is through need-based deliberation which is used very infrequently in the DCSD ASSOCC model.

Summary

The simulations both have some behaviour that is not performed during the global lockdown. In Original ASSOCC, the agents do not do grocery shopping. This happens because the agents are getting supplied with enough food to get through the global lockdown. In DCSD ASSOCC the agents do get similar amounts of food, but they are due to their deliberation settings slightly more likely to buy food, due to the setting which decreases the food-safety need level making agents more likely to grocery shop. The agents in the DCSD do not perform leisure activities during global lockdown. This is due to the leisure need in the DCSD being so high, it never becomes critical, which is necessary for the DCSD to break the quarantine rule. Despite these differences both simulations show the same pattern in the infection curves. This also shows that there is not one right way of making a simulation. Rather, there are multiple ways, and even within the same simulation there can easily appear differences in behaviour when changing the deliberation model.

6.5.5 Scalability Aspects of DCSD ASSOCC

The behaviour of the agents with the infected run and the infected with global lockdown run seems to be realistic for DCSD ASSOCC. Before going into more detail on scalability in the next section, we will already give a glimpse of the deliberation speed-up of DCSD ASSOCC. The Original ASSOCC deliberation takes 96 seconds, DCSD ASSOCC takes 5.8 seconds which is 16.5 times as fast! We also tested the execution time of deliberation when global lockdown is enabled. Here, Original ASSOCC deliberation takes 181.0 seconds while DCSD ASSOCC takes 33.2 seconds which is 5.4 times as fast, still an improvement. However, do not forget that DCSD ASSOCC has not been optimised for the global lockdown scenario. DCSD ASSOCC is the final version of the DCSD described in this document. These initial numbers show that DCSD can speed-up deliberation in an agent-based simulation. The next section will solidify these results by applying multiple runs, evaluating the difference per DCSD version, and evaluating whether the speed-up is retained with larger agent numbers.

6.5.6 Realism Conclusion

The general patterns of the Original ASSOCC simulation are retained in the DCSD ASSOCC. Agents perform their daily activities, respond to the spread of the virus (when they believe they are infected), they respond to the global lockdown, and occasionally break quarantine for a variety of reasons. The models are in general similar and since Original ASSOCC is argued to be a realistic model [19], DCSD ASSOCC is also realistic by being similar enough. There are some small differences in the behaviour of the agents; however, they do not necessarily make the behaviour less realistic. What is deemed realistic enough is highly dependent on the model's purpose. If the purpose of the model is to do exact predictions of, for example, the exact number of people getting infected or indicate exactly how many days of lockdown is the most effective. Then having the right number becomes more important for a realistic enough model. However, if the purpose is to understand the underlying dynamics that can cause certain behaviour, then it is more important the model catches the right patterns of behaviour rather than the exact number. The main purpose for models such as ASSOCC is to understand the effectiveness of policies on society. And given this purpose, it is important that the model contains not only the following of policies but also reasons for agents to break the policies to study its effect [43].

6.6 Scalability: Overall Comparison

This section will assess whether DCSD can increase scalability in a social simulation. To do this an empirical comparison between Original ASSOCC and DCSD ASSOCC is made. The Original ASSOCC model is used as a baseline as explained in Chapter 4. The first experiment will measure the deliberation execution time for different versions of the DCSD. This execution time will then be compared with Original ASSOCC to determine the speed-up. With these results it can be determined whether DCSD can solve the deliberation bottleneck in ASSOCC. The second experiment will determine whether this result is retained with greater number of agents. Based on these results an argument will be made that determines that DCSD can scale deliberative aspects in an agent-based simulation.

6.6.1 Experiment 1: DCSD Versions

In the first experiment, the deliberation execution time is measured while incrementing the deliberative aspects of the DCSD. Table 6.19 shows the detailed experiment settings. The experiment has six presets (ce-context-experiment-presets) corresponding to the different deliberation models. One for the Original ASSOCC model and the other five for the five DCSD versions. In each DCSD the most updated version of the realism sections is used. The number of households is 350, leading to 1004 agents. The run will last 240 ticks (which amounts to 60 days), it contains infections and does not have global lockdown. The runs are repeated five times which is reflected by using five different random seeds, this is to account for individual variations between runs.

Name	**Value**
ce-context-experiment-presets	*0.1 Original ASSOCC*, *1.4 DCSD-1-leisure-habits*, *2.2 DCSD-2-obligation-constraint*, *3.3 DCSD-3*, *4.1 DCSD-4*, *5.1 DCSD-5-optimisation*
ce-households-for-context-scenario	350
ce-enable-global-lockdown	false
with-infected?	true
stop-before-tick	241
random-seed	0 1 2 3 4

Table 6.19: Experimental setup for DCSD versions comparison

DCSD Versions - Speed-up

Table 6.20 shows the results of the experiment. In the first row it shows the mean results of the baseline, i.e., Original ASSOCC model with infected. The table shows the deliberation execution time in milliseconds. The speed-up compared to the baseline, i.e. the baseline time divided by the compared model time. The percentage compared to the baseline, i.e. the compared model time divided by the baseline time multiplied by 100. It can be seen in the table that the deliberation execution time decreases the more layers of the DCSD are activated. Considering the speed-up, there is a large jump between 2.2 DCSD-2 and 3.3 DCSD-3, but also between 4.1 DCSD-4 and 5.1 DCSD-5. This is not to say that the other steps do not matter. Had the other parts of the DCSD model not been there, the speed-up would probably be only half of the achieved 16.7 speed-up, since each part contributes to the speed-up. Compared to the baseline, the execution time is now only 6.0% with the final DCSD version, 5.1 DCSD-5.

	Deliberation Execution Time			**Actions Chosen By**	
Preset	**Time**	**SU**	**Perc. of**	**DCSD %**	**NBD %**
0.1 Original	96,010.5 ms	1	100 %	- %	100 %
1.4 DCSD-1	43,987.8 ms	2.2	45.8 %	58.79 %	41.21 %
2.2 DCSD-2	30,691.4 ms	3.1	32.0 %	70.17 %	29.83 %
3.3 DCSD-3	10,696.0 ms	9.0	11.1 %	93.57 %	6.43 %
4.1 DCSD-4	9492.2 ms	10.1	9.9 %	94.90 %	5.10 %
5.1 DCSD-5	5761.4 ms	16.7	6.0 %	98.97 %	1.03 %

Table 6.20: Comparing different DCSD versions with the baseline, Original ASSOCC (SU = speed-up, NBD = need-based deliberation).

The last two columns of the table show the percentage of actions chosen by DCSD or need-based deliberation (NBD). For the Original ASSOCC model, which does not contain DCSD, all actions chosen are chosen by need-based deliberation, hence 100%. The DCSD ASSOCC contains both a DCSD module, which is used first, if it does not provide an action need-based deliberation will be consulted. It can be seen that the percentage of actions chosen by the DCSD grows as the DCSD contains more layers. Up to almost 99% for the final DCSD! This is exactly what causes the large speed-up, since the DCSD requires less execution time for selecting an action than the need-based deliberation.

Detailed DCSD Execution Time

To illustrate why DCSD is so effective, a more detailed analysis of the DCSD has been performed which measures three aspects of DCSD. Those are, the total deliberation time, the need-based deliberation time, and the time for the DCSD algorithm. The latter is calculated by subtracting the need-based deliberation time from the total deliberation time. Table 6.21 shows the execution time and number of calls of these three aspects of DCSD. The table shows the results from 5.1 DCSD-5, with random seed two.

The table shows that DCSD only time is higher than need-based deliberation time. However, the number of calls is also about 90 times greater for

	Time	Time %	Calls	Calls %	Time/Call
DCSD Only	4641.6 ms	80 %	240,420	100 %	0.019 ms
Need-based	1286.5 ms	20 %	2582	1.1 %	0.528 ms
Total	5803.7 ms	100 %	240,420	100 %	0.024 ms

Table 6.21: DCSD execution time and calls in detail, 1004 agents.

DCSD alone. A single DCSD function call (excluding need-based deliberation) takes less time on average than a single need-based deliberation function call. DCSD is called 100% of the times the agents deliberate. Need-based deliberation is used in only 1.1% of the deliberation function calls. If we consider how much execution time need-based deliberation requires, it is 20%. This perfectly illustrates how the complexity by need principle achieves scalability. The DCSD uses 0.019 ms per call while, need-based deliberation uses 0.528 ms per call. The average need-based deliberation calls are about 27.8 times slower than the average DCSD calls. Most of the time DCSD is successful in choosing an action, few times it is not and then need-based deliberation has to be used. This dynamic is based on the complexity by need principle, and it keeps the system as a whole efficient.

Summary of Experiment 1

Table 6.20 showed that the DCSD model achieves a greater speed-up the more deliberative information it can use. The Full DCSD model has a speed-up of 16.7 times when compared with the Original ASSOCC model. Table 6.21 shows the DCSD execution time in more detail. Based on these results we expect the deliberation bottleneck to be solved.

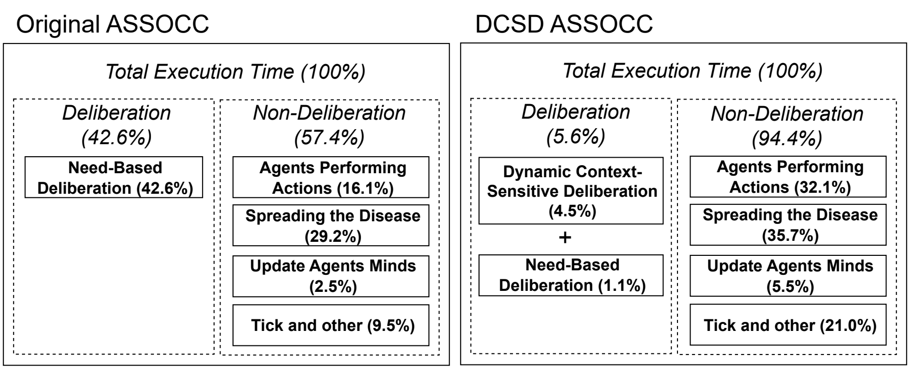


Figure 6.62: Original vs DCSD ASSOCC - Execution Time Percentages (random seed two and 350 households)

Figure 6.62 shows how the execution time percentages for the Original ASSOCC and the DCSD ASSOCC. The model on the left is the original ASSOCC model. The model on the right is the Full DCSD, DCSD ASSOCC. It shows how deliberation is no longer the main bottleneck in the DCSD ASSOCC model. It went from 42.6% to only 5.6% of the execution time. The fact that some numbers do not add up exactly comes from rounding. If one would like to increase the execution time of the model, the focus should now be on non-deliberation aspects as they take 94.4% of the total execution time in this specific example. To conclude, the deliberation bottleneck is effectively removed by using DCSD. The next section will measure whether this result holds for larger agent numbers as well.

6.6.2 Experiment 2: Increasing Number of Agents

In this section, Original ASSOCC and DCSD ASSOCC are compared with larger agent numbers. The number of agents are increased from about 1000 to about 10,000. The specific settings for this experiment are shown in Table 6.22. There are two experiment presets (ce-context-experiment-presets) for respectively Original ASSOCC and DCSD ASSOCC. The number of households (ce-households-for-context-scenario) are increased and generate a certain number of agents, which is respectively 1004, 2008, 4016, 6016, 8024 and 10,028 agents. These agent numbers are generated based on the household distribution in the ASSOCC model. Changing the random seed does not have impact on the number of starting agents generated. The run will last 240 ticks (which amounts to 60 days), it contains infections and does not have global lockdown. The runs are repeated five times which is reflected by using five different random seeds.

Name	**Value**
ce-context-experiment-presets	*0.1 Original ASSOCC*, *5.1 DCSD-5-optimisation*
ce-households-for-context-scenario	350 700 1400 2100 2800 3500
ce-enable-global-lockdown	false
with-infected?	true
stop-before-tick	241
random-seed	0 1 2 3 4

Table 6.22: Experimental setup for scalability

Deliberation Execution Time

Figure 6.63 shows for different agent numbers (x-axis) the execution time in ms of the deliberation (y-axis). Comparing the deliberation time of the Original ASSOCC and DCSD ASSOCC shows that DCSD ASSOCC retains its low execution time. Both lines are linear with an increase in the number of agents. However, DCSD ASSOCC has a much lower slope, resulting in a large time difference when compared to Original ASSOCC. For 10,028 agents the deliberation execution time is respectively, 54,941 ms versus 946,661 ms. Or 55 seconds versus 15.8 minutes. This is around a 16 times speed-up.

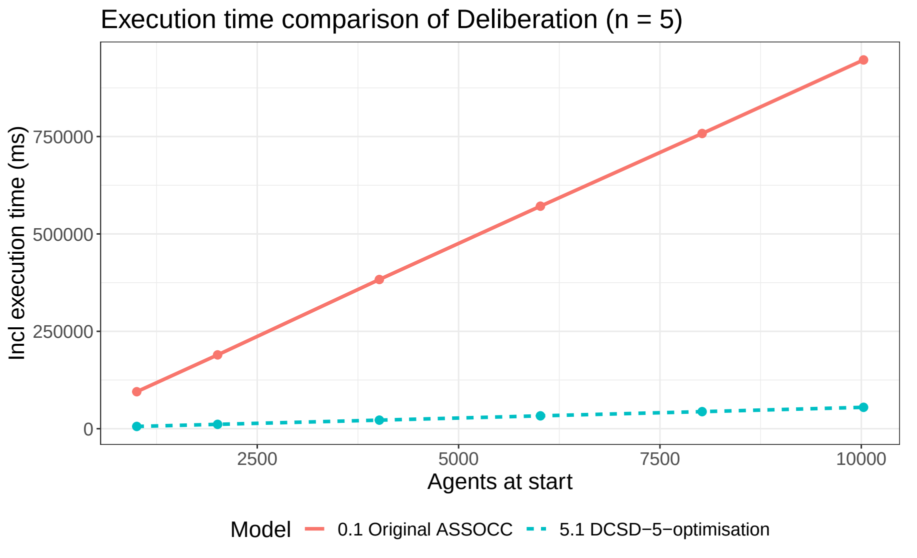


Figure 6.63: Deliberation execution time - Original VS DCSD

Table 6.23 shows the speed-up factor for each number of agents. The DCSD is generally about 16-17 times faster. Thus, from this data, it can be concluded that DCSD can make deliberation in the ASSOCC model significantly faster, even with large number of agents. The variation that can be seen between the run with 1004 agents and the other numbers is probably due to variations between runs. However, measuring 16 or 17 does not significantly affect the final conclusion, as both speed-up amounts are sufficient.

Agents	**Original ASSOCC**	**DCSD ASSOCC**	**Speed-up factor**
1004	94,977 ms	5864 ms	16.2
2008	189,516 ms	11,238 ms	16.9
4016	383,020 ms	22,048 ms	17.4
6016	571,268 ms	33,058 ms	17.3
8024	757,779 ms	43,843 ms	17.3
10,028	946,661 ms	54,941 ms	17.2

Table 6.23: Execution time with speed-up factor - Original VS DCSD

Deliberation vs Non-Deliberation

Figure 6.64 shows the deliberation time and non-deliberation time plotted over the number of agents for the DCSD ASSOCC model. It should be clear from this figure that the non-deliberation time is much higher than the deliberation time. Thus, also for larger agent numbers deliberation is no longer the bottleneck.

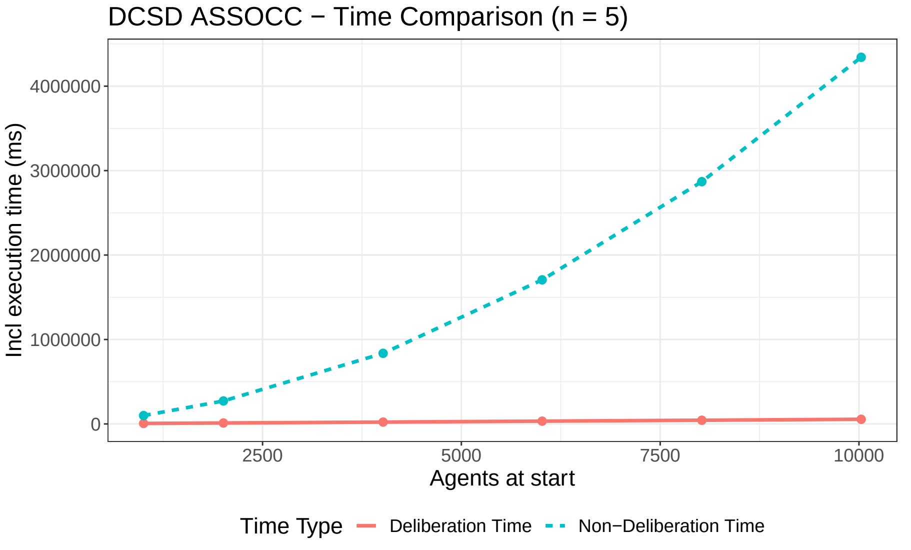


Figure 6.64: Non-deliberation versus deliberation time - DCSD ASSOCC

Now we will show how effective the use of DCSD is compared to Original ASSOCC in removing the bottleneck. By dividing the non-deliberation execution time by the deliberation execution time it becomes clear how much the non-deliberation time should be sped-up before being equal to the deliberation time. Figure 6.65 shows the result of this calculation. When simulating roughly 10,000 agents, the Original ASSOCC non-deliberation aspects can be sped up 8.7 times to make it equal to the deliberation time. This may seem like a lot; however, when using DCSD ASSOCC, the non-deliberation time has to be sped-up by 79.0 times to make it equal to the deliberation time.

6.6.3 Summary of Experiment 2

Figure 6.64 showed that both the Original ASSOCC and the DCSD ASSOCC model scale linearly. Table 6.23 shows that the speed-up factor stays as expected around 16-17 times for DCSD ASSOCC. This shows that DCSD scales well with an increasing number of agents. Considering the non-deliberation processes, it can be seen that these scale quadratically. Due to this difference

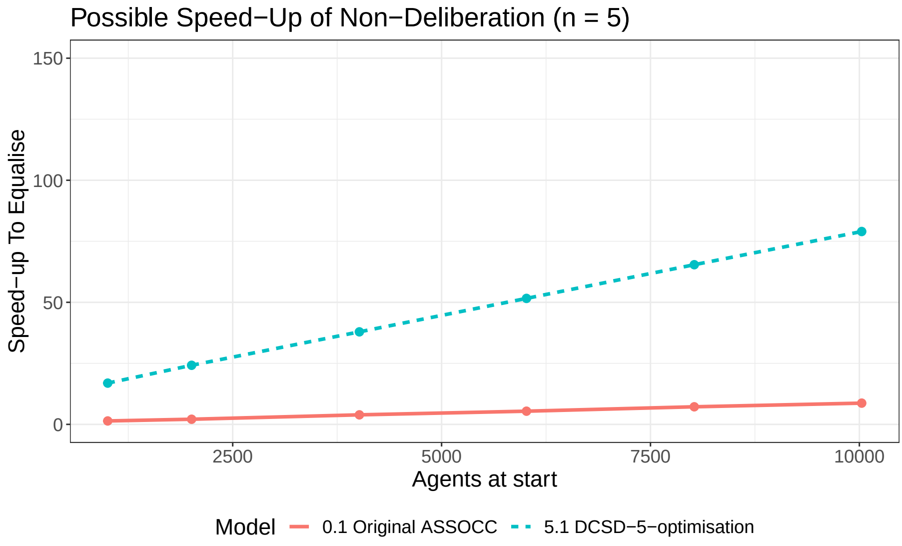


Figure 6.65: Required Non-deliberation Speed-Up

in computational complexity, the non-deliberation processes in this model do not scale well and become the bottleneck for larger agent numbers. In the end for roughly 10,000 agents the non-deliberation processes need to be scaled up by 79 times to be equal to the deliberation time. This experiment has shown that DCSD ASSOCC retains its speed-up advantage over Original ASSOCC. The deliberation bottleneck remains solved, even when increasing the number of agents to approximately 10,000. Theoretically, since the deliberation in both models scales linearly, this result is expected to be retained with even higher agent numbers.

6.6.4 Discussion: DCSD for Increased Scalability

As indicated in Figure 6.62, DCSD can effectively remove the deliberation bottleneck, since it speeds-up deliberation by about 16-17 times. This property also holds for larger agent numbers, as seen in Figure 6.63 and Table 6.23. Theoretically speaking, since both lines are linear, this speed-up will be retained for any number of agents.

Based on the empirical results, we now present an argument demonstrating how DCSD can scale deliberation in agent-based simulations that use an interdependent deliberation model. This argument is based on three cases, the simplest form of deliberation, an intermediate form using the empirical results of the ASSOCC comparison, and a hypothetical ASSOCC model with extra actions.

Case 1: DCSD in the Most Simple Deliberation Model

To assess the scalability of DCSD, we first examine the simplest deliberation model 6.66. Note that there are two actions; if one were to make a model with one action it can hardly be called a deliberation model since no deliberation has to be performed to choose that one action. This simplest deliberation model (on the left of the figure) has one parameter, i.e. whether the time is night, that is either true or false. If the time is night, the agent rests at home, if not, the agent works at the workplace. This is a very quick computation.

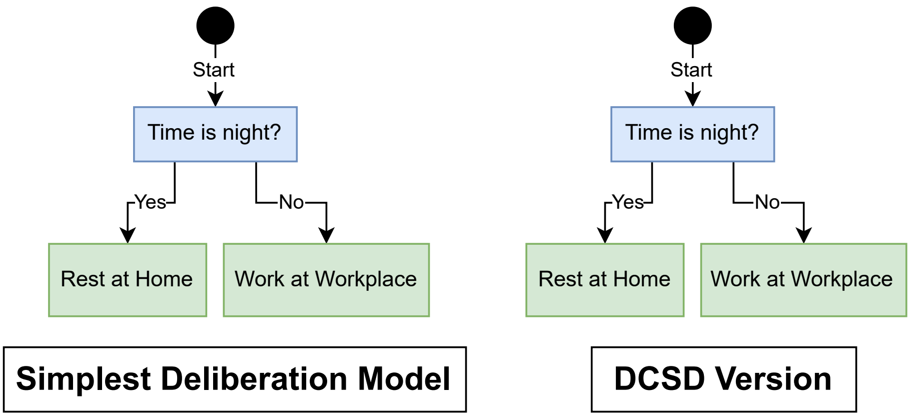


Figure 6.66: Simplest Deliberation Model vs DCSD Version

The DCSD variant of this model will look exactly the same (see on the right of the figure). While meta-deliberation is discussed in the framework, its implementation does not introduce additional computational overhead since meta-deliberation is not explicitly computed. Since the model can be implemented as a decision tree it can have the exact same form. Based on this, applying the DCSD framework to the most simple deliberation model is at least as efficient as that model.

Case 2: DCSD in ASSOCC

Applying DCSD to a more an interdependent deliberation model such as the need-based ASSOCC model can give a significant speed-up. The Original ASSOCC deliberation has many aspects that are all taken into account by need-based deliberation at the same time. The DCSD model has access to all information but usually only uses part of the information. Based on the empirical results shown in Table 6.23. DCSD can speed-up the Original ASSOCC model by about 16-17 times, even for larger agent numbers.

Case 3: DCSD in ASSOCC with more Actions

The ASSOCC need-based deliberation does not scale well when increasing the number of aspects. To calculate the best action the model takes the available actions, for example five actions when the agent has free choice. This number is then multiplied by two for adding the sub action of either social distancing or not social distancing, giving ten different actions. For each of these actions, a relatively time consuming need-based calculation is performed. There is no shortcut in the ASSOCC model except for hard cut-offs of actions at specific times, for example, no leisure activity or shopping during working hours. These actions are either considered and taken into account in the need-based calculation or are not considered at all.

If we were to extend the ASSOCC model with an extra sub action, for example, whether the agent should apply mouth masks or not, the deliberation time would be doubled. The five actions when the agent has free choice, would become ten by adding social distancing or not social distancing. Then it would become twenty by adding whether to use a mouth mask or not use a mouth mask. Since there are twenty need-based calculations instead of ten, the deliberation execution time would at least be twice as much. At least is states since it can be expected that need-based deliberation has to be expanded with a couple of extra conditions, these could take additional execution time. In the end, doubling the deliberation execution time would mean that for 10,000 agents, the time would go from 15.7 minutes, to roughly 31.4 minutes.

In the DCSD this works differently as it does not have to take into account all the information at all times. Adding the sub action of applying mouth mask can possibly be done in a similar way as performing social distancing or no social distancing. In the DCSD whether to do social distancing is calculated when the main action is selected. The calculation generally only considers two needs, risk-avoidance and, if the agent should stay at home, compliance. For applying a mouth mask it could also involve just checking a couple of needs and perhaps whether the agent believes that a mouth mask works. This will not cause a large increase in execution time, especially not doubling the execution time. It can be expected that this perhaps increases the DCSD execution time by 10%.

To summarise, the main difference when adding an action to the two deliberation models is the following. In the ASSOCC need-based deliberation model, an additional sub action can double the execution time. This is happening as it doubles the amount of actions that have to be considered by need-based deliberation. In the DCSD model, it can be expected that there is only a slight increase in execution time. The DCSD model only requires to calculate such sub action once, i.e. when the main action is selected. At that moment, only a couple of specific variables have to be considered to make a decision. Most variables are irrelevant to that specific choice anyway. For example, wearing the mouth mask does not do anything to needs such as financial safety or food safety and thus the extra information does not have to be considered. If the AS-

SOCC need-based deliberation time was doubled. And the DCSD time would only increase marginally, for example with 10%, the speed-up of DCSD would be even greater when compared with Original ASSOCC. DCSD is therefore expected to scale well, even when adding more deliberative aspects, such as actions, to the deliberation.

Conclusion: DCSD can Scale Deliberative Aspects

These three cases explain how DCSD can scale interdependent deliberative models on the number of aspects. 1) The simplest model is equal to the DCSD variant of that model. 2) As the number of aspects of the model increases, the speed-up becomes larger. As empirically shown, the ASSOCC need-based deliberation can be sped-up by about 16-17 times by using DCSD. This shows that DCSD can make deliberation more efficient. 3) It can be expected that in models that contain even more aspects than ASSOCC, DCSD can have even more speed-up benefits. By adding an additional sub action to the ASSOCC need-based deliberation, the deliberation time will double. While adding an additional sub action to the DCSD ASSOCC model, it can be expected that it only causes a marginal increase in execution time. Based on this argument, it can be expected that DCSD can be used to scale deliberative aspects in agent-based models that use interdependent deliberative models, such as ASSOCC.

6.6.5 Scalability Conclusion

This section showed the results of two experiments and discusses the results. The first experiment showed that the more complete the DCSD model, the larger the speed-up. The Full DCSD model speeds up deliberation in the ASSOCC model by about 16-17 times. This successfully removes the deliberation bottleneck in the ASSOCC model as it takes only 5.6% of the total execution time of the model when using Full DCSD.

The second experiment showed that the speed-up is retained even with larger agent numbers. There is a 16-17 times speed-up when simulating 1000 to 10,028 agents. Deliberation is no longer the bottleneck, also for higher agent numbers. The non-deliberation is now the main bottleneck and one would have to scale-up non-deliberation by 79.0 times to make it equal to the deliberation time.

In the end, it can be argued that DCSD can be expected to scale the number of deliberation aspects in interdependent deliberation in agent-based simulations. In a very simple model the DCSD will at least perform equally. As the number of deliberative aspects of the model increases the DCSD could perform better. In the ASSOCC framework, a framework with deliberation that contains many aspects, the DCSD made deliberation 16-17 times quicker. Even if the deliberative aspects in ASSOCC were expanded, it can be expected that DCSD scales well and would give an even greater speed-up compared to the all-evaluating need-based deliberation.

6.7 Conclusion of the Empirical Evaluation

In this chapter the Original ASSOCC framework and DCSD ASSOCC framework have been compared. These models have been compared based on the methods described in Chapter 4. The DCSD ASSOCC and Original ASSOCC models were compared in terms of behavioural output and the infection curve. By measuring with criteria and detailed time series it has been shown that the output of DCSD ASSOCC is similar enough to Original ASSOCC. The DCSD retains the realistic properties of ASSOCC's need-based deliberation in terms of agent behaviour and output.

In terms of scalability the DCSD can give a significant speed-up, about 16 times, over the Original ASSOCC deliberation. This speed-up solves the deliberation bottleneck as deliberation is now only using 5.6% of the total execution time when simulation about 1000 agents. The speed-up is retained when increasing the number of agents up to 10,000. Due to linear scaling of both models' deliberation, it can be expected that the same speed-up will be retained with even more agents, for example one million. It can even be expected that DCSD achieves a higher speed-up when applied to a model with even more deliberative aspects than the ASSOCC model. Due to significant speed-up results and the expected increase in speed-up when the deliberation contains more aspects, it can be claimed that DCSD can increase scalability in deliberation in agent-based simulation.

6.7.1 Answering Research Question 3: What are the trade-offs between scalability and realism in deliberation in an agent-based simulation?

This chapter provided interesting insights with regard to the trade-offs between scalability and realism in deliberation. The chapter initially showed that a very scalable model can be made by just using rigid habits (Section 6.1.1). This model is 123.5 times faster than the baseline deliberation model, the Original ASSOCC deliberation model. However, it cannot be considered realistic enough. The agents in the model do not respond to changes in their environment, such as being sick. When sick, the agents just continue their normal behaviour like nothing is going on. To conclude, one could say that this model gains a lot on the scalability but loses a lot on realism.

A more realistic model in this use case is the Habitual DCSD. This takes the advantages of habitual behaviour, i.e. specific contexts have a default action that can quickly be selected, and ties this to the need-based deliberation which can calculate an optimal action in all situations. This model is more scalable than Original ASSOCC, as it is twice as fast, but as argued retains the realism in the behaviour of the agents.

Adding more layers of DCSD can make the model even faster. If this is done carefully, the model is tweaked where necessary, and the behaviour of the agents can stay realistic enough. This has been shown in Section 6.2, where the

Strategic DCSD had to be adjusted such that very sick agents would not go to school, study, or work even though their autonomy was relatively high. This happened due to the DCSD model focusing on only the most salient need, which could be autonomy in some cases. While the need-based ASSOCC deliberation would consider all needs and see that the other needs such as health are very salient as well. The realistic behaviour was retained and the Strategic DCSD model became about three times faster.

Even at other layers such as the normative layer, it is important to not simplify too much. Section 6.3.1 shows what happens to the agent behaviour if a rigid normative model is added to the DCSD. The model is now almost ten times as quick, which is a large improvement compared to three times as fast with the Strategic DCSD. However, when inspecting the behaviour, the agents do behave undesirably during the global lockdown. The Rigid Normative DCSD causes agents to stay at home 100% of the time during the global lockdown. Causing the infection curve to die out. This is not realistic, as people in the real world would sometimes go out during the global lockdown, which retained the spread of the disease. The Normative DCSD was enabled with a more flexible normative deliberation and this model portrayed more realistic agent behaviour. The Normative DCSD retained the speed advantage, of nearly ten times as quick, while also portraying realistic enough agent behaviour.

Finally, the social layer was added for slightly more speed-up (Social DCSD, more than 10 times), and an optimisation was performed for even more speed-up (Full DCSD, about 16-17 times). This optimisation showed how a specific decision context that was frequently visited could be solved. This had a small impact on the behaviour of the agents while drastically increasing the speed up, from ten to sixteen times faster. To summarise, this chapter showed quite some considerations that have to be made in the scalability realism trade-off for deliberation in an agent-based simulation. It is easy to get a very quick model that is not so realistic. It is possible to get a realistic model that is not so scalable. However, striking the balance can be quite difficult; however, based on the results, the DCSD framework can definitely help in this regard.

Chapter 7

Conclusion

This thesis started out by describing that there is scalability vs realism trade-off in agent-based simulation. Simple agents are often more scalable, however they can lack behavioural aspects making them less realistic. More realistic agents, that do incorporate these crucial behavioural aspects are often not so scalable. This was seen in the ASSOCC framework, where deliberation is the main bottleneck. This bottleneck made it practically impossible to add more sub models and aspects to the framework.

One way to alleviate the trade-off between realism and scalability is the idea of context-sensitive deliberation. It provides more complex reasoning when needed (thus keeping realism), while saving on complexity in all cases when it is not needed (thus keeping scalability). Instead of a single complex deliberative algorithm, deliberation would slide from simple to more complex if needed. This is the complexity by need principle. Context-sensitive deliberation would use simple deliberation most of the time, making it efficient. If simple deliberation does not work, it will gradually slide to more complex deliberation and information. Determining when to use what kind of deliberation and information is determined by the context, hence the concept was called context-sensitive deliberation.

Intuitively, this approach increases scalability when the overhead of determining context and deliberation complexity is less than the gain in decreasing deliberation complexity in many cases. Of course, this also depends on the inherent complexity of the simulation. If the simulation does not require complex deliberation at all, the context-sensitive deliberation is only overhead. Similarly, if most of the time the agents need complex deliberation, the few times simple deliberation is successful will not compensate for the overhead of the context-sensitive deliberation. This dependency on the context in which the simulation is used means that the gain of the context-sensitive deliberation has to be empirically determined in some relevant cases.

To test whether context-sensitive deliberation can increase scalability while retaining realism, context-sensitive deliberation had to be empirically evaluated. Before it can be evaluated, it had to be formalised as there did not exist an appropriate context-sensitive deliberation framework. The formalisation of context-sensitive deliberation was given in Chapter 3, where **RQ1.1**, **RQ1.2**, and thus **RQ1** are answered. Chapter 4 provides the methods for testing context-sensitive deliberation. The ASSOCC framework was picked as a use case, and in the chapter, it is explained how realism and scalability are measured in the ASSOCC framework. Context-sensitive deliberation was implemented in the ASSOCC framework's deliberation function. This implementation is described in Chapter 5, which answers **RQ2**. Chapter 6, the evaluation, answers **RQ3** about the trade-offs in realism vs scalability. It shows that DCSD can retain realism and increase scalability in the ASSOCC framework. It is shown to retain realism by passing all the behavioural and infection curve criteria. DCSD is more scalable as it provided a 16-17 times speed-up on deliberation that is retained with higher agent numbers. It also explains how DCSD can be expected to scale even better in deliberative models that have more behavioural aspects than ASSOCC need-based deliberation.

7.1 Answering The Main Research Question

Since all other research questions have been answered, it is now possible to answer the main research question.

Can context-sensitive deliberation increase scalability while retaining realism in agent-based simulations?

To answer this, let us first state that the DCSD framework is a form of context-sensitive deliberation. This context-sensitive deliberation has been implemented and tested in the ASSOCC framework. The experiments and results showed that context-sensitive deliberation can increase scalability while retaining realism in an agent-based simulation. Now, it will be evaluated for both realism and scalability whether these results are generalisable to other agent-based simulations as well.

Realism can be retained, it was shown that a variety of behavioural aspects can be modelled in DCSD. The DCSD is shown to be capable of habitual behaviour (albeit without learning), rational choice behaviour in the form of comparing need levels, normative considerations, and performing actions based on the social network's preferred action. All of the DCSD versions were capable of simulating agent behaviour in ASSOCC that passes all criteria. In some specific cases, more information was required, although this did not negatively affect the execution time of the DCSD model by much.

Based on the previous paragraph, it can be expected that the DCSD model would perform similarly in terms of behaviour when applied to other agent-based simulations. If behavioural aspects can be implemented, they can also be included in the DCSD. If they do not work immediately, it is possible to add exceptions which usually do not drastically increase deliberation execution time. The DCSD is not limited to just the four CAFCA cells described in the use-case, but can also use information and deliberation types described in any of the other CAFCA cells. Since the framework contains such a wide range of behavioural aspects, it can be applied to many different agent-based simulations. Since it can use the social rules of other agent-based simulations, it can be expected to retain the realistic behaviour of those simulations.

The DCSD showed an increase in speed-up in the ASSOCC framework. As explained at the end of the Chapter 6, it can be expected to also be capable of speeding-up other agent-based deliberation systems. It can still work for simpler models, but the speed-up benefits would not be as great. And probably the efforts of making a DCSD version will not be worth it. However, for interdependent deliberation like ASSOCC, with even more aspects, it is expected to have even more speed-up. This makes DCSD scalable in terms of the number of deliberative aspects. Since DCSD is an relatively abstract framework it can be applied to many different agent-based simulations. In this sense, the result is generalisable to other agent-based simulations as long as the deliberation of that model has sufficient aspects. Otherwise, there is no benefit in using context-sensitive deliberation.

Given the above arguments, we can conclude that indeed context-sensitive deliberation can be used as a way to increase scalability while retaining realism in agent-based simulations. We have tested this extensively by implementing DCSD in ASSOCC which has agents with complex deliberations.

7.2 Limitations

There are some limitations related to this research that need to be discussed. The context-sensitive deliberation model will, for example, not be useful in every agent-based simulation. This section will discuss the limitations of the research conducted.

7.2.1 Cognitive Model

The formalisation of context-sensitive deliberation does not try to replicate the exact cognitive mechanism of sensing context in the human brain. The context-sensitive deliberation model is inspired by cognitive and behavioural science. The findings of these sciences, such as heuristics [35] and the thinking fast and thinking slow concept [46] serve as inspiration for our model. In agent-based simulation these concepts turn out to be useful, as the DCSD can increase scalability while retaining realism. However, it is not claimed that our context-sensitive deliberation framework proposes how actual humans deliberate using context. The purpose of context is to endow the framework with a sliding capability instead.

7.2.2 Scaling Number of Agents

It has been determined in Chapter 6 that DCSD can scale the number of deliberative aspects, however not much has been said on scaling of number of agents. It is not the case that DCSD directly scales the number of agents. However, it can be argued that an increase of agent numbers is possible by using the DCSD. Especially in systems where deliberation is by far the biggest bottleneck. The bigger the deliberation bottleneck, the more agents can be simulated by solving that bottleneck.

For example, if a system uses 95% of the execution time on deliberation and 5% on non-deliberation. By speeding up the deliberation by twenty times. The simulation could simulate more agents in the same amount of total execution time. If the goal is to purely scale the number of agents, it is perhaps better to use other scaling techniques. For example, scaling techniques related to actions and interactions in the environment could be considered. Some would perhaps suggest using High Performance Computing [12, 3, 53], however, one of the downsides is that it is not easy to investigate these simulations in runtime. These simulations are distributed on a high performance cluster that typically does not have an interface that can be used to inspect the simulation. Investigating a simulation during runtime, which is possible in Netlogo, enhances the debugging and development of the model. This could be desirable in a situation, such as policy testing for crises, where quick adjustments to the model and new results are required.

7.2.3 Serial vs Parallel Deliberation

A limitation of using DCSD would be that it will not provide benefit when agent deliberation is fully parallelised. This is the case in, for example, GPU-based agent frameworks [11]. These frameworks are capable of scaling agent-based simulations by using the GPU to parallelise all the agents' deliberation. Since the agent deliberation is happening in parallel rather than in series, this system can potentially scale agent-based simulations. The downside of performing deliberation in parallel is that the slowest deliberation cycle will determine the speed of the system as a whole. The strength of the DCSD gets almost completely negated by this. While most agents will have a quick deliberation, they would have to wait on the slowest deliberating agents. Even if there is only one slow agent out of a million, that one agent will still let the other 999,999 agents wait. DCSD would, however, work well in serialised deliberation (as was the case in ASSOCC) or in hybrid systems that use both serialised and parallel deliberation. In hybrid systems, when there is a cluster of slower deliberating agents, multiple faster deliberation agents can be run serially.

7.3 Future Work

This work contributed to both the expansion of knowledge on deliberation in agent-based social simulations and scalability in agent deliberation. The DCSD provides handles for implementing a deliberation system that can contain and combine different social and normative concepts. However, there are aspects that have not been investigated and could provide an interesting starting point for future research.

7.3.1 Increasing Realism with Context-Sensitive Deliberation

In the future we hope to expand the DCSD such that it can even increase the realism of behaviour. When considering the ASSOCC model, the agents are still quite rigid in their behaviour. Even if the ASSOCC agents can deviate from work by grocery shopping, there is still some human behaviour that is not modelled. For example, in real life, some people who work from home have more flexibility. They do not necessarily need to work nine to five. Some people can choose to sleep more in the morning, to work longer in the evening. Or, plan an activity in the afternoon, and then work in the evening or a bit on the weekend. This flexibility is not represented in the ASSOCC framework as the model could not be expanded.

For testing some policies, it may be of use to add such replanning of the day components. Imagine, for example, that an evening clock policy is tested. This policy was actually applied in the Netherlands, where people were not allowed to be on the streets between 21:00 and 04:30 unless they had specific permission. It can be expected that when introducing this policy that real people will actually adjust their daily schedule, especially those with flexible working hours. To realistically simulate the effects of this policy on the population, it can be argued that agents should be capable of changing their schedule. As explained above, this flexibility is not represented in the ASSOCC framework. It would require expanding the deliberative model, which was not practically possible before. The DCSD opens up this option as it makes the inclusion of deliberative aspects more scalable. Thus, these extra components such as planning could be added.

The DCSD already contains goals and plans in the meta-deliberation tuple. However, more research is required on how to formalise the combination of goals and plans with other behavioural aspects.

7.3.2 Expanding Context-Sensitive Deliberation to other Agent-Based Simulations

DCSD ASSOCC contains information and deliberation types from four of the nine CAFCA cells. The other cells can be just as relevant to include in the DCSD model if the model requires. The relevance of these other cells becomes apparent when looking at the literature. In Elsenbroich and Verhagen [30] it is described how agent-based deliberation frameworks fit in the CAFCA matrix. Consumat [40] considers information from the repetition, imitation, rational choice, and game theory cells. Multi-Agent systems consider institutional rules. The EMIL architecture [13], considers social norms. Joining-in can be represented by social practices and team reasoning by collective reasoning. The article [29] does not provide a framework that uses moral values. Values are still important in agent-based simulation for some purposes, they are considered in, for example, Heidari et al. [37] and Dechesne et al. [15]. The work by Wijermans and Verhagen [65] describes an agent framework to understanding common pool resource management, this framework uses all aspects on the strategic layer.

In the future, it could be relevant to include information and deliberation types from other cells as well in the DCSD. This will allow to create context-sensitive deliberation for agent models making use of these types of deliberations as well. Especially when the use of agent-based simulation for policy testing will grow. It can be expected that these social aspects must also be modelled.

7.4 Closing Remarks

This thesis has presented a formalisation of context-sensitive deliberation. It showed which aspects are necessary for context-sensitive deliberation and it described the Dynamic Context-Sensitive Deliberation framework. The DCSD framework has been implemented in the ASSOCC deliberation to evaluate context-sensitive deliberation. The Original ASSOCC model has been compared with the DCSD ASSOCC model to evaluate the trade-offs in terms of realism and scalability.

Based on these results, the main research question could be answered. Context-sensitive deliberation can increase scalability while retaining. The behaviour of the ASSOCC agents and the infection curve stayed similar and between the boundaries of the criteria. Therefore, the realism of the model is retained. Scalability is mainly tested in terms of deliberation execution time. The DCSD managed to speed-up the deliberation by about 16-17 times even for larger agent numbers. The DCSD is then argued to scale well with the number of deliberation aspects taken into account by the deliberation. In conclusion, context-sensitive deliberation can increase scalability while retaining realism in agent-based simulations.

Bibliography

[1] Patrycja Antosz et al. "Simulation model implementing different relevant layers of social innovation, human choice behaviour and habitual structures". In: *SMARTEES Deliverable* (2019).

[2] Tina Balke and Nigel Gilbert. "How do agents make decisions? A survey". In: *Journal of Artificial Societies and Social Simulation* (2014).

[3] Parantapa Bhattacharya et al. "PanSim+ Sim-2APL: A framework for large-scale distributed simulation with complex agents". In: *International Workshop on Engineering Multi-Agent Systems*. Springer. 2021, pp. 1–21.

[4] Ken Binmore. *Game theory: a very short introduction*. OUP Oxford, 2007.

[5] André B Bondi. "Characteristics of scalability and their impact on performance". In: *Proceedings of the 2nd international workshop on Software and performance*. 2000, pp. 195–203.

[6] Michael Bratman. "Intention, plans, and practical reason". In: (1987).

[7] Fred Brauer. "Compartmental models in epidemiology". In: *Mathematical epidemiology* (2008), pp. 19–79.

[8] Jan Broersen et al. "Goal generation in the BOID architecture". In: *Cognitive Science Quarterly* 2.3-4 (2002), pp. 428–447.

[9] Jan Broersen et al. "The BOID architecture: conflicts between beliefs, obligations, intentions and desires". In: *Proceedings of the fifth international conference on Autonomous agents*. 2001, pp. 9–16.

[10] Cristiano Castelfranchi et al. "Deliberative normative agents: Principles and architecture". In: *Intelligent Agents VI. Agent Theories, Architectures, and Languages: 6th International Workshop, ATAL'99, Orlando, Florida, USA, July 15-17, 1999. Proceedings 6*. Springer. 2000, pp. 364–378.

[11] Siddharth Chaturvedi, Ahmed El-Gazzar, and Marcel van Gerven. "Foragax: An Agent-Based Modelling Framework Based on JAX". In: *arXiv preprint arXiv:2409.06345* (2024).

[12] Nicholson Collier and Michael North. "Parallel agent-based simulation with repast for high performance computing". In: *Simulation* 89.10 (2013), pp. 1215–1235.

[13] Rosaria Conte, Giulia Andrighetto, and Marco Campennl. *Minding norms: Mechanisms and dynamics of social order in agent societies.* OUP Us, 2014.

[14] Mehdi Dastani. "2APL: a practical agent programming language". In: *Autonomous agents and multi-agent systems* 16 (2008), pp. 214–248.

[15] Francien Dechesne et al. "No smoking here: values, norms and culture in multi-agent systems". In: *Artificial intelligence and law* 21 (2013), pp. 79–107.

[16] Anind K Dey. "Understanding and using context". In: *Personal and ubiquitous computing* 5 (2001), pp. 4–7.

[17] Frank Dignum. "Foundations of social simulations for crisis situations". In: *Social simulation for a crisis: Results and lessons from simulating the COVID-19 crisis.* Springer, 2021, pp. 15–37.

[18] Frank Dignum. "Should we make predictions based on social simulations?" In: *International Journal of Social Research Methodology* 26.2 (2023), pp. 193–206.

[19] Frank Dignum. *Social Simulation for a Crisis.* Springer, 2021.

[20] Frank Dignum, Virginia Dignum, and Catholijn M Jonker. "Towards agents for policy making". In: *Multi-Agent-Based Simulation IX: International Workshop, MABS 2008, Estoril, Portugal, May 12-13, 2008, Revised Selected Papers 9.* Springer. 2009, pp. 141–153.

[21] Virginia Dignum and Frank Dignum. "Agents are dead. Long live agents!" In: *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems.* 2020, pp. 1701–1705.

[22] Virginia Dignum, Nigel Gilbert, and Michael P Wellman. "Introduction to the special issue on autonomous agents for agent-based modeling". In: *Autonomous Agents and Multi-Agent Systems* 30 (2016), pp. 1021–1022.

[23] Barbara Dunin-Keplicz and Rineke Verbrugge. *Teamwork in multi-agent systems: A formal approach.* Vol. 21. John Wiley & Sons, 2011.

[24] Bruce Edmonds. "Complexity and Context-Dependency". In: *Foundations of Science* 18 (2013), pp. 745–755.

[25] Bruce Edmonds. "Context in social simulation: why it can't be wished away". In: *Computational and Mathematical Organization Theory* 18 (2012), pp. 5–21.

[26] Bruce Edmonds. "The pragmatic roots of context". In: *International and Interdisciplinary Conference on Modeling and Using Context.* Springer. 1999, pp. 119–132.

[27] Bruce Edmonds and Lia ní Aodha. "Using agent-based modelling to inform policy–what could possibly go wrong?" In: *Multi-Agent-Based Simulation XIX: 19th International Workshop, MABS 2018, Stockholm, Sweden, July 14, 2018, Revised Selected Papers 19*. Springer. 2019, pp. 1–16.

[28] Bruce Edmonds and Emma Norling. "Integrating learning and inference in multi-agent systems using cognitive context". In: *International Workshop on Multi-Agent Systems and Agent-Based Simulation*. Springer. 2006, pp. 142–155.

[29] Corinna Elsenbroich and Harko Verhagen. "Integrating CAFCA—A Lens to Interpret Social Phenomena". In: *Conference of the European Social Simulation Association*. Springer. 2019, pp. 161–167.

[30] Corinna Elsenbroich and Harko Verhagen. "The simplicity of complex agents: a Contextual Action Framework for Computational Agents". In: *Mind & Society* 15 (2016), pp. 131–143.

[31] Joshua M Epstein. "Why model?" In: *Journal of artificial societies and social simulation* 11.4 (2008), p. 12.

[32] Marc Esteva et al. "On the formal specification of electronic institutions". In: *Agent mediated electronic commerce*. Springer, 2001, pp. 126–147.

[33] Peter C Fishburn. "Utility theory". In: *Management science* 14 (5 1968), pp. 335–378.

[34] Gerd Gigerenzer and Wolfgang Gaissmaier. "Heuristic decision making". In: *Annual review of psychology* 62.2011 (2011), pp. 451–482.

[35] Gerd Gigerenzer, Ralph Hertwig, and Thorsten Pachur. *Heuristics: The Foundations of Adaptive Behavior*. Oxford University Press, 2011.

[36] Anisha Gupta, Rivana Christie, and R Manjula. "Scalability in internet of things: features, techniques and research challenges". In: *Int. J. Comput. Intell. Res* 13.7 (2017), pp. 1617–1627.

[37] Samaneh Heidari, Maarten Jensen, and Frank Dignum. "Simulations with values". In: *Advances in Social Simulation*. Springer, 2020, pp. 201–215.

[38] Mark D Hill. "What is scalability?" In: *ACM SIGARCH Computer Architecture News* 18.4 (1990), pp. 18–21.

[39] Robert Hinch et al. "Effective configurations of a digital contact tracing app: a report to NHSX". In: *Retrieved July* 23 (2020), p. 2020.

[40] Wander Jager and Marco Janssen. "The need for and development of behaviourally realistic agents". In: *Multi-Agent-Based Simulation II: Third International Workshop, MABS 2002 Bologna, Italy, July 15–16*. Springer. 2003, pp. 36–49.

[41] Maarten Jensen, Loïs Vanhée, and Frank Dignum. "Dynamic Context-Sensitive Deliberation". In: *Multi-Agent-Based Simulation, MABS 2023*. 2023.

[42] Maarten Jensen, Loïs Vanhée, and Frank Dignum. "Dynamic Context-Sensitive Deliberation for Scalability in Realistic Social Simulations". In: *Advances in Social Simulation*. Springer, 2023.

[43] Maarten Jensen, Loïs Vanhée, and Frank Dignum. "Towards Realism for Policy Testing". In: *Advances in Social Simulation*. arXiv, 2024.

[44] Maarten Jensen, Loïs Vanhée, and Christian Kammler. "Social Simulations for Crises: From Theories to Implementation". In: *Social Simulation for a Crisis*. Springer, 2021, pp. 39–84.

[45] Maarten Jensen et al. "Towards Efficient Context-Sensitive Deliberation". In: *Advances in Social Simulation*. Springer, 2022, pp. 409–421.

[46] Daniel Kahneman. *Thinking, fast and slow*. Macmillan, 2011.

[47] William Ogilvy Kermack and Anderson G McKendrick. "A contribution to the mathematical theory of epidemics". In: *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character* 115.772 (1927), pp. 700–721.

[48] Boicho Kokinov and Maurice Grinberg. "Simulating context effects in problem solving with AMBR". In: *International and Interdisciplinary Conference on Modeling and Using Context*. Springer. 2001, pp. 221–234.

[49] F Lorig, E Johansson, and P Davidsson. "Agent-based social simulation of the COVID-19 pandemic: A systematic review". In: *JASSS: Journal of Artificial Societies and Social Simulation* 24 (3 2021).

[50] William F McColl. "Scalable computing". In: *Computer Science Today: Recent Trends and Developments* (2005), pp. 46–61.

[51] René Mellema, Maarten Jensen, and Frank Dignum. "Social rules for agent systems". In: *International Workshop on Coordination, Organizations, Institutions, Norms, and Ethics for Governance of Multi-Agent Systems*. Springer. 2020, pp. 175–180.

[52] Jan de Mooij et al. "A framework for modeling human behavior in large-scale agent-based epidemic simulations". In: *Simulation* 99.12 (2023), pp. 1183–1211.

[53] Jan de Mooij et al. "Quantifying the effects of norms on COVID-19 cases using an agent-based simulation". In: *International Workshop on Multi-Agent Systems and Agent-Based Simulation*. Springer. 2021, pp. 99–112.

[54] Hannah Muelder and Tatiana Filatova. "One theory-many formalizations: testing different code implementations of the theory of planned behaviour in energy agent-based models". In: *Jasss* (2018).

[55] Anand S Rao, Michael P Georgeff, et al. "BDI agents: from theory to practice." In: *Icmas*. Vol. 95. 1995, pp. 312–319.

[56] Diogo Rato and Rui Prada. "Towards social identity in socio-cognitive agents". In: *Sustainability* 13.20 (2021), p. 11390.

[57] Bastin Tony Roy Savarimuthu and Stephen Cranefield. "Norm creation, spreading and emergence: A survey of simulation models of norms in multi-agent systems". In: *Multiagent and Grid Systems* 7.1 (2011), pp. 21–54.

[58] Thomas C Schelling. "Models of segregation". In: *The American economic review* 59.2 (1969), pp. 488–493.

[59] Alastair Smith and Federico Varese. "Payment, protection and punishment: The role of information and reputation in the Mafia". In: *Rationality and Society* 13.3 (2001), pp. 349–393.

[60] Flaminio Squazzoni, Wander Jager, and Bruce Edmonds. "Social simulation in the social sciences: A brief overview". In: *Social Science Computer Review* 32.3 (2014), pp. 279–294.

[61] Robert Sugden. "The logic of team reasoning". In: *Philosophical explorations* (3 2003), pp. 165–181.

[62] Jason Thompson et al. "Modelling SARS-CoV-2 disease progression in Australia and New Zealand: an account of an agent-based approach to support public health decision-making". In: *Australian and New Zealand Journal of Public Health* 46.3 (2022), pp. 292–303.

[63] Loïs Vanhée. "Engineering Social Simulations for Crises". In: *Social Simulation for a Crisis: Results and Lessons from Simulating the COVID-19 Crisis*. Springer, 2021, pp. 353–378.

[64] Michael P Wellman. "Putting the agent in agent-based modeling". In: *Autonomous Agents and Multi-Agent Systems* 30 (2016), pp. 1175–1189.

[65] Nanda Wijermans and Harko Verhagen. "Fishing Together? Exploring the Murky Waters of Sociality". In: *International Workshop on Multi-Agent Systems and Agent-Based Simulation*. Springer. 2021, pp. 180–193.

[66] Renyu Yang and Jie Xu. "Computing at massive scale: Scalability and dependability challenges". In: *2016 IEEE symposium on service-oriented system engineering (SOSE)*. IEEE. 2016, pp. 386–397.

[67] Andreas Zimmermann, Andreas Lorenz, and Reinhard Oppermann. "An operational definition of context". In: *International and Interdisciplinary Conference on Modeling and Using Context*. 2007, pp. 558–571.

Chapter 8

Appendix

This appendix will show some supporting material for the thesis. Section 8.1 will show the experimental settings for the DCSD ASSOCC model. Section 8.2 shows the results of an optimisation analysis performed on the DCSD.

8.1 Experimental Settings

This section of the appendix will shortly explain how to setup the model and mention the general parameter settings for the experiments in this thesis. The ASSOCC model contains many adjustable parameters. Many of them are not relevant enough to be mentioned in the main text of this thesis. This section will explain the parameters that are of influence on the model, including those that could not have been mentioned in the main text.

8.1.1 Running the Model

The DCSD ASSOCC model can be downloaded at `https://github.com/maartenjensen/ASSOCC-context`. This includes the Original ASSOCC model. The model requires Netlogo version 6.1.1 to run properly, which is available at `https://ccl.northwestern.edu/netlogo/`. The `covid-sim.nlogo` file should be opened which can be found in the `ASSOCC-context-main/simulation_model` directory. The *preset-scenario* variable should be set to *context-assocc*, the *ce-context-esperiment-presets* can be set to the desired preset. The *load & setup* button initialises the model. To run the model, one of the buttons in the interface can be used, e.g. the *8 Weeks Run* button.

8.1.2 The Behaviour Space Settings

The behaviour space of Netlogo can automatically execute runs with varying variables. The following variables are the more important variables for the experiments in this thesis. These variables are mostly mentioned throughout the chapters, when an experiment is described. In this section of the Appendix, these variables and their default setting are quickly mentioned.

ce-context-experiment-presets indicates the experimental preset. This variable is the most influential variable in the DCSD ASSOCC implementation. Namely, this variable is used in the setup of the model and influences some other variable. This variable can adjust which deliberative model is used, whether infected are simulated, whether there is a global lockdown, and more. By default it is set at *0.1 Original ASSOCC*. This variable and setting up the model are described in more detail in the next section.

#**random-seed** indicates the seed for the stochastic function that is used for this particular run. Using a different random seed between runs, the stochastic functions return different results. This will make the run unique when compared to another run that has the same parameters, but a different random seed. Changing the random seed allows for simulation of multiple different runs that have the same base settings. This can be used to solidify results, to not just be dependent on a single run. The default is two, as this gave some representative behavioural and infection curve runs.

ce-households-for-context-scenario indicates the number of households. The number of households determines the number of agents in the simulation.

The default value is 350 which generates 1004 independent of the random seed. Other variables used here are 700, 1400, 2100, 2800, and 3500. These generate, respectively, 2008, 4016, 6016, 8024, and 10,028 agents.

ce-actions-space indicates the number of available actions for deliberation system to deliberate on. This variable was used to test the effect on the execution time when the number of actions was increased [42]. The default is six, which allows all actions. In this thesis, this variable will not be adjusted.

ce-testing-action-disabled? indicates whether the get-tested action is excluded or included in need-based deliberation. This variable was added for the scalability experiments. In the scalability experiments it is important not to let the need-based deliberation, deliberate on extra actions otherwise the comparison with DCSD would not be fair. For the realism results, having these get-tested actions included or excluded does not matter. It does not matter since the get-tested action is never selected, since all testing parameters of the model are disabled. By default, the get-tested variable is true.

ce-need-salient-threshold indicates whether a need in DCSD is considered salient. A need is salient if the need level is below this threshold. This is by default set to 0.5 and is not changed for any of the experiments.

ce-need-critical-threshold indicates whether a need in DCSD is considered critical. A need is critical if the need level is below this threshold. This is by default set to 0.1 and is not changed for any of the experiments.

ce-risk-avoidance-threshold-for-sd is used to manipulate the amount of social distancing. Adjusting this parameter can influence the steepness of the infection curve. For most actions, if the risk avoidance satisfaction of an agent is below this threshold, the agent will apply social distancing. Otherwise, the agent will not apply social distancing. The value used for all runs is 0.78.

ce-compliance-quarantine-threshold-for-sd is also used to manipulate the amount of social distancing. Adjusting this parameter can influence the steepness of the infection curve. For most actions, if the agent should stay at home, the compliance need will be checked using this parameter. If the compliance satisfaction level is below 0.57 the agent will social distance, otherwise the agent will not social distance.

ce-private-leisure-by-risk can enable a preference for private leisure activities over public leisure activities. This was the case in the Original ASSOCC model, where private leisure activities are performed about 2-3 times more frequently than public leisure activities. Dependent on the risk-avoidance need the **ce-risk-avoidance-private-leisure-preference** parameter determines the threshold, which is set to 0.65. When the agent select leisure in the DCSD, first it will be checked whether the risk-avoidance need is below the threshold. If this is the case DCSD will select the leisure at private leisure activity. If not the leisure at private or public leisure will be determined randomly.

ce-risk-avoidance-home-preference is a parameter for action selection in the free time decision tree for a salient belonging need. Resting at home or the two leisure activities can all give belonging. However, if the risk avoidance level is lower than a certain threshold. The agent will choose resting at home as action. The default of this parameter is 0.5.

ce-enable-need-balancing was added to toggle need balancing for DCSD. Need balancing proved to be not that useful in improving the behaviour of the agents in the model, unless it would be done very extensively. Since the need balancing was not used in the DCSD versions, the variable is set to false.

ce-log-agent and **ce-log-agent-id** are used for logging purposes during individual runs when inspecting the model. These should be disabled when doing batch runs, as the logging may slow down execution time and influence the scalability results. To disable logging, **ce-log-agent** is set to false. **ce-log-agent-id** is set to 596 as a default for when individual runs are inspected.

8.1.3 Experiment Presets

The section above showed many parameters that are set using the behaviour space. One of those parameters, i.e. *ce-context-experiment-presets* determines additional settings, it mainly influences which deliberation model settings are used (see `scenarios.nls`). This section will first show the parameters that it can influence. Secondly, it will show for each preset which settings these parameters will have.

with-infected? determines whether there are infected agents enabled in this run. By default this is set to false, although in most experiments the infected are enabled by setting them to true.

ce-enable-global-lockdown determines whether global lockdown is active in the model. The full functioning of global lockdown in the ASSOCC framework is described in [44]. By default, global lockdown is not active, the variable is set to false. This sets the **stop-before-tick** variable to 241, meaning the simulation will run for 240 ticks, or 60 days. If global lockdown is enabled, set to true. The **stop-before-tick** variable will be set to 481, meaning the simulation will run for 480 ticks, or 120 days. This is done since when global lockdown is enabled, it will take more time for the spread of the disease to be over.

ce-context-depth determines which deliberation model is used. By default this is set to 0, which represents the Original ASSOCC deliberation. If this is set to a value of 1 to 4, one of DCSD models is used for deliberation. Each number 1 to 4, represents the number of information layers activated in the DCSD. If this is set to -1, an alternative deliberation model is used.

ce-forced-habits determines whether the rigid habits deliberation model is enabled (set to true). **ce-forced-habits-level** determines which level of habitual deliberation is active. By default, the latter is set to 0, but it is set to 1 when rigid habits are enabled.

ce-leisure-habits is a setting used in the DCSD model. It is described in Section 6.1.4. If enabled, it will add preplanned moments of leisure activities to the deliberation model. By default, it is set to false.

ce-only-obligation-when-health-riskfree-enough is a setting used in the DCSD model. It is described in Section 6.2.2. If enabled, it will do an additional check that agents should be healthy and have a high enough risk-avoidance level to perform their obligated activities. By default, it is set to false.

ce-should-rigidly-follow-quarantine is a setting used in the DCSD model. It is described in Section 6.3.1. If enabled, the agents will never break quarantine. By default, this is set to false.

ce-enable-salient-food-luxury-forced-obligation is a setting used by the DCSD. This setting represents the optimisation done to the DCSD model, fully described in Section 5.5. By default, this is set to false.

The frequency of essential shopping has been slightly adjusted for the DCSD model. By default in the Original ASSOCC model **ce-more-likely-to-essential-shop** is set to false, and the **days-of-rations-bought** is set to 3. In DCSD ASSOCC and the other deliberation models, **ce-more-likely-to-essential-shop** is set to true and **days-of-rations-bought** is set to 4. These parameters prevents many agents that have to work from shopping on Monday morning. This happened due to the shops being closed on Sunday, this lead many agents to perform essential shopping on Monday since their food-safety need was very salient.

Original ASSOCC Presets

This section shows the settings for the Original ASSOCC presets. If the preset is set to *0.0 Original ASSOCC-no-infections*, none of the before mentioned parameters change. Table 8.1 shows the changes to the parameters for the *0.1 Original ASSOCC* preset.

Name	Value
with-infected?	true

Table 8.1: Settings for preset: *0.1 Original ASSOCC*

Table 8.1 shows the changes to the parameters for the *0.2 Original ASSOCC-lockdown* preset.

Name	Value
with-infected?	true
ce-enable-global-lockdown	true

Table 8.2: Settings for preset: *0.2 Original ASSOCC-lockdown*

Habitual Deliberation Presets

This section shows the settings for the habitual deliberation presets. This section shows the settings for both the rigid habitual deliberation model and the Habitual DCSD. Table 8.3 shows the settings for *1.1 rigid-habits-no-infected* the preset. Table 8.4 shows the *1.2 rigid-habits-infected* preset.

Name	Value
ce-context-depth	-1
ce-forced-habits	true
ce-forced-habits-level	1

Table 8.3: Settings for preset: *1.1 rigid-habits-no-infected*

Name	Value
with-infected?	true
ce-context-depth	-1
ce-forced-habits	true
ce-forced-habits-level	1

Table 8.4: Settings for preset: *1.2 rigid-habits-infected*

Table 8.5 shows the settings for *1.3 DCSD-1* the preset. Table 8.6 shows the *1.4 DCSD-1-leisure-habits* preset.

Name	Value
ce-context-depth	1

Table 8.5: Settings for preset: *1.3 DCSD-1*

Name	Value
with-infected?	true
ce-context-depth	1
ce-leisure-habits	true

Table 8.6: Settings for preset: *1.4 DCSD-1-leisure-habits*

Strategic Deliberation Presets

This section shows the settings for the strategic deliberation presets, i.e. Strategic DCSD. Table 8.7 shows the settings for *2.1 DCSD-2* the preset. Table 8.8 shows the *2.2 DCSD-2-obligation-constraint* preset.

Normative Deliberation Presets

This section shows the settings for the normative deliberation presets, i.e. (Rigid) Normative DCSD. Table 8.9 shows the settings for *3.1 DCSD-3-rigid-norms* the preset. Table 8.10 shows the *3.2 DCSD-3-rigid-norms-lockdown* preset.

Name	Value
with-infected?	true
ce-context-depth	2
ce-leisure-habits	true

Table 8.7: Settings for preset: *2.1 DCSD-2*

Name	Value
with-infected?	true
ce-context-depth	2
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true

Table 8.8: Settings for preset: *2.2 DCSD-2-obligation-constraint*

Name	Value
with-infected?	true
ce-context-depth	3
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true
ce-should-rigidly-follow-quarantine	true

Table 8.9: Settings for preset: *3.1 DCSD-3-rigid-norms*

Name	Value
with-infected?	true
ce-enable-global-lockdown	true
ce-context-depth	3
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true
ce-should-rigidly-follow-quarantine	true

Table 8.10: Settings for preset: *3.2 DCSD-3-rigid-norms-lockdown*

This section shows the settings for the normative deliberation presets, i.e. (Rigid) Normative DCSD. Table 8.11 shows the settings for *3.3 DCSD-3* the preset. Table 8.12 shows the *3.4 DCSD-3-lockdown* preset. The *ce-should-rigidly-follow-quarantine* is set to false again, since this is the default.

Name	Value
with-infected?	true
ce-context-depth	3
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true

Table 8.11: Settings for preset: *3.3 DCSD-3*

Name	Value
with-infected?	true
ce-enable-global-lockdown	true
ce-context-depth	3
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true

Table 8.12: Settings for preset: *3.4 DCSD-3-lockdown*

Social Deliberation Presets

This section shows the settings for the social deliberation preset, i.e. Social DCSD. Table 8.13 shows the settings for *4.1 DCSD-4* the preset.

Name	Value
with-infected?	true
ce-context-depth	4
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true

Table 8.13: Settings for preset: *4.1 DCSD-4*

DCSD ASSOCC Presets

This section shows the settings for the optimised DCSD presets. With these presets the Full DCSD is activated, this is the DCSD ASSOCC model. Table 8.14 shows the settings for the *5.0 DCSD-5-optimisation-no-infections* preset.

Name	Value
ce-context-depth	5
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true
ce-enable-salient-food-luxury-forced-obligation	true

Table 8.14: Settings for preset: *5.0 DCSD-5-optimisation-no-infections*

Table 8.15 shows the settings for the *5.1 DCSD-5-optimisation* preset. Table 8.16 shows the settings for the *5.2 DCSD-5-optimisation-lockdown* preset. This preset has global lockdown enabled.

Name	**Value**
with-infected?	true
ce-context-depth	5
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true
ce-enable-salient-food-luxury-forced-obligation	true

Table 8.15: Settings for preset: *5.1 DCSD-5-optimisation*

Name	**Value**
with-infected?	true
ce-enabled-global-lockdown	true
ce-context-depth	5
ce-leisure-habits	true
ce-only-obligation-when-health-riskfree-enough	true
ce-enable-salient-food-luxury-forced-obligation	true

Table 8.16: Settings for preset: *5.2 DCSD-5-optimisation-lockdown*

8.2 ASSOCC DCSD Optimisation Analysis

This section shows the results of the optimisation analysis described in Section 5.5. Table 8.17 shows the number of calls for each situation in which need-based deliberation is called. The first four functions are responsible for about 82% of the calls. Those are the CSO-FOOD-SAFETY-F, CSO-LUXURY-F, CSOWH-FOOD-SAFETY-F and CSOWH-LUXURY-F functions. They actually occur in situations where the worker agents have a salient luxury or food safety need. In most cases, the agents should actually work an ignore the salient luxury and food-safety needs, unless the salient need is critical. This gave the opportunity to optimise the obligation and obligation work home decision tree. The other 17 decision situations only account for 18% of the calls. These do not have to be considered for optimising as the effect will only be marginal.

Function name	*Calls*	*Calls %*
CSOWH-FOOD-SAFETY-F	**3504**	**26.98%**
CSOWH-LUXURY-F	**3163**	**24.36%**
CSO-FOOD-SAFETY-F	**2247**	**17.30%**
CSO-LUXURY-F	**1772**	**13.65%**
CSO-CONFORMITY-F	412	3.17%
CSFT-CONFORMITY-F	403	3.10%
CSOWH-RISK-AVOIDANCE-SLEEP-F	312	2.40%
CSOWH-CONFORMITY-F	278	2.14%
CSN-CONFORMITY-F	167	1.29%
CSFT-LUXURY-F	128	0.99%
CSFT-HABIT-LEISURE-F	118	0.91%
CSSFT-FOOD-SAFETY-F	107	0.82%
CSSOWH-FOOD-SAFETY-F	97	0.75%
CSSOWH-AUTONOMY-FINANCIAL-F	70	0.54%
CSOWH-BELONGING-F	69	0.53%
CSOWH-AUTONOMY-F	61	0.47%
CSSO-FOOD-SAFETY-F	47	0.36%
CSSOWH-CONFORMITY-F	12	0.09%
CSFT-FOOD-SAFETY-F	11	0.08%
CSSO-CONFORMITY-F	4	0.03%
CSSFT-CONFORMITY-F	3	0.02%
Total:	**12,985**	**100.00%**

Table 8.17: Function calls for the use of need-based deliberation in DCSD.