Die approbierte Originalversion dieser Diplom-/ Masterarbeit ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.

http://www.ub.tuwien.ac.at

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology.

http://www.ub.tuwien.ac.at/eng

TECHNISCHE UNIVERSITÄT WIEN

Diplomarbeit

SimSchool - Optimierung der Abläufe in Schulbauten

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Diplom-Ingenieurs unter der Leitung von

Privatdozent Dipl.Ing. Dr. Gabriel Wurzer

E272

Institut für EDV-gestützte Methoden in Architektur und Raumplanung

eingereicht an der Technischen Universität Wien

Fakultät für Architektur und Raumplanung

von

Daniel Elsner

Matr. Nr. 0927239

Wien, am

SimSchool

Optimierung der Abläufe in Schulbauten

Abstract

Der Zustand bzw. die Entwicklungen des Bildungssystems sind gegenwärtig eines der Hauptthemen des öffentlichen Diskurses. Gerade in Österreich handelt es sich dabei um ein wichtiges Thema, da sich das Bildungswesen derzeit in einer Phase des Um- bzw. Aufbruchs befindet. Mit politischen Reformen wird auf die Vielzahl neuer Erkenntnisse bezüglich Pädagogik und Lernverhalten reagiert, die sich in den letzten Jahren herauskristallisiert haben. Der Architektur kommt im Zuge dieser Reformierungen die Aufgabe und Möglichkeit zu, durch Überprüfung bestehender, sowie Entwicklung neuer Schultypen als Reaktion auf die geänderten Anforderungen, am Diskurs teilzunehmen und diesen voranzutreiben.

Diese Diplomarbeit setzt sich mit den aktuellen Tendenzen der Schulplanung, angefangen von städtebaulichen Aspekten, über Nutzungsneutralität und Erweiterbarkeit bis hin zu den Abläufen innerhalb eines Schulgebäudes auseinander. Ziel ist die Entwicklung eines Simulations-gestützten Hilfsprogramms zur Unterstützung des/r Architekten/in in frühen Planungsphasen.

Die Arbeit ist in mehrere, aufeinander folgende Abschnitte gegliedert. Zunächst werden die aktuelle Situation des Bildungswesens sowie vorhergehende Ansätze Computer-gestützter Hilfestellungen in der Architektur untersucht. Anschließend wird die Idee des Hilfsprogramms sowie deren Umsetzung erläutert. Das Programm bietet eine Planungsunterstützung auf mehreren Ebenen, von der städtebaulichen Bedarfsplanung zur Eruierung des prognostizierten Bedarfs neuer Schulbauten einer Region, bis zur Planung einzelner Schulgebäude. Hierbei werden die Abläufe in Schulbauten mittels Agenten-basierter Simulation analysiert, um daraus Nachbarschaftsdiagramme zu erstellen, welche die komplexen räumlichen Beziehungen der unterschiedlichen Funktionsbereiche innerhalb eines Schulgebäudes veranschaulichen. Als Ausgangsdaten der Simulation werden die Wege, welche SchülerInnen während einer Schulwoche zurücklegen, herangezogen. Die Nachbarschaftsdiagramme dienen in weiterer Folge zur Erstellung von Vorschlägen einer geschoßweisen Anordnung der Bereiche.

Das Hilfsprogramm findet schließlich Anwendung zur Überprüfung des prognostizierten Klassenzuwachses der Stadt Wien bis 2030 sowie als Planungsunterstützung des aktuell ausgeschriebenen Wettbewerbs für den Neubau des "Bildungscampus Nordbahnhof". Dieser folgt dem gegenwärtigen Trend, mehrere Schultypen an einem Bauplatz anzuordnen, um Austausch und Mehrfachnutzungen zu ermöglichen.

Abstract

The state and development of the education system is one of the premier topics of today's public dialogue. In Austria it is an especially important issue at the moment, since the education system is currently in a rebuilding phase. This development can be seen as a reaction to the variety of new findings concerning pedagogy and learning behavior of students, which emerged in recent years. The role of Architecture can be seen as to participate in and advance the discourse by revision of existing school typologies as well as development of new ones as a reaction to the changing requirements.

This master thesis deals with the current tendencies in schools design, including aspects of urban planning, neutrality of use, expendability and procedures within a single school building. The goal is the development of a simulation-based application to support architects in the early stages of planning school buildings.

The paper is divided in various, consecutive steps. Initially, the current situation of the education system as well as existing papers and prototypes in the field of computer-based planning-tools are examined. Afterwards the idea behind the application and its implementation are explained. The software provides assistance for planers on various layers, from the determination of the need for additional educational buildings of a region to the planning of individual schools. For that the processes in schools are used as base data for the creation of adjacency-diagrams showing the complex spatial relationships between the various functional units within school buildings. This is accomplished by analyzing the paths of students during a typical school day or full week of school using agent-based simulation. The resulting adjacency-diagrams serve as source material for the generation of recommendations for the arrangement of the functional units on floors.

In the last step, the application is utilized for the examination of the projected increase of the number of classrooms for the city of Vienna until the year 2030. Eventually, the tool is applied to demonstrate the spatial relationships of the areas of activity in the ongoing competition for the construction of "Bildungscampus Nordbahnhof". This competition follows the trend of placing various school types on one construction site to facilitate interconnectivity, communication and multiple use.

Inhaltsverzeichnis

1. Problem **8**

1.1. Situation 9

1.2. Problemstellung 27

1.3. Vorhergehende Arbeiten 27

2. Idee **34**

2.1. Regionale Bedarfsplanung 35

2.2. Lokale Ressourcenplanung und Anforderungsgenerierung 36

2.3. Ergebnis 39

3. Implementierung **40**

3.1. Regionale Bedarfsplanung 41

3.2. Lokale Ressourcenplanung und Anforderungsgenerierung 44

3.3. Ergebnis 64

4. Case Study **66**

4.1. Regionale Prognose: Klassenzuwachs in Wien 67

4.2. Einsatz Schulgenerator: Wettbewerb "Bildungscampus Nordbahnhof" 73

5. Diskussion **86**

5.1. Anwendung des Hilfsprogramms als Analysetool 87

5.2. Zusammenhang Regional - Lokal 88

5.3. Ansatzpunkte zukünftiger Arbeiten 89

6. Schluss **97**

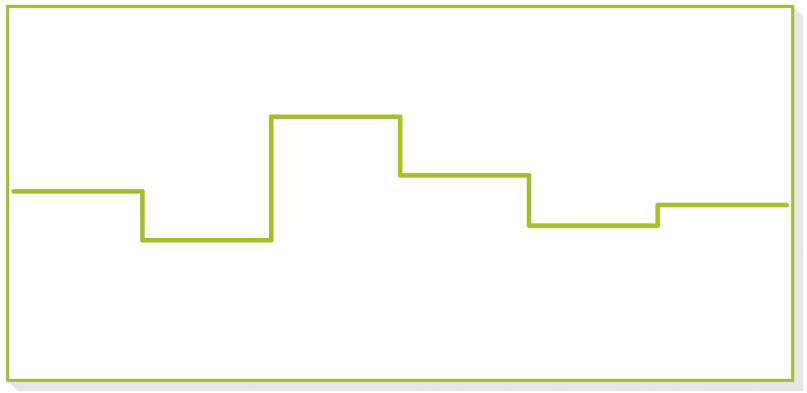
7. Anhang **98**

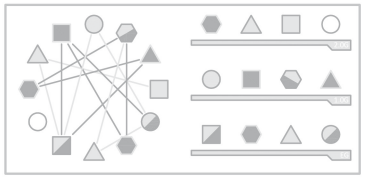
7.1. Literaturverzeichnis 99

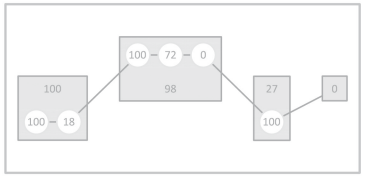
7.2. Abbildungsverzeichnis 101

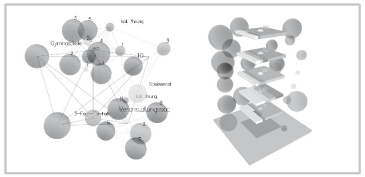
7.3. Sourcecode 104

7.4. Excel-Tabellen 163









1. Problem

Komplexität der neuen Anforderungen in der Schulplanung

1.1. Situation **9**

1.1.1. Räumliche Organisationsmodelle 20

1.1.1.1. SchülerInnenbereiche 20

1.1.1.2. Team-, Personal- und Beratungsräume 24

1.1.1.3. Gemeinschaftsbereiche 27

1.2. Problemstellung **27**

1.3. Vorhergehende Arbeiten **27**

1. Problem

Komplexität der neuen Anforderungen in der Schulplanung

1.1. Situation

Das Bildungssystem und dessen Reformierung ist aktuell eines der meistdiskutierten Themen des öffentlichen Dialogs. Erhöhte Aufmerksamkeit erhielt die Problematik durch das schlechte Abschneiden Österreichs bei der 2000 erstmals von der OECD durchgeführten PISA-Studie, einer internationalen Schulleistungsstudie. Diese seitdem alle 3 Jahre durchgeführte Studie untersucht europaweit "die Leistungen von 15-jährigen Schülerinnen und Schülern sowie deren Lernmotivation, ihre Selbsteinschätzung und ihre Lernstrategien."[1]

In den darauffolgenden Jahren begann die Politik, die gängigen Praktiken des Unterrichtens, sowie in weiterer Folge des Schulbaus zu hinterfragen. Eine Vielzahl neuer Reformen wurden in ganz Europa diskutiert und zum Teil auch umgesetzt.[2] Die Vorreiterrolle übernahmen hier vor allem die Skandinavischen Länder sowie die Niederlande, wo die Forderung der Funktion des Raumes als "drittem Pädagogen" am besten umgesetzt wurde. Diese Formulierung geht ursprünglich auf den Italiener Loris Malaguzzi, den Begründer der "Reggio-Pädagogik", zurück, welcher die drei Pädagogen in der Schule als die anderen Schüler, den Lehrer und eben den Raum definierte.

Die Reformen im deutschsprachigen Raum nahmen unterschiedliche Ausmaße an. Sie reichten von der Verringerung der Schülerzahlen pro Klasse über die Verbesserung der Bezahlung der Lehrkräfte bis zu einer Veränderung der gängigen Praktiken des Schulbaus. Das Konzept der Flurschule, das seit fast zwei Jahrhunderten bevorzugte Modell in der Schulplanung, bei dem Klassenräume gleicher Größe an Erschließungsgängen aufgereihten werden, verlor seine Vormachtstellung. An seine Stelle trat eine Vielzahl neuer räumlicher Organisationsmodelle, welche zeitgenössische pädagogische Konzepte deutlich besser umsetzten. In den Vordergrund rückten nun differenziertere Raumangebote, bei denen die Klassenräume etwa in größere Einheiten, sogenannte Lerncluster, zusammengefasst werden, Erschließungsgänge sich in Lernstraßen verwandeln oder sich gänzlich in offene Lernzonen auflösen. [3]

[1] http://www.oecd.org/berlin/themen/pisa-hintergrund.htm, 1.3.2016, 14:22

[2] Lederer, A., Pampe, B., (2011): Vergleich ausgewählter Richtlinien zum Schulbau - Kurzfassung. Montag Stiftung Urbane Räume, Bonn, Rahmen und Richtlinien für einen leistungsfähigen Schulbau in Deutschland, Heft 1

[3] Kühn, C., (2013): Cluster und Lernstraße statt Gang und Klassenzimmer, Merkur, Heft 766, März 2013, 245-250

Diese Entwicklungen gingen mit neu formulierten Konzepten der Pädagogik einher. Einige der Prämissen heißen hier "Förderung statt Selektion", Hinwendung zu ganztägigen Schulformen, Variation der Lernmethoden, mehr Eigenverantwortung der SchülerInnen oder auch mehr fächerübergreifende Kooperation. Diese Ansichten lassen sich mit den etablierten Bausteinen des Schulsystems, wie der Halbtagsschule, der 50-Minuten-Stunde, dem fast ausschließlichen Frontalunterricht oder des klassischen, neun mal sieben Meter großen Klassenraumes kaum mehr vereinbaren.

Die, in diesem Kapitel untersuchten, unterschiedlichen Tendenzen im Bildungswesen sind nicht isoliert zu betrachten, sondern stehen in einer Wechselwirkung zueinander und funktionieren am effizientesten in ihrer Kombination.

Neue Richtlinien für den Schulbau

In den letzten Jahren wurden in weiten Teilen Europas neue Richtlinien bezüglich der Schulplanung eingeführt, um den geänderten pädagogischen und organisatorischen Anforderungen gerecht zu werden.

In Deutschland etwa gibt es eine Vielzahl unterschiedlicher Regelungssysteme, die von Bundesland zu Bundesland variieren können. Einige davon haben die veralteten Schulbaurichtlinien und dazugehörigen Musterraumprogramme ersatzlos abgeschafft, andere sich hingegen für Neuformulierungen entschieden.[4]

In Österreich gibt es Richtlinien für den Schulbau, die vom Österreichischen Institut für Schul- und Sportstättenbau (ÖISS) in regelmäßigen Abständen veröffentlicht werden. In der aktuellen Ausgabe aus dem Jahr 2014 werden folgende Vorgaben bezüglich der räumlichen Anforderungen von Unterrichtsräumen definiert: *"Die Größe von Unterrichtsräumen wird projektspezifisch in den jeweiligen Raum- und Funktionsprogrammen definiert. In Zusammenhang mit unterschiedlichen Schülerzahlen in einzelnen Klassenverbänden bzw. Teilungsgruppen erscheint es sinnvoll, Unterrichtsräume mit differenzierten Raumgrößen zu schaffen."*[5] Darüber hinaus werden zahlreiche detaillierte Anforderungen bezüglich der Positionierung von Möbeln, Sichtbedingungen in den Unterrichtsräumen, Raumakustik und weiteren Variablen der Schulplanung als Richtwerte oder Empfehlungen erläutert. Von der Stadt Wien werden Raumblätter sowie Musterpläne für die Planung verschiedener Schultypen zur Verfügung gestellt. In den aktuellen Raumblättern für die Bauaufgabe "Neue Mittelschule" finden sich Größenvorgaben für verschiedene Arten von Lernbereiche. Für

[4] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[5] ÖISS (2014): Räumliche Anforderungen für Unterrichtsräume. Kapitel 6, Richtlinien für den Schulbau, S.2

Unterrichtsräume, die als ständige Aufenthaltsbereiche für 25 SchülerInnen dienen sollen, wird darin eine Größe von 63 m² veranschlagt und für Gruppenräume, welche auf den Zusatzunterricht von bis zu 15 Personen ausgelegt sind, 42 m².[6]

Aufgrund der geänderten Anforderungen an die Schulplanung geht es heute nicht mehr nur darum, einen quantitativen Flächenmehrbedarf zeitgemäßer Lernumgebungen zu begründen, sondern auch, neue Qualitätskriterien und veränderte räumliche Organisationsmodelle zu diskutieren. Die diversen deutschsprachigen Regelwerke weisen zum Teil große Unterschiede auf und beschreiben meist eher Grenzwerte der Raumgrößen als Qualitätsmerkmale.[7] Neuere Richt- und Leitlinien verzichten zunehmend auf exakte Festlegungen von Flächen für verschiedene Funktionsbereiche und definieren eher Flächenkennwerte, die dann nach den lokalen Gegebenheiten und abhängig des pädagogischen Konzepts der jeweiligen Schule flexibel adaptiert werden können.[8]

Lernen und Unterrichten

"Lernen heißt nicht bloße Reproduktion, Spiegelung von Vorhandenem, sondern aktive Konstruktion durch die Lernenden - im eigenen Kopf, im eigenen Körper und gemeinsam mit anderen."[9]

Jedes Kind, jeder Jugendliche, jeder Mensch ist und lernt verschieden; im eigenen Tempo, an unterschiedlichen Themen, auf verschiedenen Wegen. Eine darauf reagierende Lern- und Unterrichtskultur braucht geeignete Lernumgebungen. Um dieser Vielfalt unterschiedlicher Lernwege und Lernhandlungen, vom individuellen Lernen über Teamarbeit in Kleingruppen bis zu Lerneinheiten in Großgruppen, gerecht zu werden, muss eine Schule ein differenziertes Angebot unterschiedlicher, anregender Lernsituationen bieten. Das traditionelle Klassenzimmer verliert im Zuge dieser Entwicklungen seine zentrale Stellung als Instruktionsraum und wird eine Raum unter anderen. Die Rolle des Lehrenden unterzieht sich ebenso einem Wandel; er wird von Instrukteur einer Klasse zum Mentor unterschiedlicher Individuen. [10]

[6] Magistratsabteilung 34: Bau- und Gebäudemanagement (2016): Raumblätter für Neue Mittelschulen der Stadt Wien, Wien

[7] Lederer, A., Pampe, B., (2011): Vergleich ausgewählter Richtlinien zum Schulbau - Kurzfassung. Montag Stiftung Urbane Räume, Bonn, Rahmen und Richtlinien für einen leistungsfähigen Schulbau in Deutschland, Heft 1

[8] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

[9] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn, S.29

[10] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

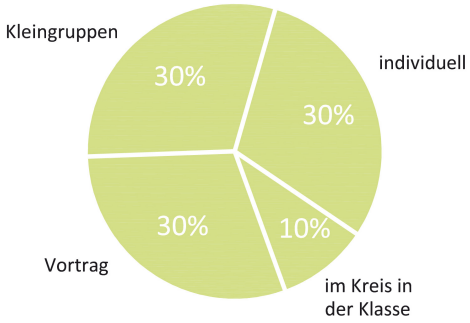


Abb. 1.1 Methodenmix

Entsprechend aktueller pädagogischer Konzepte sollte im Unterricht ein Methodenmix aus Frontalunterricht, Gruppen- und Einzelarbeit angewendet werden. Als "Faustformel" gilt dabei ein Verhältnis von 30% Vortrag, 30% Arbeit in Kleingruppen, 30% individuellem Arbeiten und 10% Arbeit im Kreis in der Klasse.[11] (siehe Abb. 1.1) Darüber hinaus gewinnen fächerübergreifende Lernprozessen zunehmend an Bedeutung. Um den Methodenmix und die verstärkte Zusammenarbeit im Unterricht umzusetzen, müssen die zur Verfügung gestellten Räume möglichst flexibel veränderbar und auf unterschiedliche Weise nutzbar sein.[12] (siehe Abb. 1.2)

In den letzten Jahrzehnten gab es eine Entwicklung weg vom unreflektierten Auswendiglernen hin zu einem Lernen des Verstehens. In diesem Sinne lernen SchülerInnen selbstständiges Lösen von Problemen anstatt fertige Problemlösungen nachzuahmen. Im Computerzeitalter ist

[11] Seydel, O. (2013): Die kleine Schule in der großen Schule. Lehren & Lernen, Heft 12 2013, 7-13

[12] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

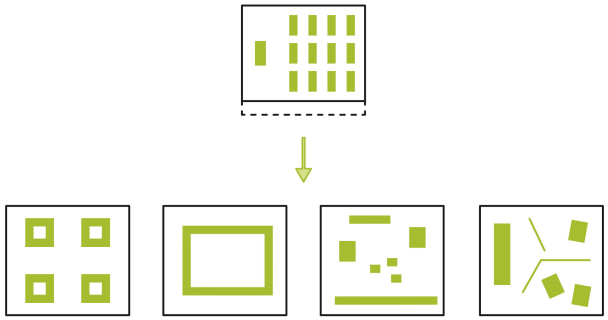


Abb. 1.2 Unterschiedliche Raumnutzungen

dabei vor allem die Fähigkeit der intelligenten Informationsbeschaffung und -verarbeitung von wesentlicher Bedeutung. Als Basis dafür müssen die SchülerInnen allerdings weiterhin auf ein Grundwissen aus diversen Bildungsbereichen zurückgreifen können, das in der Schule vermittelt werden sollte. Aktuelle Ergebnisse der Lernforschung belegen eindeutig, dass gerade vielfältiges Lernen die beste Voraussetzung für den Lernerfolg ist.[13] Fächerübergreifendes, problemorientiertes Lernen hilft dabei ebenso wie Selbst- und Gruppenlernphasen. Um dieses Lernverständnis in der Schule umsetzen zu können, muss den SchülerInnen ein angemessenes Raum- und Zeitkonzept geboten werden. Ein weiterer wesentlicher Ansatz ist das sogenannte Beziehungslernen durch die Auseinandersetzung und den Dialog mit anderen. Dieses tritt besonders

[13] Arnold, R., Schüßler, I. (1998): Wandel der Lernkulturen. Ideen und Bausteine für ein lebendiges Lernen. Darmstadt

bei Lerneinheiten in Klein- bis Großgruppen abseits des Frontalunterrichts ein.[14]

Ganztag

Immer mehr Schulen greifen auf das System des ganztägigen Unterrichts zurück. Dieses bietet einerseits viele Vorteile im Vergleich zum halbtägigen, stellt alle Beteiligten (PlanerInnen, SchülerInnen, Lehrpersonal) andererseits aber vor Herausforderungen in der Umsetzung. Grundsätzlich lassen sich die, im letzten Abschnitt besprochenen, neuen Lern- und Lehrmethoden in ganztägigen Schulformen besser umsetzen, da Lernen als aktives Lernen mehr Zeit und Raum benötigt. [15]

Das österreichische Modell der Neuen Mittelschule basiert auf dem System der ganztägigen Schule. Ein pädagogischer Vollbetrieb ist von 7:45 bis 16:00 Uhr vorgesehen. Zusätzlich gibt es einen Frühdienst von 6:00 - 7:45 Uhr und einen Schlussdienst von 16:00 bis 18:00 Uhr.[16] Generell setzt sich der Tagesablauf einer Ganztagsschule aus sechs variablem Phasen zusammen (Ankommen in der Schule, Unterricht, Essen, Bewegung, individuelle Interessensbildung, gemeinsame Projekte, Verlassen der Schule), welche je nach Schule modifiziert oder unterschiedlich gewichtet werden können. [17]

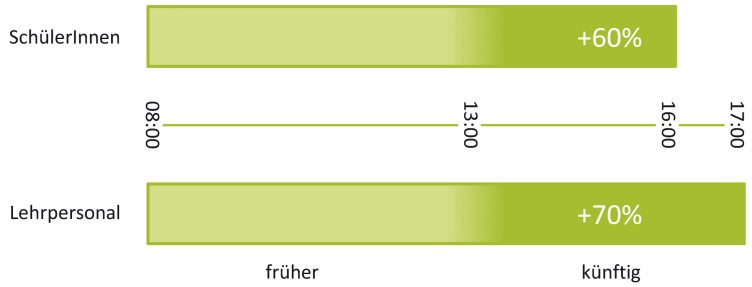


Abb. 1.3 Zeit in der Schule

Da alle SchulnutzerInnen mehr Zeit am Schulstandort verbringen; SchülerInnen etwa 60% und LehrerInnen etwa 70%,[18] (siehe Abb. 1.3) müssen sich Schulbauten von Lern- zu Lebensorten entwickeln, die den unaufwändigen Wechsel zwischen Orten und Phasen der Konzentration und der Regeneration erlauben.[19] Um dies zu gewährleisten, müssen sie den SchülerInnen und LehrerInnen neben Lern- und Arbeitsbereichen

[14] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[15] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[16] Magistratsabteilung 34: Bau- und Gebäudemanagement (2014): Raumbuch Campus+: Bildungseinrichtung mit Mehrwert der Stadt Wien, Wien

[17] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[18] Huven, K. (2007): Pädagogische Architektur. Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bonn

[19] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

auch Bereiche der Entspannung zur Verfügung stellen. Die unterschiedlichen Funktionen sollten räumlich allerdings nicht voneinander getrennt angeordnet werden, sondern ineinander verwoben und kombinierbar sein. Ganztagsaufenthaltsbereiche sind daher am besten in die Lehrbereiche zu integrieren, um die Flächennutzung zu maximieren. Ebenso sollten sie flexibel genug ausgebildet sein, sodass ein Bereich über den Schultag hinweg mehrere Funktionen übernehmen kann, um den Flächenbedarf aufgrund der großen Anzahl unterschiedlicher Aktivitäten nicht zu überschreiten. Eine besonders wichtige Stellung nehmen Gemeinschaftsbereiche, wie Mensa oder Bibliothek, sowie Freiflächen als soziale Treffpunkte ein.

Mit dem Konzept der Ganztagsschule geht auch eine Veränderung der LehrerInnenrolle einher. Ihr Schulalltag wird durch die längere Anwesenheit und veränderte Arbeitsanforderungen vielfältiger. Damit verbunden ist auch eine notwendige Verbesserung der entsprechenden räumlichen Bedingungen hinsichtlich individueller Arbeitsplätze und gemeinsamer Aufenthaltsbereiche für das Lehrpersonal.[20]

Laut Gesundheitsfolgenabschätzung ist durch die Ganztagsschule eine positive Auswirkung für 10- bis 14-jährige SchülerInnen in Österreich hinsichtlich sozialer Aspekte sowie Fördermöglichkeit und Schulleistung, zusätzlich zu eine erwarteten Verbesserung des familiären Klimas, zu erwarten.[21]

Rhythmisierung des Unterrichts

Einhergehend mit der Forderung nach der Ganztagschule ist auch das Thema der Rhythmisierung des Unterrichts und des Tagesablaufs, um das volle Potenzial des ganztägigen Konzepts auszuschöpfen.

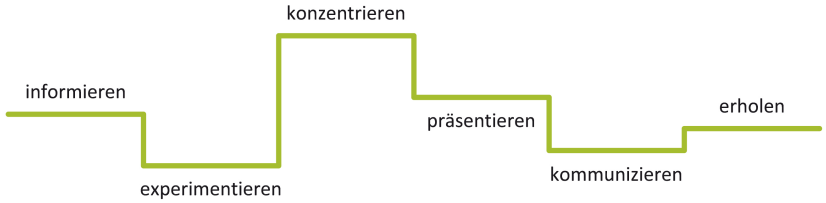


Abb. 1.4 Rhythmisierung des Schultages

Der natürliche Tagesrhythmus der SchülerInnen sollte bei der Gestaltung des Schultages auf jeden Fall Berücksichtigung finden. Dieser umfasst eine hohe Leistungsphase am Vormittag, ein Leistungstief um die Mittagszeit

[20] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[21] Flaschberger, E., Grandy, S., Hofmann, F., Lehner, L., Teutsch, F., Vogl, S., Felder-Puig, R.(2015): Die zeitliche Gestaltung des Schulalltags. Bestandsaufnahme und Empfehlungen zur Umsetzung in Österreich. Wien: LBIHPR Forschungsbericht.

und eine weitere Leistungsphase am Nachmittag.[22] Der "kindgerechte" Rhythmus zeichnet sich durch ein Wechselspiel aus Bewegungs- versus Ruhephasen oder Spiel- versus Arbeitsphasen aus.[23] Auf den Schultag umgelegt zeigt sich die Bedeutung der Rhythmisierung zwischen Schulstunden und dazwischenliegenden Pausen zur Erholung und Bewegung. Es gilt das bestmögliche Verhältnis dieser Phasen zu untersuchen. (siehe Abb. 1.4)

Ein erster Ansatzpunkt ist der Zeitpunkt des Schulbeginns in der Früh. Aus Sicht der Chronobiologie ist es für SchülerInnen in der Pubertät zielführend, den Unterrichtsbeginn weiter nach hinten zu verlegen, um deren Leistungsfähigkeit über den gesamten Tag zu steigern. Mehrere Studien aus den USA zeigen bereits bei einer Verschiebung des Schulbeginns von 8:00 auf 8:25 oder 8:30 signifikante Verbesserungen hinsichtlich Schlafdauer, Laune, Tagesmüdigkeit und Koffeingebrauch bei den SchülerInnen.[24] Eine weitere Möglichkeit besteht darin, eine offene Beginnphase zwischen 7:45 und 8:30 einzuführen. Allerdings gibt es auch Kritiker der Anpassung des Schultages an den natürlichen Biorhythmus von Kindern und Jugendlichen, da sie diesen als anpassungsfähig bezeichnen.[25] Als ein Nachteil eines späteren Unterrichtsbeginns wird zumeist erwähnt, dass der Schultag dann auch später enden würde.[26]

Ein weiterer wesentlicher Punkt der Rhythmisierung des Schultages ist die Länge der einzelnen Unterrichtseinheiten. Alternative Zeitrhythmen, jenseits des in Österreich herkömmlich 50-Minuten-Takts, bieten diesem gegenüber einige Vor- und Nachteile. In längeren Lerneinheiten, etwa durch Blockungen von zwei oder mehr Unterrichtsstunden, kann eine größere Variation von Unterrichtsmethoden zum Einsatz gebracht werden. Ebenso ermöglicht diese Rhythmisierung vermehrt fächerübergreifende Formen des Projektunterrichts.[27] Die Einführung längerer Unterrichtseinheiten muss in jedem Fall mit einem Umdenken bezüglich der Unterrichtsmethoden gekoppelt sein. Innerhalb einer Einheit müssen sich Phasen des Frontalunterrichts mit Phasen der Gruppenarbeit und Einzelarbeit abwechseln, um das volle Potenzial der Blockung auszuschöpfen. DeutschlehrerInnen beschreiben als einen Vorteil der Blockung beispielsweise, dass durch Ausdehnung der Taktung, Textproduktion und darauf bezogene Gruppenarbeit in einer Einheit

[22] Kolbe, F.U., Rabenstein, K., Reh, S. (2006): Expertise „Rhythmisierung“. Hinweise für die Planung von Fortbildungsmodulen für Moderatoren. Berlin, Mainz: Johannes-Gutenberg Universität Mainz, Technische Universität Berlin.

[23] Holtappels, H.G. (1995): Zeitgestaltung und Schulqualität: Entwicklung der Lernkultur in "Vollen Halbtagsschulen" Niedersachsens. In: Holtappels, H:G: (Hg.): Ganztagserziehuung in der Schule. Schule und Gesellschaft 10. Opladen: Leske + Budrich, S. 209-229

[24] Owens, J.A., Belon, K., Moss, P. (2010): Impact of delaying school start time on adolescent sleep, mood, and behavior. Archives of Pediatrics & Adolescent Medicine, 164 (7), 608-614.

[25] Vogel, J. (2012): Zeitstruktur entwickeln - Schritt für Schritt. IV. Fachtagung für Schulen mit Ganztagsangeboten/Sek. I im Land Brandenburg "Ganztag als Merkmal zukunftsfähiger Schulen" (14.03.2012)

[26] Flaschberger, E., Grandy, S., Hofmann, F., Lehner, L., Teutsch, F., Vogl, S., Felder-Puig, R.(2015): Die zeitliche Gestaltung des Schulalltags. Bestandsaufnahme und Empfehlungen zur Umsetzung in Österreich. Wien: LBIHPR Forschungsbericht.

[27] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

ermöglicht werden. Darüber hinaus wird der Zeitverlust durch administrative Aufgaben verringert. Als Problemstellungen werden oft ein insgesamter Mehraufwand für LehrerInnen oder auch die begrenzte Konzentrationsfähigkeit der Schüler genannt. Diese kommt in diesem Fall allerdings nicht zum Tragen, da sie altersabhängig zwischen zehn und 20 Minuten liegt und somit auch in einer 50-Minuten-Einheit bereits deutlich überschritten wird.[28] Generell kann man zwischen zeitlicher und thematischer Blockung unterscheiden. Unter thematischem Blocken versteht man die Aneinanderreihung unterschiedlicher Unterrichtsfächer, die sich allerdings mit demselben Thema beschäftigen.[29]

Gerade bei der Debatte längerer Unterrichtseinheiten ist die Länge der Pause dazwischen als Phase der Entspannung für SchülerInnen, genauso wie für LehrerInnen, ein ebenso wichtiges Thema. Neben flexiblen, kurzen Pausen innerhalb geblockter Einheiten sollten zwischen längeren Lernphasen auch längere Unterbrechungen von 20-25 Minuten ermöglicht werden. Pausen übernehmen eine Vielzahl wichtiger Funktionen, wozu neben der Möglichkeit zur Bewegung als Ausgleich auch eine wichtige soziale Komponente zählt. In Ganztagsschulen ist es generell einfacher, mehr Pausenzeit zu schaffen, da insgesamt über den Tag verteilt mehr Zeit zur Verfügung steht.[30]

Maria Montessori, eine Vertreterin des selbstständigen Lernens von Kindern und Jugendlichen, erkannte früh die Wichtigkeit von Pausen zur Bewegung im Schulalltag unter dem, von ihr entwickelten, Konzept der "bewegten Schule": *"When mental development is under discussion, there are many who say: " How does movement come into it? We are talking about the mind." And when we think of intellectual activity, we always imagine people sitting still, motionless. But mental development must be connected with movement and be dependent on it. It is vital that education theory and practice should be informed by that idea."*[31]

Inklusion

Das Thema der Inklusion steht derzeit im Mittelpunkt der bildungspolitischen und pädagogischen Diskussion. Soziale Inklusion bezeichnet die Einbeziehung aller Menschen in die Gesellschaft. Auf das Bildungssystem umgelegt, bedeutet Inklusion die Umsetzung eines gemeinsamen Schulsystems für alle SchülerInnen, ungeachtet ihres Leistungsniveaus oder sozialen Hintergrunds. (siehe Abb. 1.5) Derzeit gibt

[28] Wicki, W., Dietrich, C. (2007): Expertise Zeitstrukturen Basel-Stadt; Luzern: Forschungsbericht Nr. 6 der Pädagogischen Hochschule Zentralschweiz, Hochschule Luzern.

[29] Flaschberger, E., Grandy, S., Hofmann, F., Lehner, L., Teutsch, F., Vogl, S., Felder-Puig, R.(2015): Die zeitliche Gestaltung des Schulalltags. Bestandsaufnahme und Empfehlungen zur Umsetzung in Österreich. Wien: LBIHPR Forschungsbericht.

[30] Flaschberger, E., Grandy, S., Hofmann, F., Lehner, L., Teutsch, F., Vogl, S., Felder-Puig, R.(2015): Die zeitliche Gestaltung des Schulalltags. Bestandsaufnahme und Empfehlungen zur Umsetzung in Österreich. Wien: LBIHPR Forschungsbericht.

[31] Montessori, M. (1949): The Absorbent Mind, The Theosophical Publishing House, Adyar, S. 203

es eine weltweite pädagogische Entwicklung zu leistungsfähigen heterogenen Lernsystemen. Das österreichische Modell der Neuen Mittelschule basiert auch auf diesem System. Die Lehrenden in solchen Schulen sollten zusätzlich durch sozialpädagogisch, psychologisch und medizinisch geschultes Personal unterstützt werden.[32]

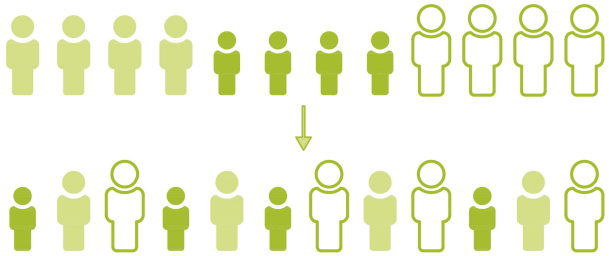


Abb. 1.5 Inklusion

Es ist die Aufgabe der Schule Benachteiligungen von jungen Menschen, egal ob Herkunft, Geschlecht, oder Behinderungen, auszugleichen und dabei jedem Lernenden eine optimale Förderung zukommen zu lassen.[33] Die Inklusionsforderung erfährt ihre Begründung allerdings nicht nur durch die Schaffung von Chancengleichheit für alle SchülerInnen, sondern bietet darüber hinaus viele Vorteile im Vergleich zu selektiven Schulsystemen, welche wertvolles Bildungspotenzial verschenken und im Umkehrschluss die Chancenungleichheit verstärken.[34] Die heterogene Zusammensetzung von Lerngruppen bzw. Schulklassen durch Unterschiede der Leistungspotenziale, sozialer Rahmenbedingungen, angeborener Voraussetzungen sowie Interessen der SchülerInnen ist kein Nachteil, sondern eine Chance. Es tritt der Effekt des Perspektivenwechsels ein, bei dem die Lernenden den anderen gegenüber unterschiedliche Rollen einnehmen (Lernende, Lehrende, Zuhörer, Redner, Beobachter,...), wodurch sie sich der Verbesserung ihrer persönlichen Lernmöglichkeiten bewusster werden.[35] Die SchülerInnen lernen also nicht nur durch die Instruktion einer Lehrkraft, sondern ebenso durch den Dialog und die Auseinandersetzung untereinander. Die PISA-Studie hat eindeutig gezeigt, dass inklusive Bildungssysteme, in denen heterogenen Gruppen gemeinsam über einen längeren Zeitraum unterrichtet werden, ohne eine frühe Selektion vorzunehmen, zu besseren Bildungsergebnissen führen. Eine frühe Selektion, etwa nach

[32] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[33] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[34] Huven, K. (2007): Pädagogische Architektur. Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bonn

[35] Malina, B., Busch, C., (2010): Inklusion: Leitlinien für die BIldungspolitik. Deutsche UNESCO-Kommission, Bonn

Leistungsniveau, führt im Gegensatz nur dazu, die soziale Leistungsdifferenz zu vergrößern.[36]

Die internationalen Organisationen UNESCO und OECD weisen nach, dass sich Schulsysteme weltweit mit dem Thema der "Heterogenität und Diversität" auseinandersetzen müssen. Die Architektur kann die pädagogische Idee der Inklusion unterstützen, indem sie genügend und gut ausgestattete Räume zur Verfügung stellt.

Vielseitigkeit, Veränderbarkeit und Langlebigkeit

Flexibilität, Anpassbarkeit, Mehrfachnutzungen und Schaltbarkeit sind häufig genutzte Schlagwörter zeitgenössischer Schulplanung. Damit gemeint ist einerseits die kurzzeitige Flexibilität, um einzelne Funktionsbereiche ohne großen Aufwand an verschiedene Anforderungen anpassen zu können, andererseits auch die langfristige Anpassbarkeit der Schulbauten, um diese über mehrere Jahrzehnte hinweg, auch bei Änderung von pädagogischen Anforderungen, optimal nutzen zu können. Schulgebäude haben schließlich eine Nutzungsdauer von 50 bis 90 Jahren, wodurch sie als Gebäude für mehrere Generationen bezeichnet werden können.[37] Zeitgenössische Schulgrundrisse schaffen Flexibilität, indem sie unterschiedliche Raumsituationen für unterschiedliche Lernszenarien bieten. Unter dem Begriff der "räumlichen Polyvalenz" werden Mehrfachnutzungen ermöglicht, ohne dabei eigenschaftslose Räume ohne Identität zu schaffen. (siehe Abb. 1.6)

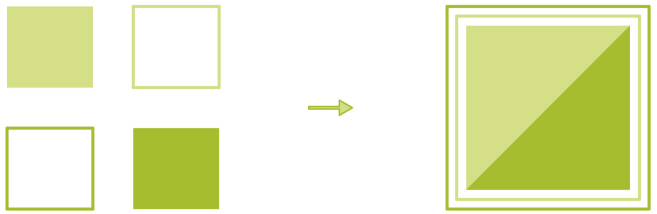


Abb. 1.6 Räumliche Polyvalenz

Durch einfache Methoden lassen sich Lernräume an verschiedenste Lernsituationen anpassen. Dies geschieht etwa durch Umstellen flexibler Möbel oder die Trennung bzw. Zusammenschaltung von Bereichen mittels mobiler Trennelemente. Besonders gut eignen sich Schultische mit Freiform- Tischplatten zur Schaffung unterschiedlicher Raumszenarien.[38] Flexibilität bedeutet aber nicht zwangsläufig die Anwendung mobiler Trennelemente, welche die NutzerInnen vor Probleme in der Handhabung oder Schalltechnik stellen können. Oft reicht die bloße zur Verfügung

[36] OECD (2010): PISA 2009 Results: Overcoming Social Background - Equality in Learning Opportunitiers and Outcomes (Volume II), http://dx.doi.org/10.1787/9789264091504-en

[37] Huven, K. (2007): Pädagogische Architektur. Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bonn

[38] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

Stellung mehrerer verschieden großer Räume, um die Anwendung der gesamten Vielfalt der Lernmethoden zu unterstützen. Erschließungsbereiche übernehmen dabei je nach gewählter Raumvariante (Lerncluster, offene Lernlandschaft,...) auch weitere Funktionen, so können sie beispielsweise den Lernbereichen zugeschalten werden, Platz für spontane Lernsituationen bieten oder vollständig in Lernlandschaften integriert werden.[39]

Bildungscampus

Die Anordnung mehrerer Schulen an einem Standort oder in unmittelbarer Umgebung ermöglicht Synergien durch die gemeinsame Nutzung ausgewählter Funktionsbereiche. (siehe Abb. 1.7) Ebenso bieten lokale Bildungslandschaften die Möglichkeit, weitere Funktionen der Nachbarschaft in sich aufzunehmen und damit einen wichtigen Beitrag als soziale Treffpunkte außerschulischer Gruppen zu leisten.[40]

Beim österreichischen Modell des sogenannten "Campus+", einem Ableger des Bildungscampus-Systems, kommt es zu einer Vernetzung von Kindergarten-, Schul-, und Freizeitpädagogik an einem Standort. Es handelt sich hierbei um eine gesamtheitliche, integrative Bildungseinrichtung für Null- bis Zehnjährige. Der Mehrwert dieses Systems ergibt sich aus diversen Synergien im Bildungsalltag. Pädagogische Einrichten (Therapiebereich, Kreativbereich, Sportbereich) sowie Nebenräume (Küche, Lager, Verwaltung) können gemeinsam verwendet werden. Die Auslastung der pädagogischen Flächen kann dadurch von unter 50% auf insgesamt über 60% erhöht werden. Ein weiterer wesentlicher Aspekt des Systems ist die Ermöglichung neuer Lern-und Bildungsformen. Darüber hinaus kommt es zu einem Mehrwert durch Öffnung des Campus zur Stadt und Integration von Stadtteilfunktionen (Freiraumbereich, Veranstaltungssaal, Sporthallen). Generelles Planungsziel ist auch die Reduktion der Bauplatzgrößen durch den effizienten Umgang mit Raum- und Freizeitressourcen.[41]

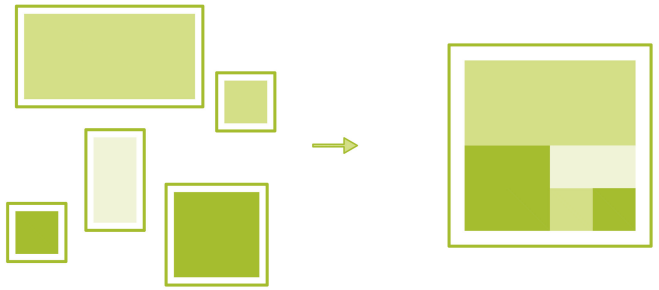


Abb. 1.7 Bildungscampus

[39] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[40] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[41] Magistratsabteilung 34: Bau- und Gebäudemanagement (2014): Raumbuch Campus+: Bildungseinrichtung mit Mehrwert der Stadt Wien, Wien

1.1.1. Räumliche Organisationsmodelle

Grundsätzlich gibt es die Unterscheidung zwischen verschiedenen Modellen zur räumlichen Organisation von Schulbauten. Beim klassischen Modell sind die verschiedenen Funktionsbereiche, wie allgemeine Lehrbereiche, spezielle Lehrbereiche, Gemeinschaftsbereiche sowie Team- und Personalräume räumlich und funktional separiert. A Im Teilintegrierten Modell sind bereits einige davon in die Lern- und Unterrichtsbereiche integriert. Beim Integrierten Modell sind schließlich alle Funktionsbereiche Bestandteile einer umfassenden, multifunktionalen Lernlandschaft. B [42] (siehe Abb. 1.8)

1.1.1.1. SchülerInnenbereiche

Im Zuge der Auflösung der klassischen Aneinanderreihung von Klassenräumen entlang eines Erschließungsgangs C entwickelten sich mehrere prototypische räumliche Organisationsmodelle, die sich mehr oder weniger radikal mit der Problematik auseinandersetzen. In diesem Abschnitt werden drei dieser neuen Raumtypologien, nämlich "Klassenraum Plus", "Lerncluster" und "Offene Lernlandschaft" vergleichend besprochen. (siehe Abb. 1.9)

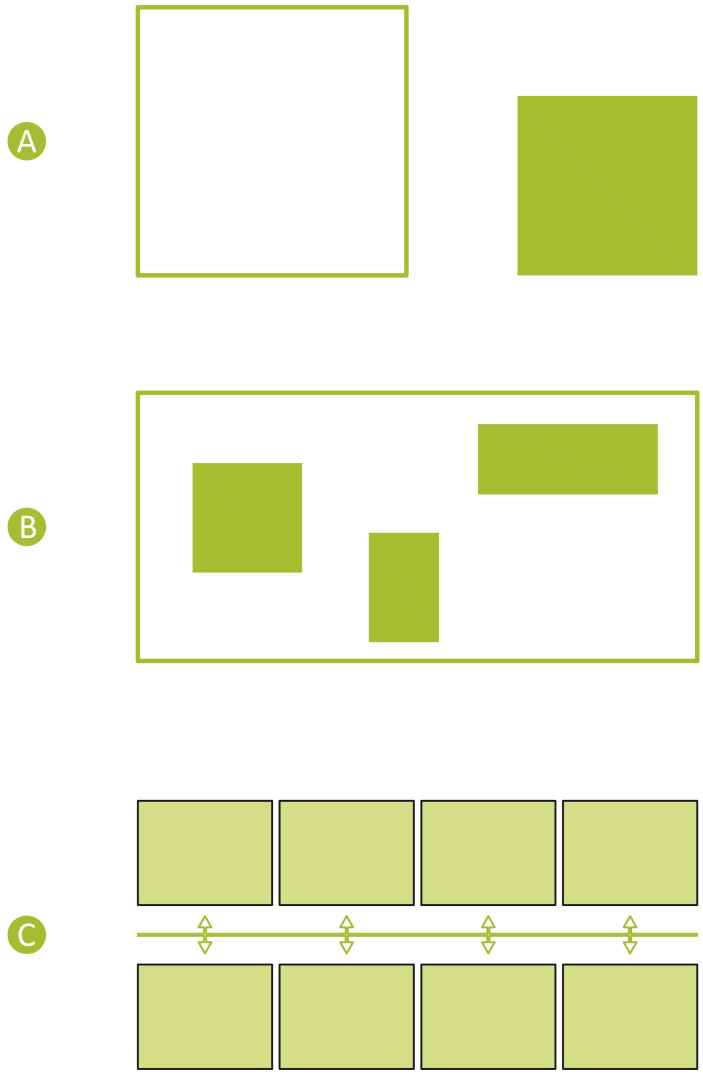


Abb. 1.8 Räumliche Organisationsmodelle I

[42] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

Aus heutiger Sicht ist die, über Jahrzehnte hinweg gültige, Faustformel für Unterrichtsräume von 2m² Flächenbedarf pro SchülerIn (= 65m² bei 25 SchülerInnen) nicht mehr ausreichend, um unterschiedliche Lernformen abseits des Frontalunterrichts zum Einsatz zu bringen. Durch die Addition von Räumen für Differenzierungsbereiche und Ganztagesbetreuung erscheint ein Flächenbedarf von 4-5m² pro SchülerIn realistischer.[43] Umsetzbar wird diese Forderung nach zusätzlicher Fläche beispielsweise durch die geschickte Integration von Erschließungsflächen in die Lernbereiche. Um die Parallelarbeit von Einzelnen und Kleingruppen zu gewährleisten, müssen die verschiedenen Bereichen mittels Transparenz oder Durchlässigkeit räumlich vernetzt sein. Diese Verflechtung erfordert eine Perforation der Grenzen, die von offenen Türen über transparente Wandelemente bis zu Schiebe-Elementen reichen kann.[44]

Grundsätzlich kommt dem Klassenzimmer innerhalb des Schulgebäudes eine große Bedeutung zu. Johann Heinrich Pestalozzi bezeichnete es als "Wohnstube", in der sich SchülerInnen und LehrerInnen gleichermaßen wohlfühlen sollen.[45]

[43] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[44] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[45] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

1.1.1.1.1. Allgemeine Lern- und Unterrichtsbereiche

- Klassenraum Plus Ⓓ

Beim räumlichen Organisationsmodell des "Klassenraum Plus" wird die Fläche im bzw. um den Klassenraum erweitert, um dem Bedarf an zusätzlichen multifunktionalen Nutzräumen für unterschiedliche Lernformen gerecht zu werden. Beim "Rucksackprinzip" erhalten zumeist zwei Klassenräume einen zwischengeschalteten Gruppenraum, den sie für Unterrichtsformen abseits des Frontalunterrichts zusätzlich nutzen können. Eine wesentliche Herausforderung hierbei ist die Ausführung der Schnittstelle der Stammklassen mit dem "Appendix". Zwischen den Räumen sollte eine visuelle Verbindung bestehen, um dem Lehrpersonal die Kontrolle zu erleichtern und somit Parallelunterricht zu ermöglichen, ohne dass der Blickkontakt verloren geht. Somit entsteht trotz notwendiger akustischer Trennung ein Raumkontinuum. Die Erschließung der Gruppenräume kann über mehrere Zugänge vom Erschließungsbereich oder direkt über die Klassenräume erfolgen. Maßgebliche Verbesserungen bezüglich der Flexibilität der Nutzungsformen eines Klassenraumes bei einer Belegung durch 25 SchülerInnen ergeben sich ab einer Raumgröße von 70 m².[46]

[46] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

- Lerncluster E

Als Lerncluster wird der Zusammenschluss von meist zwei bis sechs Klassen oder Lerngruppen zu einem größeren, gemeinsamen Unterrichtsbereich bezeichnet. Dabei besitzt jede Schulklasse zwar weiterhin einen eigenen Raum, der allerdings über einen zentralen Gemeinschaftsbereich mit allen anderen Räumen des Clusters verbunden ist. Die Ausführung dieses allgemeinen Bereichs ist ein entscheidender Aspekt für die Qualität des Raumangebots. Er kann unterschiedliche Formen annehmen, von einem verbreiterten Gang mit Sitznischen bis zu einem frei geformten, vielfältig bespielbaren Multifunktionsbereich. Auf jeden Fall bietet er das Potenzial, neben der Erschließungsfunktion auch noch andere Funktionen, etwa des Lernbereichs, Spielbereichs und sozialen Treffpunkts für die SchülerInnen zu übernehmen. Durch die sinnvolle Integration der Erschließungszonen ist eine Erhöhung der verfügbaren Flächen im Unterrichtsbereich möglich. [47]

Generell gibt es verschiedene Arten der Clusterbildung.[48] Beim Jahrgangscluster werden alle SchülerInnen eines Jahrgangs in einem gemeinsamen Bereich zusammengefasst, während sich bei der jahrgangsübergreifenden Variante (auch "Lernhaus" genannt) Klassen unterschiedlicher Jahrgänge einen Bereich teilen. Eine dritte Möglichkeit bietet der sogenannte Fachraum-Cluster, bei dem die jeweiligen Bereiche von Gruppen fachlich verwandter Unterrichtsgegenstände gebildet werden.

Abb. 1.9 Varianten der Clusterbildung

So entstehen etwa Sprachen-Cluster, in denen alle Sprachen unterrichtet werden, Kunst-Cluster oder naturwissenschaftliche Cluster. (siehe Abb. 1.10) Im Unterschied zu den beiden vorher genannten Varianten, bewegen sich beim Prinzip des Fachraum-Clusters die SchülerInnen nach dem klassisch-angloamerikanischen Departmentsystem zum Lehrenden und nicht umgekehrt. Der Vorteil dieses Systems liegt in der Möglichkeit für das Lehrpersonal, sich den persönlichen, dauerhaften Arbeitsbereich bestmöglich einzurichten, da dieser nicht nach jeder Einheit gewechselt werden muss. Als Nachteil kann betrachtet werden, dass in diesem Fall

[47] Seydel, O. (2013): Die kleine Schule in der großen Schule. Lehren & Lernen, Heft 12 2013, 7-13

[48] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

den SchülerInnen, in Ermangelung einer eigenen Stammklasse, ihr persönlicher Bereich als "Homebase" in der Schule abhandenkommt.

Die Ausbildung der Schnittstelle zwischen den Unterrichtsräumen und dem dazwischenliegenden Gemeinschaftsbereich ist auch bei der Planung von Lernclustern von zentraler Bedeutung.

- Offene Lernlandschaft F

Die offene Lernlandschaft ist das Ergebnis einer vollständigen Loslösung vom typischen Konzept des Klassenzimmers und kann somit als der fortschrittlichste Typ der räumlichen Organisationsmodelle bezeichnet werden. Bei dieser Typologie gibt es keine Klassenräume im herkömmlichen Sinn mehr, sondern stattdessen eine Vielzahl an Bereichen unterschiedlicher Größe und Zuschnitts, die sich für verschiedene Unterrichtsmethoden bzw. Gruppengrößen eignen.

Dieses Konzept ist die unmittelbarste räumliche Abbildung der Idee von Individualisierung und Eigenverantwortung der SchülerInnen. Anwendung findet es vor allem in höheren Jahrgangsstufen, da es ein hohes Maß an Eigenverantwortung voraussetzt. Zu diesem räumlichen Organisationmodell zählen auch die in Skandinavien verbreiteten "Base"-Konzepte. Bei diesen sind die SchülerInnen nicht mehr in herkömmliche Klassenverbände eingeteilt, sondern in Großgruppen von 60 bis 120 Personen, welche sich über den Schultag oder die Schulwoche hinweg

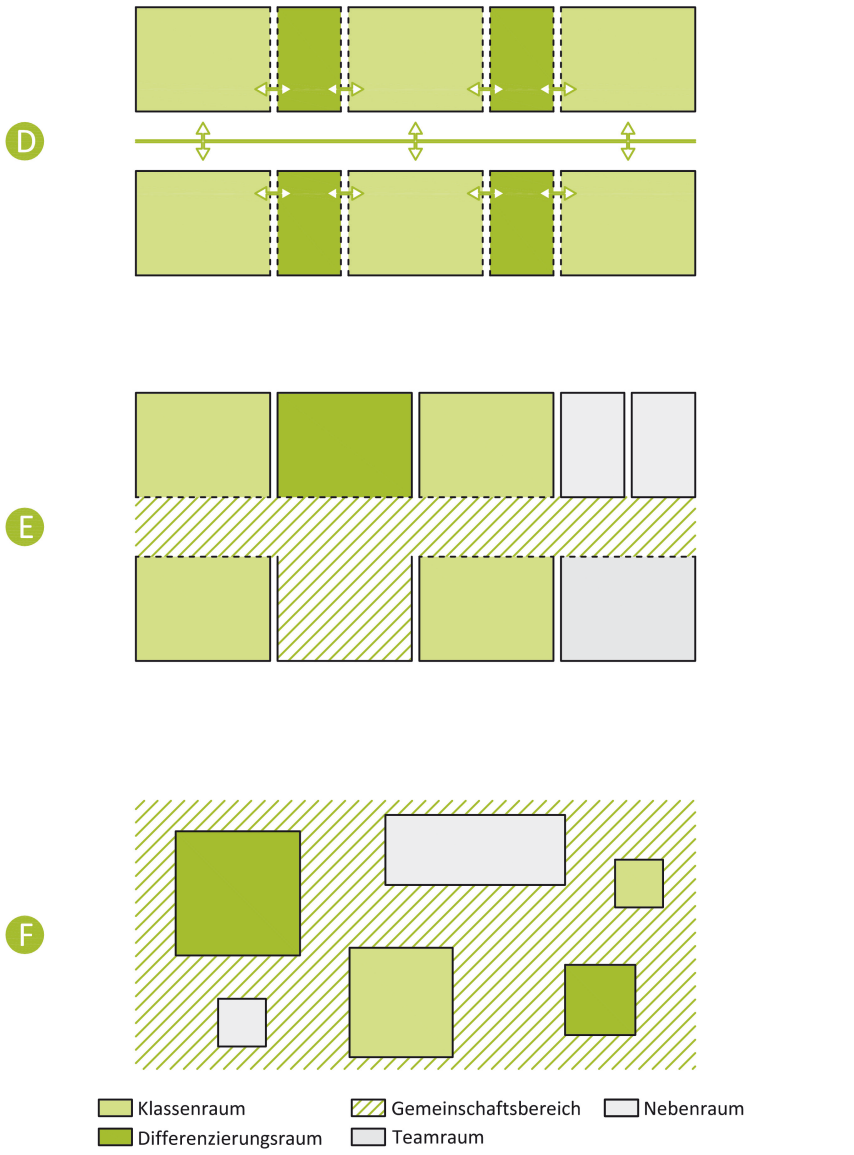


Abb. 1.10 Räumliche Organisationsmodelle II

immer wieder in verschiedenen Gruppenkonstellationen zusammensetzen.[49]

1.1.1.1.2. Spezialisierte Lern- und Unterrichtsbereiche

Unterrichtsbereiche für naturwissenschaftliche Fächer, Musik, Kunst und Gestaltung, Technik, Hauswirtschaft oder Sport und Bewegung benötigen spezialisierte Grundausstattungen. Zusätzlich gelten für sie zumeist erhöhte Anforderungen betreffend Lüftung, Akustik, Bodenbelag oder Infrastruktur. Um sie möglichst effizient nutzen zu können, ist es sinnvoll, sie weitgehend fächerübergreifend auszubilden, etwa als Labors für naturwissenschaftliche Fächer oder als Ateliers für künstlerische Fächer. Darüber hinaus sollten diese Lernumgebungen Raumgruppen bilden, um fächerübergreifenden Projektunterricht zu erleichtern und gegebenenfalls Synergien bei der Nutzung von Vorbereitungs- oder Lagerräumen zu ermöglichen.[50]

1.1.1.2. Team-, Personal- und Beratungsräume

Bei der Planung neuer Schulbauten gewinnt auch die Ausführung von Personal- und Verwaltungsbereichen an Bedeutung, vor allem in Erwägung der zunehmenden Hinwendung zu ganztägigen Schulformen und der damit verbundenen Veränderung des Berufsalltags von Lehrenden. Das klassische, auf die Halbtagsschule ausgerichtete, Lehrerzimmer mit eineinhalb bis zwei m² Fläche pro Person, wird den geänderten Anforderung nicht mehr gerecht und durch andere, zentrale oder dezentrale, räumliche Organisationsmodelle abgelöst.[51] (siehe Abb. 1.11) Arbeitsstättenrichtlinien sehen etwa für reguläre Büroarbeitsplätze Flächen von acht bis zwölf m² pro Person vor.[52] Zusätzlich sollte die steigende Bedeutung von Teamarbeit unter den Lehrenden berücksichtigt werden. Ebenso sollten es zu einer funktionalen Entflechtung der Arbeitsbereiche der LehrerInnen kommen; so sind etwa für individuelle Vorbereitung oder Ablage, Besprechungen oder für den informellen Austausch, voneinander abgegrenzte Flächen vorzusehen. Ein entscheidender Aspekt ist hierbei auch die akustische Entkopplung der unterschiedlichen Nutzungsbereiche, um wechselseitige Beeinträchtigungen zu vermeiden. Jedem Lehrenden muss innerhalb des Schulgebäudes genug Platz zur Verfügung stehen, um auch in der unterrichtsfreien Zeit (Pausen, Freistunden) produktives Arbeiten zu

[49] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

[50] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[51] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

[52] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

ermöglichen, wofür ein Flächenbedarf von 4 - 5 m² pro Person eingeplant werden sollte.[53]

Zur Situierung dieser Bereiche im Gebäude gibt es verschiedenen Ansätze:

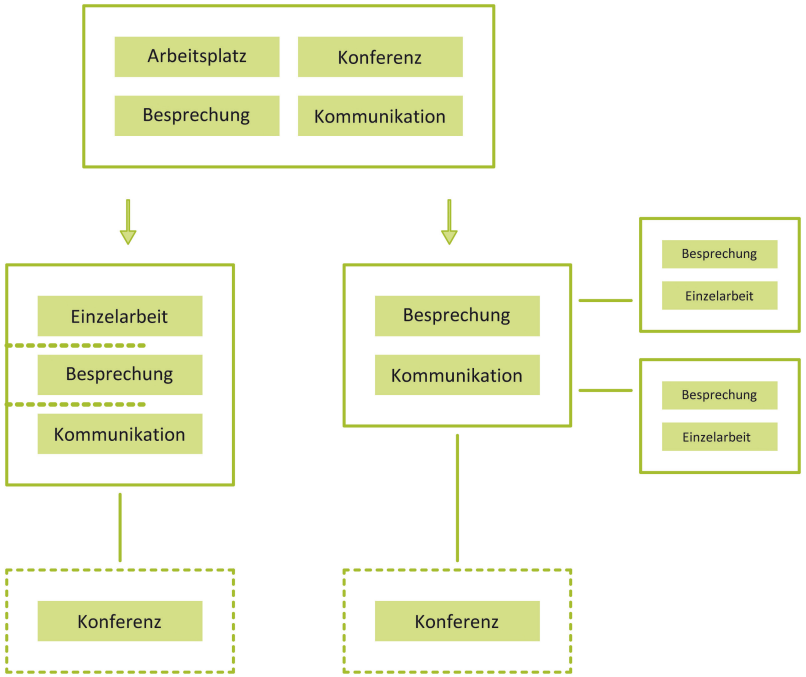


Abb. 1.11 Verwaltungs- und Teambereiche[54]

- Zentraler Verwaltungsbereich

Eine zentrale Lösung kommt häufig für kleinere Schulbauten zum Einsatz, in denen sich alle Pädagogen noch in einem gemeinsamen Raum treffen. Dennoch sollten auch hier klar differenzierte Bereiche für unterschiedliche Nutzungen eingeplant werden, wobei die Transparenz und Überschaubarkeit erhalten bleiben sollte.

- Dezentrale Teambereiche

Als Alternative zum zentralen Organisationsmodell finden gerade in größeren Schulbauten dezentrale Teambereiche zunehmend Aufmerksamkeit. Dadurch wird den Pädagogen die Zusammenarbeit in kleinen, überschaubaren Einheiten, meist in Jahrgangs- oder Fachteams ermöglicht, wodurch die Teamstrukturen zusätzlich gestärkt werden.

Dezentrale, oft direkt in Lernclustern integrierte Teamstützpunkte setzen sich meist aus bis zu acht individuellen Arbeitsplätzen und zugeschalteten Besprechungsbereichen zusammen. An manchen Schulen kommt eine Mischform als Kombination aus zentralen Kommunikationszonen mit dezentralen Arbeits-und Besprechungszonen zum Einsatz.[55]

[53] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[54] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

[55] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

PRIMARSTUFE

Raumart	Flächenbedarf pro Schüler	Bemerkungen
Allgemeine Lern- und Unterrichtsbereiche Basisräume, Gruppenräume, Rückzugsräume (Inklusion), Mehrzweckräume (dezentrale Ganztagsbereiche), Aufenthalts bereiche (einschließlich pädagogisch nutzbarer Erschließungsbereiche)	4,5 – 5,5 m²	aufteilbar gemäß dem pädagogischen Konzept
Spezialisierte Lern- und Unterrichtsbereiche Naturwissenschaften, Musik, Kunst und Gestaltung. Aufenthaltsbereiche (einschließlich pädagogisch nutzbarer Erschließungsbereiche)	0,5 – 1,5 m²	ohne Lernbereiche für Sport und Bewegung teilweise kombinierbar
Gemeinschaftsbereiche Foyer, Aula, Forum, Mensa, Cafeteria und Bibliothek	1,5 – 2,5 m²	ohne Außenareale teilweise kombinierbar
Team-,Personal- und Beratungsräume Gruppenräume, Teambüros, Besprechungs, Aufenthaltsbereiche, Rezeption, Sekretariat, Schulleitung, Hausmeister, Therapieräume, Gesundheitsstation, Räume für Sozialarbeit und Beratung, Schülerselbstverwaltung	1,5 – 2,0 m²	für Arbeitsplätze: 4–5 m² pro Beschäftigte
Sonstige Funktionsbereiche Sanitärräume, Garderobenbereiche, Serverraum, Kopierraum, Materialräume, Gebäudetechnik und -unterhaltung, reine Erschließungsbereiche, weitere Lager- und Nebenräume		teilweise dezentral und kombinierbar

SEKUNDARSTUFE I

Raumart	Flächenbedarf pro Schüler	Bemerkungen
Allgemeine Lern- und Unterrichtsbereiche Basisräume, Gruppenräume, Rückzugsräume (Inklusion), Mehrzweckräume (dezentrale Ganztagsbereiche), Aufenthalts bereiche (einschließlich pädagogisch nutzbarer Erschließungsbereiche)	4,5 – 5,5 m²	aufteilbar gemäß dem pädagogischen Konzept
Spezialisierte Lern- und Unterrichtsbereiche Naturwissenschaften, Musik, Kunst und Gestaltung. Aufenthaltsbereiche (einschließlich pädagogisch nutzbarer Erschließungsbereiche)	2,0 – 3,0 m²	ohne Lernbereiche für Sport und Bewegung teilweise kombinierbar
Gemeinschaftsbereiche Foyer, Aula, Forum, Mensa, Cafeteria und Bibliothek	1,5 – 2,5 m²	ohne Außenareale teilweise kombinierbar
Team-,Personal- und Beratungsräume Gruppenräume, Teambüros, Besprechung, Aufenthaltsbereiche, Rezeption, Sekretariat, Schulleitung, Hausmeister, Therapieräume, Gesundheitsstation, Räume für Sozialarbeit und Beratung, Schülerselbstverwaltung	1,2 – 1,8 m²	für Arbeitsplätze: 4–5 m² pro Beschäftigte
Sonstige Funktionsbereiche Sanitärräume, Garderobenbereiche, Serverraum, Kopierraum, Materialräume, Gebäudetechnik und -unterhaltung, reine Erschließungsbereiche, weitere Lager- und Nebenräume		teilweise dezentral und kombinierbar

Abb. 1.12 Raumbedarfe[56]

[56] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

1.1.1.3. Gemeinschaftsbereiche

Gemeinschaftsräume fungieren in einem Schulbau als Orte des Austausches und der Kommunikation, wodurch sie eine wesentliche Funktion der sozialen Interaktion übernehmen. Auch für sie gilt das Primat der Mehrfachnutzung. So sollten Foyer, Aula/Forum, Mensa/Cafeteria, Bibliothek sowie Außenareale, um nur einige der zentralen Gemeinschaftsbereiche von Schulbauten zu nennen, möglichst flexibel nutzbar und bei Bedarf kombinierbar sein. Je nach Schulgröße können sich diese Funktionsbereiche auch räumlich überlagern, um ihre Auslastung zu maximieren.[57] So kann bei kleinen Schulen eine Kombination aus Mensa und Aula sinnvoll sein, um im Veranstaltungsfall beide Räume gemeinsam nutzen zu können. Gerade die Mensa nimmt in vielen Schulen ein zentrale Rolle ein. Durch multifunktionale Nutzungen abseits der reinen Nahrungsaufnahme, etwa als kommunikativer Treffpunkt, durch Leselounges, Internetcafés oder für nicht-schulische Veranstaltungen, kommt es zu einer ganztägigen Aktivierung des Bereichs.[58] Die vormals klaren Grenzen zwischen Arbeits- und Aufenthaltsbereichen, Gemeinschaftsräumen und Erschließungsflächen verschwimmen zunehmend. (siehe Abb. 1.12)

[57] Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

[58] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

1.2. Problemstellung

Eine Problemstellung der Planungsaufgabe Schulbau ist deren hohes Maß an Komplexität. Diese ergibt sich einerseits aus der Menge an unterschiedlichen Funktionen, die ein Schulgebäude in sich aufnimmt und die in der Planung berücksichtigt werden müssen, andererseits aus den diversen Nutzergruppen, die jeweils variierende Anforderungen an das Gebäude stellen. In den letzten Jahren hat die Komplexität, gerade durch die Vielzahl neuer Ansätze bezüglich veränderter pädagogischer und räumlicher Anforderungen, die in diesem Kapitel besprochen wurden, noch weiter zugenommen.

1.3. Vorhergehende Arbeiten

An Literatur bezüglich der Planung von Schulbauten mangelt es nicht; eine ganze Reihe von Werken verschiedener Autoren befasst sich mit dem Thema, so auch mit der Umsetzung der neuen Tendenzen in der Schulplanung. Diese wurden im Abschnitt der "Situation" bereits abgehandelt. Darüber hinaus gibt es eine Vielzahl von Arbeiten zum Thema des sogenannten "Space Layout Planning", dem Versuch, den architektonischen Planungsprozess mit Hilfe von Computer-gestützten Anwendungen zur automatisierten Erstellung von Grundrissplänen und Entwürfen zu unterstützen. Diese Methode könnte auch bei der Planungsaufgabe Schulbau auf verschiedene Arten Anwendung finden.

Lobos und Donauth (2010) beschäftigen sich mit der Problematik des "Space Layout Planning" und definieren im Zuge dessen 13 Kriterien, welche bei der Umsetzung eines Raumprogramms in einen architektonischen Grundriss eine Rolle spielen. Dabei unterscheiden sie zwischen rationalen, also messbaren und generellen, nicht messbaren Kriterien, deren Gewichtung vom konkreten Einzelfall abhängt.

Rationale Kriterien:

- Belichtung
- Ausblick
- Erreichbarkeit
- Verwandte Funktionen
- Minimierung der Wege
- Effizienz (Verhältnis Nutzfläche zu Restfläche)
- Effizienz (Volumen)
- Größe

Generelle Kriterien:

- Geometrische Komposition
- Goldener Schnitt
- 3D Form zur Füllung
- Nachhaltigkeit
- Andere: Ökonomisch, strukturell, spirituell, ...

Diese Menge an zu berücksichtigenden Anforderungen bedingt das hohe Maß an Komplexität im Planungsprozess, worin gleichzeitig die Schwierigkeit bei der Generierung von architektonischen Grundrissplänen und Entwürfen liegt. Die Komplexität ist gleichsam auch der primäre Grund für das Fehlen präziser, universeller Methoden für Lösungen. Zwar können sie mittels Schemata oder Diagrammen veranschaulicht werden, doch liegt das Hauptproblem in der Umsetzung dieser Darstellungen in konkrete räumliche Designs, in denen jeweils nur einige der Kriterien wirklich erfüllt werden können. [59]

Die Planungsmethode des "Space Layout Planning" existiert zwar seit mehr als 50 Jahren, doch findet sie bis heute im architektonischen Planungsprozess keine konkrete Anwendung. Ziel des "Space Layout Planning" ist, anhand einiger der architektonischen Kriterien, Computer-gestützte Hilfestellungen im Planungsprozess anzubieten oder gänzlich automatisiert Grundrisse zu generieren. Die Methoden dahinter lassen sich grob in vier verschiedene Ansätze gliedern: "Generative", "Constraint-based", "Shape Grammar" und "Expert Systems". Die ersten, die die Methodik auf die Architektur anwandten, waren Mitchell und Dillon im Jahr 1972. In den letzten Jahren war eine genereller Trend zur

[59] Lobos, D., Donath, D. (2010): The problem of space layout in architecture: A survey and reflections. arquiteturarevista, Ausgabe 6, Nummer 2, Juli/ Dezember 2010, 136 - 161

Generierung von Grundrissen sowie Optimierung von Lösungen erkennbar.

In der Architektur gibt es prinzipiell weniger Hilfsmittel zur räumlichen Planung als in anderen Forschungsgebieten, doch sind deren Erkenntnisse nicht einfach auf die Architektur übertragbar, da sie teilweise andere Anforderungen und Kriterien aufweisen, die sich mit den architektonischen nicht unbedingt decken.

Im folgenden Abschnitt werden einige der wesentlichsten Lösungsansatze des "Space Layout Planning" der letzten Jahre in der Architektur, eingeteilt in drei Kategorien (Prototypen und Forschung, Kommerzielle CAAD-Software und Kommerzielle BIM-Software) veranschaulicht.

Prototypen und Forschung:

Elezkurtaj und Frank (1999) entwickeln ein System, das die interaktive Erstellung architektonischer Grundrisspläne unterstützt und automatisierte Lösungen generiert. Bei einer Arbeit von Arvin und House (1999) kommt eine "Dynamic Physik Simulation" zum Einsatz, bei der die Nachbarschaftverhältnisse als Kräfte auf elastische Bänder zwischen den Funktionen eines Raumprogramms einwirken. Einen ähnlichen Ansatz verfolgen auch Wolfgang E. Lorenz, Martin Bicher und Gabriel X. Wurzer (2015). Deren Arbeit " Adjacency in hospital planning" befasst sich ebenfalls mit dem Einfluss von Nachbarschaftsverhältnissen in der Planung von Gebäuden. Nachbarschaftsverhältnisse beschreiben grundsätzlich die Beziehungen unterschiedlicher Einheiten eines Gebäudes zueinander; wie nah oder weit sie voneinander entfernt sein sollten. Nachbarschaftsmatrizen können in frühen Planungsphasen zum Einsatz kommen, um vorläufige räumliche Layouts zu generieren, die, ohne die finale Form vorwegzunehmen, in weiterer Folge das Design beeinflussen können. Die Räume werden in dieser Arbeit als Punktmassen und die Beziehungen dazwischen als Federn, auf die Kräfte wirken, abgebildet. Dasselbe System kann auch angewendet werden, um finale Grundrisspläne bezüglich ihrer Nachbarschaftsverhältnisse zu analysieren.[60] Hsu (2000) befasst sich mit dem Thema der "Constraint-Based"-Methodik bei der Erstellung von dreidimensionalen "Wand-Modellen", basierend auf einer Datenbank der Beziehungen zwischen Räumen und deren Umgebung (Lage, Sonne, Wind, ...). In dieselbe Kerbe schlagen auch Donath und Gonzalez (2006-2008), die "Constraint-Based Design" zur Unterstützung eines Planungsprozesses von Häusern in Lateinamerika anwenden. Li et al. (2000) entwickeln ein nicht-lineares Programm, dass mehrere Lösungen in einer rechteckigen Umgrenzung erstellt. Dabei entstehen optimale und sub-optimale Lösungen, um dem Planenden unterschiedliche Optionen als Inspiration zu bieten. Nilaew (2006) greift hingegen auf einen "Genetic Algorithm" zurück, um, durch

[60] Lorenz, W., Bicher, M., Wurzer, G., (2015): Adjacency in hospital planning

die Analyse des Verhältnisses von Raumgrößen zu Raumbeziehungen, Lösungen zu generieren, die auf architektonischem Wissen basieren. Doulgerakis (2007) nutzt einen ähnlichen Ansatz zur Implementierung von Computer-basierten Methoden der Generierung und Optimierung von Grundrissplänen unter Berücksichtigung der räumlichen Konfiguration und der Zuordnung von Aktivitäten. Duarte (2003) entwickelt ein interaktives Systems zur Generierung von Lösungen basierend auf einer sogenannten "discursive grammar", die sich aus "programming" und "designing grammar" zusammensetzt. Wolfram Research nutzen ein Optimierungs-Tool, bei dem Grundrisspläne eines einfachen Hauses mit zwei Apartments nach drei Kriterien, nämlich Größe der Erschließung, Distanzen zwischen den Räumen und der geometrischen Komplexität, beurteilt werden. Diese drei Parameter können von dem/der NutzerIn unterschiedlich gewichtet werden. Michalak et al. (2002) beschreiben die Entwicklung einer Methode, die mathematische Optimierung und subjektive Entscheidungen in der Konzept-Phase kombiniert. Eine Arbeit von Keatruangkamala and Sinapiromsaram (2005) beschäftigt sich hingegen mit Entwürfen von Häusern mit vier bis acht Räumen. Ziel ist die Minimierung der Distanzen zwischen den Räumen bei gleichzeitiger Maximierung der Raumgrößen. Beim Ansatz von Loemker (2006) wird der architektonische Planungsprozess in Form mathematischer Regeln beschrieben. Sein Prototyp kann sowohl für Neubauten, als auch zur Revitalisierung bestehender Gebäude eingesetzt werden. Die Arbeit von Del Rio-Cidoncha et al. (2007) verbindet Ansätze der Architektur und des Ingenieurswesens, um ein Modell für die Implementierung von "Facility Layout design" in der Architektur zu entwickeln.

Generell lässt sich feststellen, dass auf dem Gebiet der Prototypen und Forschung in den letzten Jahren große Anstrengungen unternommen wurden, das Problem der automatisierten Generierung von Grundrissplänen mittels Ansätzen unterschiedlicher Forschungsgebiete zu lösen. In den meisten Fällen kommen dabei "Constraint-based" oder "Genetic" Algorithmen zum Einsatz. Die Möglichkeiten der Implementierung sind vielfältig, wobei Graphen selten Anwendung finden. Es gibt allerdings keine Beweise, dass diese Ansätze auch in der Praxis umgesetzt werden; ausschließlich Medjou und Jannou sowie Duarte geben dies an. Alle Ansätze verbindet die Tatsache, dass sie Grundrisspläne produzieren, die aus einer äußeren Form (Wohnung oder Stockwerk) und darin enthaltenen kleineren Einheiten (Räume) bestehen.[61]

Kommerzielle CAAD Software

Derzeit gibt es keine kommerzielle Software, welche die automatisierte Generierung von architektonischen Grundrissplänen unterstützt. Zum Teil

[61] Lobos, D., Donath, D. (2010): The problem of space layout in architecture: A survey and reflections. arquiteturarevista, Ausgabe 6, Nummer 2, Juli/ Dezember 2010, 136 - 161

erzeugen die Programme zwar Lösungsvorschläge, doch sind diese entweder nicht akzeptabel (Alberti (AcadGraph, 1998)), beschreiben nur das schematische Layout, ohne dieses zu generieren (Affinity 5.0 -5.6 (Trelligence, 2006-2009)), oder erfordern eine manuelle Anpassung (Vectorworks10 (Nemetscheck, 2004)). Der aktuelle Trend im Bezug auf kommerzielle Software geht in Richtung einfacher Unterstützung des Planungsprozesses ohne die Generierung kompletter Entwürfe.

Kommerzielle BIM Software

Auch die Programme dieser Kategorie (Archicad, Autodesk, Revit, Microstation, Allplan, ...) bieten keine Möglichkeit einer automatisierten Generierung von Grundrissplänen. Sie sind vor allem dann nützlich, wenn es um die Visualisierung und Kommunikation von Entscheidungen geht. Ein großer Vorteil der BIM-Software liegt darin, dass sie eine große Anzahl komplexer Informationen für den kompletten Lebenszyklus eines Gebäudes speichern, darstellen und manipulieren kann.

Conclusio der vorhergehenden Arbeiten

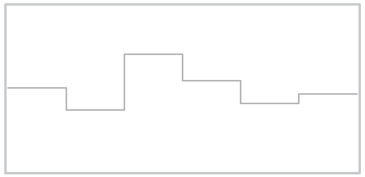
Die meisten Ansätze zum "Space Layout Planning" lassen sich in zwei Kategorien einteilen: Computer-unterstütztes und Computer-generiertes "Space Layout". Ein Großteil der verfügbaren kommerziellen Software zählt zur ersten Kategorie; sie unterstützen die Erstellung von Entwürfen nur durch manuelle Methoden, etwa die Manipulation von Räumen (Translation, Rotation, Größenänderung, ...). Nur wenige prototypische Programme generieren wirklich automatisiert Entwürfe und dienen damit meistens nur zur akademischen Forschung.

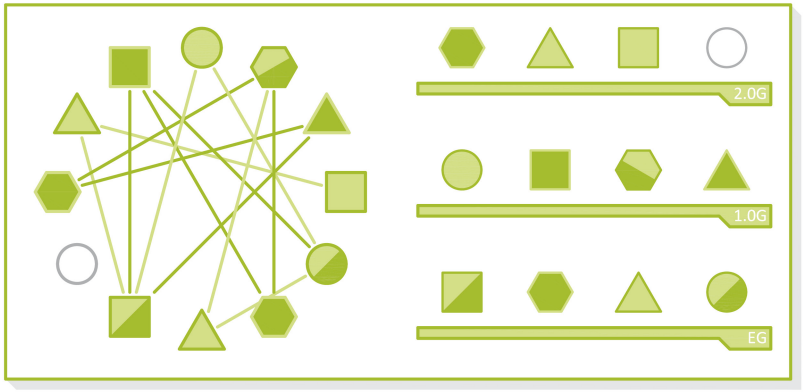
Heute, nach mehr als 50 Jahren Forschungsarbeit auf dem Gebiet steht ArchitektInnen kein Programm zur Verfügung, das sie im Planungsprozess ganzheitlich durch die automatisierte Generierung von Grundrissen unterstützt. Generell nehmen die meisten Forschungsarbeiten auf dem Gebiet keine Bezug auf wichtige Vertreter der Architektur, ebenso wenig wie auf architektonische Literatur, was zur Folge haben kann, dass sie architektonische Probleme missverstehen und dadurch in weiterer Folge unbrauchbare Lösungsansätzen produzieren. Grundsätzlich gibt es keine universelle Lösung des "Space Layout Planning", da jede Bauaufgabe unterschiedliche Voraussetzungen und Anforderungen mit sich bringt. Dass Computer allerdings die Möglichkeit bieten, Probleme in der Generierung von Grundrissplänen zu lösen, wurde bereits bewiesen, allerdings kommen die Resultate in der architektonischen Praxis nicht zum Einsatz, was sich auch im Fehlen entsprechender kommerzieller Software für die tägliche Arbeit eines/r Architekten/in wiederspiegelt. Eine Steigerung der Nachfrage nach manuellen Methoden, die eine schnelle,

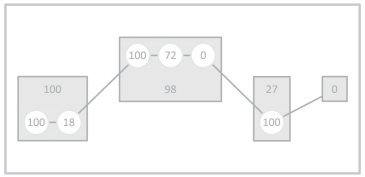
wenn auch nicht komplett automatisierte Hilfestellung in frühen Planungsphasen bieten, ist erkennbar.[62]

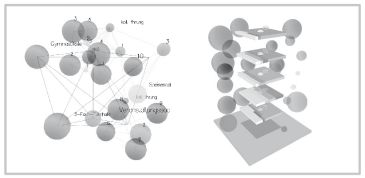
Die, in diesem Abschnitt abgehandelten, Arbeiten sind entweder auf keine konkrete Bauaufgabe spezialisiert, oder beschäftigen sich mit einem Planungsprozess abseits des Schulbaus. Ziel dieser Diplomarbeit ist, einige der Ansätze des "Space Layout Planning" auf den Schulbau anzuwenden, um das hohe Maß an Komplexität der Planungsaufgabe durch computergestützte Hilfestellungen für den Planenden in frühen Phasen des Planungsprozesses zu reduzieren.

[62] Lobos, D., Donath, D. (2010): The problem of space layout in architecture: A survey and reflections. arquiteturarevista, Ausgabe 6, Nummer 2, Juli/ Dezember 2010, 136 - 161









2. Idee

Ansatz zur Planungsunterstützung basierend auf prognostiziertem Bedarf

2.1. Regionale Bedarfsplanung **35**

2.1.1. demographische Prognosedaten 35

2.1.2. Bedarfsdeckung zusätzlicher Klassen 35

2.2. Lokale Ressourcenplanung und Anforderungsgenerierung **36**

2.2.1. Aufbau und AkteurInnen einer Schule 37

2.2.2. Nachbarschaftsdiagramm der Funktionsbereiche 38

2.2.3. Anordnung der Funktionsbereiche 39

2.3. Ergebnis **39**

2. Idee

Ansatz zur Planungsunterstützung von Schulbauten basierend auf prognostiziertem Bedarf

Diese Diplomarbeit greift einige der, im Abschnitt "Vorhergehende Arbeiten" besprochenen, Ansätze des "Space Layout Planning" auf und wendet sie auf den Planungsprozess der Bauaufgabe "Schulbau" an, um deren hohes Maß an Komplexität in frühen Planungsphasen zu reduzieren. Das Ziel ist dabei nicht die vollständige Automatisierung des Prozesses, sondern die Erstellung eines Simulationstools zur Unterstützung im Planungsprozess von Schulbauten, deren Bedarf anhand regionaler Prognose-Daten eruiert werden kann.

Die Planungsunterstützung geschieht auf verschiedenen Ebenen und in mehreren, aufeinander folgende Schritten. Diese reichen von der Überprüfung der raumplanerischen Situation des Schulwesens einer bestimmten Region bis zur räumlichen Anordnung der Funktionsbereiche eines konkreten Schulgebäudes. In einem ersten Schritt der "Regionalen Bedarfsplanung" wird der Bedarf an zukünftig notwendigen, neuen Schulklassen und Schulgebäuden einer Region mit Hilfe von demographischen Daten, bezüglich der prognostizierten Entwicklung der SchülerInnenzahlen, ermittelt. Im darauffolgenden Schritt der "Lokalen Ressourcenplanung und Anforderungsgenerierung" werden planungsrelevante Prozesse innerhalb eines dieser potentiellen Neubauten simuliert. Anhand der daraus resultierenden Daten der nachbarschaftlichen Beziehungen des Schulbaus wird schließlich ein Vorschlag zur räumlichen Situierung der unterschiedlichen Funktionsbereiche der Schule erstellt.

2.1. Regionale Bedarfsplanung

Im Zuge der "Regionalen Bedarfsplanung" wird für eine Region mittels demographischer Prognosedaten die Anzahl zusätzlich benötigter Klassen über einen beliebigen Zeitraum eruiert.

2.1.1. Demographische Prognosedaten

Als Input für die Simulation dienen prognostizierte Daten der Entwicklung der SchülerInnenzahlen der untersuchten Region über einen bestimmten Zeitraum. Mithilfe dieser Daten werden die erforderlichen Klassenzahlen für alle Schultypen aller Unterbereiche der Region ermittelt.

2.1.2. Bedarfsdeckung zusätzlicher Klassen

In der Simulation werden dazu in jedem zu simulierenden Jahr die SchülerInnenzahlen mit den zur Verfügung stehenden Klassenzahlen abgeglichen. Dabei gilt das Prinzip, dass Klassen niemals mit mehr als 25

SchülerInnen belegt sein dürfen.[63] Steigt die durchschnittliche Anzahl von SchülerInnen pro Klasse für einen Schultyp eines Bereichs der Region in einem Jahr über diesen Grenzwert, wird die Anzahl an Klassen für besagten Schultyp so lange erhöht, bis der Grenzwert unterschritten wird. (siehe Abb. 2.1)

Das Resultat der Simulation gibt einen Überblick über die, nach Bereichen, Schultypen und Jahren aufgeschlüsselten, zusätzlich benötigten Klassen im simulierten Zeitraum für alle Schultypen aller Bereiche der simulierten Region. Diese Daten können in weiterer Folge als Grundlage für die folgenden Simulationen zur "Lokalen Ressourcenplanung und Anforderungsgenerierung" eingesetzt werden.

2.2. Lokale Ressourcenplanung und Anforderungsgenerierung

In der "Lokalen Ressourcenplanung und Anforderungsgenerierung" werden mit Hilfe von Simulationen der Bewegungsabläufe in Schulgebäuden die räumlichen Nachbarschaftsbeziehungen der unterschiedlichen Funktionsbereiche analysiert und daraus ein Vorschlag einer Verteilung auf Geschoßebenen erstellt.

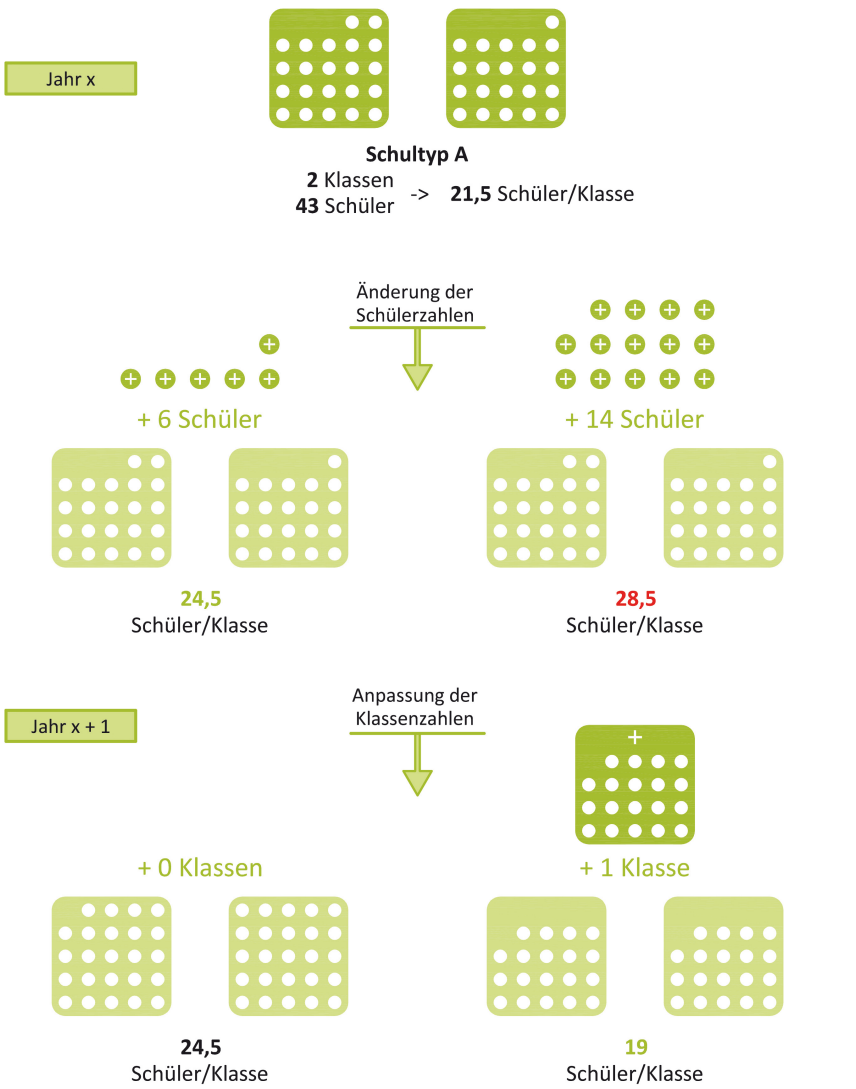


Abb. 2.1 Bedarfsdeckung zusätzlicher Klassen

63 Magistratsabteilung 34: Bau- und Gebäudemanagement (2014): Raumbuch Campus+: Bildungseinrichtung mit Mehrwert der Stadt Wien, Wien

2.2.1. Aufbau und Akteure einer Schule

Zunächst müssen die unterschiedlichen Bereiche einer Schule als Grundlage der Simulation festgelegt werden. Generell lässt sich der Aufbau eines Schulgebäudes in einige Funktionsbereiche zusammenfassen. Dazu zählen Unterrichtsbereiche, Verwaltungsbereiche, Gemeinschaftsbereiche und sonstige Bereiche. Abhängig vom (pädagogischen) Konzept der Schule kann der konkrete Aufbau dieser einzelnen Einheiten variieren oder sie sich teilweise überlagern. Dem aktuellen Trend in der Schulplanung entsprechend werden die Unterrichtsbereiche zumeist zu multifunktionalen Lernclustern zusammengefasst. Die Verwaltungsbereiche können dabei in einem zentralen Bereich untergebracht werden oder als Teamstützpunkte in unmittelbarer Nähe der jeweiligen Lerncluster angeordnet werden. Dasselbe gilt auch für die Nebenräume der Schule.[64] (siehe Abb. 2.2)

Neben dem Aufbau der Schulgebäudes spielen natürlich auch die späteren NutzerInnen eine wesentliche Rolle im Planungsprozess. Die Akteure der Schule lassen sich in mehrere Gruppen einteilen. In Abhängigkeit des (pädagogisches) Konzepts können diese in Kategorien von administrative MitarbeiterInnen, Eltern und BesucherInnen sowie StadtteilbewohnerInnen, die das Gebäude im außerschulischen Betrieb

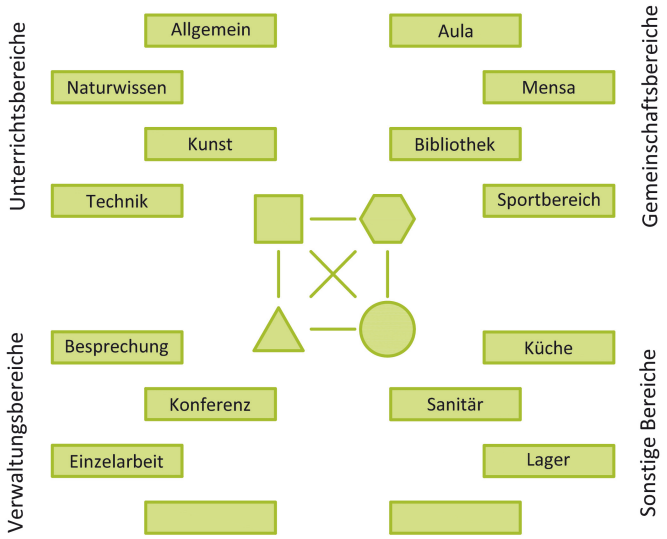


Abb. 2.2 Aufbau einer Schule

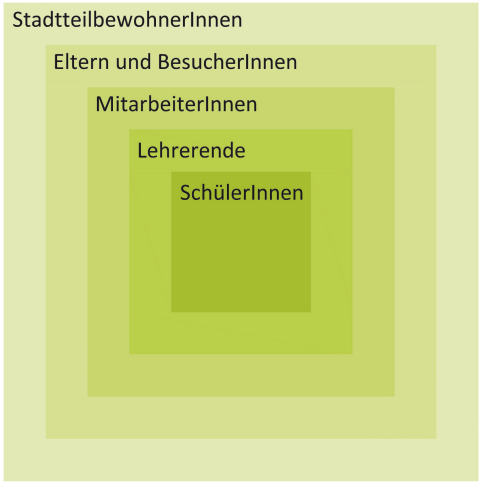


Abb. 2.3 Akteure einer Schule

[64] Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

nutzen, zusammengefasst werden.[65] Die zahlenmäßig größte Gruppe stellen die SchülerInnen, wodurch sie als die Hauptakteure der Schule bezeichnet werden können, auf deren Bedürfnisse in der Planung besonderes Augenmerk zu legen ist. (siehe Abb. 2.3)

2.2.2. Nachbarschaftsdiagramm der Funktionsbereiche eines Schulbaus

In einem ersten Schritt der Simulation werden die Prozesse innerhalb des Schulgebäudes in einer Schulwoche simuliert. Zu diesem Zweck werden die Bewegungsabläufe der Schulklassen im Gebäude über besagten Zeitraum herangezogen. Diese suchen in jeder Zeiteinheit einen der Funktionsbereiche der zu simulierenden Schule auf, in dem sie ein noch verbleibendes Unterrichtsfach ihrer Stundentafel absolvieren können. Kommt es dabei zu einer Bewegung von einem Bereich zu einem anderen, wird eine Verbindung zwischen den beiden hergestellt bzw. diese verstärkt. Dieser Teil der Simulation wird so lange ausgeführt, bis alle Schulklassen alle Unterrichtsgegenstände ihrer Stundentafeln komplett abgearbeitet und ihre simulierte Schulwoche damit abgeschlossen haben. (siehe Abb. 2.4)

Anhand der variierenden Stärke der Verbindungen zwischen den Funktionsbereichen der Schule zeigt sich demnach die Häufigkeit, mit der

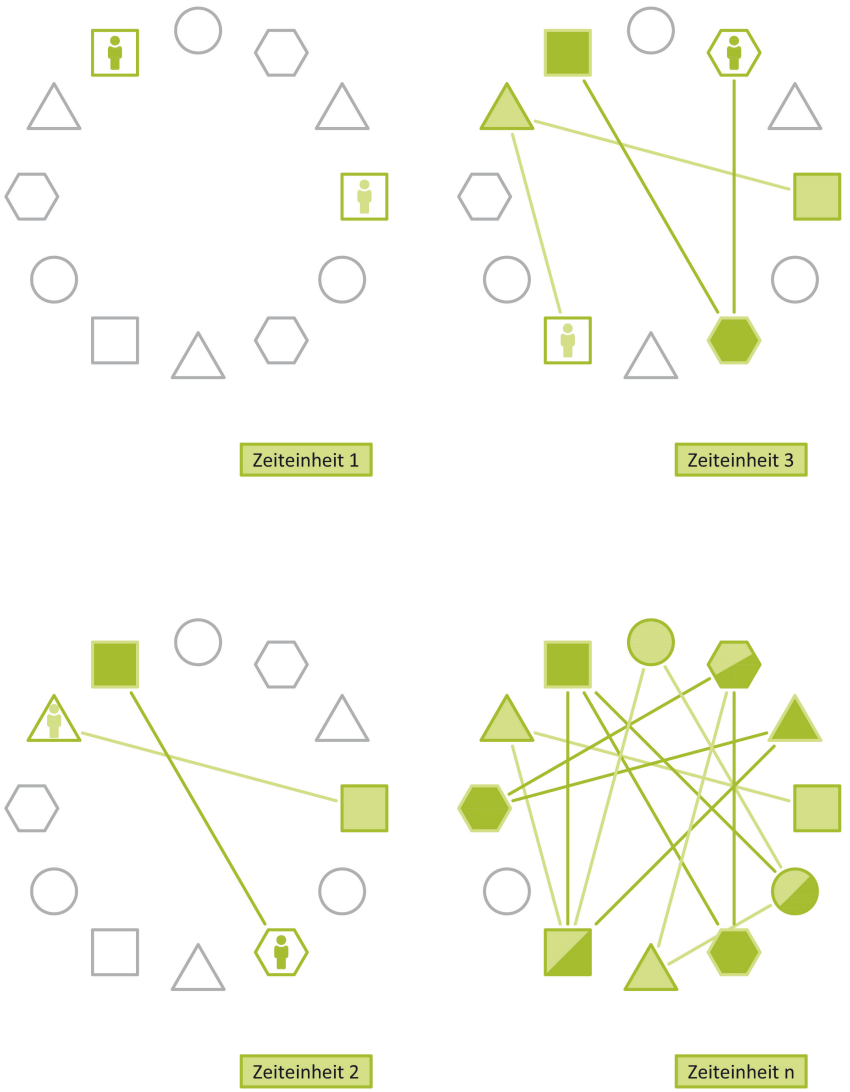


Abb. 2.4 Simulation der SchülerInnenwege

[65] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

diese von Schulklassen in der simulierten Schulwoche benutzt wurden. In diesem Diagramm lassen sich daraus die Nachbarschaftsverhältnisse innerhalb des simulierten Schulgebäudes ablesen.

2.2.3. Anordnung der Funktionsbereiche

Dieses Nachbarschaftsdiagramm dient in weiterer Folge als Grundlage für die Verteilung der Bereiche auf die Geschoße des Gebäudes. Dazu muss zunächst die Anzahl der Geschoße sowie die maximale Geschoßfläche von dem/der NutzerIn festgelegt werden. Andere Parameter wie beispielsweise eine freibleibende Restfläche auf den Geschoßen oder ähnliches können ebenso in die Simulation einfließen. Im letzten Schritt der Planungsunterstützung werden die Bereiche mit Hilfe eines Algorithmus, basierend auf dem zuvor erstellten Nachbarschaftsdiagramm, auf die vorgegeben Geschoße angeordnet. Dabei sucht sich in jeder Zeiteinheit ein Funktionsbereich einen Platz im Gebäude, bis alle Bereiche verteilt sind. (siehe Abb. 2.5)

2.3. Ergebnis

Das Endergebnis der Simulationen ist ein räumliches Nachbarschaftsdiagramm aller Funktionsbereiche des simulierten Schulgebäudes inklusive eines Vorschlages der daraus resultierenden bestmöglichen Anordnung dieser Bereiche auf Geschoßebenen.

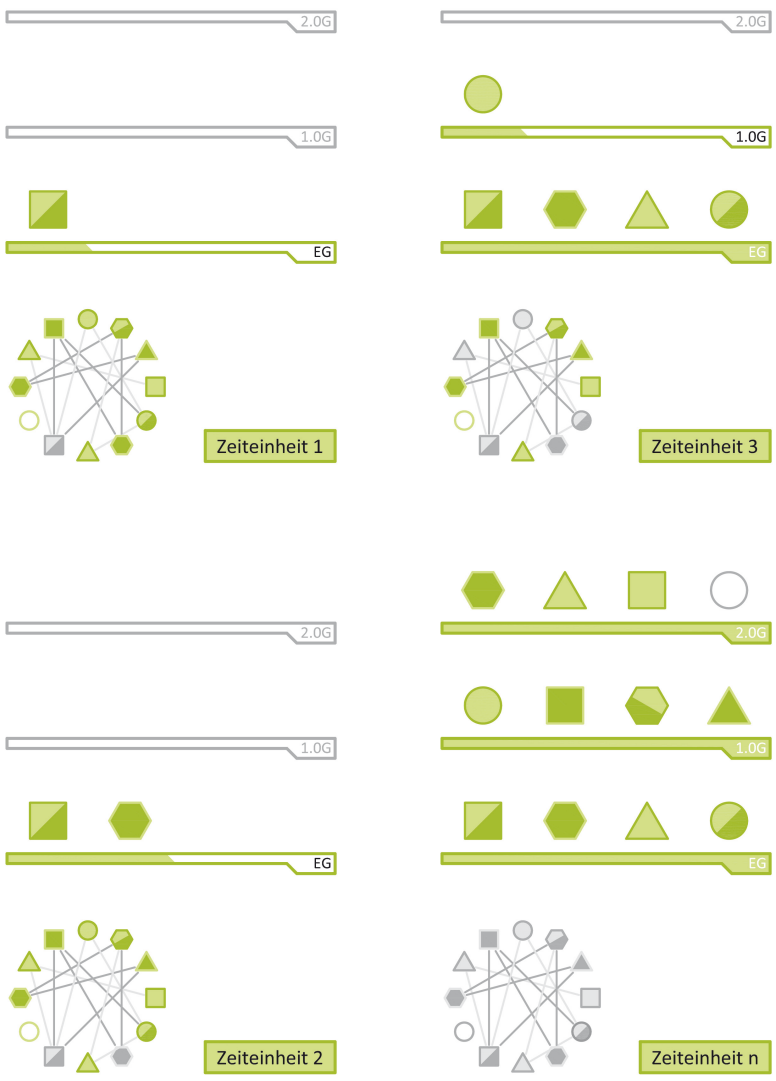
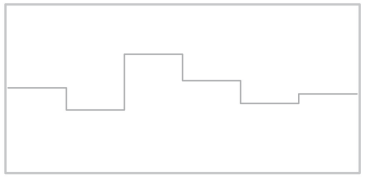
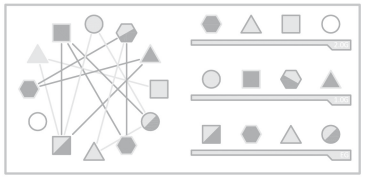
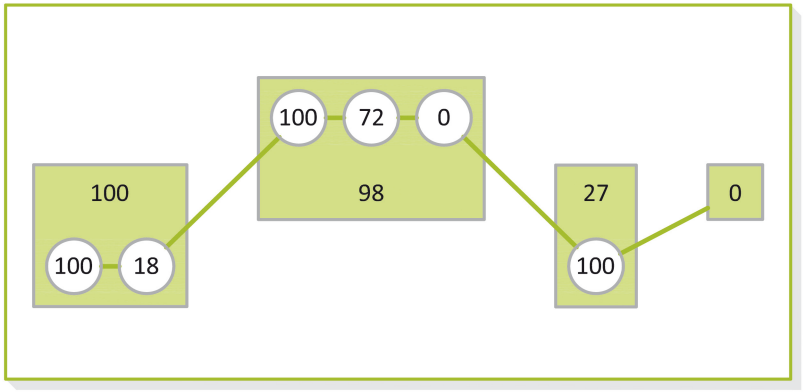
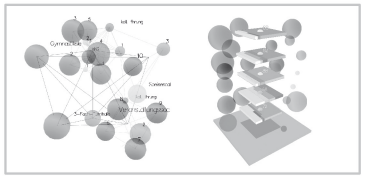


Abb. 2.5 Anordnung der Bereiche auf Geschoßebene









3. Implementierung

3.1. Regionale Bedarfsplanung **41**

3.1.1. Input 41

3.1.2. Parametrisierung 42

3.1.3. Ebenen der Simulation 42

3.1.4. Bedarfsdeckung zusätzlicher Klassen 42

3.1.5. Graphischer Aufbau der Simulation 43

3.1.6. Ergebnis 44

3.2. Lokale Ressourcenplanung und Anforderungsgenerierung **44**

3.2.1. Begriffsklärung 44

3.2.2. Input 45

3.2.2.1. Hierarchischer Aufbau des Schulgebäudes 45

3.2.2.2. Pädagogisches Konzept 46

3.2.2.3. AkteurInnen der Schule als Agenten der Simulation 47

3.2.3. Erstellung eines Nachbarschaftsdiagramms 48

3.2.3.1. Parametrisierung 48

3.2.3.2. Graphischer Aufbau der Simulation 49

3.2.3.3. Simulation der Bewegungsabläufe in einer Schulwoche 50

3.2.4. Algorithmus zur Verteilung 54

3.2.4.1. Parametrisierung 54

3.2.4.2. Verschiedene Methoden der Verteilung 55

3.2.4.3. Implementierter Verteilungsalgorithmus 59

3.2.4.4. Graphischer Aufbau der Simulation 64

3.2.4.5. Anpassung der Geschoßfläche und Rerun 64

3.2.5. Ergebnis 64

3. Implementierung

Dieses Kapitel beschäftigt sich mit der Umsetzung der, im vorhergehenden Abschnitt erläuterten, Idee, ein Hilfsprogramm als Unterstützung in frühen Phasen des Planungsprozess von Schulbauten zu entwickeln. Zur Implementierung kommt die Anwendung "NetLogo" zum Einsatz, ein Programm, das zur Simulation von Agenten-basierten Prozessen dient. Um die exakte Reproduzierbarkeit der Simulationen zu ermöglichen, kann eine Wert für einen sogenannten "Random-Seed" festgelegt werden. Bei gleichem Wert und gleicher Programmversion kommt es immer zum selben Ergebnis.

Im einem ersten Schritt der "Regionalen Bedarfsplanung" wird zunächst in einer städtebaulich aufgebauten Simulation der Bedarf an zusätzlich erforderlichen Klassenräumen einer Region anhand prognostizierter Entwicklungen der SchülerInnenzahlen eruiert. Anschließend kommt im darauffolgenden Schritt der "Lokalen Ressourcenplanung und Anforderungsgenerierung" das eigentliche Hilfsprogramm zu Einsatz, um eine Unterstützung in der Planung einzelner Schulbauten, etwa aus der vorhergehenden Bedarfsplanung resultierend, zu bieten. Dabei werden schlussendlich Vorschläge für die räumliche Anordnung der Funktionsbereiche des simulierten Schulgebäudes erstellt.

3.1. Regionale Bedarfsplanung

Die regionale Bedarfsplanung dient der Ermittlung des Bedarf an neuen Klassenräumen bzw. Schulbauten über einen bestimmten Zeitraum anhand von regionalen Prognose-Daten zur erwarteten Entwicklung der SchülerInnenzahlen einer Region.

3.1.1. Input

Die Simulation wird mit Excel-Listen als Input gespeist, die als Informationen die SchülerInnenzahlen sowie die entsprechenden Klassenzahlen der zu simulierenden Region für das Ausgangsjahr der Simulation enthalten müssen. Zusätzlich benötigt das Programm prognostizierte Daten der Entwicklungen der SchülerInnenzahlen über den gesamten zu simulierenden Zeitraum. Sofern diese Informationen nicht für alle Jahre gesondert zur Verfügung stehen, werden die fehlenden Werte bei der Importierung automatisch aus den vorliegenden berechnet. Um ein bestmögliches Ergebnis zu erzielen, sollten die Daten nach Unterbereichen der Region und nach Schultypen sortiert sein. Zusätzlich kann eine Karte der Region importiert werden, was die graphische Lesbarkeit der Simulation erleichtert.

3.1.2. Parametrisierung

Der/die NutzerIn kann wahlweise alle Schultypen der Region auf einmal simulieren lassen oder bewusst einzelne davon auswählen. Ebenso gibt es die Option, die Simulation entweder chronologisch durchlaufen zu lassen, oder durch die Betätigung eines Sliders einzelne Jahre gezielt aufzurufen.

3.1.3. Ebenen der Simulation

Der Aufbau der Simulation besteht aus mehreren Ebenen: die gesamte Region, deren Unterbereiche, Schultypen, Klassenzahlen, SchülerInnenzahlen und Jahre. Diese Schichten greifen ineinander und beeinflussen sich gegenseitig. Kommt es zu einer Änderung der Daten in einer Ebene, hat dies wechselseitige Auswirkungen auf die jeweils anderen. (siehe Abb. 3.1)

3.1.4. Bedarfsdeckung zusätzlicher Klassen

Die Simulation wird in Zeiteinheiten durchgeführt, die den zu simulierenden Jahren entsprechen. In der ersten Zeiteinheit gleicht die Simulation die SchülerInnenzahlen sowie die Klassenzahlen aller Schultypen in allen Bereichen der Region im Ausgangsjahr der Daten miteinander ab. Sollte dabei der Grenzwert von durchschnittlich 25 SchülerInnen pro Klasse überschritten werden, kommt es zu einer Aufstockung der Klassenzahlen für jenen Schultyp in jenem Bereich, in

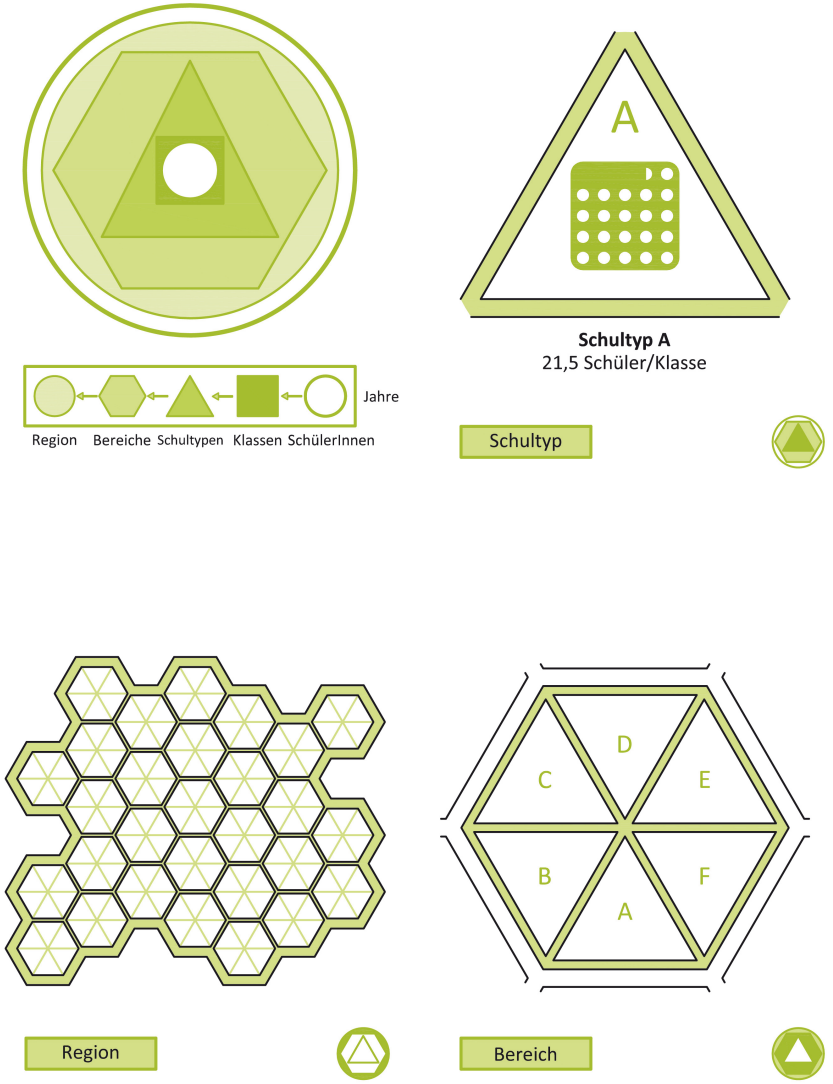


Abb. 3.1 Ebenen der Simulation

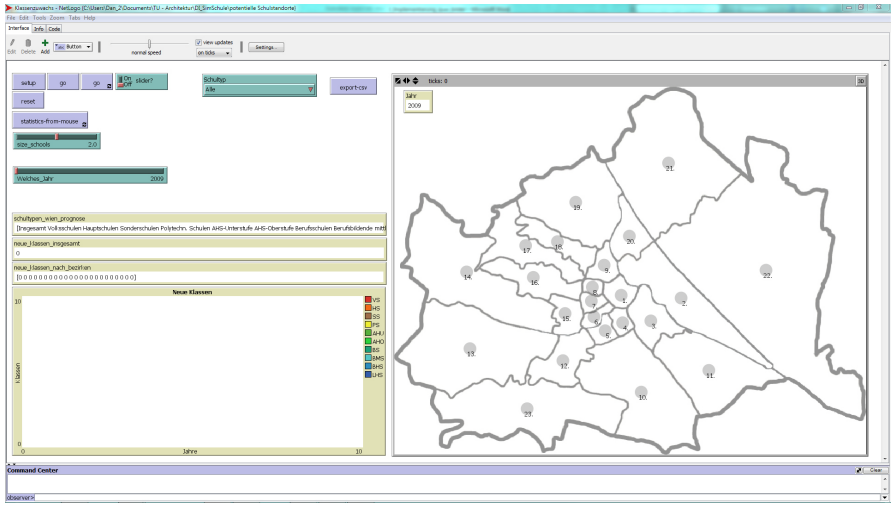


Abb. 3.2 Benutzeroberfläche: Bedarfssimulation

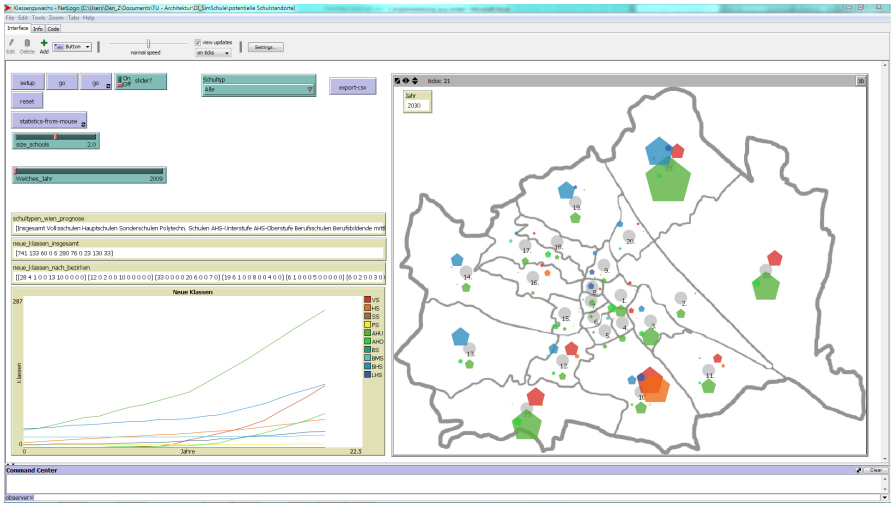


Abb. 3.3 Ergebnis Bedarfssimulation

dem der Wert überschritten wurde, bis der Grenzwert unterschritten wird. In der darauffolgenden Zeiteinheit werden die prognostizierten SchülerInnenzahlen des Folgejahres mit den bereits aktualisierten Klassenzahlen, inklusive der in der letzten Zeiteinheit hinzugefügten Klassen, abgeglichen. Abermals werden die Klassenzahlen angepasst, falls der Grenzwert überschritten wird. Dies geschieht schließlich für alle Jahre des zu simulierenden Zeitraums. Sowohl die Gesamtzahl der hinzugefügten Klassen, als auch deren Aufteilung auf die einzelnen Unterbereiche und Schultypen ist stets erkennbar.

3.1.5. Graphischer Aufbau der Simulation

Als graphische Oberfläche der Simulation kann eine importierte Karte der Region, welche auch die Unterbereiche zeigt, verwendet werden. (siehe Abb. 3.2) Die einzelnen Bereiche stellen in jeder Zeiteinheit farblich dar, ob und wie viele Klassen zusätzlich im entsprechenden Jahr gebraucht werden.

Ebenso werden die jeweils neu hinzugefügten Klassenräume in ihren jeweiligen Bereichen, nach Schultypen unterscheidbar, abgebildet. Zusätzlich hat der/die NutzerIn die Option, einzelne Unterbereiche mit

dem Maus-Cursor auszuwählen, um genauere Information der entsprechenden Klassenzahlen zu erhalten.

3.1.6. Ergebnis

Schlussendlich zeigt die Simulation den Bedarf an neuen Klassen für alle Schultypen und Bereiche der Region für jedes Jahr gesondert bzw. über den gesamten untersuchten Zeitraum gesammelt. (siehe Abb. 3.3)

Die Ergebnisse werden wiederum als Excel-Listen ausgegeben, wobei die Output-Daten hierbei beliebig nach Jahren, Bereichen und Schultypen ausgewertet werden können. Diese Daten können in der städtebaulichen Planung als Entscheidungsgrundlage bezüglich Lage, Größe und erforderlicher Schultypen zukünftiger Bildungsbauten fungieren.

3.2. Lokale Ressourcenplanung und Anforderungsgenerierung

Das in diesem Kapitel besprochene Hilfstool der "Lokalen Ressourcenplanung und Anforderungsgenerierung" kann zur Erstellung räumlicher, geschoßweise gegliederter Nachbarschaftsdiagramme als Planungsunterstützung für Bildungsbauten eingesetzt werden. Es ist in mehreren aufeinanderfolgenden Simulationen aufgebaut, die der Übersichtlichkeit halber aufgeteilt und der Reihe nach abgehandelt werden. In einer ersten Simulation wird ein dreidimensionales Diagramm der räumlichen Nachbarschaftsverhältnisse der simulierten Schule erstellt. Anschließend werden die diversen Funktionsbereiche des Gebäudes anhand der zuvor gewonnen Daten auf Geschoßebenen angeordnet. Das Hilfsprogramm kann zwar auch zur Analyse der Beziehungen der Funktionsbereiche von Schulbauten eines einzelnen Schultyps eingesetzt werden, entfaltet sein volles Potenzial allerdings bei der Simulation eines komplexeren Bildungscampus, bei dem mehrere Schultypen in räumlicher Nähe, etwa auf einem Grundstück, geplant werden.

3.2.1. Begriffsklärung (siehe Abb. 3.4)

Throughput (TP) Ⓐ

"Throughput" bezeichnet den Durchsatz der Kanten zwischen Knotenpunkten in Graphen. In der folgenden Simulation ist damit die Häufigkeit gemeint, mit der Schulklassen die jeweilige Verbindung auf ihren Wegen zwischen den Funktionsbereichen benutzen.

Closeness-Centrality (CC) Ⓑ

"Closeness-Centrality" bezeichnet den Kehrwert der durchschnittlichen Entfernung eines Knotenpunktes zu allen anderen mit ihm verbunden in einem Graphen. Je kleiner der Wert, desto zentraler liegt der Knoten. Ein Knotenpunkt, der keinerlei Verbindungen zu anderen aufweist, besitzt dabei immer eine "Closeness-Centrality" von 0.

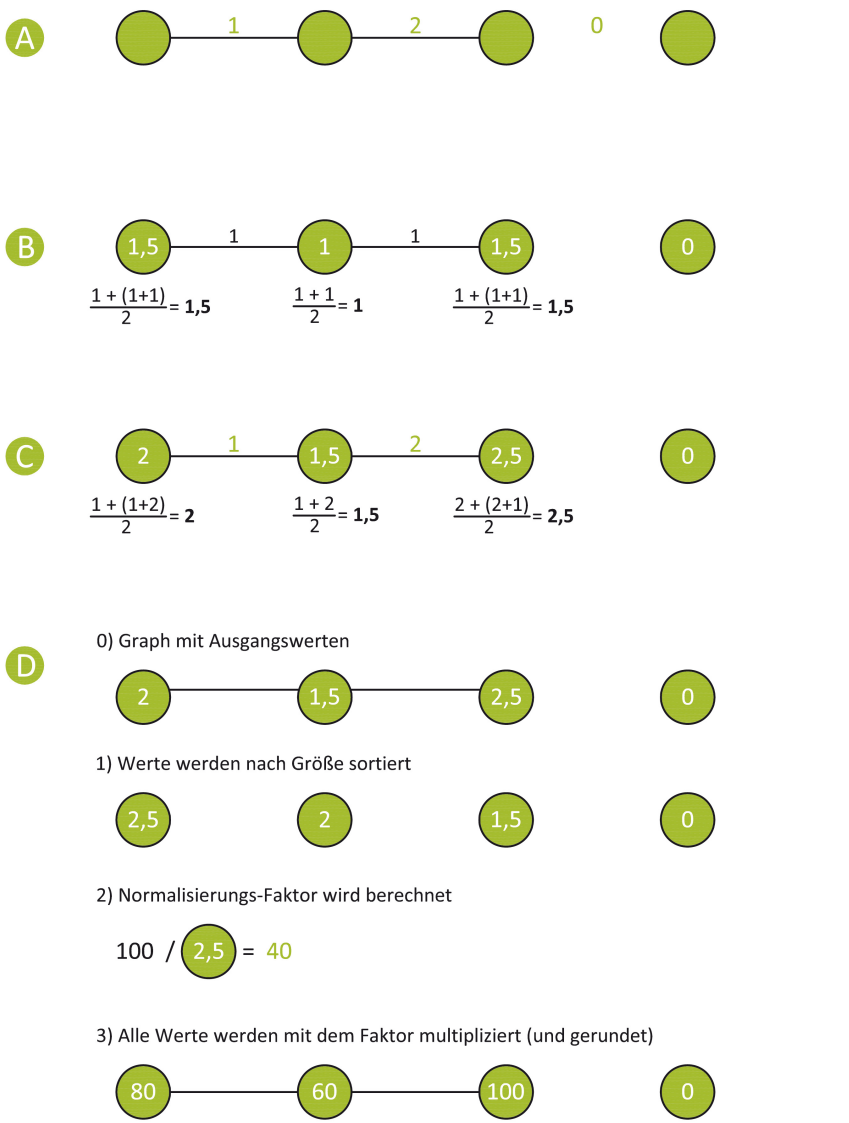


Abb. 3.4 Begriffsklärung

Im Zuge der Simulation wird mit dem Kehrwert der "Closeness-Centrality" gearbeitet, wodurch zentral liegende Knoten höhere Werte aufweisen, als dezentralere.

Weighted Closeness-Centrality (WCC) C

Zur Ermittlung der "Weighted Closeness-Centrality" wird anstelle der Entfernungen der Knotenpunkte zueinander ein anderer Wert der Kanten eingesetzt. In der folgenden Simulation wird dieser Wert der "Throughput" der Verbindungen sein.

Normalisierung D

Bei einer Normalisierung werden Werte mit einem Normalisierungsfaktor multipliziert, sodass sich ihr Verhältnis zueinander nicht ändert, sie jedoch auf einer beliebig wählbaren Skala liegen.

3.2.2. Input

Als Input für die Simulation dienen einerseits der Aufbau des zu simulierenden Gebäudes, andererseits die Anzahl der, nach Schultypen geordneten, SchülerInnen der Schule.

3.2.2.1. Hierarchischer Aufbau des Schulgebäudes

Das Schulgebäude wird in der Simulation grundsätzlich aus hierarchisch gegliederten Einheiten, bestehend aus höherrangigen Über- und

untergeordneten Unterbereichen, aufgebaut. (siehe Abb. 3.5) Abhängig von den Eigenschaften der konkreten Planungsaufgabe bezüglich Größe und pädagogischem Konzept, können diese Einheiten unterschiedlich definiert werden. Prinzipiell ist eine Unterteilung der Funktionsbereiche in kleinere Einheiten nicht zwangsläufig notwendig, doch bietet der hierarchische Aufbau die Möglichkeit, bereits bekannte oder geforderte Nachbarschaftsverhältnisse des zu simulierenden Gebäudes schon vorweg darzustellen und in die Simulation einfließen zu lassen. In einem kleineren Schulbau können beispielsweise Lerncluster oder Verwaltungsbereiche als Überbereiche definiert werden und die einzelnen Räume, aus denen sich diese zusammensetzen, also Klassenräume, Gruppenräume, oder Differenzierungsflächen als deren Unterbereiche. Genauso verhält es sich mit Verwaltungsbereichen oder Gemeinschaftsflächen, die in dem Fall ebenfalls als Überbereiche zu klassifizieren wären. In größeren Schulen oder einem Bildungscampus kann diese Einteilung anders vorgenommen werden. Eine Möglichkeit bestünde darin, ganze Schulzweige oder Schultypen als Überbereiche festzulegen und die darin enthaltenen Raumgruppen, beispielsweise Lerncluster oder Verwaltungsbereiche, als deren Unterbereiche.

Ein weiterer essentieller Input für die Simulation ist in jedem Fall die vorgegebene Raumgröße aller Bereiche, die im zweiten Abschnitt des Hilfsprogramms, der Anordnung auf Geschoßebenen, eine wesentliche Rolle einnehmen wird. Zusätzlich muss für jeden Funktionsbereich des Schulgebäudes exakt festgelegt werden, welche Unterrichtsfächer bzw. -methoden in ihm durchgeführt werden können.

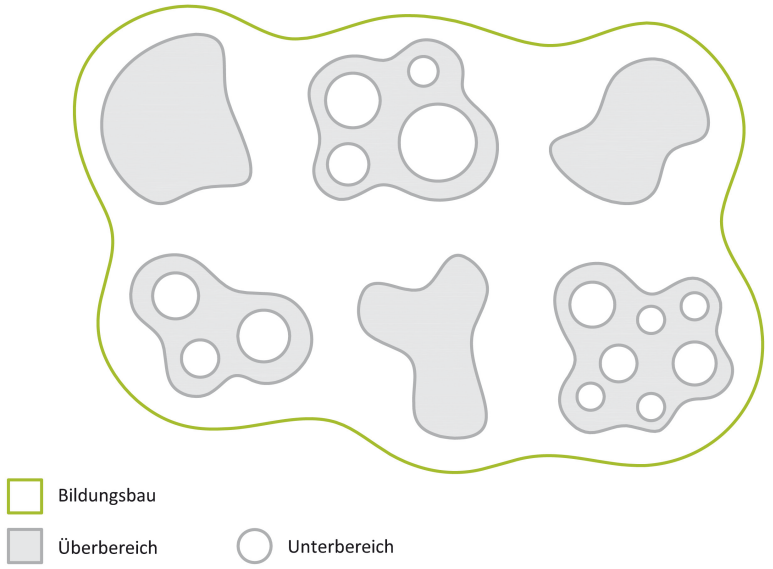


Abb. 3.5 Hierarchischer Aufbau

3.2.2.2. Pädagogisches Konzept

Einfluss auf die Simulation hat darüber hinaus auch das pädagogische Konzept der Schule. Dieses kann beispielsweise beeinflussen, ob alle Schulklassen einen eigenen Stammraum als "Homebase" besitzen, in dem sie so viele Unterrichtsfächer ihrer Stundentafel wie möglich abarbeiten,

oder diese stattdessen in jedem vergleichbaren Raum abgehalten werden können. Ebenso hat es Auswirkungen auf die Simulation vieler Funktionsbereiche des Gebäudes. Sind die Unterrichtsbereiche der zu simulierenden Schule etwa in Lerncluster zusammengefasst, stellt sich zusätzlich die Frage, welche Variante der Clusterbildung zum Einsatz kommt.

3.2.2.3. AkteurInnen der Schule als Agenten der Simulation

In einer Schule gibt es eine Vielzahl verschiedenerer AkteurInnen, die das Gebäude zeitlich und funktional unterschiedlich nutzen. Die folgende Simulation befasst sich primär mit den HauptakteurInnen und der zahlenmäßig größten dieser Gruppen, den SchülerInnen. Dies geschieht aus mehreren Überlegungen heraus. Ihre Bewegungsabläufe in einer Schulwoche lassen sich aufgrund der festgesetzten Stundentafeln für verschiedene Schulformen und Jahrgangsstufen, im Hinblick auf die räumlichen Nachbarschaftsverhältnisse eines Schulbaus, am gezieltesten simulieren. Daher bieten sie großes Potenzial zur Generierung eines Entwurfs für eine langjährig gut funktionierenden Schule. Im Vergleich dazu bieten etwa die Bewegungsabläufe des Lehrpersonals deutlich weniger Potenzial zur Feststellung oder Überprüfung der funktionalen Beziehungen innerhalb eines Schulgebäudes, da die Zusammensetzung des Lehrkörpers von Schule zu Schule und Jahr zu Jahr viel stärker variieren kann, was die Ableitung über einen langen Zeitraum gültiger Rückschlüsse erschwert. Zusätzlich kann die Nutzergruppe der SchülerInnen als die wichtigste bezeichnet werden, da sie in einer entscheidenden Entwicklungsphase durch ihre Erfahrungen in der Schule für ihr weiteres Leben geprägt werden.[66]

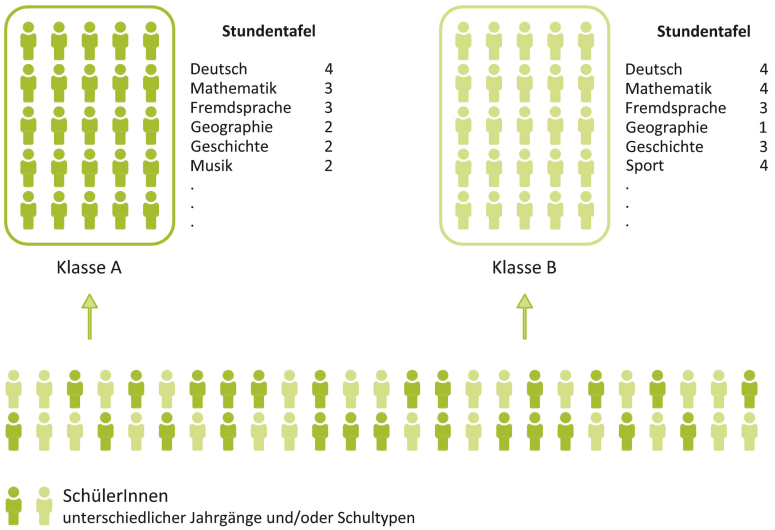


Abb. 3.6 Klassen als Agenten

Jede Schulklasse hat in Abhängigkeit ihres Schultyps und ihres Jahrgangs eine fixierte Stundentafel, welche ihr alle Unterrichtsfächer, inklusive der genauen Stundenanzahl, vorgibt, die sie in einer Schulwoche zu

[66] Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

absolvieren hat. (siehe Abb. 3.6) Die Stundentafeln variieren somit im Bezug der durchzuführenden Unterrichtsgegenstände, der Anzahl der Stunden der einzelnen Fächer sowie der Gesamtstunden. Einige der Unterrichtsfächer lassen sich nur in speziellen Räumlichkeiten abhalten, während andere relativ flexibel in verschiedenen Lernsituationen durchführbar sind. So erfordern Unterrichtsgegenstände wie Deutsch oder Mathematik anders als Sport und Bewegung oder Physik keine spezifischen Lernumgebungen.

3.2.3. Erstellung eines Nachbarschaftsdiagramms

3.2.3.1. Parametrisierung

Der/die NutzerIn hat die Möglichkeit festzulegen, in welcher zeitlichen Rhythmisierung die Schulklassen sich in der Simulation durch das Gebäude bewegen. (siehe Abb. 3.7)

- Potentieller Bereichswechsel nach jeder Zeiteinheit

In dieser Variante der Simulation suchen sich die Klassen in jeder Zeiteinheit ein neues Unterrichtsfach und damit potenziell einen neuen Bereich. Vorstellbar wären genauso andere Methoden, die sich auch aus dem pädagogischen Konzept der Schule ergeben können.

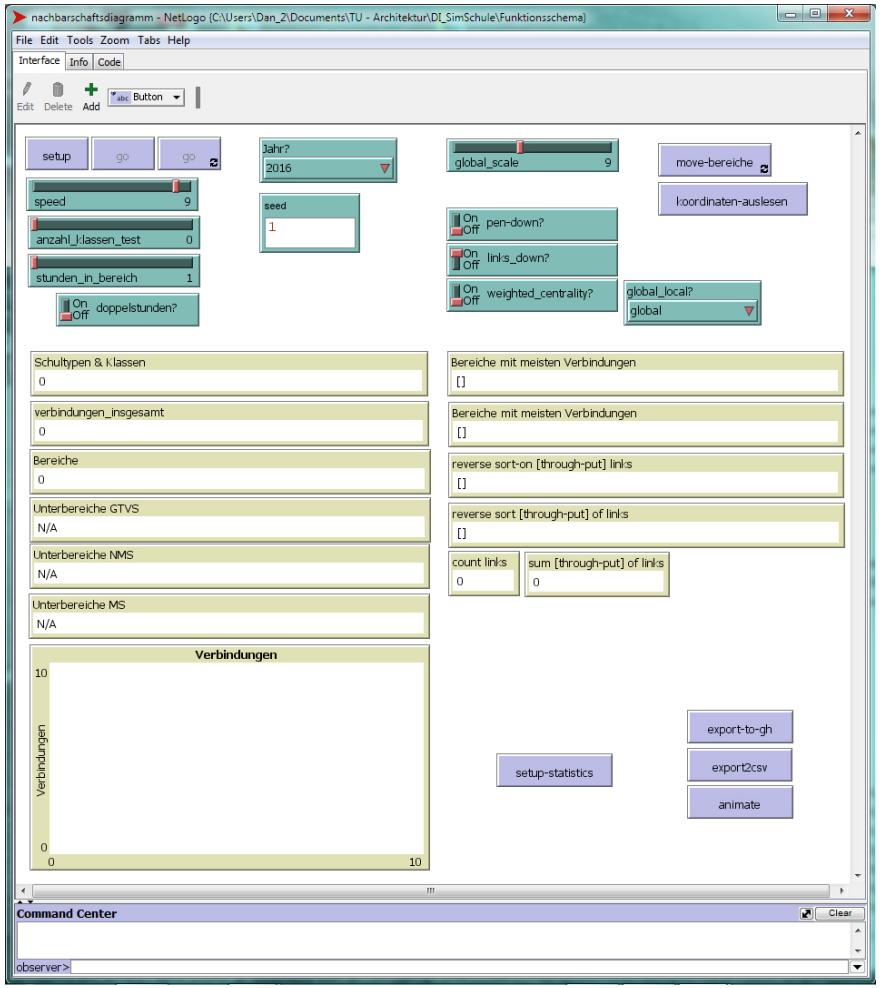


Abb. 3.7 Benutzeroberfläche: Simulation der SchülerInnenwege

- "Doppelstunden" des gewählten Unterrichtsfachs

Eine andere Möglichkeit besteht etwa darin, soweit möglich, immer Doppelstunden desselben Unterrichtsgegenstandes durchzuführen, während denen die Klassen ihren Bereich niemals wechseln. Ausnahmen sind nur dann zulässig, wenn nur mehr eine Stunde des gewählten Fachs in der Stundentafel enthalten ist.

- Längere Aufenthaltseinheiten im jeweiligen Bereich

Darüber hinaus können auch längere Aufenthalte in den Bereichen vorgesehen werden, in denen die Klassen zwar unterschiedliche Unterrichtsfächer absolvieren, aber dazwischen keinen Ortswechsel vollziehen. Ein vorzeitiger Bereichswechsel muss nur vorgenommen werden, falls in der Stundentafel keine Unterrichtsgegenstände mehr vorhanden sind, die im aktuellen Bereich durchgeführt werden können. Der maximalen Anzahl an Stunden, die Schulklassen in einem Bereich verbringen, sind dabei grundsätzlich keine Grenzen gesetzt, doch macht eine Begrenzung auf die Stundenanzahl eines Schultages Sinn.

3.2.3.2. Graphischer Aufbau der Simulation

Ein erster Schritt zur Erstellung eines Nachbarschaftsdiagramms des zu simulierenden Schulgebäudes ist die Umsetzung des Inputs in einen dreidimensionalen Graphen, wobei die einzelnen Funktionsbereiche die Rolle der Knotenpunkte einnehmen. Sie werden als Kugeln, entsprechend ihrer vorgegebenen Raumgröße in einem dreidimensionalen Raum definierter Größe verteilt. Ihre Anordnung zueinander basiert zu diesem Zeitpunkt rein auf einer möglichst gleichmäßigen Verteilung der hierarchisch gegliederten Bereiche. Dabei werden zunächst die Überbereiche der Schule in gleichbleibenden Abständen voneinander platziert. Im nächsten Schritt suchen sich die Unterbereiche jedes Überbereichs, wiederum mit gleichbleibenden Distanzen dazwischen, in unmittelbarer räumlicher Nähe ihrer jeweiligen Überbereiche einen Platz. All jene Überbereiche, denen Unterbereiche zugeordnet sind, dienen nur als "Platzhalter" und werden zu diesem Zeitpunkt graphisch und funktional von diesen ersetzt. In jedem Funktionsbereich sind Daten bezüglich seiner hierarchischen Beziehungen, seiner Funktion und der Unterrichtsgegenstände, zu deren Durchführung er sich eignet, gespeichert.

In einem nächsten Schritt werden alle SchülerInnen des zu simulierenden Schulbaus, bereits in Klassen zusammengefasst und gemäß ihres Schultyps eingefärbt, zunächst willkürlich als Agenten im dreidimensionalen Raum verteilt. Jede Klasse hat die Daten ihrer fixen Stundentafel sowie, wenn vom pädagogischen Konzept vorgegeben, eine "Homebase" in der Schule in sich gespeichert. Zu diesem Zeitpunkt gilt

das Setup als abgeschlossen und die Simulation kann durchgeführt werden.

Der/die NutzerIn hat die Option, sich während des Simulationsvorgangs entweder die zurückgelegten Wege der Klassen oder bereits die daraus abgeleiteten Verbindungen der Funktionsbereiche anzeigen zu lassen. Ebenso besteht die Möglichkeit, die Bereiche entsprechend ihrer "Weighted Closeness-Centrality"-Werte farblich darzustellen.

3.2.3.3. Simulation der Bewegungsabläufe in einer Schulwoche

Nach dem Abschluss des Grundsetups beginnt die eigentliche Simulation zur Erstellung des Nachbarschaftsdiagramms. In dieser suchen sich alle Klassen in jeder Zeiteinheit einen Bereich des Schulgebäudes, in dem sie ein noch offenes Unterrichtsfach ihrer Stundentafel absolvieren können. (siehe Abb. 3.8) Dazu wählen sie zunächst nach dem Zufallsprinzip einen Bereich, der sich für zumindest eines der Fächer eignet, die sie noch nicht erledigt haben. Sollte der gewählte Bereich die Durchführung mehrerer Unterrichtsgegenstände zulassen, die von der Klasse noch nicht absolviert wurden, wählen sie jenes davon, das zuerst in ihrer Stundentafel vorkommt. Danach löschen sie eine Stunde des in dieser Zeiteinheit gewählten Fachs aus ihrer aktualisierten Stundentafel. Für den Falls, dass dadurch keine Stunde des jeweiligen Unterrichtsgegenstandes mehr übrig ist, gilt das gesamte Unterrichtsfach für die simulierte Schulwoche als erledigt.

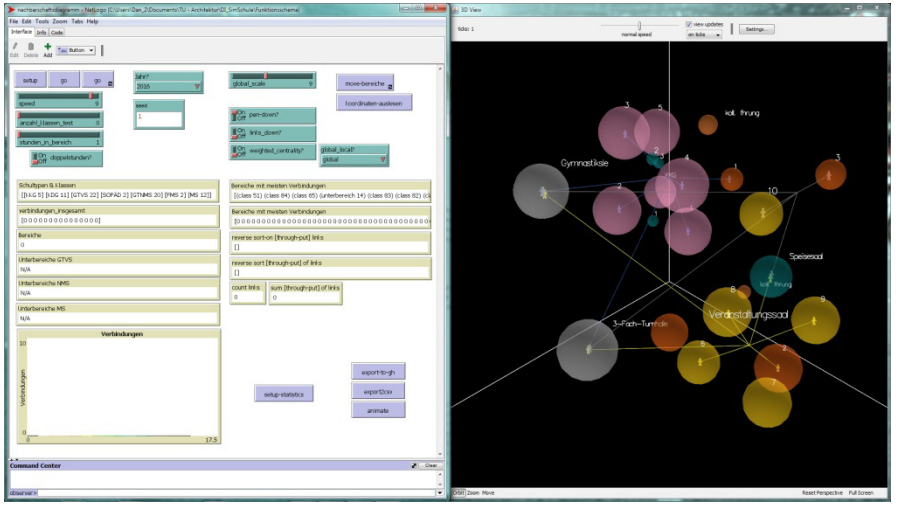


Abb. 3.8 Bewegungen der Klassen nach der ersten Zeiteinheit

Jedes Mal, wenn sich eine Klasse von einem Bereich zu einem anderen bewegt, wird eine Verbindung zwischen diesen hergestellt bzw. im Fall einer bereits bestehenden Verbindung, diese verstärkt. Der "Throughput" ist der Wert der Verbindung, der die Häufigkeit ihrer Verwendung angibt. Je öfter Klassen diese Verbindung nutzen, desto größer ist besagter Wert. Die Stärke ("Thickness") der Verbindungen werden graphisch in Abhängigkeit des "Throughputs" in jeder Zeiteinheit aktualisiert , wodurch Änderungen in Echtzeit von dem/der NutzerIn mitverfolgt werden können. Grundsätzlich kommt es zu einer Unterscheidung von drei

verschiedenen Verbindungsarten: jene zwischen zwei Überbereichen, zwischen zwei Unterbereichen desselben Überbereichs, sowie zwischen einem Überbereich und einem Unterbereich (eines anderen Bereichs). (siehe Abb. 3.9)

- Bewegt sich eine Klasse von einem Überbereich zu einem anderen Überbereich wird der "Throughput" der Verbindung der beiden erhöht. A

- Bewegt sich eine Klasse von einem Unterbereich zu einem anderen Unterbereich desselben Überbereichs, wird der "Throughput" der Verbindung der beiden erhöht. B

- Bewegt sich eine Klasse zwischen zwei Unterbereichen unterschiedlicher Überbereiche, wird ausschließlich der "Throughput" der Verbindung zwischen den beiden Überbereichen der Unterbereiche erhöht. C

- Bewegt sich eine Klasse von einem Überbereich zu einem Unterbereich (eines anderen Überbereichs) (oder umgekehrt) wird der "Throughput" der Verbindung zwischen dem Unterbereich und dem Überbereich sowie zwischen dem Überbereich und dem Überbereich des Unterbereichs erhöht. D

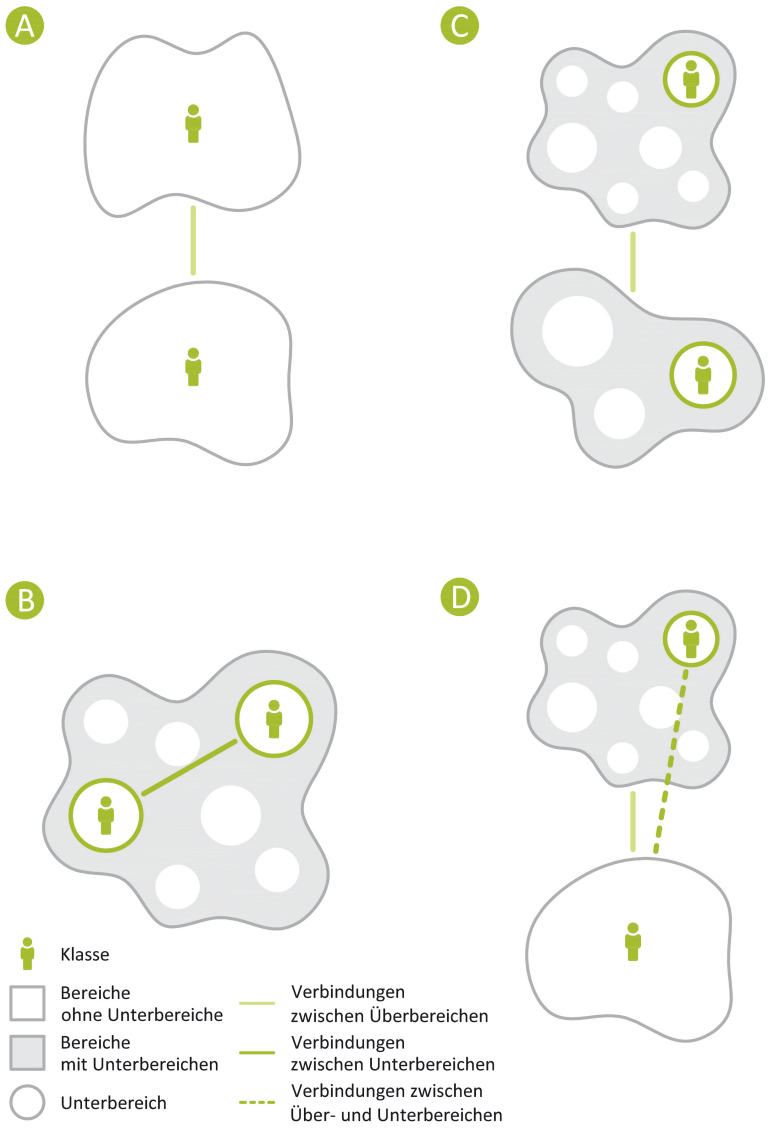


Abb. 3.9 Verbindungsarten

Diese unterschiedlichen Verbindungsarten gehen auf den hierarchischen Aufbau der simulierten Schule zurück und sorgen in weiterer Folge dafür, dass dieser auch in den nachfolgenden Simulationen nicht verloren geht. Sie werden graphisch zur leichteren Unterscheidung in verschiedenen Farben dargestellt. Die "Throughput"-Werte der Verbindungen kommen in weiterer Folge zur Berechnung der "Weighted Closeness-Centrality" aller Bereiche zum Einsatz. Zur Berechnung dieser Werte dienen die Funktionsbereiche des Schulbaus als Knotenpunkte und der durchschnittliche "Throughput" der Verbindungen wird anstelle der durchschnittlichen Entfernung eingesetzt. Jeder Bereich erhält zwei verschiedene "WCC"-Werte, einen globalen, der den durchschnittlichen "Throughput" zu allen anderen Bereichen anzeigt mit denen er verbunden ist, sowie einen lokalen, der für Überbereiche den durchschnittlichen "Throughput" zu allen anderen Überbereichen und bei Unterbereichen zu allen anderen Unterbereichen desselben Überbereichs angibt. (siehe Abb. 3.10) An diesem Punkt kommt der Vorteil der Unterscheidung der verschiedenen Verbindungsarten zum Einsatz, da zur Berechnung der Werte jeweils nur bestimmte Verbindungen herangezogen werden. Alle "WCC"-Werte werden der Übersichtlichkeit halber normalisiert, wodurch sie auf einer Skala zwischen 0 und 100 liegen. Das bedeutet, dass der Wert des Bereich mit der größten "Weighted Closeness-Centrality" auf 100 gesetzt wird und die anderen verhältnismäßig angepasst werden. Das

Abb. 3.10 Vergleich: Globale (links) und Lokale (rechts) WCC

geschieht sowohl für die globalen Werte aller Bereiche als auch für die lokalen innerhalb der jeweiligen hierarchischen Einheiten. Grundsätzlich gibt der "WCC"-Wert Auskunft darüber, wie zentral ein Bereich im Verhältnis zu den anderen, mit ihm verbundenen, im Gebäude liegen sollte. Je größer der Wert, desto leichter sollte der Bereich von allen verbundenen aus erreichbar sein. Dieser Durchlauf der Simulation gilt als abgeschlossen, wenn alle Klassen alle Stunden ihrer Stundentafeln absolviert und somit ihre simulierte Schulwoche gänzlich absolviert haben. (siehe Abb. 3.11)

Da die Abfolge, in der die Klassen ihre Stundentafeln abarbeiten, auch Auswirkungen auf die Reihenfolge hat, in der sie sich zwischen den

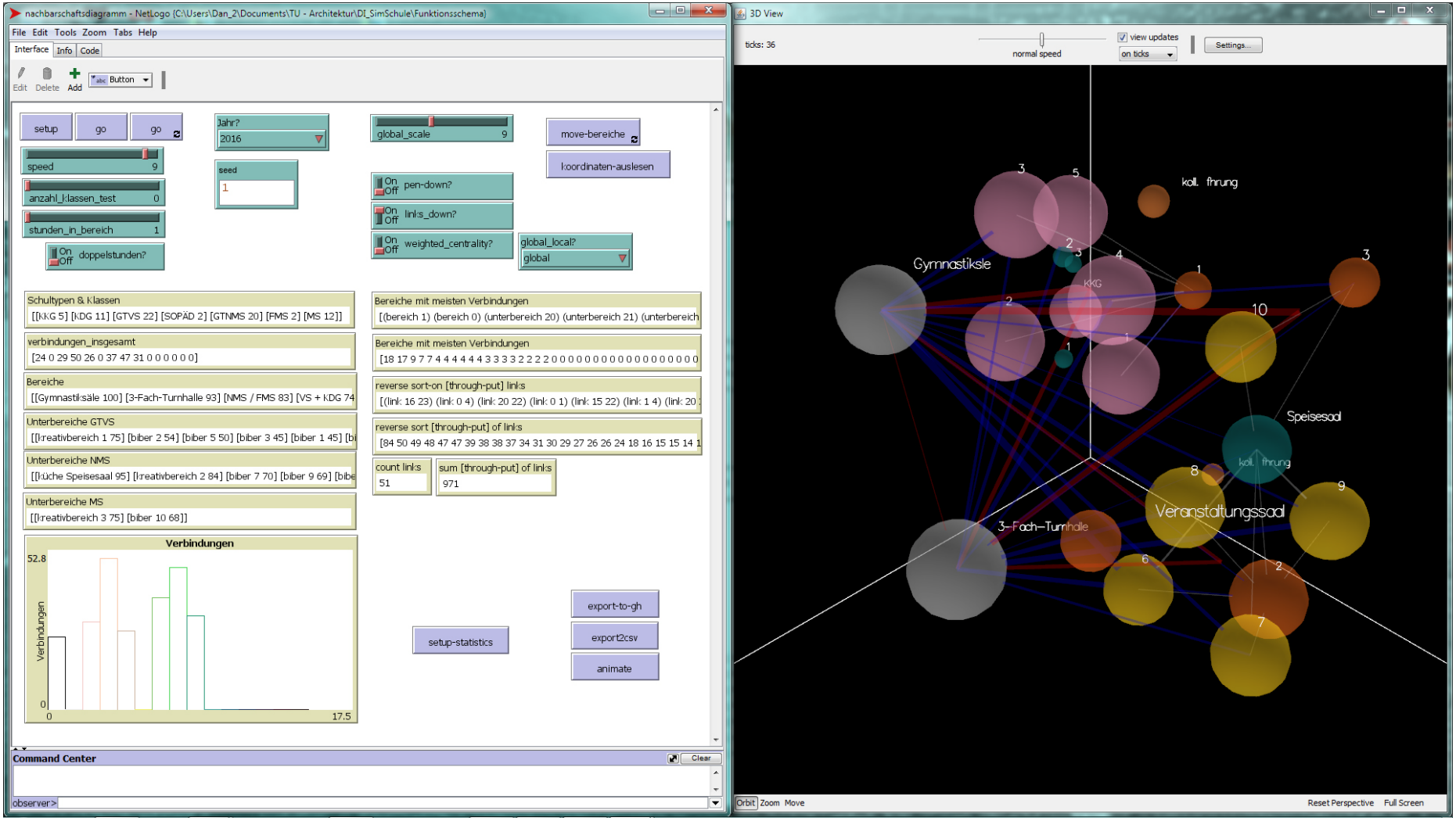


Abb. 3.11 Verbindungen am Ende der Simulation

Bereichen bewegen und dadurch die Verbindungen erstellen, ist es sinnvoll, diese Simulation oftmals durchzuführen, um Durchschnittswerte für die "Throughputs" der Verbindungen zu erhalten. Somit werden im Grunde die Stundentafeln der Klassen wieder und wieder zu veränderten Stundenplänen zusammengesetzt, anhand derer ihre Bewegungsabläufe in der Schulwoche simuliert werden. Dies ist in einer echten Schule genauso der Fall, da sich auch hier die Stundenpläne der Klassen über den gesamten Lebenszyklus des Gebäudes von Jahr zu Jahr ändern, aber stets aus über viele Jahre gleichbleibenden Stundentafeln gebildet werden.

3.2.4. Algorithmus zur Verteilung

Im nächsten Abschnitt des Hilfsprogramms werden die gewonnen Daten bezüglich der Häufigkeit der genutzten Verbindungen der Bereiche des Schulgebäudes weiter verarbeitet, um einen Vorschlag einer geschoßweisen räumlichen Anordnung zu erstellen.

3.2.4.1. Parametrisierung

In dieser Phase der Simulation hat der/die NutzerIn verschiedene Möglichkeiten der Beeinflussung des letztendlichen Ergebnisses. Als Eingabeparameter benötigt dieser Teil des Hilfsprogramms zunächst die zuvor ermittelten Daten bezüglich der Nachbarschaftsverhältnisse innerhalb des Schulgebäudes. Als Schnittstelle dienen CSV-Dateien (comma-separated-values), welche alle relevanten Informationen

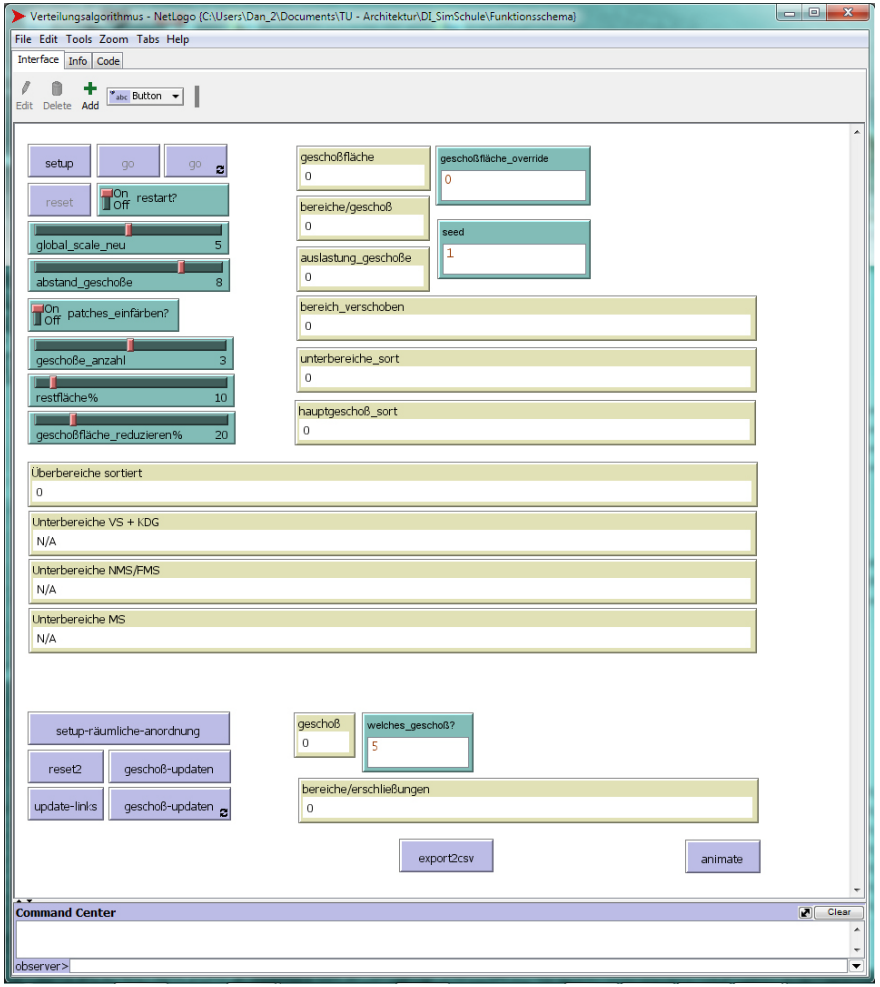


Abb. 3.12 Benutzeroberfläche: Verteilungssimulation

bezüglich der Nachbarschaften, die "Throughput"-Werte aller Verbindungen sowie die globalen und lokalen WCC-Werte aller Bereiche

enthalten. Als zusätzlicher Input dienen einige von dem/der NutzerIn festzulegende Parameter. Dazu zählen die Anzahl an Geschoßebenen des Schulgebäudes, die maximale Geschoßfläche und eine prozentual anzugebende Restfläche, die auf jedem Geschoß mindestens frei bleiben muss. Darüber hinaus besteht die Möglichkeit, einen ebenso in Prozent anzugebenden Wert für die potenzielle Verkleinerung der Geschoßfläche anzugeben. Dieser Wert kommt aber ausschließlich am Ende jeder Simulationsrunde zum Einsatz. (siehe Abb. 3.12)

In einem Grundsetup werden zunächst diese Parameter in eine dreidimensionale Darstellung umgewandelt. Darin sind die Geschoßanzahl, deren Fläche sowie das Verhältnis zur Grundstücksgröße ablesbar. Die von dem/der NutzerIn eingegebenen Parameter werden dabei auf eine bestimmte Anzahl von "Patches" umgerechnet. Dabei handelt es sich um die räumlichen Einheiten des Simulationsprogramms, dreidimensionale Würfel, aus denen der gesamte Raum aufgebaut ist und die nun entsprechend eingefärbt werden.

3.2.4.2. Verschiedene Methoden der Verteilung

Grundsätzlich gibt es eine Vielzahl unterschiedlicher Methoden und Algorithmen, die zur Anordnung der Bereiche eingesetzt werden können und den anderen gegenüber jeweils gewisse Vor- und Nachteile bieten. Dieser Abschnitt beschäftigt sich mit einigen davon, bevor im Anschluss die zur Implementierung gewählte Methode der Simulation im Detail erläutert wird. Alle hier besprochenen Algorithmen verbindet, dass in der Simulation jeweils ein Bereich pro Zeiteinheit auf einem Geschoß platziert wird, bis alle ein passendes gefunden haben. Ein wesentlicher Aspekt der Algorithmen ist die Ermittlung der Reihenfolge, in der die Bereiche auf die Geschoßebenen verteilt werden. Die maximale Auslastung der Geschoße wird aus der von dem/der NutzerIn vorgegebenen Geschoßfläche, abgezogen des prozentualen Werts für die freibleibende Restfläche, berechnet und darf für keines der Geschoße überschritten werden. Die Simulation gilt jeweils als abgeschlossen, wenn alle Funktionsbereiche der Schule einen Platz auf einem Geschoß gefunden haben.

- Ausgangsgeschoß

Eine grundlegende Frage der Anordnung ist das Ausgangsgeschoß bzw. auf welchem Geschoß in der ersten Zeiteinheit der anzuordnende Bereich platziert wird. Als Optionen stehen das unterste (Erdgeschoß) und das jeweils oberste Geschoß zur Verfügung. In den folgenden Algorithmen zur Verteilung wurde stets das unterste Geschoß als Ausgangspunkt gewählt, da sich im Zuge der Simulation kaum Unterschiede zwischen dieser und anderen Optionen herausstellten.

- Verteilung nach "Globaler WCC"

Dieser Algorithmus zur Verteilung betrachtet alle Bereiche der simulierten Schule als gleichberechtigt, egal ob es sich dabei um Über- oder Unterbereiche handelt. Daher kommen zur Bestimmung der Reihenfolge, in der sie auf Geschoße verteilt werden, bei allen die globalen WCC-Werte zum Einsatz. In einem ersten Schritt werden alle Bereiche nach diesen Werten sortiert, also danach wie leicht sie im Schulgebäude erreichbar sein sollten.

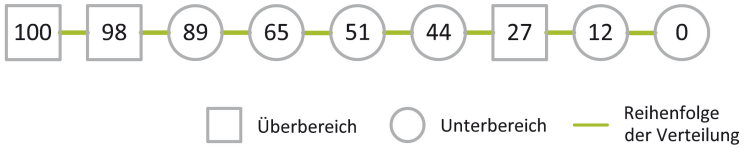


Abb. 3.13 Verteilung nach "Globaler WCC"

Anschließend werden die Bereiche der Sortierung entsprechend, in absteigender Reihenfolge ihrer WCC-Werte auf den Geschoßen platziert. In der ersten Zeiteinheit wird demnach der Bereich mit dem größten Wert auf die unterste Ebene, das Erdgeschoß, gesetzt. An diesem Punkt kommt erstmals die Raumgröße der Funktionsbereiche zur Anwendung. Diese wird von der maximalen Geschoßfläche des Erdgeschoßes abgezogen, um dessen Auslastung zu überprüfen. Danach wird weiterhin ein Bereich pro Zeiteinheit auf dem Erdgeschoß platziert, bis dessen maximale Auslastung überschritten wird. Zu diesem Zeitpunkt wird der aktuelle Bereich stattdessen auf das nächsthöhere Geschoß verschoben. Die Verteilung der Bereiche gilt als abgeschlossen, wenn der letzte, also der Bereich mit dem niedrigsten WCC-Wert platziert wurde. (siehe Abb. 3.13)

Der augenscheinlichste Nachteil dieses Anordnungsalgorithmus ist die Gleichbehandlung von Über- und Unterbereichen, wodurch der hierarchische Aufbau des Schulgebäudes weitgehend ignoriert wird, was zur Folge haben kann, dass die einzelnen Unterbereiche desselben Überbereichs über viele Geschoße verteilt sind. Somit kann es vorkommen, dass bei diesem Algorithmus vor allem Verwaltungs- oder Mitarbeiterbereiche weit entfernt von anderen Bereichen liegen, die eigentlich in unmittelbarer räumlicher Nähe situiert sein sollten. Der Grund hierfür liegt in der Simulation zur Erstellung des Nachbarschaftsdiagramms, welches als Grundlage der Verteilung dient. Dieses berücksichtigt zwar die zurückgelegten Wege der SchülerInnen, erfasst aber nicht jene der anderen AkteurInnen der Schule.

- Verteilung nach "Link-Neighbors"

Eine weitere Möglichkeit der Verteilung basiert auf sogenannten "Link-Neighbors". Als "Link-Neighbors" werden alle Bereiche bezeichnet, die mit einem Bereich direkt verbunden sind. Im diesem konkreten Fall des

Nachbarschaftsdiagramms trifft dies für jeden Bereich auf alle jene anderen Bereiche zu, von denen sich zu ihm hin oder zu denen sich von ihm weg mindestens eine Klasse bewegt hat.

Bei diesem Algorithmus werden zunächst wieder alle Bereiche nach ihren globalen WCC-Werten sortiert. Danach sucht sich in der ersten Zeiteinheit, analog zum letzten abgehandelten Algorithmus, ebenso jener Bereich mit dem größten Wert ein Geschoß. Nun kommt es zur ersten Abwandlung der Methodik. Im nächsten Schritt werden die "Link-Neighbors" des zuletzt platzierten Bereichs nach ihren globalen WCC-Werten sortiert. Der Reihe nach wird jetzt in jeder Zeiteinheit jener von ihnen mit dem nächsthöheren Wert auf einem Geschoß platziert, bis alle einen Platz gefunden haben. Dabei gilt wiederum der Grundsatz, dass bei kompletter Auslastung eines Geschoßes das nächsthöhere genutzt wird.

Sind auf diese Weise alle "Link-Neighbors" des ersten verschobenen Bereichs platziert, sucht sich jener Bereich mit dem größten globalen WCC-Wert, der noch nicht angeordnet wurde, seinen Platz. Daraufhin kommt es wieder in gleicher Weise zur Verteilung dessen "Link-Neighbors". (siehe Abb. 3.14)

Zwar können auch bei diesem Algorithmus die einzelnen Unterbereiche desselben Überbereichs über viele Geschoße verstreut sein, doch sind im Gegensatz zum ersten Algorithmus die meisten Bereiche zwischen denen sich Klassen bewegen, in unmittelbarer Umgebung. Ein Problem tritt allerdings auf, falls ein Bereich zu den "Link-Neighbors" mehrerer verschobener Bereiche zählt. In diesem Fall gesellt er sich räumlich zum jeweils zuerst verschobenen, wodurch er von den anderen weiter entfernt sein kann.

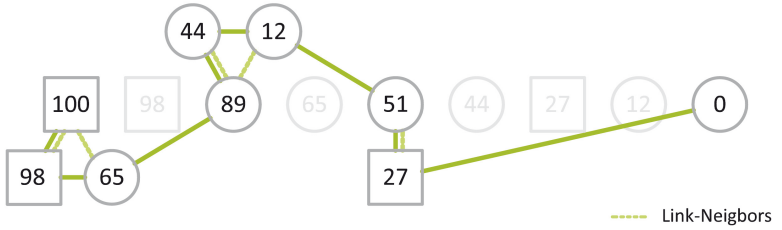


Abb. 3.14 Verteilung nach "Link-Neighbors"

- Hierarchische Verteilung

Eine dritte Möglichkeit bietet ein Algorithmus, der sowohl die Hierarchie der Bereiche als auch die globalen und lokalen WCC-Werte zur Verteilung nutzt. In diesem Fall werden im ersten Schritt nur die Überbereiche nach ihren WCC-Werten sortiert. In der ersten Zeiteinheit sucht sich nun der Überbereich mit dem größten Wert einen Platz auf einem Geschoß. Besitzt der platzierte Überbereich keine Unterbereiche, wird in der nächsten Zeiteinheit der Überbereich mit dem nächsthöheren Wert verschoben. Besitzt er allerdings Unterbereiche, so werden diese nach

ihren lokalen WCC-Werten sortiert und anschließend in absteigender Reihenfolge auf Geschoßen platziert. Nach der Anordnung aller Unterbereiche des verschobenen Bereichs, sucht sich wiederum der Überbereich mit dem nächsthöchsten Wert ein Geschoß. (siehe Abb. 3.15)

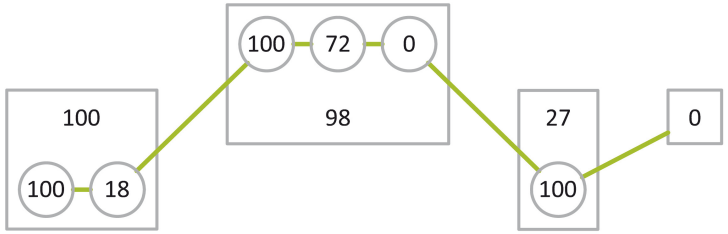


Abb. 3.15 Hierarchische Verteilung

Der große Vorteil dieses Algorithmus zur Verteilung der Bereiche liegt in der Tatsache, dass hier, durch die Übernahme des hierarchischen Aufbaus der Schule, auch jene Unterbereiche richtig platziert werden, die im Diagramm der Nachbarschaftsverhältnisse keine Verbindungen zu anderen aufweisen, aber dennoch in deren Nähe angeordnet sein sollten. Davon betroffen sind vor allem Verwaltungs- und Mitarbeiterbereiche, die aufgrund der ausschließlichen Erfassung der Bewegungsabläufe der SchülerInnen in der Erstellung des Nachbarschaftsdiagramms, nicht gleichwertig in die Simulation eingebunden sind.

Ein Problem kann bei der Verteilung dann auftreten, wenn ganze Überbereiche der Schule keine Verbindungen zu anderen aufweisen, aber dennoch zentral angeordnet sein sollten. In diesem Fall führt kein Mittel an einem manuellen Eingriff in die Simulation vorbei.

- Manuelle Verteilung

Falls spezielle Rahmenbedingungen für Bereiche des Schulgebäudes gelten, welche in der Simulation der Schülerwege keine Berücksichtigung finden, können diese Bereiche auch manuell auf passenden Geschoßen angeordnet werden. (siehe Abb. 3.16) Dieser Fall kann beispielsweise eintreten, falls in den Ausschreibungsunterlagen eines Wettbewerbs für einen Schulbau gewisse räumliche Zusammenhänge als zwingend notwendig vorgegeben werden.

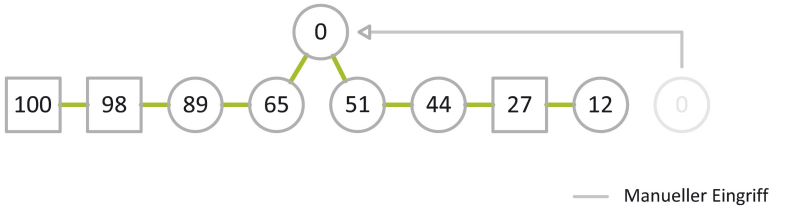


Abb. 3.16 Manuelle Verteilung

- Hauptgeschoß

Die Idee des Hauptgeschoßes bietet einen Lösungsansatz für ein Problem, das bei den anderen Algorithmen der Verteilung auftreten kann. Theoretisch besteht bei diesen die Möglichkeit, dass der Unterbereich mit dem größtem lokalen WCC-Wert, also jener Bereich, der von den anderen Unterbereichen desselben Überbereichs am leichtesten zu erreichen sein sollte, auf einem anderen Geschoß liegt als die übrigen.

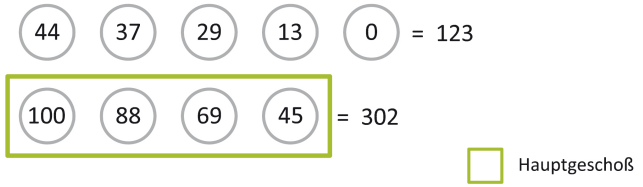


Abb. 3.17 Hauptgeschoß

Ein Lösungsansatz für diese Problemstellung ist die Festlegung eines sogenannten Hauptgeschoßes für jeden Überbereich. Dieses wird als jenes Geschoß definiert, auf dem sich der Großteil der Gesamtsumme aller lokalen WCC-Werte der Unterbereiche ihres Überbereichs befindet. Die Unterbereiche werden danach immer ausgehend vom Hauptgeschoß ihres Überbereichs verteilt, was in aufsteigender Reihenfolge ihrer lokalen WCC-Werte geschieht. Das bedeutet, dass immer die Unterbereiche mit den kleinsten WCC-Werten zuerst auf andere Geschoße verschoben werden. Dadurch besteht im Umkehrschluss die Gewährleistung, dass sich die Unterbereiche mit den größten lokalen WCC-Werten immer auf dem Hauptgeschoß befinden. (siehe Abb. 3.17)

3.2.4.3. Implementierter Verteilungsalgorithmus (siehe Abb. 3.18)

Der letztendlich gewählte Algorithmus zur Verteilung der Bereiche kann als Kombination der hierarchischen Anordnung und der Idee des Hauptgeschoßes betrachtet werden, wobei die Möglichkeit manueller Eingriffe besteht, falls sie erforderlich sind. Gewählt wurde dieser Algorithmus aus mehreren Gründen. Erstens schöpft er das volle Potenzial des hierarchischen Aufbaus des Schulbaus in Über- und Unterbereiche aus, wodurch alle Unterbereiche eines Überbereichs in unmittelbarer räumlicher Nähe zueinander liegen. Zweitens gewährleistet die Umsetzung der Idee des Hauptgeschoßes, dass die am meisten genutzten Unterbereiche jedes Überbereichs innerhalb dessen so zentral und leicht erreichbar sind, wie möglich. Drittens bietet der Algorithmus genug Flexibilität, um dem/der NutzerIn im Fall der Fälle manuelle Eingriffe zu ermöglichen.

Analog zur Hierarchischen Verteilung werden in einem ersten Schritt alle Überbereiche nach ihren WCC-Werten sortiert. Anschließend wird der Überbereich mit dem größten Wert mitsamt seiner Unterbereiche, falls welche vorhanden sind, auf dem untersten Geschoß, auf dem noch Fläche

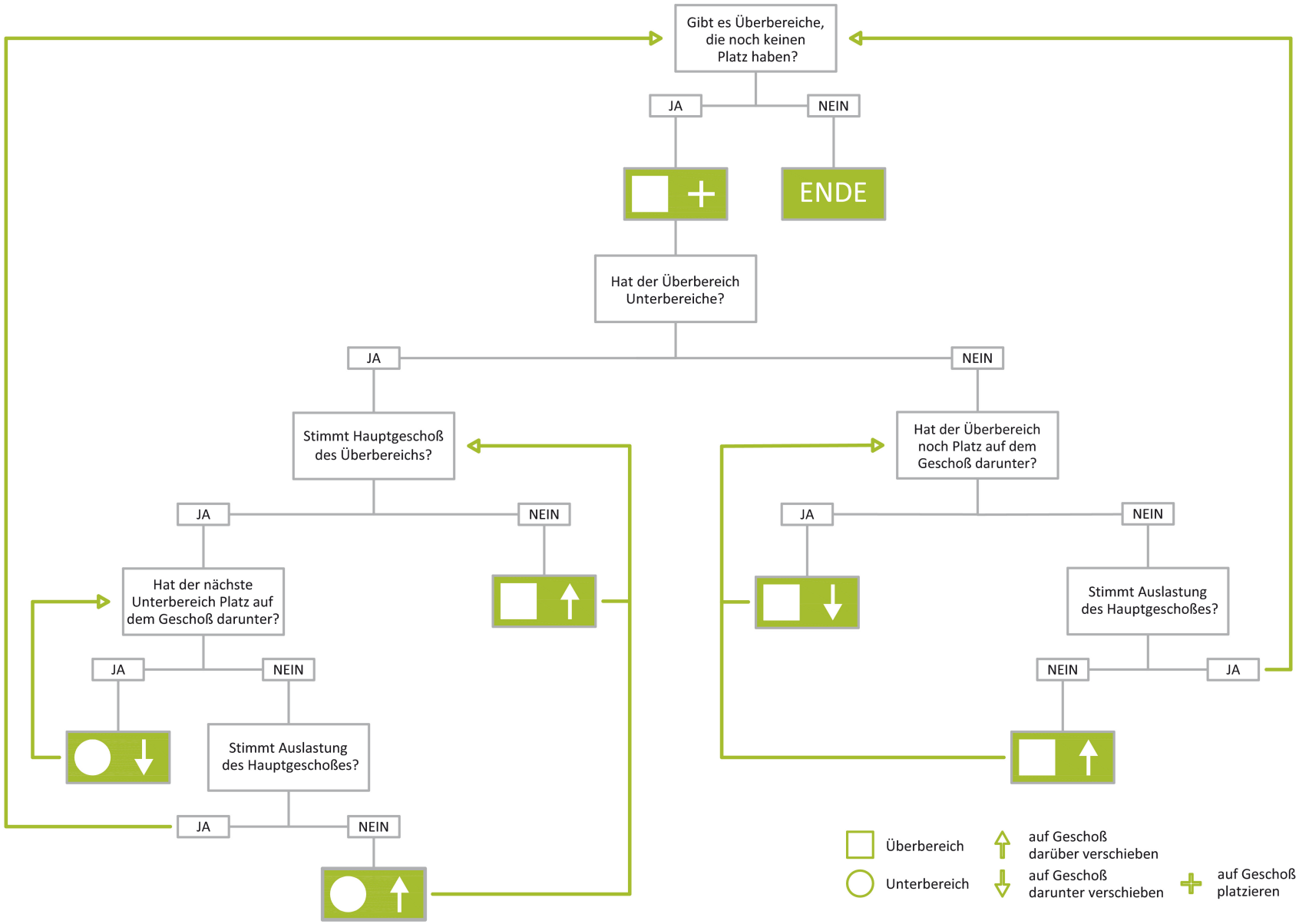


Abb. 3.18 Aktionsdiagramm des Verteilalgorithmus

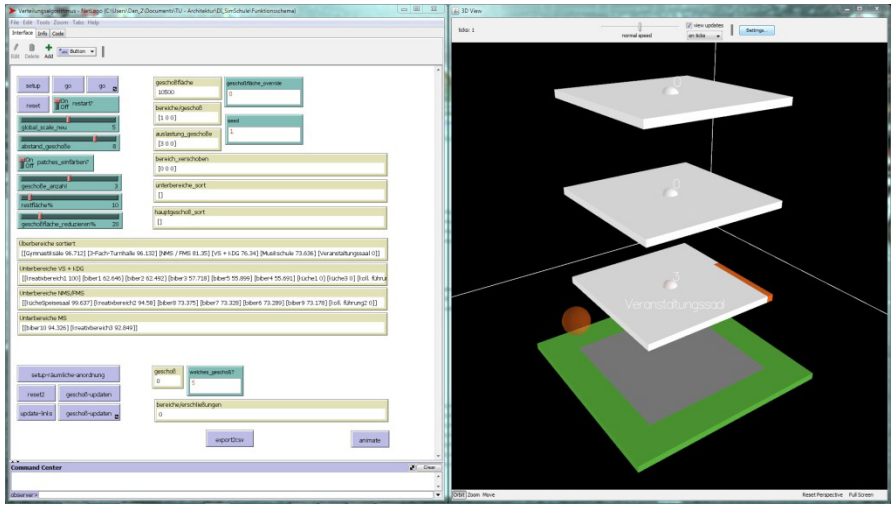


Abb. 3.19 Anordnung nach der ersten Zeiteinheit

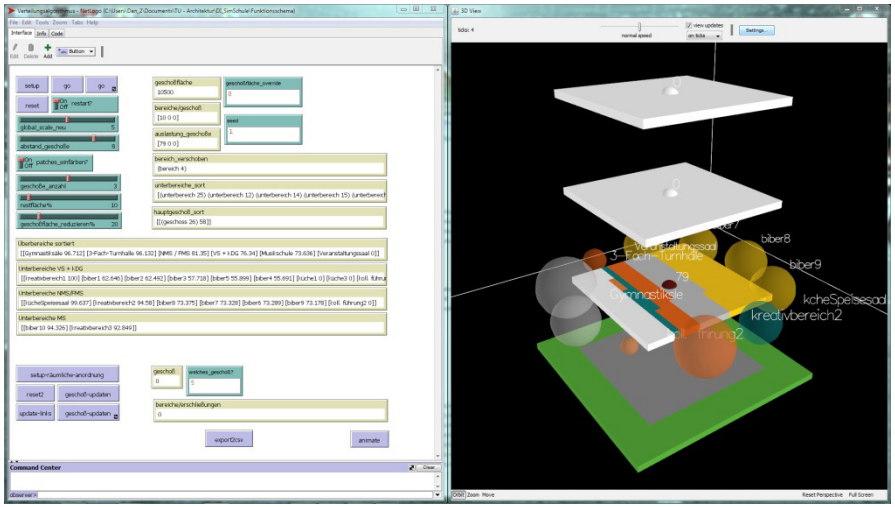


Abb. 3.20 Anordnung nach den ersten platzierten Unterbereichen

frei ist, platziert. (siehe Abb. 3.19) In der ersten Zeiteinheit handelt es sich dabei zwangsläufig um das absolut unterste Geschoß, da zuvor noch keine Bereiche platziert wurden. Dieses gilt nun als dessen aktuelles Hauptgeschoß, da sich der Großteil der Summe der lokalen WCC-Werte seiner Unterbereiche auf diesem Geschoß befindet (zu diesem Zeitpunkt sogar die Gesamtheit).

Nun wird überprüft, ob die maximale Auslastung des Geschoßes nicht überschritten wird, indem die Summe der Flächen aller Bereiche auf dem Geschoß mit der Geschoßfläche abgeglichen wird. Übersteigt die Gesamtfläche der Bereiche die maximal zulässige Geschoßfläche, so müssen Unterbereiche auf andere Geschoße verschoben werden, bis die maximal zulässige Auslastung des Geschoßes unterschritten wird. (siehe Abb. 3.20)

Dazu werden zunächst alle Unterbereiche des platzierten Überbereichs auf dem Geschoß nach ihren lokalen WCC-Werten sortiert. In der nächsten Zeiteinheit wird der Unterbereich mit dem kleinsten Wert auf ein anderes Geschoß verschoben. Hier wird zuerst überprüft, ob am Geschoß unter dem Hauptgeschoß noch entsprechend freie Fläche für den zu verschiebenden Bereich zur Verfügung steht. Ist dies nicht der Fall oder ist das aktuelle Hauptgeschoß bereits das unterste (was hier zutrifft), so wird der Unterbereich stattdessen auf das nächsthöhere Geschoß verschoben.

Anschließend wird überprüft, ob nach dieser Verschiebung eine Änderung des Hauptgeschoßes des Überbereichs eintritt. Befindet sich nun der Großteil der Summe der lokalen WCC-Werte der Unterbereiche des zuletzt verschobenen Überbereichs auf einem anderen Geschoß als dem bisherigen Hauptgeschoß, würde der gesamte Überbereich mitsamt seiner Unterbereiche auf das neue Hauptgeschoß verlegt werden und die Verschiebung der Unterbereiche von diesem Geschoß aus neu beginnen.

Tritt keine Änderung des Hauptgeschoßes ein, wird in der nächsten Zeiteinheit abermals überprüft, ob die maximale Auslastung des Hauptgeschoßes des Überbereichs unterschritten ist. Ist dies nicht der Fall, wird der Unterbereich mit dem nächst kleineren lokalen WCC-Wert auf ein anderes Geschoß verschoben. Danach kommt es wiederum zur Überprüfung der Korrektheit des Hauptgeschoßes. Ist die maximale Auslastung des Hauptgeschoßes zu diesem Zeitpunkt unterschritten, gilt der gesamte Überbereich mitsamt seiner Unterbereiche als korrekt platziert.

Anschließend wird der Überbereich mit dem nächst höheren WCC-Wert mitsamt seiner Unterbereiche wiederum auf dem niedrigsten Geschoß platziert, auf dem Fläche frei ist. Danach werden gegebenenfalls wieder solange dessen Unterbereiche verschoben, bis sich das Hauptgeschoß ändert und die Verschiebung der Unterbereiche von neuem beginnt, oder

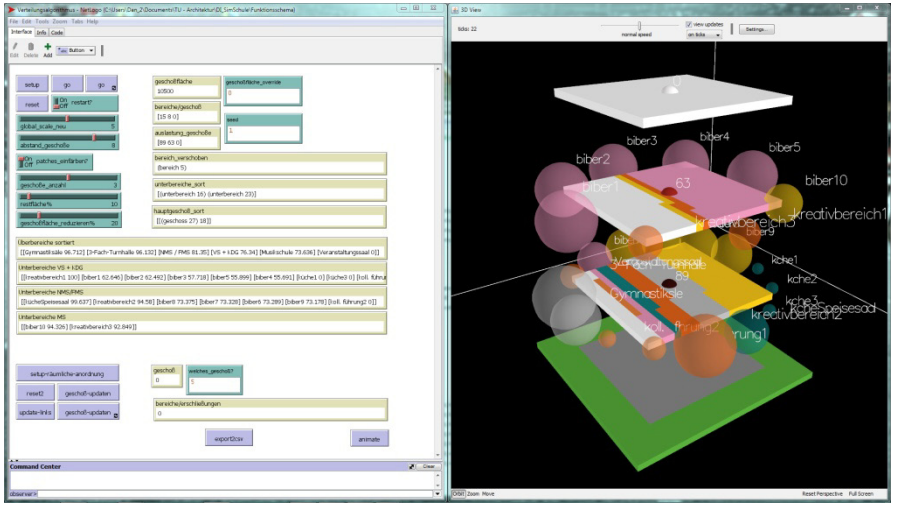


Abb. 3.21 Anordnung nach der ersten Simulationsrunde

die maximale Auslastung des Hauptgeschoßes unterschritten wird. Durch diese Art der Umsetzung der Idee des Hauptgeschoßes sind Unterbereiche nie mehr als ein Geschoß vom Hauptgeschoß ihres Überbereichs entfernt.

Dieser Teil der Simulation zur Anordnung der Bereiche gilt als abgeschlossen, wenn alle Bereiche der simulierten Schule einen Platz auf einem Geschoß gefunden haben und die maximale Auslastung keines der Geschoße überschritten wird. (siehe Abb. 3.21)

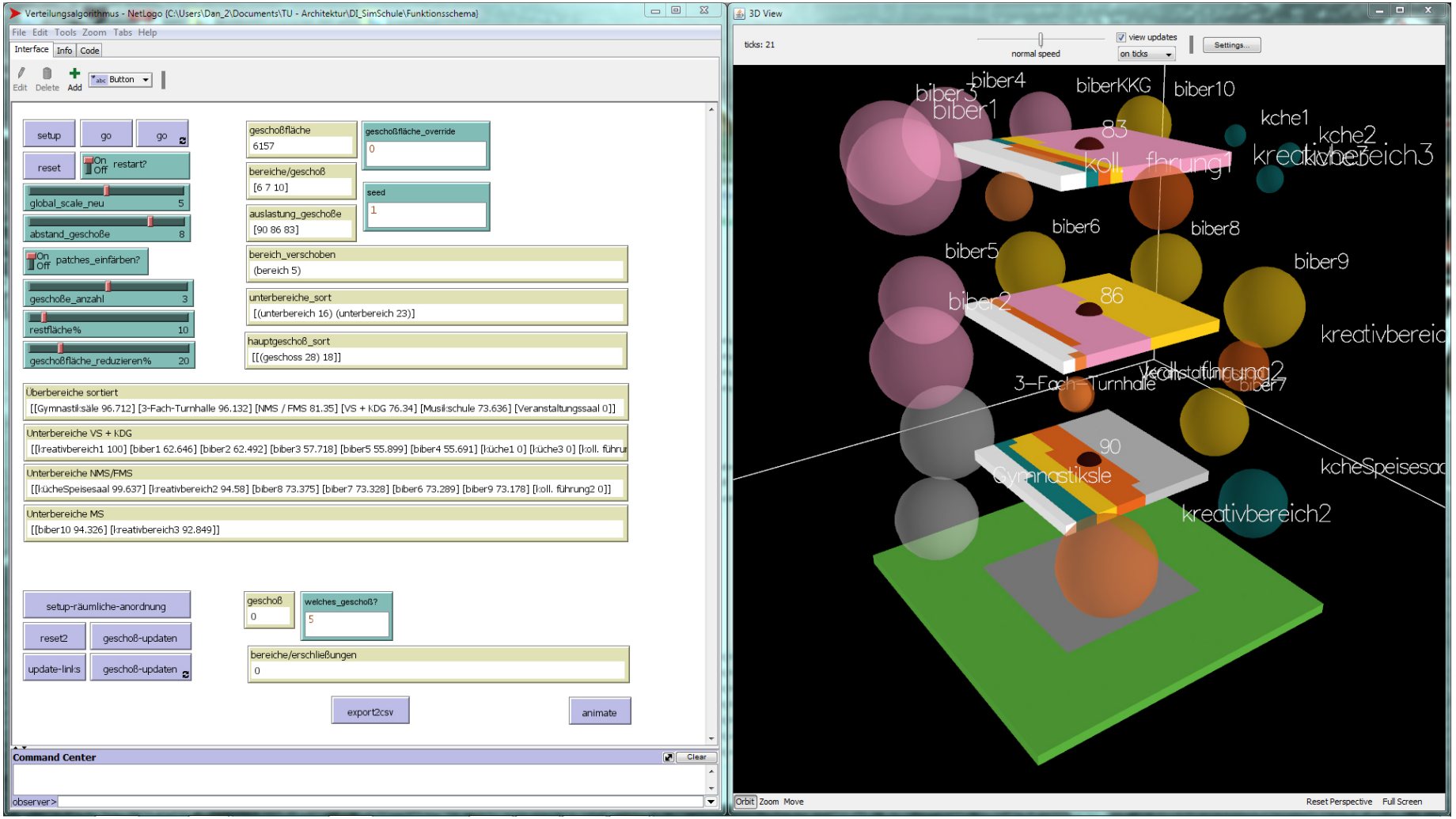


Abb. 3.22 Ergebnis des Verteilalgorithmus für 3 Geschoße

3.2.4.4. Graphischer Aufbau der Simulation

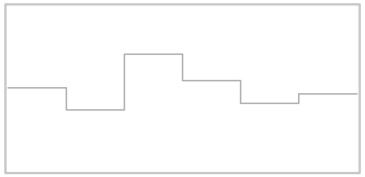
Die bereits platzierten Funktionsbereiche werden ihren jeweiligen Geschoßen graphisch zugeordnet, indem sie in der dreidimensionalen Darstellung auf derselben Höhe wie diese angeordnet werden. Die Auslastung der einzelnen Geschoßebenen im Vergleich zu ihren verfügbaren Flächen ist unmittelbar ablesbar. Alle platzierten Bereiche belegen farblich einen, entsprechend ihrer Raumgröße, verhältnismäßig großen Anteil der Fläche ihres Geschoßes.

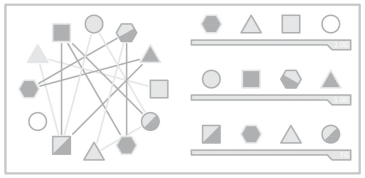
3.2.4.5. Anpassung der Geschoßfläche und Rerun

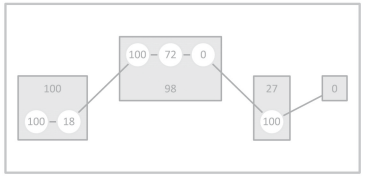
Sollte die von dem/der NutzerIn eingegebene maximale Geschoßfläche in Kombination mit der Geschoßzahl nicht ausreichen, um alle Bereiche zu platzieren, erhält er eine Fehlermeldung, in der er dazu aufgefordert wird, entweder die maximale Geschoßfläche oder die Geschoßzahl (oder beides) zu erhöhen. Für den Fall, dass die Geschoße nicht optimal ausgelastet sind, weil die von dem/der NutzerIn angegebene maximale Geschoßfläche zu groß gewählt wurde, wird diese im nächsten Schritt der Simulation entsprechend angepasst. Wenn das oberste Geschoß der Schule noch komplett unausgelastet ist, kommt der, vor der Simulation, von dem/der NutzerIn prozentual eingegebene Wert der "Verkleinerung der Geschoßfläche" zur Anpassung der Geschoßflächengröße zum Einsatz. Ist das oberste Geschoß hingegen bereits zum Teil ausgelastet, wird automatisch vom Hilfsprogramm, in Abhängigkeit der Auslastung des Geschoßes, ein Wert für die Verkleinerung errechnet, der verhindert, dass die Geschoßfläche zu stark reduziert wird, was beim nächsten Durchlauf eine Fehlermeldung hervorrufen würde. Anschließend wird die Simulation zur Verteilung der Bereiche automatisch mit dem aktualisierten Wert der Geschoßfläche neu gestartet. Die gesamte Simulation zur Verteilung der Bereiche gilt als abgeschlossen, wenn die Auslastung des obersten Geschoßes maximal 10% unter der maximal zulässigen Auslastung liegt.

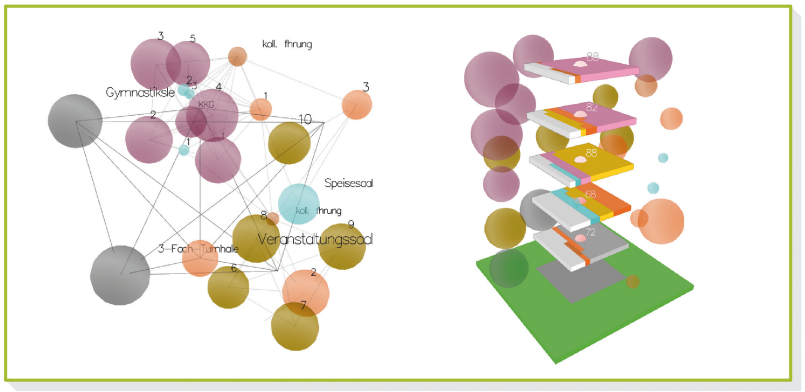
3.3. Ergebnis

Das Ergebnis des Hilfsprogramms bietet einen Vorschlag einer geschoßweisen Anordnung der verschiedenen Funktionsbereiche des simulierten Schulgebäudes. (siehe Abb. 3.22) Es verbindet dabei die, durch die Bewegungsabläufe der Schulklassen erstellten, Nachbarschaftsverhältnisse der Bereiche mit einer räumlichen Verteilung, trifft allerdings keine Aussagen über die exakte Positionierung der Bereiche auf ihren zugewiesenen Geschoßen. Das letztendlich abgebildete dreidimensionale Diagramm bietet dem/r PlanerIn eine erste Orientierung in der frühen Entwurfsphase, ohne dabei die finale Form zu definieren und ihn/sie somit einzuschränken. Die simulierte Anordnung der Bereiche kann wiederum in Form von Excel-Tabellen ausgegeben werden, um eine Weiterverarbeitung der Daten zu ermöglichen. In den Dateien enthalten sind alle relevanten Informationen der Anordnung der Bereiche und ihrer Nachbarschaftverhältnisse im Gebäude.









4. Case Study

4.1. Regionale Prognose: Klassenzuwachs in Wien **67**

4.1.1. Parametrisierung 67

4.1.2. Bedarfsdeckung zusätzlicher Klassen für Wien 69

4.1.3. Ergebnis 73

4.2. Einsatz Schulgenerator: Wettbewerb **73**
" Bildungscampus Nordbahnhof"

4.2.1. Wettbewerb Bildungscampus Nordbahnhof 73

4.2.2. Parametrisierung 75

4.2.2.1. Aufbau der Schule 75

4.2.2.2. AkteurInnen der Schule 75

4.2.3. Simulation der Abläufe in einer Schulwoche 76

4.2.4. Verteilung der Bereiche auf Geschoße 79

4.2.5. Ergebnis 83

4. Case Study

4.1. **Regionale Prognose:** Klassenzuwachs in Wien

Als Case Study für die Simulation der Regionalen Prognose wird der Zustand der Schulwesens der Stadt Wien herangezogen. Durch Anwendung des Hilfsprogramms wird für zehn Schultypen und alle 23 Gemeindebezirke der Stadt der Bedarf zusätzlich notwendiger Schulklassen bis zum Jahr 2030 ermittelt.

4.1.1. Parametrisierung

Als Input für das Programm dienen prognostizierte Daten der SchülerInnen- und Klassenzahlen des Ausgangsjahres[67] der Simulation sowie der Folgejahre[68], die von der Statistik Austria in Form von PDFs und Excel-Dateien zu Verfügung gestellt werden. Die aktuellsten erhältlichen Information bezüglich der Entwicklung der SchülerInnenzahlen für Wien stammen aus dem Jahr 2012 und haben das Schuljahr 2009/10 als Ausgangsjahr. In drei Tabellen wird davon ausgehend die prognostizierte Entwicklung für die Schuljahre 2020/21 und 2030/31 dargestellt. (siehe Abb. 4.1) Abgeglichen werden diese Informationen mit den Klassenzahlen des Ausgangsjahres, also dem Schuljahr 2009/10. (siehe Abb. 4.4)

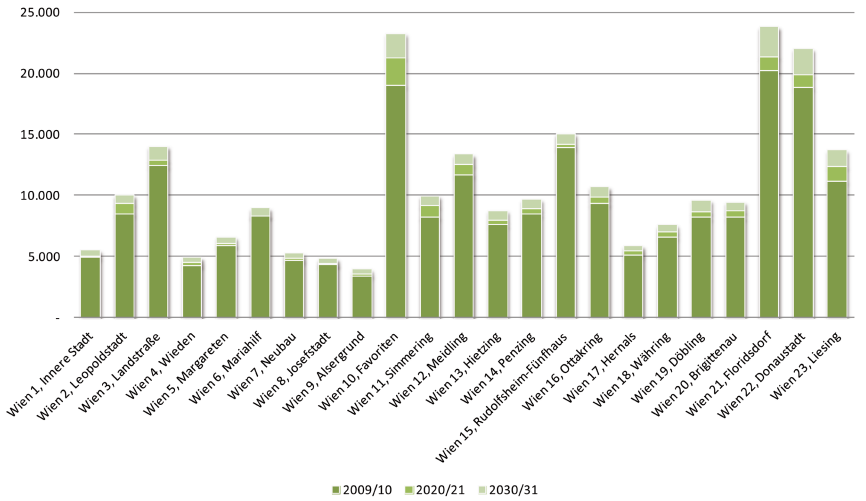


Abb. 4.1 Entwicklungen der SchülerInnenzahlen bis 2030/31 nach Bezirken

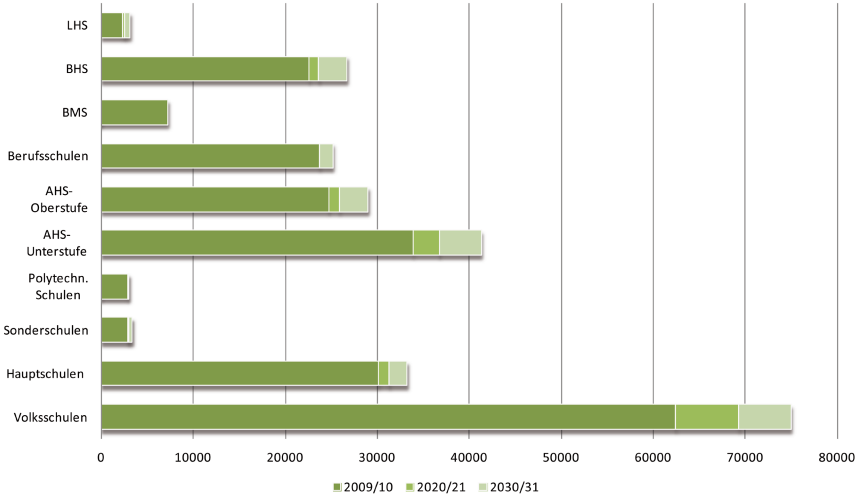


Abb. 4.2 Entwicklungen der SchülerInnenzahlen bis 2030/31 nach Schultypen

[67] Schulstatistik; Klassen ab 2006, Statistik Austria

[68] Prognose der Schülerinnen- und Schülerzahl für die Schuljahre 2020/21 und 2030/31 nach Politischen Bezirken, Statistik Austria

Alle Daten sind nach 10 Schultypen (Volksschulen, Hauptschulen, Sonderschulen, Polytechnische Schulen, AHS-Unterstufe, AHS-Oberstufe, Berufsschulen, Berufsbildende mittlere Schulen, Berufsbildende höhere Schulen und Lehrerbildende höhere Schulen) und den 23 politischen Bezirken Wiens aufgeschlüsselt verfügbar. (siehe Abb. 4.2)

Vor dem Simulationsdurchgang müssen aufgrund der eingeschränkten Verfügbarkeit der Daten Angleichungen an diesen vorgenommen werden. Zunächst werden die Tabellen der SchülerInnenzahlen an jene der Klassenzahlen angepasst, da diese wesentlich präzisere Unterscheidungen bezüglich der Schultypen anbieten. So sind die Klassenzahlen nach 27 unterschiedlichen Schulformen sortiert, die SchülerInnenzahlen hingegen nur nach zehn. In den Tabellen der Klassenzahlen sind beispielsweise vier verschiedene Schultypen angegeben, die in jener der SchülerInnenzahlen unter der Bezeichnung "Berufsbildende höhere Schulen" zusammengefasst sind. Dieselben Differenzen bezüglich der Aufteilungen bestehen in ähnlicher Weise bei den meisten Schulformen.

Eine weitere Anpassung ist erforderlich, da jene Schulklassen, die am Schulversuch der Neuen Mittelschule teilnehmen, in den SchülerInnenzahlen verhältnismäßig auf "Hauptschule" und "AHS-Unterstufe" aufgeteilt sind, in den Klassenzahlen aber als eigener Schultyp existieren. Dieses Klassen wurden in ihren jeweiligen Bezirken je zur Hälfte auf "Hauptschulen" und "AHS-Unterstufe" aufgeteilt.

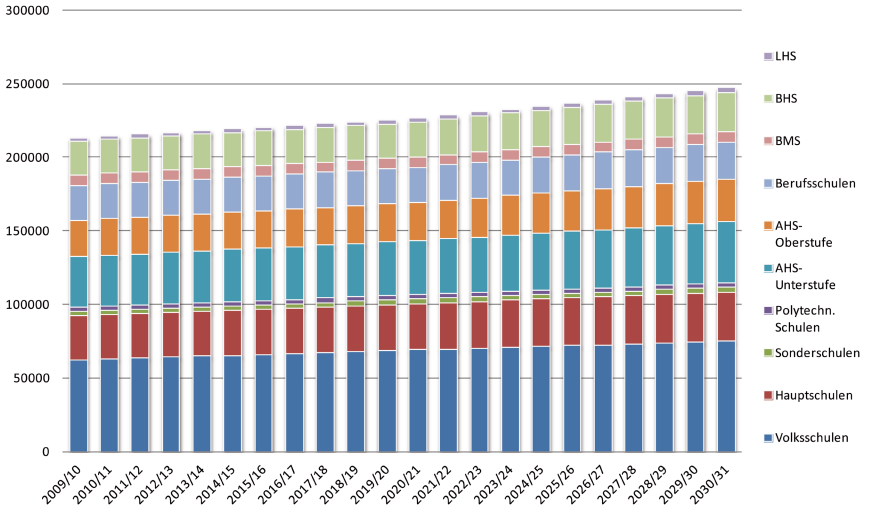


Abb. 4.3 Berechnete Entwicklungen der SchülerInnenzahlen bis 2030/31

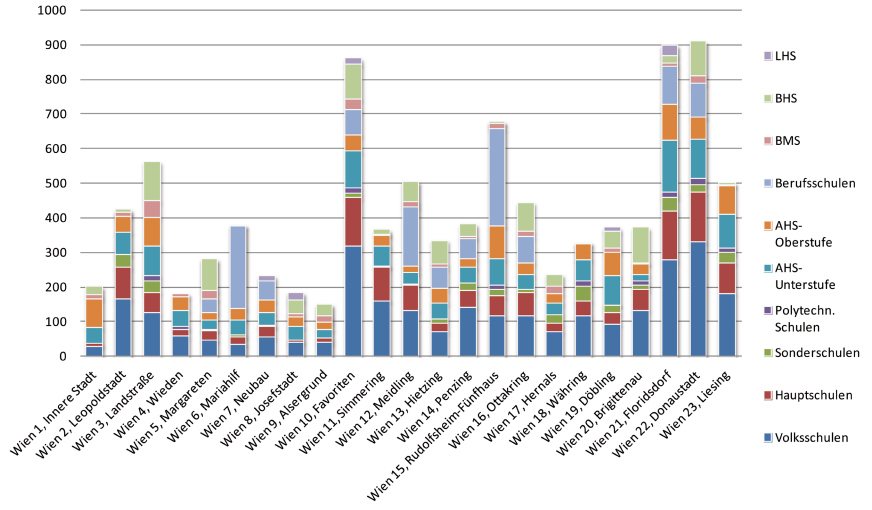


Abb. 4.4 Klassenzahlen im Ausgangsjahr der Simulation (2009/10)

Nachdem die Tabellen in Form von Excel-Dateien in die Applikation importiert wurden, berechnet diese in einem ersten Schritt automatisiert die Informationen der fehlenden Jahre der SchülerInnenzahlen aus den vorliegenden. Dazu werden aus den Listen der Jahre 2009/10 und 2020/21 die Daten der dazwischenliegenden Schuljahre ebenso kalkuliert wie zwischen 2020/21 und 2030/31. Das Programm subtrahiert die Zahlen des jeweils späteren Schuljahres von denen des früheren und dividiert sie durch die Anzahl dazwischenliegender Jahre. Dieser Vorgang wird für alle 23 Bezirke und jeweils 10 Schultypen durchgeführt. (siehe Abb. 4.3) Im Endeffekt entsteht dadurch eine Datenbank von über 5000 Werten.[69]

Als zusätzlicher Input kommt eine schwarz-weiß Karte der Stadt Wien zur Anwendung, in der auch die politischen Bezirke abgebildet sind. Diese dient in der Simulation als graphischer Hintergrund und zur besseren Orientierung des Nutzers. (siehe Abb. 4.5)

4.1.2. Bedarfsdeckung zusätzlicher Klassen für Wien

Im Zuge der Bedarfsdeckung zusätzlicher Klassen für Wien werden ausgehend vom Schuljahr 2009/10 für alle Jahre bis 2030/31, alle 23 Bezirke und 10 Schultypen die Anzahl notwendiger Schulklassen berechnet, sodass der Grenzwert von maximal 25 SchülerInnen pro Klasse nie überschritten wird. (siehe Abb. 4.6)

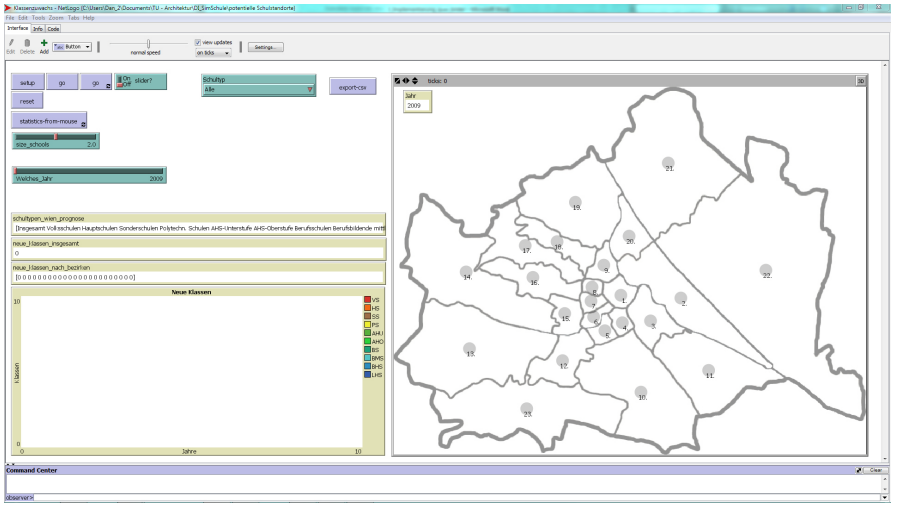


Abb. 4.5 Setup der Simulation zur Bedarfsdeckung für Wien

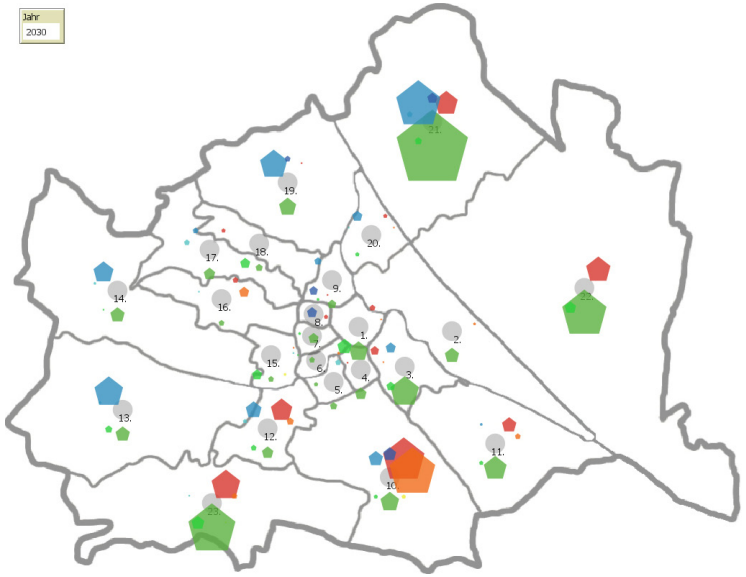


Abb. 4.6 Resultat der Simulation zur Bedarfsdeckung zusätzlicher Klassen

[69] Anm.: 22 Jahre mit je 23 Bezirken und 10 Schultypen -> 5060 Werte

Im Ausgangsjahr der Simulation werden insgesamt bereits 106 neue Klassen benötigt. Davon zählen 36 zu "Berufsbildenden höheren Schulen", 33 zu "AHS-Unterstufe", 19 zu "Berufsbildende mittleren Schulen", neun zu "Hauptschulen", fünf zu "Polytechnische Schulen" und vier zu "Lehrerbildende höheren Schulen". (siehe Abb. 4.7) Für die üblichen Schultypen wird der Grenzwert in diesem Jahr in keinem der Bezirke überschritten. An erster Stelle der Bezirke, in denen im Schuljahr 2009/10 der größte Klassenzuwachs zustande kommt, liegt der 21. Wiener Gemeindebezirk, Floridsdorf, mit einem Bedarf von insgesamt 36 zusätzlichen Klassen, von denen 22 dem Schultyp "Berufsbildende höhere Schulen" angehören, 10 der "AHS-Unterstufe" und die restlichen der "Berufsbildenden mittleren Schulen". Die weiteren Bezirke, in denen in dieser Zeiteinheit für zumindest einen Schultyp der Grenzwert überschritten wird, sind, in absteigender Reihenfolge der Anzahl hinzugekommener Klassen, Favoriten (11 Klassen), Meidling (10), Penzing (7), Donaustadt (5), Innere Stadt, Landstraße, Hietzing, Hernals, Liesing (jeweils 4), Wieden, Döbling (jeweils 3), Mariahilf, Josefstadt, Rudolfsheim-Fünfhaus (jeweils 2), Neubau, Alsergrund, Simmering, Ottakring, Brigittenau (jeweils 1). Die einzigen Bezirke, in denen kein Bedarf an neuen Schulklassen besteht, sind Leopoldstadt, Margareten und Währing. (siehe Abb. 4.8)

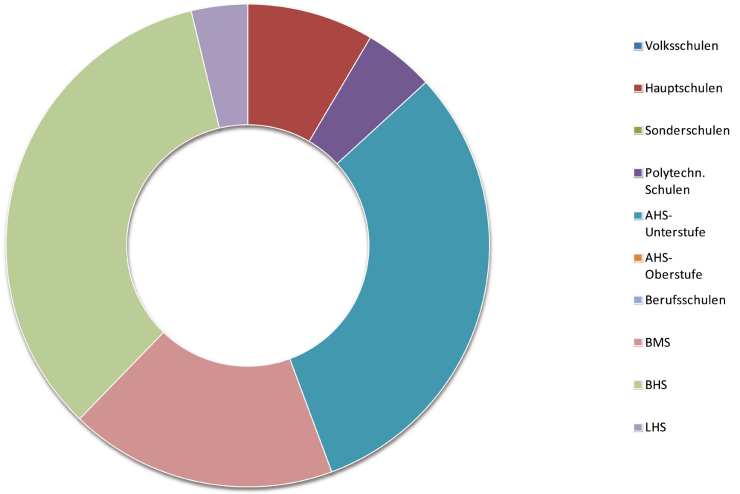


Abb. 4.7 Hinzugefügte Klassen im Schuljahr 2009/10 nach Schultypen

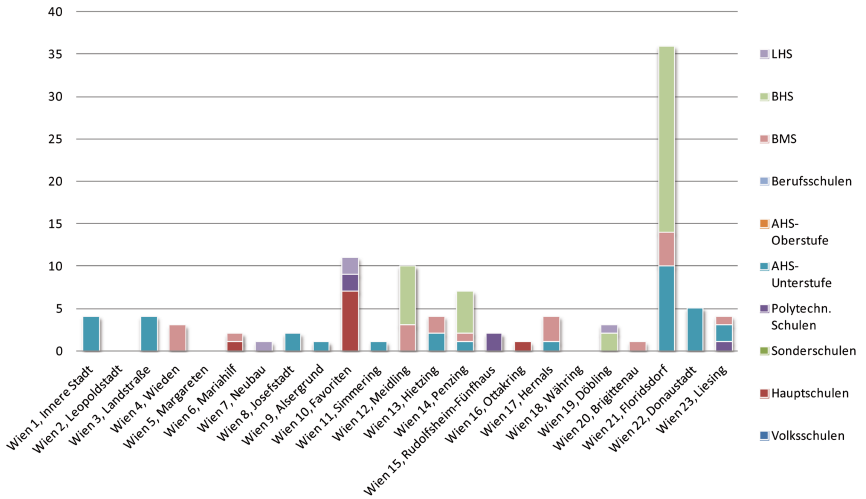


Abb. 4.8 Hinzugefügte Klassen im Schuljahr 2009/10 nach Bezirken

Der Grenzwert von 25 SchülerInnen pro Klasse wird im Schuljahr 2009/10 bereits insgesamt 33 Mal in 20 verschiedenen Gemeindebezirken Wiens überschritten. Spitzenreiter der Bezirke sind in dieser Statistik der 10., 21. und 23. mit jeweils 3 erhöhten Werten. Der Schultyp, in dem es zu den meisten Überschreitungen kommt, ist "AHS-Unterstufe", für den dies in elf Bezirken der Fall ist. Die totalen Durchschnittswerte aller Schultypen jedes Bezirks sowie die aller Bezirke jedes Schultyps liegen allerdings unter dem Grenzwert.

Über den gesamten simulierten Zeitraum bis 2030/31 werden im 21. Wiener Gemeindebezirk, Floridsdorf, mit insgesamt 113 zusätzlichen Klassen am meisten hinzugefügt, wie dies bereits in der ersten Zeiteinheit der Fall war. Ein Großteil davon fällt auf den Schultyp "AHS-Unterstufe" (50) ab, gefolgt von "Berufsbildende mittlere Schulen" (31) und "Volksschulen" (16). Die einzigen beiden Bezirke, in denen die Anzahl zusätzlicher Klassen einstellig bleibt, sind der 5. Bezirk, Margareten, und der 6. Bezirk, Mariahilf. (siehe Abb. 4.9)

Der Schultyp, der insgesamt den größten Zuwachs neuer Klassen verzeichnet, ist "AHS-Unterstufe" (280), gefolgt von "Volksschulen" (133) und "Berufsbildende höherer Schulen" (130). Zwei Schultypen verzeichnen keinerlei Zuwachs aufgrund der prognostizierten Daten bis 2030/31, nämlich "Sonderschule" und "Polytechnische Schulen". (siehe Abb. 4.10)

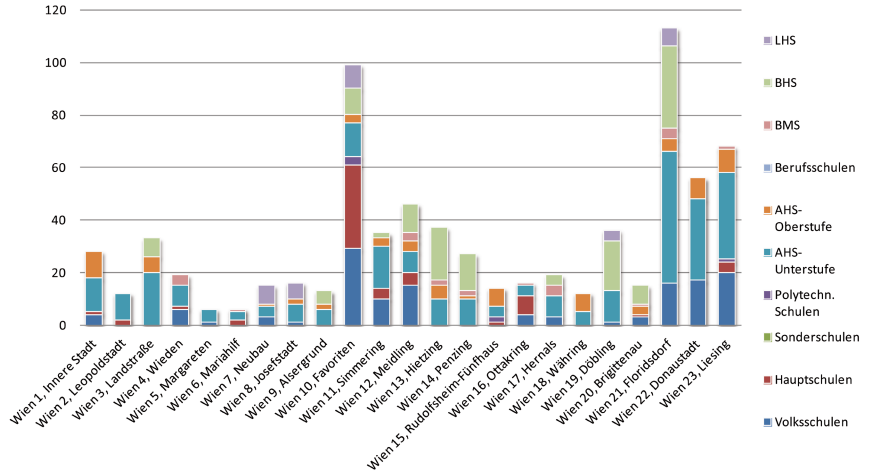


Abb. 4.9 Bis 2030/31 hinzugefügte Klassen nach Bezirken und Schultypen

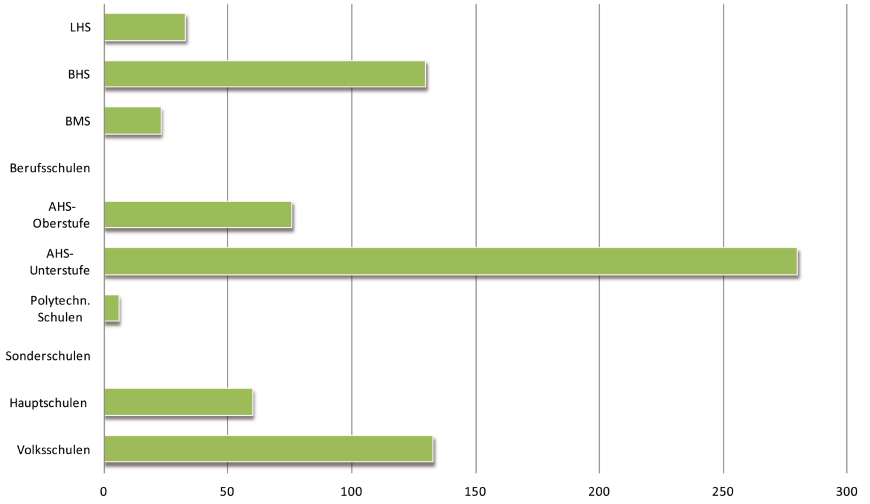


Abb. 4.10 Insgesamt hinzugefügte Klassen nach Schultypen

Zum größten jährlichen Klassenzuwachs kommt es im Ausgangsjahr der Simulation (106 Klassen), gefolgt vom letzten Jahr, 2030/31 (67). Dazwischen nimmt die Zahl hinzugefügter Klassen relativ konstant von Jahr zu Jahr geringfügig zu. Die einzigen Ausreißer nach unten lassen sich in den Jahren 2014/15 (6) und 2017/18 (10) erkennen. Daran lässt sich ablesen, dass bereits im Schuljahr 2009/10 ein großer Bedarf an zusätzlichen Schulklassen bestand. Nach dessen vorläufiger Deckung kommt es danach aber weiterhin zu einer Zunahme der SchülerInnenzahlen und damit verbundenen Steigerung der Klassenzahlen in den meisten Bezirken und Schultypen. (siehe Abb. 4.11)

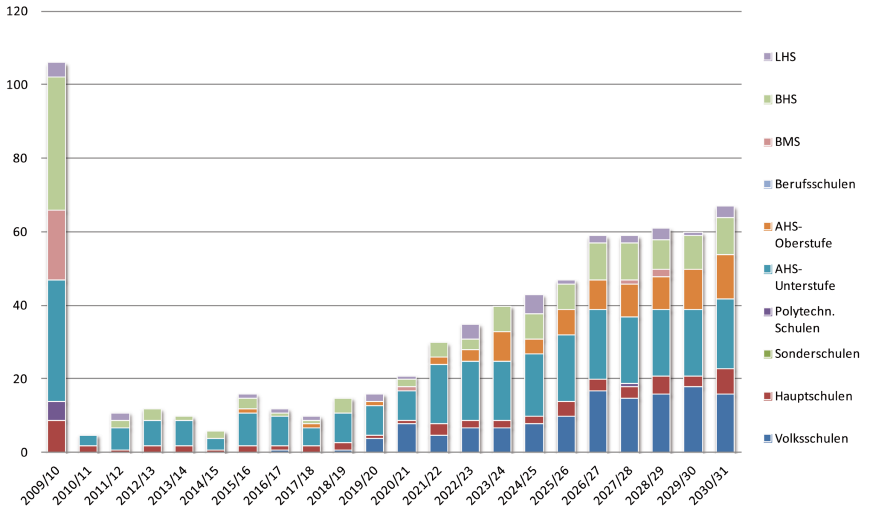


Abb. 4.11 Hinzugefügte Kassen nach Jahr und Schultypen

Für den theoretischen Fall, dass bis 2030/31 keine neuen Klassen in Wien hinzukommen, würde es, den prognostizierten Daten zufolge, in allen 23 Gemeindebezirken zu insgesamt 89 Überschreitungen des Grenzwertes der maximalen Klassenbelegung kommen. Für den Schultyp "AHS-Unterstufe" würde der Wert sogar in allen Bezirken mit Ausnahme des 20. erhöht sein. Die Durchschnittswerte aller Schultypen von 13 Bezirken und aller Bezirke von fünf Schultypen würden ebenfalls über 25 SchülerInnen pro Klasse liegen. (siehe Abb. 4.12)

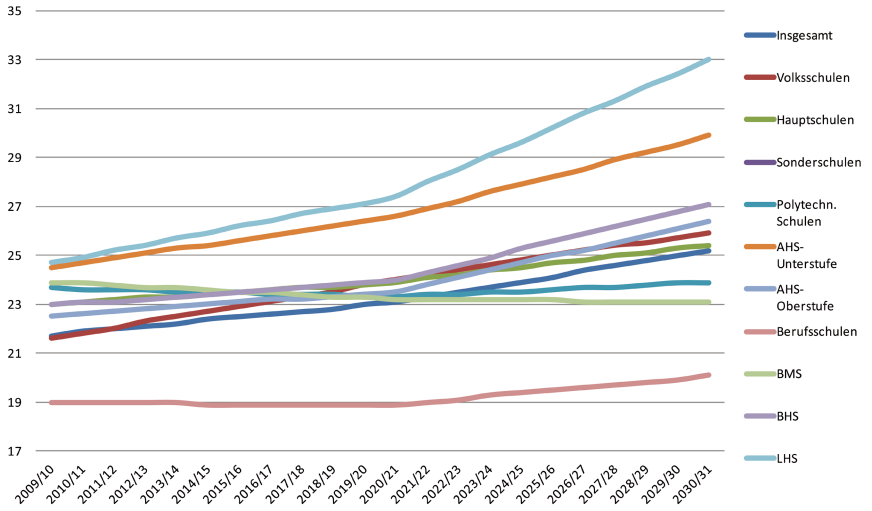


Abb. 4.12 Entwicklungen der SchülerInnen / Klasse exkl. neuer Klassen

4.1.3. Ergebnis

Alle aus der Simulation resultierenden Daten bezüglich des Bedarfs zusätzlicher Schulklassen werden anschließend automatisch in sieben Excel-Dateien aufteilt ausgegeben. Diese zeigen in Tabellen Informationen der, über den gesamten simulierten Zeitraum, hinzugefügten Schulklassen, entweder nach Jahren oder nach Bezirken gefiltert. Zusätzlich stellen sie die Entwicklung der Werte der Klassenbelegungen, sowohl inklusive als auch exklusive der neuen Klassen, ebenfalls nach Bezirken oder Jahren sortiert dar. (siehe Abb. 4.13)

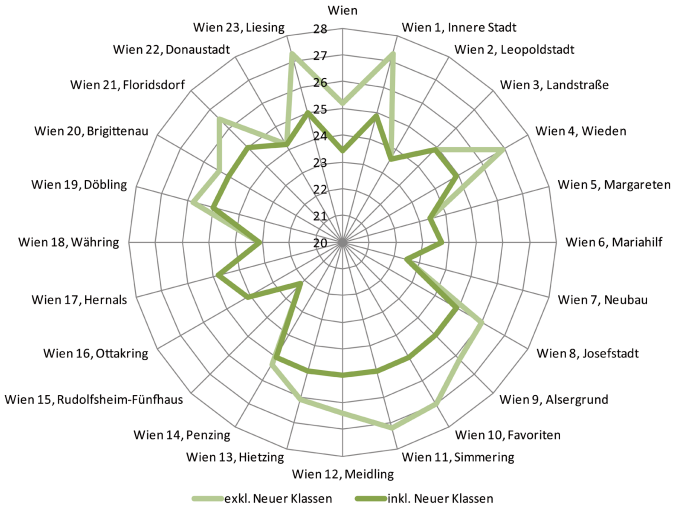


Abb. 4.13 Entwicklungen der SchülerInnen / Klasse inkl. neuer Klassen

4.2. **Einsatz Schulgenerator:** Wettbewerb "Bildungscampus Nordbahnhof"

Das Hilfsprogramm zur Unterstützung in der Planung von Schulbauten wird auf den von der Stadt Wien ausgeschriebenen Wettbewerb für den Neubau eines Bildungscampus am Gelände des Nordbahnhofs angewendet. Für diesen wird durch Simulationen in einem ersten Schritt ein Nachbarschaftsdiagramm erzeugt, welches die räumlichen Beziehungen seiner Funktionsbereiche zueinander darstellt. Anschließend wird, darauf aufbauend, ein Vorschlag einer räumlichen Anordnung der Bereiche auf Geschoßebenen generiert.

4.2.1. Wettbewerb Bildungscampus Nordbahnhof

Beim gewählten Wettbewerb handelt es sich um einen EU-weiten, offenen, zweistufigen Realisierungswettbewerb, der im November 2015 von der Stadt Wien ausgeschrieben wurde. Planungsaufgabe ist der Neubau einer integrativen Bildungseinrichtung für Null- bis Fünfzehnjährige und einer Musikschule im 2. Wiener Gemeindebezirk.

Der Bildungscampus besteht aus einer Volksschule inklusive Kindergarten, einer Neuen Mittelschule kombiniert mit einer Fachmittelschule, einer Musikschule und sonstigen allgemeinen Funktionsbereiche, die eine 3-Fach-Turnhalle, Gymnastiksäle und einen Veranstaltungssaal umfassen und gemeinschaftlich von allen Schulklassen genutzt werden. Der

Schulbau bietet Platz für 44 Klassen, fünf Kleinkinder- und elf Kindergartengruppen. Er ist für insgesamt 1875 Kinder unterschiedlichen Alters, 234 Pädagogen und 50 Personen als Betriebspersonal ausgelegt.

Das Wettbewerbsgebiet hat eine Größe von insgesamt 22990 m². Die zulässige Gebäudehöhe von fünf oberirdischen Geschoßen sollte gänzlich ausgenutzt und eine maximal zulässige bebaute Fläche von 10500 m² nicht überschritten werden. In den Ausschreibungsunterlagen ist ein exaktes Funktionsprogramm enthalten, welches sowohl die erforderlichen Raumgrößen aller Bereiche als auch deren Raumhöhen vorgibt. Zusätzlich gibt es eine Liste der Funktionsbereiche, die im Erdgeschoß situiert sein müssen.[70]

Ein weiterer Bestanteil der Ausschreibungsunterlagen sind Raumbücher und Raumblätter zum Campus+-Modell, dem das zu planende Schulgebäude angehört, in denen einige wesentliche Grundsätze dieser Bauaufgaben beschrieben werden. Das Modell des Wiener Campus+ beschreibt Bildungseinrichtungen, die Kindergarten, Schul- und Freizeitpädagogik an einem Standort vereinen. Das Modell setzt dabei verstärkt auf die Vernetzung der einzelnen Altersgruppen und Bildungseinrichtungen untereinander, wodurch sich Nutzungssynergien ergeben. Ein weiterer wesentlicher Aspekt ist auch die Prämisse der Mehrfachnutzung. Möglichst viele Räume sollten einer großen Anzahl unterschiedlicher Nutzergruppen offenstehen; sowohl im regulären Schulbetrieb als auch außerhalb dessen. Eine essentielle Forderung im Raumbuch ist darüber hinaus eine Anordnung des Baukörpers, die sowohl die funktionalen Zusammenhänge zwischen den Bildungsbereichen und den gemeinschaftlichen Flächen als auch die Wegeführung innerhalb des Gebäudes optimiert.[71]

Der Wettbewerb wurde aus mehreren Gründen als Case Study für diese Diplomarbeit gewählt. Zunächst war die Ausschreibung gerade im Gange, als sie verfasst wurde. Außerdem bietet ein Entwurf dieses Umfangs, bestehend aus verschiedenen Schultypen und Funktionseinheiten an einem Standort, in Kombination mit den sehr detaillierten Ausschreibungsunterlagen, die optimalen Voraussetzungen zur Erprobung des Simulationsprogramms. Eine weitere Entscheidungsgrundlage war die Tatsache, dass bei diesem Wettbewerb viele neue Ansätze der Schulplanung, sowohl pädagogische, wie die soziale Inklusion als auch räumliche, wie die Clusterbildung oder Mehrfachnutzungen, Anwendung finden.

[70] Auslobung Neubau Bildungscampus Nordbahnhof, Magistratsabteilung 19 der Stadt Wien, 2015

[71] Magistratsabteilung 34: Bau- und Gebäudemanagement (2014): Raumbuch Campus+: Bildungseinrichtung mit Mehrwert der Stadt Wien, Wien

4.2.2. Parametrisierung

4.2.2.1. Aufbau der Schule

Als Ausgangspunkt für den Aufbau des Bildungscampus in der Simulation dient ein Schema aus den Ausschreibungsunterlagen, welches eine hierarchische Gliederung aller Funktionsbereiche graphisch abbildet. (siehe Abb. 4.14) Daraus lassen sich sechs höherwertige Überbereiche ablesen, zu denen die drei Schultypen (Volksschule und Kindergarten (VS + KDG), Neue Mittelschule und Fachmittelschule (NMS / FMS) und Musikschule (MS)) sowie drei allgemeine Funktionsbereiche (Gymnastiksäle, 3-Fach-Turnhalle und Veranstaltungssaal) zählen. Sowohl für den Bereich der Volksschule und des Kindergartens als auch für die Neue Mittelschule kombiniert mit der Fachmittelschule werden zugehörige Unterbereiche dargestellt. Zu diesen gehören allgemeine Bildungsbereiche (Biber), Kreativbereiche, in denen alle Sonderfächer, zu deren Durchführung spezielle räumliche Grundausstattungen vonnöten sind, abgehalten werden, und Verwaltungsbereiche (Koll. Führung). Die Darstellung zeigt darüber hinaus ebenfalls dienende Funktionen, etwa Küchen, sowie deren räumliche Beziehungen zu den Bildungsbereichen. Diese Informationen bezüglich des hierarchischen Aufbaus mitsamt der entsprechenden Raumgrößen werden als Input für das Hilfsprogramm herangezogen. Der Übersichtlichkeit wegen wird auch die farbliche

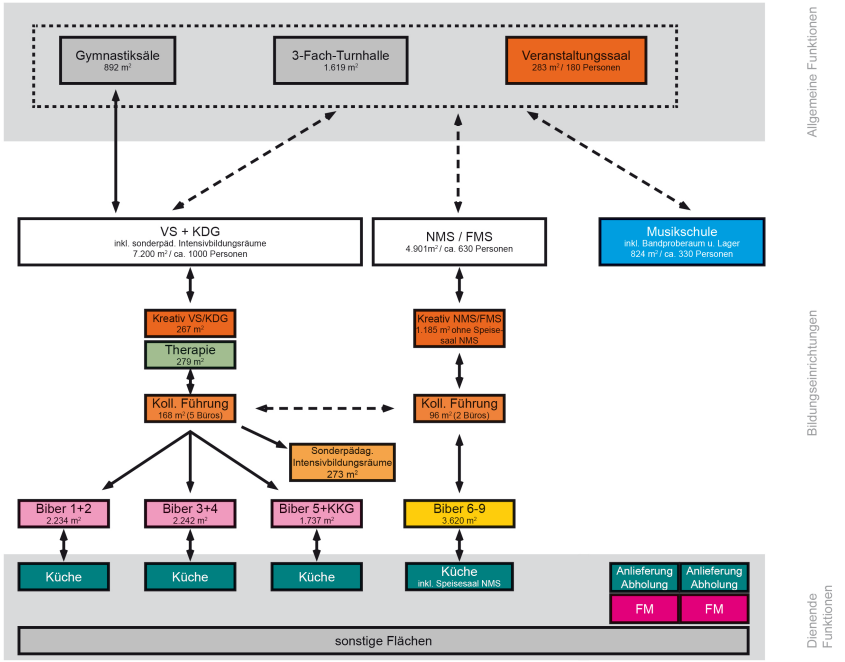


Abb. 4.14 Funktionsdiagramm[72]

Darstellung der unterschiedlichen Funktionsbereiche des Diagramms in der Simulation beibehalten.

4.2.2.2. AkteurInnen der Schule

Als AkteurInnen der Schule und zugleich Agenten in der Simulation kommen ebenfalls in den Ausschreibungsunterlagen definierte

[72] Auslobung Neubau Bildungscampus Nordbahnhof, Magistratsabteilung 19 der Stadt Wien, 2015, S. 20

Schulklassen bzw. Kindergruppen zum Einsatz. Dazu zählen 5 Kleinkindergruppen (KKG), 11 Kindergartengruppen (KDG), 22 Volksschulklassen, 20 Klassen der Neuen Mittelschule, 2 Klassen der Fachmittelschule und 12 Klassen der Musikschule.

Allen Klassen und Gruppen wird als "Homebase", in Übereinstimmung mit den Auslobungsunterlagen, einer der Bildungsbereiche zugewiesen. Zusätzlich erhalten sie in Abhängigkeit ihres Schultyps und Jahrgangs eine Stundentafel, die sie in der folgenden Simulation abarbeiten müssen. Die Stundentafeln variieren, bezüglich der zu absolvierenden Gesamtstundenanzahl, zwischen 22 (1. und 2. Jahrgang Volksschule)[73] und 31 Stunden (3. und 4. Schulstufe Neue Mittelschule)[74]. Alle Unterrichtsgegenstände, für deren Durchführung keine speziellen räumlichen Vorrausetzungen gefordert werden, absolvieren die Klassen in ihren persönlichen Bildungsbereichen, die übrigen Fächer, auf die diese Beschreibung nicht zutrifft, entweder im dazugehörigen Kreativbereich oder in einem der allgemeinen Funktionsbereiche. Zusätzlich zu den Stundentafeln wird für jeden Schultag eine Stunde Essenszeit eingeplant, da der Bildungscampus auf dem Modell der ganztägigen Schulform aufgebaut ist. Die Klassen der Volksschule und die Gruppen des Kindergarten führen diese Aktivität in ihren entsprechenden

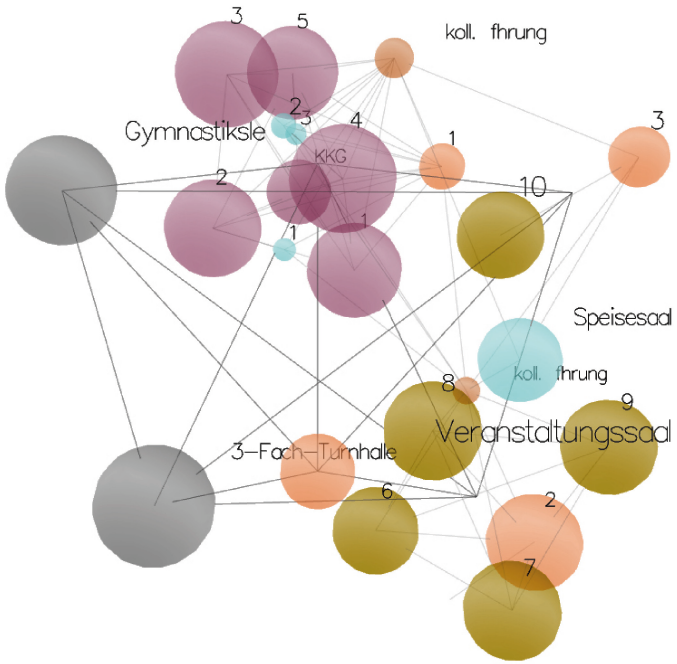


Abb. 4.15 Dreidimensionale Anordnung der Funktionsbereiche

Bildungsbereichen durch, die restlichen Klassen in einem gemeinsam genutzten Speisesaal.

4.2.3. Simulation der Abläufe in einer Schulwoche

Nachdem das Hilfsprogramm mit allen Daten bezüglich des Aufbaus des Bildungscampus sowie der Anzahl und Zusammensetzung der Schulklassen gespeist wurde, werden zunächst alle Funktionsbereiche in Abhängigkeit ihrer hierarchischen Beziehungen zueinander

[73] Lehrplan der Volksschule, Artikel I und II, Stand: BGBl. II Nr. 303/2012, September 2012
[74] Lehrpläne - Neue Mittelschule, BGBl. II Nr. 185/2012, September 2015

dreidimensional angeordnet. Die Überbereiche "VS + KDG", "NMS / FMS" und "MS" werden in weiterer Folge nur mehr durch ihre Unterbereiche abgebildet. (siehe Abb. 4.15)

Anschließend beginnt die Simulation der Abläufe in einer Schulwoche. In dieser bewegen sich alle Klassen von Zeiteinheit zu Zeiteinheit zwischen den Funktionsbereichen des Bildungscampus, bis sie schließlich ihre Stundentafeln vollständig absolviert haben, was für die erste von ihnen nach 27 und für die letzte nach 36 Zeiteinheiten der Fall ist. (siehe Abb. 4.16) Während des Simulationsvorgangs werden anhand der Bewegungen der Klassen sowohl die Durchsätze aller Verbindungen als auch die normalisierten lokalen und globalen WCC-Werte aller Funktionsbereiche berechnet. Nach einem ersten Durchlauf[75] weist die 3-Fach-Sporthalle den größten globalen WCC-Wert (100) auf, gefolgt von den "Gymnastiksälen" (98), dem Speisesaal (94), dem Kreativbereich der Neuen Mittelschule (84) und dem Kreativbereichen der Volksschule (80). Die einzigen Bereiche mit einem globalen Wert von 0 sind der Veranstaltungssaal sowie die Verwaltungsbereiche und Küchen, da in der Simulation in diesen keine Aktivitäten aus den Stundentafeln der SchülerInnen durchgeführt werden. Die meistgenutzten Unterbereiche der jeweiligen Überbereiche sind der Kreativbereich (VS), der Speisesaal (NMS), und ein Bildungsbereich (MS). (siehe Abb. 4.17)

[75] Anm.: mit Random-Seed 1

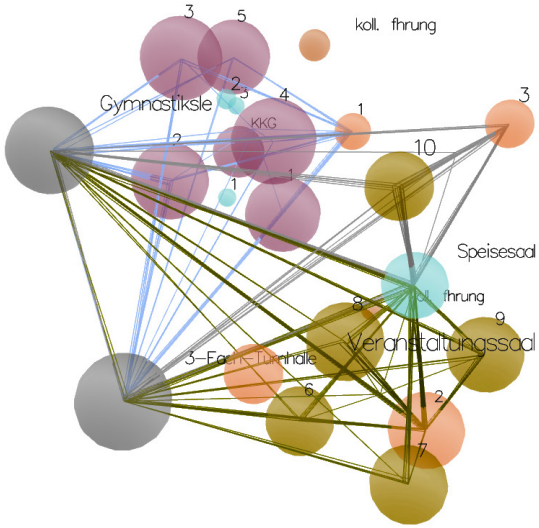


Abb. 4.16 Nachbarschaftsdiagramm inkl. der zurückgelegten Klassenwege

Bereiche mit den meisten Verbindungen	
[[Gymnastiksäle 18] [3-Fach-Turnhalle 18] [kücheSpeisesaal 9] [kreativbereich2 7] [kreativbereich1 7] [biber10 4] [biber6 4]	
Meistgenutzte Verbindungen	
[[[biber10 kreativbereich3] 88] [[kücheSpeisesaal kreativbereich2] 58] [[Gymnastiksäle NMS / FMS] 57] [[3-Fach-Turnhalle N	
WCC der Bereiche	
[[3-Fach-Turnhalle 100] [Gymnastiksäle 95] [NMS / FMS 84] [VS + KDG 77] [Musikschule 76] [Veranstaltungssaal 0]]	
WCC der Unterbereiche GTVS	
[[kreativbereich 1 100] [biber 2 64] [biber 4 62] [biber 3 59] [biber 1 59] [biber 5 50] [küche 3 0] [küche 2 0] [biber KKG 0]	
WCC der Unterbereiche NMS	
[[küche Speisesaal 100] [kreativbereich 2 91] [biber 6 72] [biber 7 71] [biber 8 71] [biber 9 71] [koll. führung 2 0]]	
WCC der Unterbereiche MS	
[[biber 10 100] [kreativbereich 3 94]]	
Verbindungen insgesamt	Summe der Through-Put aller Verbindungen
52	1016

Abb. 4.17 Werte als Resultat der Simulation

Der Überbereich mit der größten Zentralität, "3-Fach-Sporthalle", ist zugleich auch jener, der die meisten Verbindungen zu anderen Funktionsbereichen aufweist (insgesamt 18), gleichaufliegend mit den "Gymnastiksälen". An dritter Stelle dieses Klassements rangiert der Speisesaal (9). Insgesamt werden in der simulierten Schulwoche anhand der Klassenbewegungen 52 Verbindungen zwischen den Bereichen erstellt, die insgesamt 1016 mal benutzt wurden. (siehe Abb. 4.18) Die meistgenutzte Verbindungen zwischen Überbereichen sind jene zwischen den Gymnastiksälen und der Neuen Mittelschule sowie zwischen der 3-Fach-Turnhalle und der Neuen Mittelschule.

Nach zahlreichen Durchläufen der Simulation, mit variierenden "Random-Seeds", generiert das Hilfsprogramm schließlich Werte für Durchsätze und Zentralität, die sich Durchschnittswerten annähern. Im Zuge der Case Study wurde die Simulation automatisiert 1000 mal durchgeführt.[76] Die Resultate basieren daher auf ebenso vielen simulierten Schulwochen, in denen sich die Klassen anhand von Stundenplänen, die in jedem Durchlauf neu aus den Stundentafeln zusammengesetzt werden, zwischen den Funktionsbereichen bewegen. Aus diesen Informationen lässt sich ablesen, dass pro Schulwoche im Schnitt 51,7 Verbindungen von den Schulklassen erstellt werden, deren Gesamtsumme der Durchsätze im Schnitt 1009 beträgt. Jede einzelne Verbindung hat daher einen

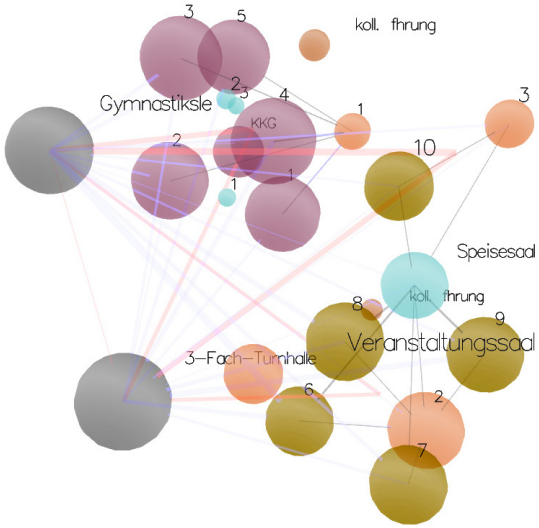


Abb. 4.18 Nachbarschaftsdiagramm inkl. der erstellten Verbindungen

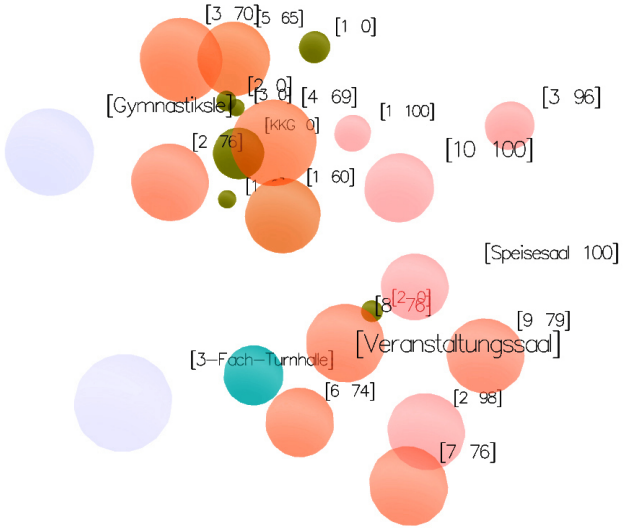


Abb. 4.19 Funktionsbereiche eingefärbt nach lokaler Zentralität

[76] Anm.: mit Random-Seeds 1 - 1000

durchschnittlichen Durchsatz von 19,5, was bedeutet, dass sie ebenso oft von Klassen auf ihrem Weg durch das simulierte Schulgebäude benutzt wird. Der Bereich mit der größten durchschnittlichen globalen WCC ist nun "Gymnastiksäle" (96,7). In absteigender Reihenfolge sind die darauffolgenden die "3-Fach-Sporthalle" (96,1), der Speisesaal (92) und der Kreativbereich der Neuen Mittelschule (85,9). (siehe Abb. 4.19 und 4.20)

Diese Informationen bezüglich der räumlichen Nachbarschaftsverhältnisse innerhalb des Bildungscampus kommen in einem nächsten Schritt zur Anordnung der Funktionsbereiche auf Geschoßebenen zum Einsatz.

4.2.4. Verteilung der Bereiche auf Geschoße

Der/die NutzerIn hat in dieser Phase die Möglichkeit, eine beliebig wählbare Anzahl von Geschoßen sowie eine maximale Geschoßfläche zu definieren. Für die Wettbewerbsaufgabe dieser Case Study werden zunächst fünf Geschoßebenen und die, aus den Auslobungsunterlagen ersichtliche, maximal bebaubare Fläche von 10500 m² als Geschoßfläche für die erste Simulationsrunde festgelegt. Zusätzlich wird prozentual eine Restfläche angegeben, welche auf jedem Geschoß mindestens frei bleiben muss, etwa als Erschließungsfläche oder für Bereiche, die in der Simulation nicht eingebunden sind. Als Wert hierfür werden 10% angenommen. Die maximal zur Verfügung stehende Fläche auf jeder

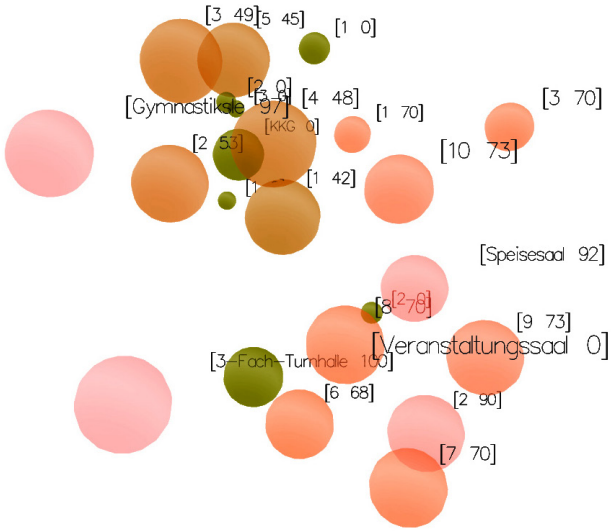


Abb. 4.20 Funktionsbereiche eingefärbt nach globaler Zentralität

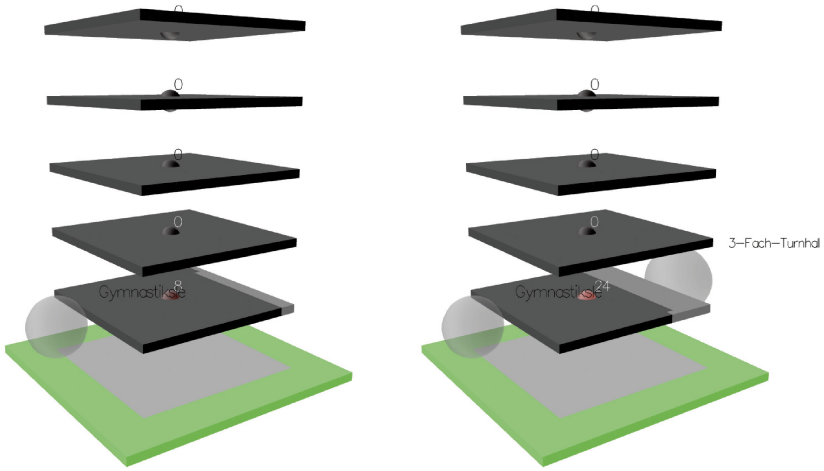


Abb. 4.21 1. und 2. Zeiteinheit der Verteilung

Ebene sind daher 9450 m². Darüber hinaus hat der/die NutzerIn die Option, als weiteren Parameter der Simulation einen Wert für die Verkleinerung der Geschoßfläche zu definieren, der am Ende jeder Simulationsrunde potenziell zur Anwendung kommen kann. Im Zuge dieser Case Study wird dafür ein Wert von 20% festgelegt, der allerdings auch während der laufenden Simulation noch angepasst werden kann.

Nach einem Grundsetup wird der Input in einem ersten Schritt räumlich dargestellt. Erkennbar sind die fünf Geschoßebenen, deren Fläche sowie ihr Verhältnis zur Grundstücksfläche, die in den Ausschreibungsunterlagen mit 22 990 m² beziffert wird.

Die Simulation läuft ab hier entsprechend des im Kapitel "Implementierung" erläuterten Algorithmus der Verteilung ab.[77] Zunächst werden alle Überbereiche in absteigender Reihenfolge nach den Werten ihrer Zentralität geordnet. Danach wird jener von ihnen mit dem größten Wert auf dem untersten Geschoß platziert. Dies ist der Funktionsbereich "Gymnastiksäle" (globale WCC: 96,7), der mit seiner Raumgröße von 892m² gerundet 8 Prozent der Geschoßfläche belegt.[78]

Nun wird der Sortierung entsprechend, in jeder Zeiteinheit ein weiterer Überbereich auf dem Geschoß deponiert, solange die maximale

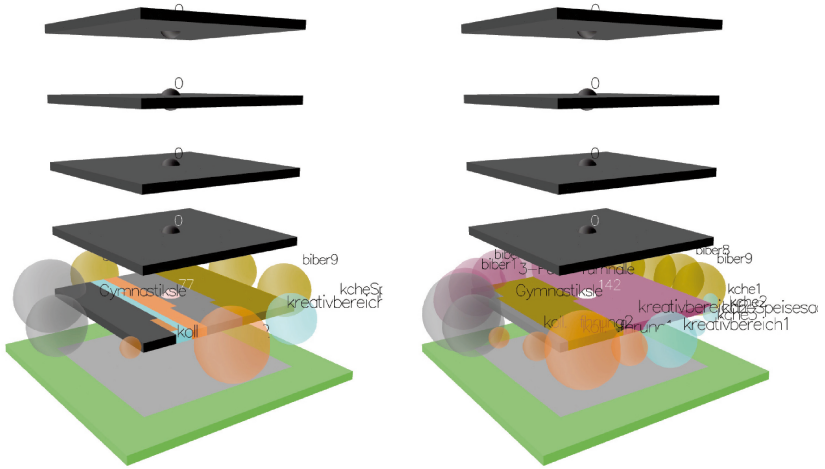


Abb. 4.22 3. und 4. Zeiteinheit der Verteilung

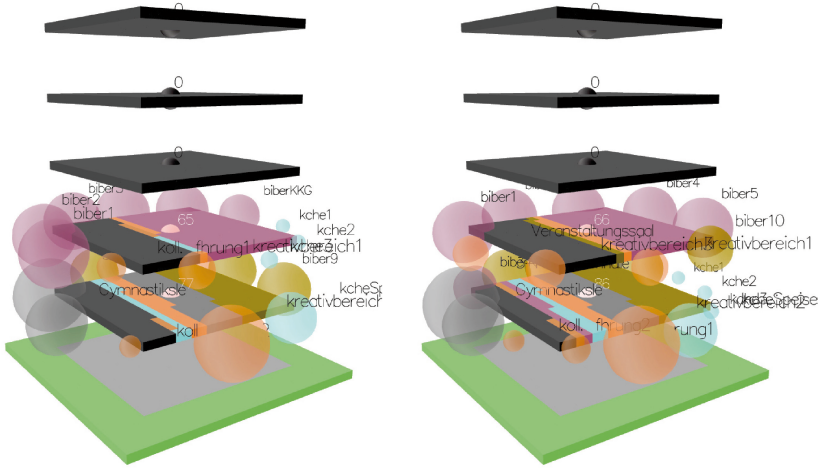


Abb. 4.23 14. und 23. (letzte) Zeiteinheit der Verteilung

[77] Anm.: mit Random-Seed 1

[78] Anm.: 892 / (10500 / 100) = 8,49

Auslastung des Geschoßes nicht überschritten wird. Der zweite Bereich in der Rangliste ist die 3-Fach-Sporthalle, der die Auslastung des Erdgeschoßes auf 24% erhöht. (siehe Abb. 4.21) In der dritten Zeiteinheit wird mit der Neuen Mittelschule bzw. Fachmittelschule erstmals ein Bereich platziert, der zugehörige Unterbereiche aufweist. Da danach die Auslastung des Geschoßes mit 77 Prozent immer noch unter der maximal zulässigen liegt, bleibt der Überbereich mitsamt allen Unterbereichen auf dem Erdgeschoß. Durch die Anordnung des nächsten Überbereichs, der Volksschule, wird die zur Verfügung stehende Fläche mit 142% allerdings erstmals überschritten. (siehe Abb. 4.22) Daher werden solange Unterbereiche der Volksschule, sortiert nach aufsteigenden lokalen WCC-Werten, auf das nächsthöhere Geschoß verschoben, bis die Auslastung des untersten Geschoßes korrigiert ist oder sich ein Großteil der Summe der lokalen WCC-Werte des Überbereichs am darüber liegenden Geschoß befindet. In diesem Fall tritt zweiteres zuerst ein, wodurch der gesamte Überbereich sein Hauptgeschoß auf das nächsthöhere verlegt und die Verteilung seiner Unterbereiche von dort erneut vollzieht. (siehe Abb. 4.23)

Nachdem schließlich alle Bereiche verteilt und der Algorithmus nach 23 Zeiteinheiten beendet ist, kann es zu einer ersten Überprüfung des Resultats kommen. Dabei fällt zunächst auf, dass die Geschoßfläche zu groß gewählt wurde. Nur zwei der fünf Geschoßebenen sind durch

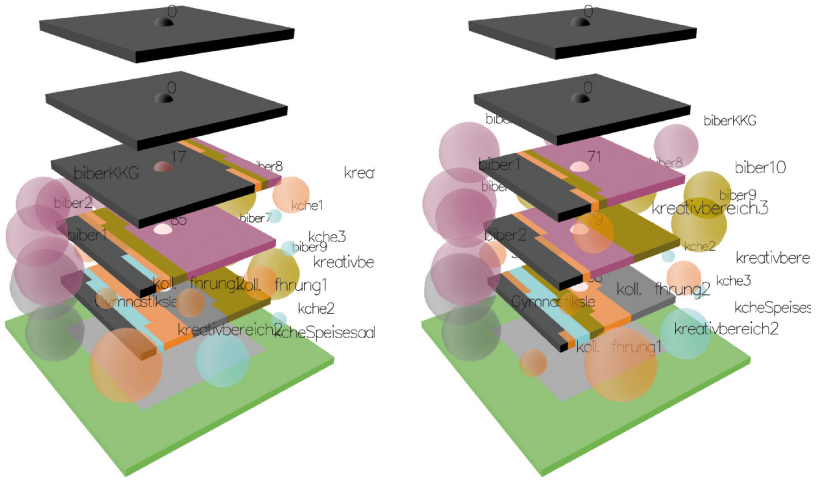


Abb. 4.24 Anordnung nach der 2. und 3. Simulationsrunde

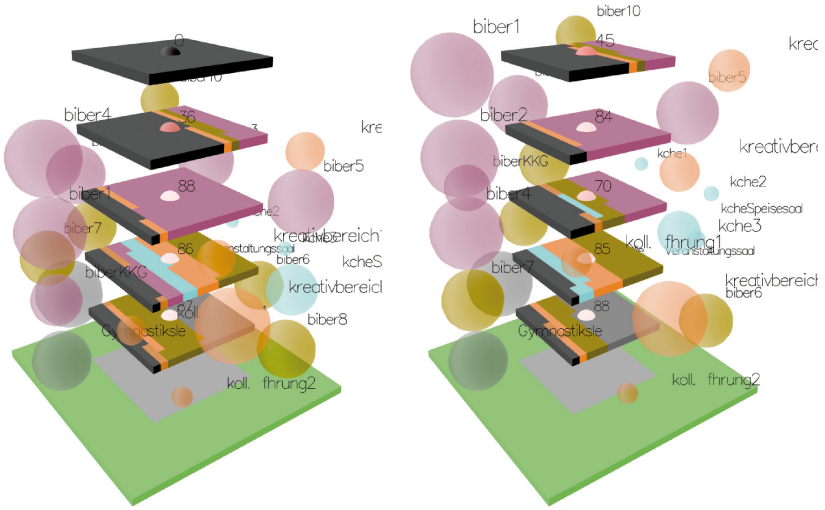


Abb. 4.25 Anordnung nach der 4. und 5. Simulationsrunde

Bereiche belegt, aber auch nicht voll ausgelastet (86 und 66%). Daher wird die Geschoßfläche für den nächsten Durchlauf automatisch um den von dem/der NutzerIn eingegebenen Wert der Verkleinerung, also 20% reduziert. Eine weitere Problemstellung, die bei genauerer Betrachtung auffällt, ist die Positionierung des Veranstaltungssaals. Dieser wurde als allerletzter Bereich angeordnet, da er von allen Überbereichen den kleinsten WCC-Wert aufweist. Demnach befindet er sich im ersten Obergeschoß, sollte aber laut Ausschreibungsunterlagen, zur leichteren Erreichbarkeit bei der Durchführung außerschulischer Veranstaltungen, im Erdgeschoß angeordnet sein. An dieser Stelle kann der/die NutzerIn einen manueller Eingriff betreffend der Reihenfolge der Verteilung vollziehen, der dafür sorgt, dass dieser Bereich ab der nächsten Simulationsrunde immer als erster ein Geschoß wählt, wodurch er zwangsläufig auf der untersten Geschoßebene platziert wird.

Der nächste Simulationsdurchlauf beginnt mit einer, auf 8400 m², reduzierten Geschoßfläche. Durch den manuellen Eingriff wird in der ersten Zeiteinheit diesmal der Veranstaltungssaal situiert. Danach läuft die Simulation abermals dem Verteilungs-Algorithmus entsprechend ab. Nach dieser zweiten Runde sind bereits drei der Geschoße ausgelastet. (siehe Abb. 4.24) In weiterer Folge wird die Geschoßfläche insgesamt sechs Mal (20, 20, 20, 20,4.4, 2.7) angepasst, bis nach sieben Simulationsrunden eine optimale Auslastung aller fünf Geschoße

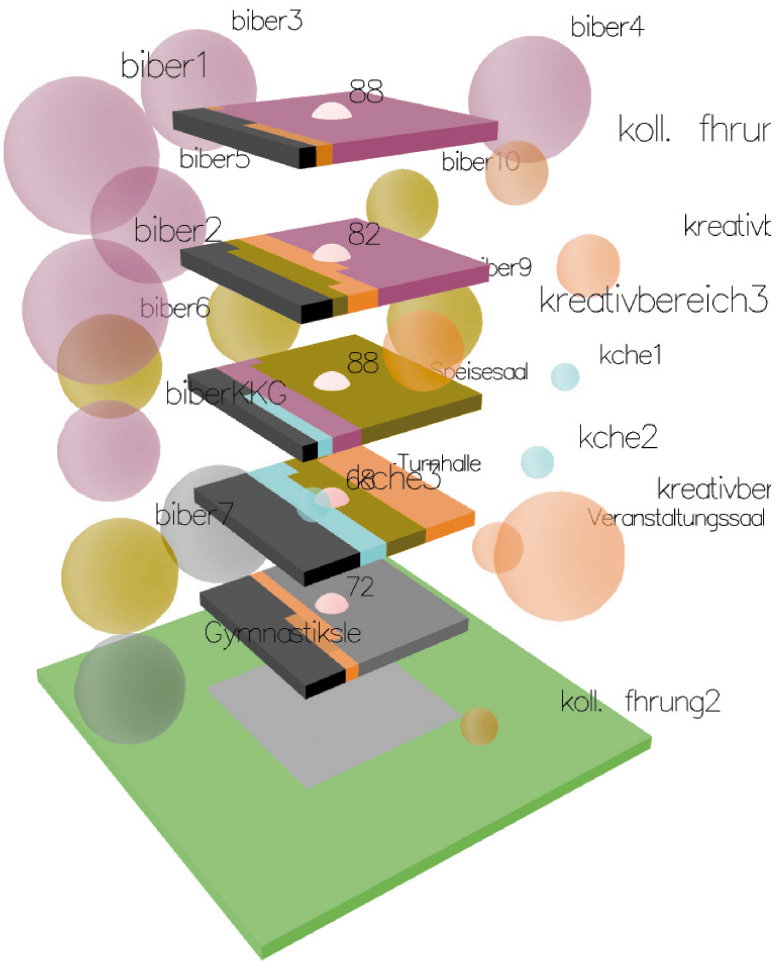


Abb. 4.26 Resultat des Verteilung für 5 Geschoße

zustande kommt. Sobald sich zumindest ein Bereich auf dem obersten Geschoß befindet (siehe Abb. 4.25), kalkuliert das Programm, anstelle des von dem/der NutzerIn festgelegten Werts, automatischen einen Verkleinerungsfaktor, der verhindert, dass die Fläche zu stark reduziert wird und eine Fehlermeldung hervorruft.

Im Zuge dieser Case Study wurde das Hilfsprogramm auch zur Simulation der Verteilung der Funktionsbereiche auf zwei bis vier Geschoßebenen angewendet. (siehe Abb. 4.27)

4.2.5. Ergebnis

Die letztendliche Geschoßfläche beträgt bei fünf Geschoßen 3999 m², was einer totalen Nutzfläche von 19995 m² entspricht. Die jeweiligen Geschoße sind zwischen 68 und 88% ausgelastet. Auf dem untersten Geschoß befinden sich mit der 3-Fach-Sporthalle, den Gymnastiksälen und dem Veranstaltungssaal alle allgemeinen Funktionsbereiche und zugleich alle Überbereiche, die keine hierarchisch untergeordneten Einheiten besitzen. Die restlichen Bereiche sind auf die vier darüber liegenden Ebenen aufgeteilt. Die Unterbereiche der Neuen Mittelschule erstrecken sich vom ersten bis zum dritten Obergeschoß, wobei sich der Großteil auf der dazwischenliegenden Ebene befindet, die dadurch als sein Hauptgeschoß definiert ist. Die Volksschule liegt mit ihren

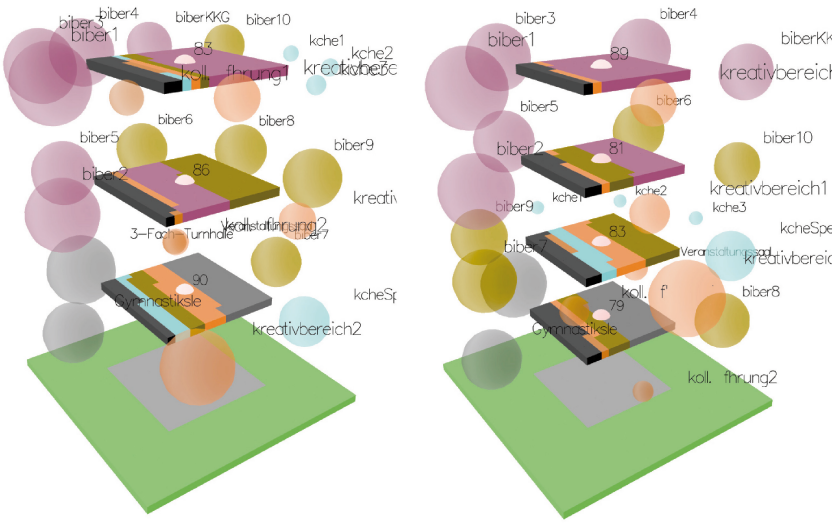


Abb. 4.27 Anordnung für 3 und 4 Geschoße

Geschoss	Funktion	Breed	Mein Bereich	Farbe	Fläche	Höhe	globale WCC	lokale WCC
0	"3-Fach-Turnhalle"	überbereiche	"3-Fach-Turnhalle"	[192 192 192 150]	1619	7.5	100	100
0	"Gymnastiksäle"	überbereiche	"GymnastiksÃ¤le"	[192 192 192 150]	892	5.5	99	99
0	"kücheSpeisesaal"	unterbereiche	"NMS / FMS"	[18 128 129 150]	649	3	54	65
0	"koll. führung2"	unterbereiche	"NMS / FMS"	[238 131 59 150]	96	3	0	0
0	"Veranstaltungssaal"	überbereiche	"Veranstaltungssaal"	[234 102 28 150]	283	4	0	0
1	"kreativbereich2"	unterbereiche	"NMS / FMS"	[234 102 28 150]	1185	3	82	100
1	"biber7"	unterbereiche	"NMS / FMS"	[253 204 26 150]	896	3	67	82
1	"biber8"	unterbereiche	"NMS / FMS"	[253 204 26 150]	914	3	66	80
2	"biber9"	unterbereiche	"NMS / FMS"	[253 204 26 150]	914	3	65	80
2	"biber6"	unterbereiche	"NMS / FMS"	[253 204 26 150]	896	3	63	77
2	"küche3"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
2	"biberKKG"	unterbereiche	"VS + KDG"	[240 154 191 150]	622	3	0	0
2	"küche1"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
2	"koll. führung1"	unterbereiche	"VS + KDG"	[238 131 59 150]	168	3	0	0
2	"küche2"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
3	"kreativbereich1"	unterbereiche	"VS + KDG"	[234 102 28 150]	267	3	85	100
3	"biber3"	unterbereiche	"VS + KDG"	[240 154 191 150]	1121	3	50	58
3	"biber2"	unterbereiche	"VS + KDG"	[240 154 191 150]	1117	3	50	58
3	"kreativbereich3"	unterbereiche	"Musikschule"	[234 102 28 150]	300	3	59	100
3	"biber10"	unterbereiche	"Musikschule"	[253 204 26 150]	500	3	58	98
4	"biber4"	unterbereiche	"VS + KDG"	[240 154 191 150]	1121	3	48	57
4	"biber5"	unterbereiche	"VS + KDG"	[240 154 191 150]	1115	3	48	57
4	"biber1"	unterbereiche	"VS + KDG"	[240 154 191 150]	1117	3	31	36

Abb. 4.28 Ausgabe als Excel-Dateien

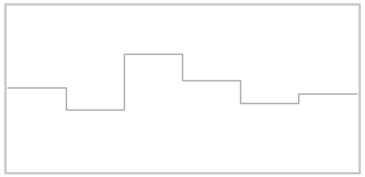
Unterbereichen auf den beiden obersten Geschoßen, die Musikschule nur am vierten. (siehe Abb. 4.26)

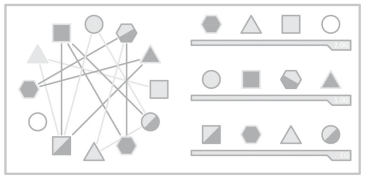
Bei der Anordnung auf vier Geschoßebenen wird die Ausgangsgeschoßfläche in sechs Simulationsrunden fünf Mal bis zu einer letztendlichen Fläche von 4832 m² reduziert. Die Auslastung der Ebenen beträgt in diesem Fall zwischen 79 und 89%.

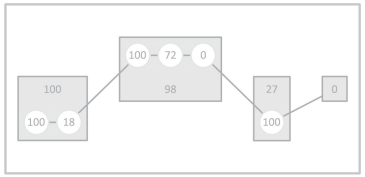
Wird die Anzahl der Geschoße auf drei herabgesetzt, ergibt sich aus der Simulation nach viermaligem Durchlauf eine Geschoßfläche von 6157 m² bei einer Auslastung zwischen 83 und 90%.

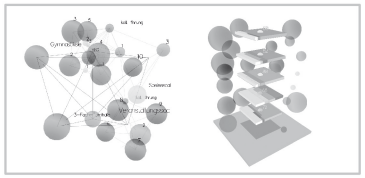
Im Zuge einer Simulation der Anordnung der Funktionsbereiche auf zwei Geschoßebenen ist eine bestmögliche Auslastung bei einer Geschoßfläche von 9 476 m² gegeben.

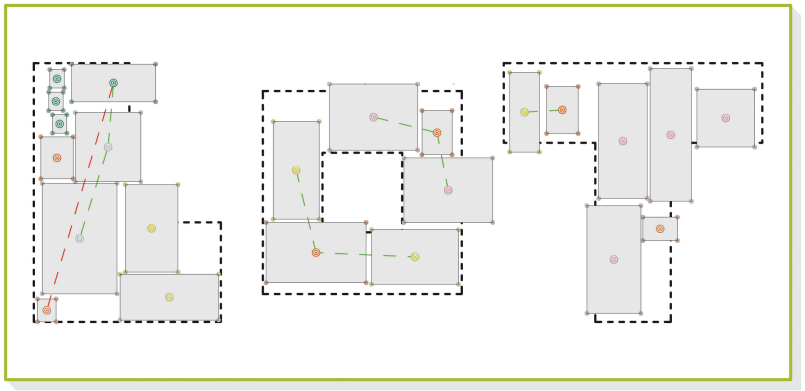
Die finalen Vorschläge der Anordnung der Funktionsbereiche auf Geschoßebenen werden abermals in Excel-Listen ausgegeben und zur Weiterverarbeitung zur Verfügung gestellt. (siehe Abb. 4.28)











5. Diskussion

5.1. Anwendung des Hilfsprogramms als Analysetool **87**

5.1.1. Vergleich pädagogischer Konzepte 87

5.1.2. Analyse finaler Entwürfe und bestehender Bauten 87

5.2. Zusammenhang Regional - Lokal **88**

5.2.1. Bildungscampus Nordbahnhof 2030 88

5.3. Ansatzpunkte zukünftiger Arbeiten **89**

5.3.1. Input der Regionalen Bedarfsplanung 89

5.3.2. Parametrisierung im Schulgenerator 90

5.3.3. Simulation anderer AkteurInnen als Agenten 90

5.3.4. Erweiterung des hierarchischen Aufbaus 91

5.3.5. Exakte Situierung auf Geschoßebenen 91

5.3.6. Diplomarbeit "Multikriterielle Grundrissgenerierung" 92

5. Diskussion

Das im Zuge dieser Diplomarbeit entwickelte Hilfsprogramm (siehe Kapitel 3.2) und die Simulation der Regionalen Bedarfsplanung (siehe Kapitel 3.1) bieten einen ersten Ansatz der Computer- und Simulations-basierten Planungsunterstützung für den Schulbau. In diesem Kapitel wird zunächst die Möglichkeit der Anwendung des Hilfsprogramms als Analysetool erläutert. Der darauffolgenden Abschnitt beschäftigt sich mit der gegenseitigen Abhängigkeit der regionalen und lokalen Planungsebenen. Anschließend werden potenzielle Anknüpfungspunkte für weiterführende Arbeiten besprochen, die sich mit der Problematik der Schulplanung auseinandersetzen.

5.1. Anwendung des Hilfsprogramms als Analysetool

5.1.1. Vergleich pädagogischer Konzepte

Das Hilfsprogramm kann in frühen Phasen der Planung nicht nur zur Generation räumlicher Anordnungsdiagramme eingesetzt werden, sondern ebenso dazu, verschiedene Methoden der pädagogischen Konzeption zu evaluieren. Diese können für zeitgenössische Bildungsbauten, die auf aktuelle Tendenzen der Schulplanung (siehe Kapitel 1.1) Bezug nehmen, von der Ausbildung der Bildungsbereiche bis zur Rhythmisierung des Schultages reichen. Die verschiedenen Varianten der Lernclusterbildung bieten beispielsweise je nach Ausführung als Jahrgangscluster, jahrgangsübergreifend oder als Fachraumcluster, sowohl unterschiedliche Potenziale als auch damit verbundene variierende Bewegungsabfolgen aller AkteurInnen. Diese können mit dem Hilfsprogramm simuliert und anhand verschiedener Gesichtspunkte, etwa der zurückgelegten Wege der SchülerInnen, verglichen werden.

5.1.2. Analyse finaler Entwürfe und bestehender Bauten

Das Hilfsprogramm könnte allerdings ebenso zur Analyse finaler Entwürfe oder bestehender Schulgebäude angewendet werden. In diesem Fall würden die vorliegenden Nachbarschaftsverhältnisse der, hier räumlich bereits fixierten, Funktionsbereiche in finalen Grundrissplänen anhand der Simulation der SchülerInnenwege überprüft. Diese Art der Anwendung des Hilfsprogramms wäre darüber hinaus auch für bereits bestehende Schulbauten umsetzbar. Erfolgt die Simulation in einem konkreten Schulbau anhand fixer Stundenpläne aller Klassen, anstelle randomisiert zusammengesetzter Stundentafeln, könnte die hinsichtlich der Reduzierung der Wege optimierte Zuordnung der Räume als Stammklassen der SchülerInnen erfolgen. Im Umkehrschluss wäre auch bei fixierter räumlicher Zuordnung eine Simulation zur Erstellung der Stundenpläne aller Klassen durchführbar, die entweder die Auslastung aller Bereiche oder die von den SchülerInnen in einer Schulwoche zurückgelegten Wege optimieren kann. In diesem Fall wäre auch die

Einbindung einer Gewichtung möglich, um etwa das Alter der SchülerInnen entsprechend zu berücksichtigen.

5.2. Zusammenhang Regional - Lokal

Die städtebauliche Simulation zur Ermittlung des prognostizierten Bedarfs zusätzlicher Klassen einer Region und das weiterführende Hilfsprogramm zur Unterstützung im Planungsprozess von Schulbauten bieten zusammen eine umfassende Hilfestellung auf mehreren Ebenen des Schulwesens, von der großmaßstäblichen regionalen Raumplanung bis zum konkreten Entwurf einzelner Gebäude. Die resultierenden Daten der Regionalen Bedarfsplanung können dazu als Input des Schulgenerators eingesetzt werden.

5.2.1. Bildungscampus Nordbahnhof 2030

Für den als Case Study herangezogenen Wettbewerb des Neubaus eines Bildungscampus am Gelände des Nordbahnhofs im 2. Wiener Gemeindebezirk können die Ergebnisse der "Regionalen Bedarfsplanung" für die Stadt Wien herangezogen werden, um dessen Größe anhand der prognostizierten Daten der SchülerInnenzahlen im Schuljahr 2030/31 zu überprüfen. (siehe Abb. 5.1)

Im Jahr 2016 bietet der Bildungscampus Platz für insgesamt 6,2% aller SchülerInnen Wiens. Aufgeteilt auf die verschiedenen Schultypen liegen

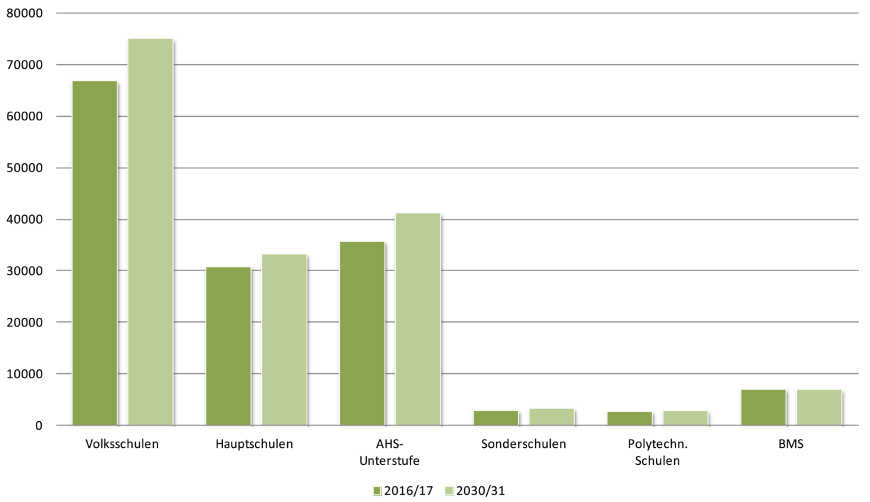


Abb. 5.1 Vergleich prognostizierter SchülerInnenzahlen 2016/17 und 2030/31

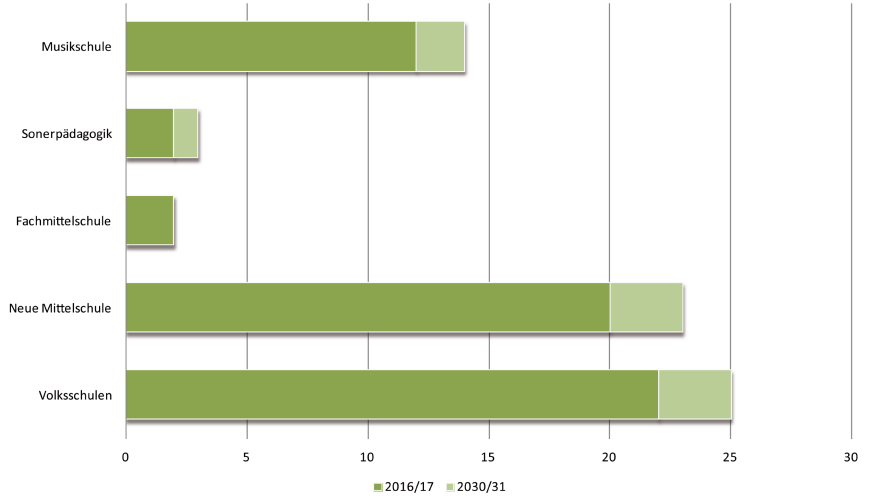


Abb. 5.2 Notwendige Klassen des Bildungscampus im Schuljahr 2030/31

die Werte bei 6,8% für Volksschulen (22 Klassen), 2,6% für AHS-Unterstufe bzw. Hauptschule (32), 5% für Sonderschulen (2) und 4,9% für Berufsbildende Mittlere Schulen (2). Die Schultypen "AHS-Unterstufe" und "Hauptschule" wurden zusammengefasst, da der Bildungscampus mit der Neuen Mittelschule eine Schulform anbietet, die keine Unterscheidung der beiden trifft. Die folgenden Berechnungen basieren auf der Annahme, dass das Gebäude im Jahr 2030 denselben Anteil der SchülerInnen der Stadt in sich aufnehmen wird, wie dies im Jahr 2016, dem Jahr seiner Planung, der Fall ist.

Das Hilfsprogramm kalkuliert dazu, anhand der zuvor berechneten Faktoren und der prognostizierten Entwicklungen, die Anzahl der SchülerInnen, die der Bildungscampus im Jahr 2030 aufnehmen müsste. Basierend auf den prognostizierten Daten müsste die Schule insgesamt 170 zusätzlichen SchülerInnen Platz bieten, von denen mit 89 der Großteil auf den Schultyp "AHS-Unterstufe" bzw. "Hauptschule" abfällt, gefolgt von "Volksschule" (60). Die Zahlen der Schulbesuche von "Berufbildende Mittlere Schulen" sind rückläufig, daher müssten für diesen sechs SchülerInnen weniger aufgenommen werden.

Auf die notwendigen Klassenzahlen umgelegt, um keine davon mit mehr als 25 SchülerInnen belegen zu müssen, bedeuten diese Zahlen eine Zunahme von insgesamt 9 Klassen (3 VS, 3 NMS, 2 MS, 1 SoPäd). (siehe Abb. 5.2)

Das Hilfsprogramm kann demnach nicht nur dazu verwendet werden, die notwendige Größe von Schulbauten anhand prognostizierter Entwicklungen der SchülerInnezahlen zu überprüfen, sondern bietet darüber hinaus auch die Möglichkeit, ein zusätzliches Nachbarschaftsdiagramm, basierend auf den aktualisierten Informationen der SchülerInnen- und Klassenzahlen des Jahres 2030 zu erstellen.

Die aus der Simulation der SchülerInnenbewegungen abgeleiteten Werte der Funktionsbereiche und Verbindungen, bezüglich des Durchsatzes und der Zentralität, sind von diesen Änderungen allerdings kaum betroffen und weisen nur minimale Abweichungen auf.

5.3. Ansatzpunkte zukünftiger Arbeiten

Dieses Kapitel dient der Abklärung von potenziellen Ansatzpunkten weiterführender Arbeiten, die auf die Ergebnisse dieser Diplomarbeit aufbauen.

5.3.1. Input der Regionalen Bedarfsplanung

Grundsätzlich gilt, dass die Resultate der Simulationen stark vom Input abhängen. Je besser der von dem/der NutzerIn zur Verfügung gestellte Input, desto bessere Ergebnisse liefert das Programm. Eine zentrale Stellung nehmen vor allem die prognostizierten Daten in der Simulation zur Regionalen Bedarfsplanung ein. Die aktuellsten öffentlich verfügbaren

Informationen hierzu für die Stadt Wien stammen aus dem Jahr 2012 und haben das Schuljahr 2009/10 als Ausgangsjahr. Sie weichen bereits für das Schuljahr 2014/15, je nach Schulform, mehr oder weniger stark von den tatsächlichen Werten der, auf verschiedene Schultypen aufgeteilten, SchülerInnenzahlen ab. (siehe Abb. 5.3)

5.3.2. Parametrisierung im Schulgenerator

In der ersten Phase des Schulgenerators, der Simulation der SchülerInnenwege innerhalb einer Schulwoche zur Erstellung des Nachbarschaftdiagramms, stellt die Untersuchung der Auswirkungen unterschiedlicher Rhythmisierungen auf die Simulation einen Ansatzpunkt für weiterführende Arbeiten dar. Es gilt zu klären, inwieweit die Ergebnisse durch unterschiedlich lange Aufenthalte der Klassen in einzelnen Bereichen oder durch die Blockung von Unterrichtsstunden variieren.

In der, darauf aufbauenden, zweiten Phase der Simulation, zur Anordnung der Funktionsbereiche auf Geschoßebenen, wurde ein Algorithmus gewählt, für den als Parametrisierung eine exakte Geschoßanzahl und eine maximale Geschoßfläche, die über mehrere Simulationsrunden automatisiert angepasst wird, von dem/der NutzerIn festzulegen sind. Andere Methoden der Parametrisierung, etwa die Definition einer fixen Geschoßfläche mit variabel anpassbarer Geschoßanzahl, könnten ebenso

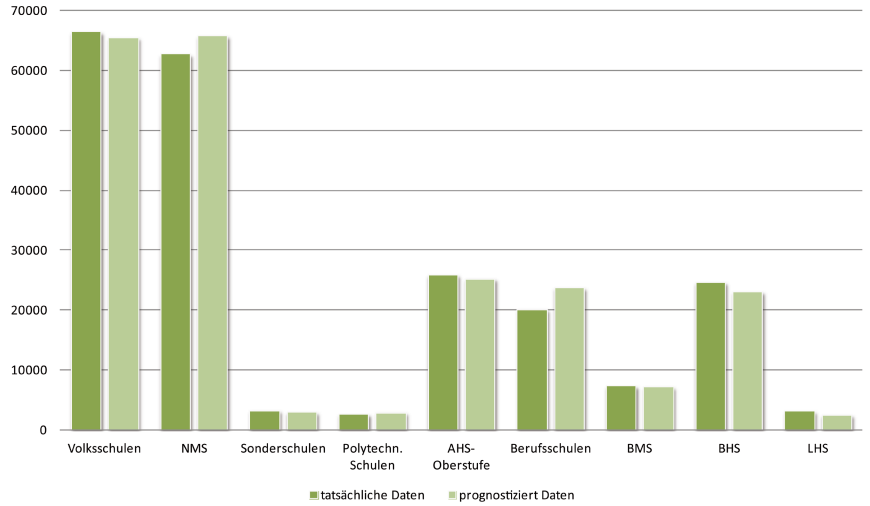


Abb. 5.3 Vergleich prognostizierter und tatsächlicher SchülerInnenzahlen (2014/15)

zur Anwendung kommen. Dabei würde in der Simulation nicht die Geschoßfläche, sondern die Anzahl der Ebenen je nach Flächenbedarf abgeändert werden.

5.3.3. Simulation anderer AkteurInnen als Agenten

Eine Einschränkung der Simulation zur Erstellung des Nachbarschaftsdiagramms stellt die alleinige Berücksichtigung der SchülerInnen und deren Bewegungsabläufe in einer Schulwoche dar. Diese Problematik wird, soweit möglich, durch die hierarchische Gliederung der Funktionsbereiche ausgeglichen, die dafür sorgt, dass auch jene Bereiche, die im Diagramm keine Verbindungen aufweisen, zumeist

korrekt platziert werden. Zur Ermittlung umfassender räumlicher Nachbarschaftsbeziehungen könnten in weiteren Simulationen zusätzliche Nutzergruppen der Schule, etwa das Lehrpersonal oder MitarbeiterInnen, als Agenten implementiert werden, um ihre Abläufe zu simulieren. Durch entsprechende Gewichtung der resultierenden Verbindungen wäre dadurch eine weitgehende Optimierung der Abläufe und Wege aller AkteurInnen der Schule möglich.

5.3.4. Erweiterung des hierarchischen Aufbaus

Die Ausweitung des hierarchischen Aufbaus der Simulation bietet einen weiteren Ansatzpunkt für zukünftige Arbeiten. Dieser könnte beispielsweise vom gesamten Schulbau, in mehreren Schichten, bis zu den atomaren Einheiten des Gebäudes, in Form einzelner Räume, ausgedehnt werden. Dabei würden in der Simulation auf allen hierarchischen Ebenen Anordnungen der jeweils untergeordneten Einheiten parallel zueinander ablaufen.

5.3.5. Exakte Situierung auf Geschoßebenen

Eine andere Möglichkeit der Weiterentwicklung des Hilfsprogramms besteht darin, die Anordnung der Funktionsbereiche nicht auf Geschoßebenen zu begrenzen, sonder deren exakte Lage auf ihren jeweils zugewiesenen Geschoßen zu definieren. Zu diesem Zweck gibt es eine Vielzahl unterschiedlicher Lösungswege. Eine Variante wäre die

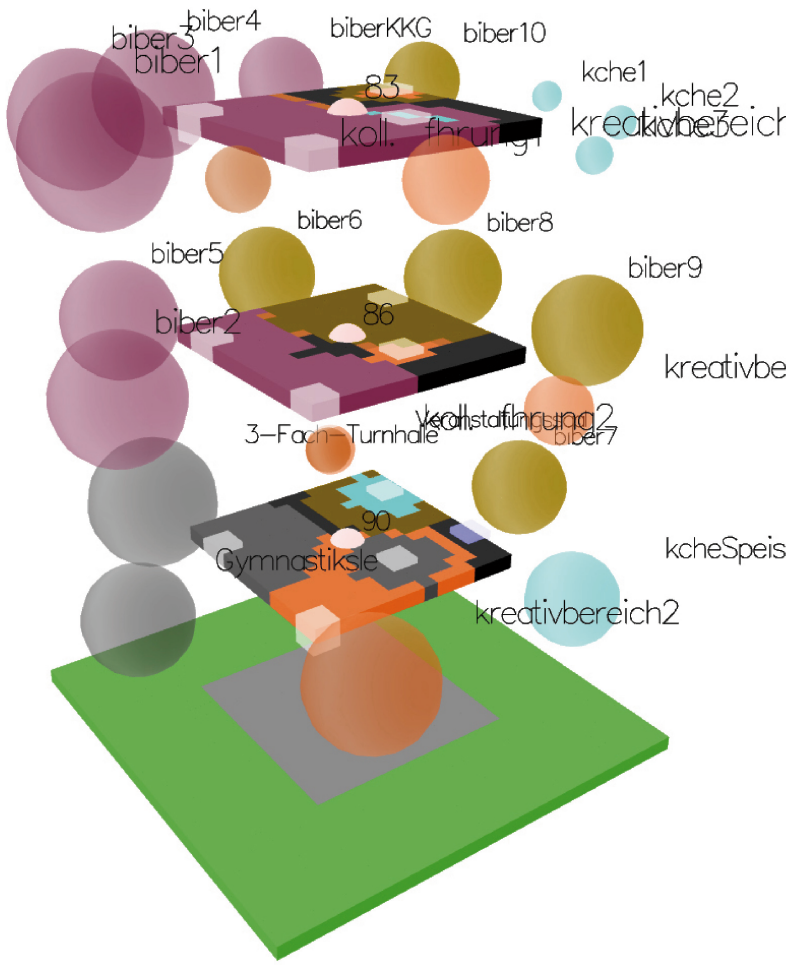


Abb. 5.4 Anordnung nach Hotspots

Festlegung von, über mehrere Ebenen synchronisierter, Hotspots auf allen Geschoßen, in deren Nähe sich Bereiche, in Abhängigkeit der Werte ihrer geforderten Zentralität, positionieren. Je größer besagte Werte, desto näher befinden sie sich räumlich an den Hotspots. Durch die Synchronisierung über alle Ebenen des Diagramms wären zusätzlich geschoßübergreifende Anordnungen durchführbar, indem alle Unterbereiche desselben Überbereichs die räumliche Zuordnung zu miteinander verknüpften Hotspots wählen. (siehe Abb. 5.4)

5.3.6. Diplomarbeit "Multikriterielle Grundrissgenerierung"[79]

Eine Möglichkeit der Weiterverarbeitung der Daten, die als Ergebnis der Simulationen dieser Arbeit entstehen, beschreibt Bernadette Arendt in ihrer Diplomarbeit "Multikriterielle Grundrissgenerierung in der Anwendung an Schulgebäuden Nutzen, Grenzen und Potenziale" zum Thema automatisierter responsiver Layoutfindung, die parallel zu dieser entwickelt wurde. Die bestmögliche Verteilung der Bereiche des Bildungscampus auf Geschoße, die als Output des Geschoßaufteilungsalgorithmus generiert wird, dient als Grundlage dafür.

Als Schnittstelle zur Übergabe der Daten des Ergebnisses der Simulation zur Verteilung auf Geschoßebenen kommen CSV-Dateien (comma-

[79] Arendt, Bernadette 2016 [im druck], "Multikriterielle Grundrissgenerierung in der Anwendung an Schulgebäuden Nutzen, Grenzen und Potenziale", Diplomarbeit, TU Wien, Fakultät für Architektur und Raumplanung

Geschoss	Funktion	Breed	Mein Bereich	Farbe	Fläche	Höhe	globale WCC	lokale WCC
0	"3-Fach-Turnhalle"	überbereiche	"3-Fach-Turnhalle"	[192 192 192 150]	1619	7.5	100	100
0	"Gymnastiksäle"	überbereiche	"GymnastiksÃ¤le"	[192 192 192 150]	892	5.5	99	99
0	"kücheSpeisesaal"	unterbereiche	"NMS / FMS"	[18 128 129 150]	649	3	54	65
0	"koll. führung2"	unterbereiche	"NMS / FMS"	[238 131 59 150]	96	3	0	0
0	"Veranstaltungssaal"	überbereiche	"Veranstaltungssaal"	[234 102 28 150]	283	4	0	0
1	"kreativbereich2"	unterbereiche	"NMS / FMS"	[234 102 28 150]	1185	3	82	100
1	"biber7"	unterbereiche	"NMS / FMS"	[253 204 26 150]	896	3	67	82
1	"biber8"	unterbereiche	"NMS / FMS"	[253 204 26 150]	914	3	66	80
2	"biber9"	unterbereiche	"NMS / FMS"	[253 204 26 150]	914	3	65	80
2	"biber6"	unterbereiche	"NMS / FMS"	[253 204 26 150]	896	3	63	77
2	"küche3"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
2	"biberKKG"	unterbereiche	"VS + KDG"	[240 154 191 150]	622	3	0	0
2	"küche1"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
2	"koll. führung1"	unterbereiche	"VS + KDG"	[238 131 59 150]	168	3	0	0
2	"küche2"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
3	"kreativbereich1"	unterbereiche	"VS + KDG"	[234 102 28 150]	267	3	85	100
3	"biber3"	unterbereiche	"VS + KDG"	[240 154 191 150]	1121	3	50	58
3	"biber2"	unterbereiche	"VS + KDG"	[240 154 191 150]	1117	3	50	58
3	"kreativbereich3"	unterbereiche	"Musikschule"	[234 102 28 150]	300	3	59	100
3	"biber10"	unterbereiche	"Musikschule"	[253 204 26 150]	500	3	58	98
4	"biber4"	unterbereiche	"VS + KDG"	[240 154 191 150]	1121	3	48	57
4	"biber5"	unterbereiche	"VS + KDG"	[240 154 191 150]	1115	3	48	57
4	"biber1"	unterbereiche	"VS + KDG"	[240 154 191 150]	1117	3	31	36

Abb. 5.5 Übergabe mittels CSV-Dateien (5 Geschoße)

Geschoss	Funktion	Breed	Mein Bereich	Farbe	Fläche	Höhe	globale WCC	lokale WCC
0	"3-Fach-Turnhalle"	bereiche	"3-Fach-Turnhalle"	[192 192 192 150]	1619	7.5	100	100
0	"GymnastiksÃ¤le"	bereiche	"GymnastiksÃ¤le"	[192 192 192 150]	892	5.5	99	99
0	"kreativbereich2"	unterbereiche	"NMS / FMS"	[234 102 28 150]	1185	3	82	100
0	"biber7"	unterbereiche	"NMS / FMS"	[253 204 26 150]	896	3	67	82
0	"biber8"	unterbereiche	"NMS / FMS"	[253 204 26 150]	914	3	66	80
0	"biber9"	unterbereiche	"NMS / FMS"	[253 204 26 150]	914	3	65	80
0	"biber6"	unterbereiche	"NMS / FMS"	[253 204 26 150]	896	3	63	77
0	"kÃ¼cheSpeisesaal"	unterbereiche	"NMS / FMS"	[18 128 129 150]	649	3	54	65
0	"koll. fÃ¼hrung2"	unterbereiche	"NMS / FMS"	[238 131 59 150]	96	3	0	0
0	"Veranstaltungssaal"	bereiche	"Veranstaltungssaal"	[234 102 28 150]	283	4	0	0
1	"kreativbereich1"	unterbereiche	"VS + KDG"	[234 102 28 150]	267	3	85	100
1	"biber3"	unterbereiche	"VS + KDG"	[240 154 191 150]	1121	3	50	58
1	"biber2"	unterbereiche	"VS + KDG"	[240 154 191 150]	1117	3	50	58
1	"biber4"	unterbereiche	"VS + KDG"	[240 154 191 150]	1121	3	48	57
1	"biber5"	unterbereiche	"VS + KDG"	[240 154 191 150]	1115	3	48	57
1	"biber1"	unterbereiche	"VS + KDG"	[240 154 191 150]	1117	3	31	36
1	"kÃ¼che3"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
1	"kÃ¼che1"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
1	"biberKKG"	unterbereiche	"VS + KDG"	[240 154 191 150]	622	3	0	0
1	"kÃ¼che2"	unterbereiche	"VS + KDG"	[18 128 129 150]	63	3	0	0
1	"koll. fÃ¼hrung1"	unterbereiche	"VS + KDG"	[238 131 59 150]	168	3	0	0
1	"kreativbereich3"	unterbereiche	"Musikschule"	[234 102 28 150]	300	3	59	100
1	"biber10"	unterbereiche	"Musikschule"	[253 204 26 150]	500	3	58	98

Abb. 5.6 Anordnung auf 2 Geschoßen

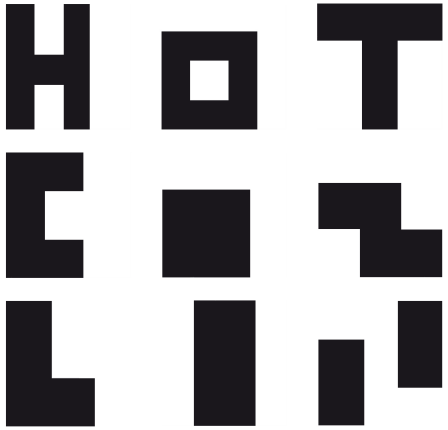


Abb. 5.7 9 Grundrisstypologien

separated values) zum Einsatz, welche alle relevanten Daten der Funktionsbereiche bezüglich Raumbezeichnung, Größe, Nachbarschaftsverhältnisse und Geschoß enthalten. Darüber hinaus werden, aus Gründen der graphischen Konsistenz, zusätzlich Informationen der farblichen Gestaltung weitergereicht. (siehe Abb. 5.5 und 5.6) Diese Daten werden daraufhin über ein VBA-Programm (Visual Basic für Applications) mit Angaben zu den jeweiligen Seitenverhältnissen der Funktionsbereiche ergänzt und geschoßweise erneut gespeichert. Als zusätzlicher Input dienen neun verschiedene Grundrisstypologien, die als Basis der folgenden Verteilung der Funktionsbereiche dienen. (siehe Abb. 5.7) Anschließend werden alle verfügbaren Informationen in das Layout-Programm geladen (siehe Abb. 5.8), woraufhin ein Algorithmus, unter Berücksichtigung der Anforderungen zu Größe und Proportion, die, zunächst randomisiert in der Programmwelt geladenen, Funktionsbereiche hinsichtlich dreier Kriterien auf den Geschoßebenen bewegt:

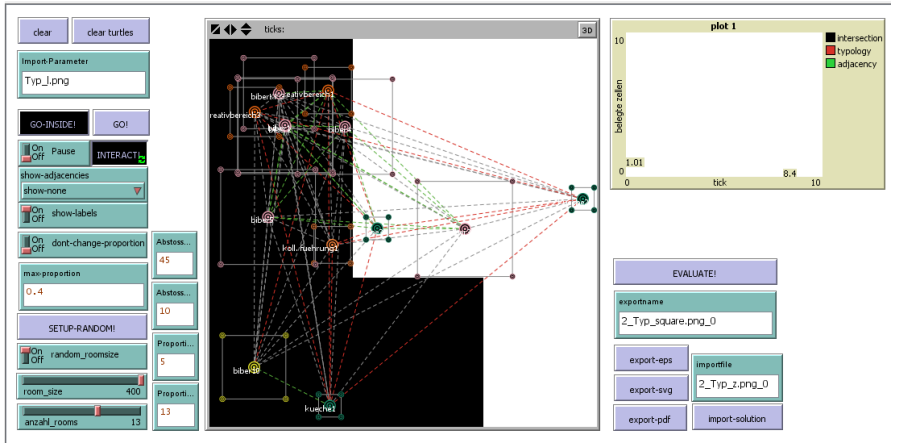


Abb. 5.8 Ausgangspunkt des Algorithmus

1. Räume wollen sich gegenseitig nicht überschneiden
2. Räume wollen innerhalb einer von dem/der NutzerIn geladenen Gebäudetypologie sein
3. Positive Nachbarschaften ziehen sich an, negative stoßen sich ab

Hinsichtlich dieser drei Ziele ergeben sich für alle Räume permanent neue Vektoren, in deren Richtungen sie sich bewegen oder auf deren Basis sie eine Proportionsänderung vornehmen. Während der Raumbewegung kann der/die NutzerIn jederzeit Eingriffe vornehmen und die Positionen der Räume mittels Drag & Drop nach seinen Wünschen anpassen. Das Programm reagiert darauf in Echtzeit und fährt mit dem Algorithmus auf Basis der geänderten Situation fort. Kann die Lösung nicht mehr weiter verbessert werden oder befindet der/die NutzerIn die Lösung als

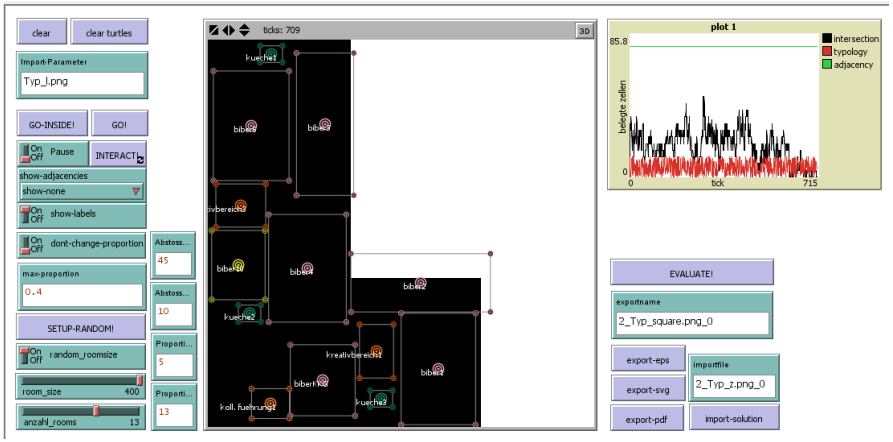


Abb. 5.9 Anordnung der Funktionsbereiche auf einer L-Typologie

ausreichend, wird eine Evaluierung zur Zentralität gestartet. (siehe Abb. 5.9)

Dabei entsteht eine Vielzahl unterschiedlicher Grundrissvarianten abhängig der gewählten Grundrisstypologie und Geschoßanzahl. (siehe Abb. 5.10 - 5.14)

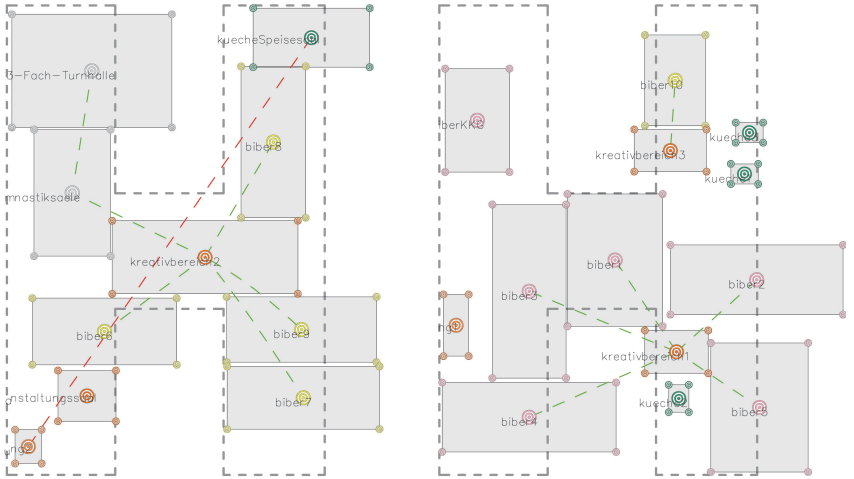


Abb. 5.10 Anordnung auf 2 Geschoßen - Grundrissvariante H

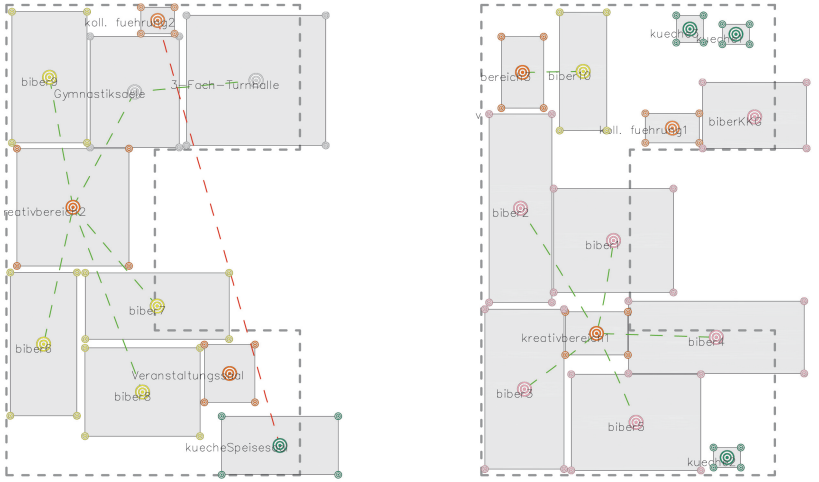


Abb. 5.11 Anordnung auf 2 Geschoßen - Grundrissvariante C

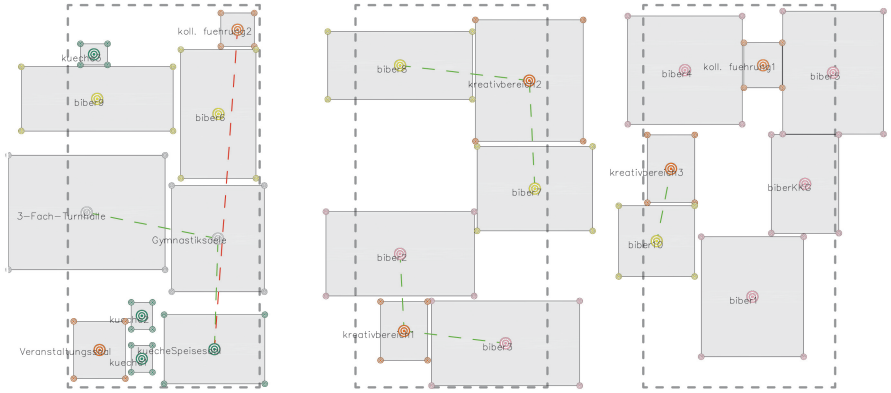


Abb. 5.12 Anordnung auf 3 Geschoßen - Grundrissvariante Riegel

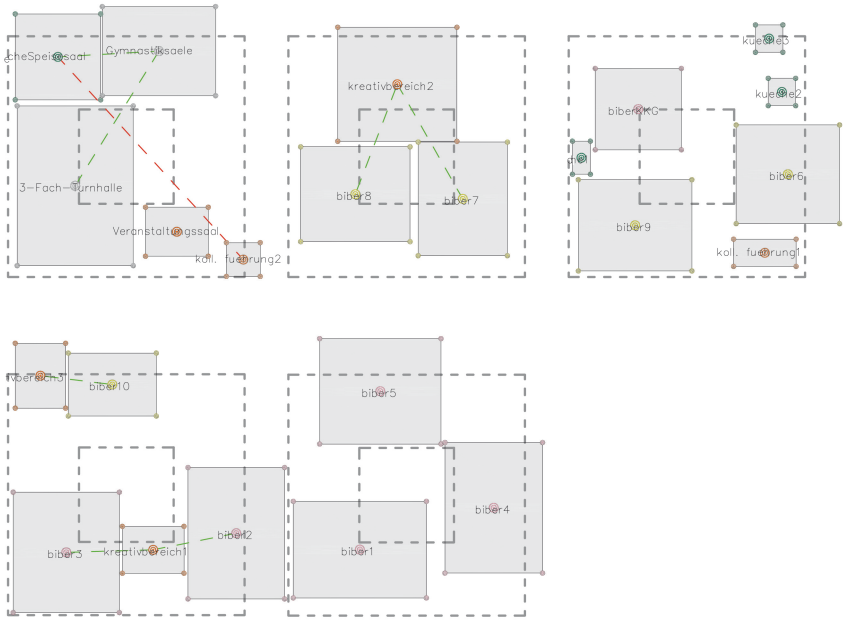


Abb. 5.13 Anordnung auf 5 Geschoßen - Grundrissvariante Hof

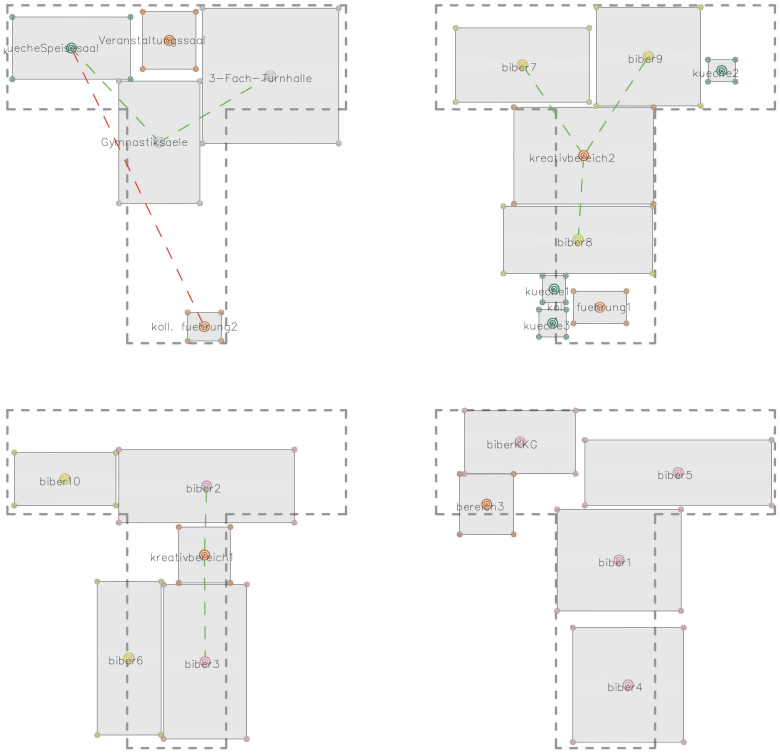


Abb. 5.14 Anordnung auf 4 Geschoßen - Grundrissvariante T

6. Schluss

Im Zuge dieser Diplomarbeit wurde ein Simulations-basiertes Hilfsprogramm entwickelt, das primär in frühen Planungsphasen eingesetzt werden kann, um die räumlichen Nachbarschaftsverhältnisse der Funktionsbereiche von Schulbauten in den architektonischen Entwurfsprozess einfließen zu lassen. Dabei kommt eine umfassende Analyse mit anschließender Illustration der Ergebnisse auf mehreren Planungsebenen zur Anwendung. Zunächst wird die Situation des Schulsystems einer Region anhand prognostizierter Daten der Entwicklung der SchülerInnenzahlen überprüft, wodurch der Bedarf zusätzlicher Bildungsbauten eruiert wird. Anschließend wird das Hauptaugenmerk auf spezifische, einzelne Planungsaufgaben gelegt, für die, aus der Simulation der Abläufe von Schulklassen in einer Schulwoche, optimierte räumliche Beziehungen abgeleitet und dreidimensional dargestellt werden. In einem letzten Schritt des Hilfsprogramms dienen die ermittelten Nachbarschaftsverhältnisse zur Erstellung eines Vorschlages der Anordnung der Funktionsbereiche auf eine von dem/der NutzerIn definierte Anzahl von Geschoßebenen. (siehe Kapitel 3)

Das Hilfsprogramm wurde im Zuge einer Case Study einerseits zur Abbildung des zu erwartenden zukünftigen Klassenzuwachs der Stadt Wien angewendet, andererseits zur Generierung eines Entwurfsansatzes für den Wettbewerb eines Bildungscampus am Gelände des Nordbahnhofs im zweiten Wiener Gemeindebezirk. (siehe Kapitel 4)

Diese Diplomarbeit bietet nicht nur einen ersten Ansatz der Computer- und Simulations-basierten Planungsunterstützung für den Schulbau, sondern auch Potenzial zur Weiterentwicklung auf verschiedene Anwendungsbereiche. So kann das Programm etwa eingesetzt werden, um die räumlichen Auswirkungen unterschiedlicher pädagogischer Konzepte, die in den letzten Jahren und Jahrzehnten entstanden, zu untersuchen und zu vergleichen. Ebenso ist ein Einsatz als Analysetool fertiger Entwürfe oder bestehender Bildungsbauten denkbar. Darüber hinaus gibt es einige Aspekte, die als Ansatzpunkte weiterführender Arbeiten aufgegriffen werden können. Dazu zählt beispielsweise die Integration anderer AkteurInnen der Schule (Lehrpersonal, MitarbeiterInnen,...) zur Ermittlung universeller räumlicher Zusammenhänge. Ebenso besteht die Möglichkeit zur Erweiterung des Hilfsprogramms mittels exakter Situierung der Funktionsbereiche auf den Geschoßebenen. (siehe Kapitel 5)

Durch Adaption kann der vorliegende Ansatz auch in anderen Bereichen abseits der Bauaufgabe Schule angewandt werden, um anhand der Bewegungsabläufe unterschiedlicher Nutzergruppen der jeweiligen Planungsaufgabe, Nachbarschaftsbeziehungen aufzuzeigen und in Entwürfe zu integrieren.

7. Anhang

7.1. Literaturverzeichnis **99**

7.2. Abbildungsverzeichnis **101**

7.3. Sourcecode **104**

7.3.1. Klassenzuwachs 104

7.3.2. Schulgenerator 124

7.3.2.1. Simulation der Bewegungsabläufe 124

7.3.2.2. Verteilalgorithmus 146

7.4. Excel-Tabellen: Klassenzuwachs in Wien bis 2030 **163**

7.1. Literaturverzeichnis

1. http://www.oecd.org/berlin/themen/pisa-hintergrund.htm, 1.3.2016, 14:22

2. Lederer, A., Pampe, B., (2011): Vergleich ausgewählter Richtlinien zum Schulbau - Kurzfassung. Montag Stiftung Urbane Räume, Bonn, Rahmen und Richtlinien für einen leistungsfähigen Schulbau in Deutschland, Heft 1

3. Kühn, C., (2013): Cluster und Lernstraße statt Gang und Klassenzimmer, Merkur,Heft 766, März 2013, 245-250

4. Haas, D.E., (2013) Leitlinien für leistungsfähigen Schulbauten in Deutschland. Montag Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bund Deutscher Architekten, Verband Bildung und Erziehung, Bonn, Berlin

5. ÖISS (2014): Räumliche Anforderungen für Unterrichtsräume. Kapitel 6, Richtlinien für den Schulbau, S.2

6. Magistratsabteilung 34: Bau- und Gebäudemanagement (2016): Raumblätter für Neue Mittelschulen der Stadt Wien, Wien

7. Meuser. N (2014): Handbuch und Planungshilfe Schulbauten. DOM publishers, Berlin

8. Becker, T., Eckmann, C. (2012): Schulen planen und bauen. Grundlagen und Prozesse. Montag Stiftung Jugend und Gesellschaft, Montag Stiftung Urbane Räume, Bonn

9. Seydel, O. (2013): Die kleine Schule in der großen Schule. Lehren & Lernen, Heft 12 2013, 7-13

10. Arnold, R., Schüßler, I. (1998): Wandel der Lernkulturen. Ideen und Bausteine für ein lebendiges Lernen. Darmstadt

11. Magistratsabteilung 34: Bau- und Gebäudemanagement (2014): Raumbuch Campus+: Bildungseinrichtung mit Mehrwert der Stadt Wien, Wien

12. Huven, K. (2007): Pädagogische Architektur. Stiftung Urbane Räume, Montag Stiftung Jugend und Gesellschaft, Bonn

13. Flaschberger, E., Grandy, S., Hofmann, F., Lehner, L., Teutsch, F., Vogl, S., Felder-Puig, R.(2015): Die zeitliche Gestaltung des Schulalltags. Bestandsaufnahme und Empfehlungen zur Umsetzung in Österreich. Wien: LBIHPR Forschungsbericht.

14. Kolbe, F.U., Rabenstein, K., Reh, S. (2006): Expertise „Rhythmisierung“. Hinweise für die Planung von Fortbildungsmodulen für Moderatoren. Berlin, Mainz: Johannes-Gutenberg Universität Mainz, Technische Universität Berlin.

15. Holtappels, H.G. (1995): Zeitgestaltung und Schulqualität: Entwicklung der Lernkultur in "Vollen Halbtagsschulen" Niedersachsens. In:

Holtappels, H:G: (Hg.): Ganztagserziehuung in der Schule. Schule und Gesellschaft 10. Opladen: Leske + Budrich, S. 209-229

16. Owens, J.A., Belon, K., Moss, P. (2010): Impact of delaying school start time on adolescent sleep, mood, and behavior. Archives of Pediatrics & Adolescent Medicine, 164 (7), 608-614.

17. Vogel, J. (2012): Zeitstruktur entwickeln - Schritt für Schritt. IV. Fachtagung für Schulen mit Ganztagsangeboten/Sek. I im Land Brandenburg "Ganztag als Merkmal zukunftsfähiger Schulen" (14.03.2012)

18. Wicki, W., Dietrich, C. (2007): Expertise Zeitstrukturen Basel-Stadt; Luzern: Forschungsbericht Nr. 6 der Pädagogischen Hochschule Zentralschweiz, Hochschule Luzern.

19. Montessori, M. (1949): The Absorbent Mind, The Theosophical Publishing House, Adyar

20. Malina, B., Busch, C., (2010): Inklusion: Leitlinien für die BIldungspolitik. Deutsche UNESCO-Kommission, Bonn

21. OECD (2010): PISA 2009 Results: Overcoming Social Background - Equality in Learning Opportunitiers and Outcomes (Volume II), http://dx.doi.org/10.1787/9789264091504-en

22. Lobos, D., Donath, D. (2010): The problem of space layout in architecture: A survey and reflections. arquiteturarevista, Ausgabe 6, Nummer 2, Juli/ Dezember 2010, 136 - 161

23. Lorenz, W., Bicher, M., Wurzer, G. (2015): Adjacency in hospital planning

24. Arendt, Bernadette (2016): [im druck], "Multikriterielle Grundrissgenerierung in der Anwendung an Schulgebäuden Nutzen, Grenzen und Potenziale", Diplomarbeit, TU Wien, Fakultät für Architektur und Raumplanung

7.2. Abbildungsverzeichnis

Abb.1.1 Methodenmix 12
Abb.1.2 Unterschiedliche Raumnutzungen 12
Abb.1.3 Zeit in der Schule 13
Abb.1.4 Rhythmisierung des Schultages 14
Abb.1.5 Inklusion 17
Abb.1.6 Räumliche Polyvalenz 18
Abb.1.7 Bildungscampus 19
Abb.1.8 Räumliche Organisationsmodelle I 20
Abb.1.9 Varianten der Clusterbildung 22
Abb.1.10 Räumliche Organisationsmodelle II 23
Abb.1.11 Verwaltungs- und Teambereiche 25
Abb.1.12 Raumbedarfe 26
Abb.2.1 Bedarfsdeckung zusätzlicher Klassen 36
Abb.2.2 Aufbau einer Schule 37
Abb.2.3 Akteure einer Schule 37
Abb.2.4 Simulation der SchülerInnenwege 38
Abb.2.5 Anordnung der Bereiche auf Geschoßebenen 39
Abb.3.1 Ebenen der Simulation 42
Abb.3.2 Benutzeroberfläche: Bedarfssimulation 43
Abb.3.3 Ergebnis Bedarfssimulation 43
Abb.3.4 Begriffsklärung 45
Abb.3.5 Hierarchischer Aufbau 46
Abb.3.6 Klassen als Agenten 47
Abb.3.7 Benutzeroberfläche: Simulation der SchülerInnenwege 48
Abb.3.8 Bewegungen der Klassen nach der ersten Zeiteinheit 50
Abb.3.9 Verbindungsarten 51
Abb.3.10 Vergleich: Globale (links) und Lokale (rechts) WCC 52

Abb.3.11 Verbindungen am Ende der Simulation 53
Abb.3.12 Benutzeroberfläche: Verteilungssimulation 54
Abb.3.13 Verteilung nach "Globaler WCC" 56
Abb.3.14 Verteilung nach "Link-Neighbors" 57
Abb.3.15 Hierarchische Verteilung 58
Abb.3.16 Manuelle Verteilung 58
Abb.3.17 Hauptgeschoß 59
Abb.3.18 Aktionsdiagramm der Verteilalgorithmus 60
Abb.3.19 Anordnung nach der ersten Zeiteinheit 61
Abb.3.20 Anordnung nach den ersten platzierten Unterbereichen 61
Abb.3.21 Anordnung nach der ersten Simulationsrunde 62
Abb.3.22 Ergebnis des Verteilalgorithmus für 3 Geschoße 63
Abb.4.1 Entwicklungen der SchülerInnenzahlen bis 2030/31 nach Bezirken 67
Abb.4.2 Entwicklungen der SchülerInnenzahlen bis 2030/31 nach Schultypen 67
Abb.4.3 Berechnete Entwicklungen der SchülerInnenzahlen bis 2030/31 68
Abb.4.4 Klassenzahlen im Ausgangsjahr der Simulation (2009/10) 68
Abb.4.5 Setup der Simulation zur Bedarfsdeckung für Wien 69
Abb.4.6 Resultat der Simulation zur Bedarfsdeckung zusätzlicher Klassen 69
Abb.4.7 Hinzugefügte Klassen im Schuljahr 2009/10 nach Schultypen 70
Abb.4.8 Hinzugefügte Klassen im Schuljahr 2009/10 nach Bezirken 70
Abb.4.9 Bis 2030/31 hinzugefügte Klassen nach Bezirken und Schultypen 71
Abb.4.10 Insgesamt hinzugefügte Klassen nach Schultypen 71
Abb.4.11 Hinzugefügte Kassen nach Jahr und Schultypen 72
Abb.4.12 Entwicklungen der SchülerInnen / Klasse exkl. neuer Klassen 72
Abb.4.13 Entwicklungen der SchülerInnen / Klasse inkl. neuer Klassen 73
Abb.4.14 Funktionsdiagramm 75
Abb.4.15 Dreidimensionale Anordnung der Funktionsbereiche 76
Abb.4.16 Nachbarschaftsdiagramm inkl. der zurückgelegten Klassenwege 77
Abb.4.17 Werte als Resultat der Simulation 77
Abb.4.18 Nachbarschaftsdiagramm inkl. der erstellten Verbindungen 78

Abb.4.19 Funktionsbereiche eingefärbt nach lokaler Zentralität 78
Abb.4.20 Funktionsbereiche eingefärbt nach globaler Zentralität 79
Abb.4.21 1. und 2. Zeiteinheit der Verteilung 79
Abb.4.22 3. und 4. Zeiteinheit der Verteilung 80
Abb.4.23 14. und 23. (letzte) Zeiteinheit der Verteilung 80
Abb.4.24 Anordnung nach der 2. und 3. Simulationsrunde 81
Abb.4.25 Anordnung nach der 4. und 5. Simulationsrunde 81
Abb.4.26 Resultat des Verteilung für 5 Geschoße 82
Abb.4.27 Anordnung für 3 und 4 Geschoße 83
Abb.4.28 Ausgabe als Excel-Dateien 83
Abb.5.1 Vergleich prognostizierter SchülerInnenzahlen 2016/17 und 2030/31 88
Abb.5.2 Notwendige Klassen des Bildungscampus im Schuljahr 2030/31 88
Abb.5.3 Vergleich prognostizierter und tatsächlicher SchülerInnenzahlen (2014/15) 90
Abb.5.4 Anordnung nach Hotspots 91
Abb.5.5 Übergabe mittels CSV-Dateien (5 Geschoße) 92
Abb.5.6 Anordnung auf 2 Geschoßen 92
Abb.5.7 9 Grundrisstypologien 93
Abb.5.8 Ausgangspunkt des Algorithmus 93
Abb.5.9 Anordnung der Funktionsbereiche auf einer L-Typologie 94
Abb.5.10 Anordnung auf 2 Geschoßen - Grundrissvariante H 94
Abb.5.11 Anordnung auf 2 Geschoßen - Grundrissvariante C 94
Abb.5.12 Anordnung auf 3 Geschoßen - Grundrissvariante Riegel 95
Abb.5.13 Anordnung auf 5 Geschoßen - Grundrissvariante Hof 95
Abb.5.14 Anordnung auf 4 Geschoßen - Grundrissvariante T 95

7.3. Sourcecode

7.3.1. Klassenzuwachs

https://gist.github.com/anonymous/45fed0c684fc278cf05a81411e3063ff

```
__includes ["schulbesuche_prognose.nls"]

extensions [csv]

globals [
  centers_districts
  output
  schultypen_wien_prognose
  schulbesuche_wien
  schüler_wien_2014
  klassen_wien_2014
  klassen_wien
  schulen_wien_2014
  schüler/klasse
  colors_bezirke
  jahre
  neue_schulen
  neue_schulen_districts
  neue_klassen_districts
  neue_schulen_insgesamt
  neue_schulen_insgesamt_list
  neue_klassen_insgesamt_list
  schüler/klasse_districts
  schultyp_nummer
  colors_schultypen
]

breed [districts district]
breed [cursors cursor]
breed [schools school]

districts-own    [zahl    schüler/klasse_district    schüler/klasse_district_list
neue_schule neue_schule_list neue_klassen? neue_klassen neue_klassen_list]
schools-own [mein_bezirk mein_schultyp klassen size_alt]
patches-own [district_number]


to setup
  ca
  random-seed 0
  setup-globals
  setup-statistics
  setup-patches
  setup-districts
  setup-cursor
  reset-ticks
end

to reset
  ct
  random-seed 0
  set jahre 2009
  set klassen_wien klassen_wien_2009
  set neue_schulen []
  repeat (length bezirke_wien - 1)[
    set neue_schulen lput 0 neue_schulen
  ]
  setup-districts
  setup-cursor
  clear-all-plots
  reset-ticks
end

to setup-globals
  set jahre 2009
  set schultypen_wien_prognose (list
    "Insgesamt"
    "Volksschulen"
    "Hauptschulen"
    "Sonderschulen"
    "Polytechn. Schulen"
    "AHS-Unterstufe"
    "AHS-Oberstufe"
    "Berufsschulen"
    "Berufsbildende mittlere Schulen"
    "Berufsbildende hoehere Schulen"
    "Lehrerbildende hoehere Schulen"
    )
  set bezirke_wien (list
    "Alle"
    "1. Bezirk"
    "2. Bezirk"
    "3. Bezirk"
    "4. Bezirk"
    "5. Bezirk"
```

```
  "6. Bezirk"
  "7. Bezirk"
  "8. Bezirk"
  "9. Bezirk"
  "10. Bezirk"
  "11. Bezirk"
  "12. Bezirk"
  "13. Bezirk"
  "14. Bezirk"
  "15. Bezirk"
  "16. Bezirk"
  "17. Bezirk"
  "18. Bezirk"
  "19. Bezirk"
  "20. Bezirk"
  "21. Bezirk"
  "22. Bezirk"
  "23. Bezirk"
  )

set neue_schulen_insgesamt_list [
  [106 0 9 0 5 33 0 0 19 36 4]
  [111 0 11 0 5 36 0 0 19 36 4]
  [122 0 12 0 5 42 0 0 19 38 6]
  [134 0 14 0 5 49 0 0 19 41 6]
  [144 0 16 0 5 56 0 0 19 42 6]
  [150 0 17 0 5 59 0 0 19 44 6]
  [166 0 19 0 5 68 1 0 19 47 7]
  [178 1 20 0 5 76 1 0 19 48 8]
  [188 1 22 0 5 81 2 0 19 49 9]
  [203 2 24 0 5 89 2 0 19 53 9]
  [219 6 25 0 5 97 3 0 19 53 11]
  [240 14 26 0 5 105 3 0 20 55 12]
  [270 19 29 0 5 121 5 0 20 59 12]
  [305 26 31 0 5 137 8 0 20 62 16]
  [345 33 33 0 5 153 16 0 20 69 16]
  [388 41 35 0 5 170 20 0 20 76 21]
  [435 51 39 0 5 188 27 0 20 83 22]
  [494 68 42 0 5 207 35 0 20 93 24]
  [553 83 45 0 6 225 44 0 21 103 26]
  [614 99 50 0 6 243 53 0 23 111 29]
  [674 117 53 0 6 261 64 0 23 120 30]
  [741 133 60 0 6 280 76 0 23 130 33]
]

set neue_klassen_insgesamt_list [
  [106 0 9 0 5 33 0 0 19 36 4]
  [5 0 2 0 0 3 0 0 0 0 0]
  [11 0 1 0 0 6 0 0 0 2 2]
  [12 0 2 0 0 7 0 0 0 3 0]
  [10 0 2 0 0 7 0 0 0 1 0]
  [6 0 1 0 0 3 0 0 0 2 0]
  [16 0 2 0 0 9 1 0 0 3 1]
  [12 1 1 0 0 8 0 0 0 1 1]
  [10 0 2 0 0 5 1 0 0 1 1]
  [15 1 2 0 0 8 0 0 0 4 0]
  [16 4 1 0 0 8 1 0 0 0 2]
  [21 8 1 0 0 8 0 0 1 2 1]
  [30 5 3 0 0 16 2 0 0 4 0]
  [35 7 2 0 0 16 3 0 0 3 4]
  [40 7 2 0 0 16 8 0 0 7 0]
  [43 8 2 0 0 17 4 0 0 7 5]
  [47 10 4 0 0 18 7 0 0 7 1]
  [59 17 3 0 0 19 8 0 0 10 2]
  [59 15 3 0 1 18 9 0 1 10 2]
  [61 16 5 0 0 18 9 0 2 8 3]
  [60 18 3 0 0 18 11 0 0 9 1]
  [67 16 7 0 0 19 12 0 0 10 3]
]

  set neue_klassen_districts [
    [[4 0 0 0 0 4 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [1 1 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [2
1 0 0 0 0 1 0 0 0 0] [3 0 0 0 0 1 2 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [3 0 0 0 0
1 2 0 0 0 0] [3 1 0 0 0 1 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [2 0 0 0 0 1 1 0 0
0 0] [1 0 0 0 0 0 1 0 0 0 0] [4 1 1 0 0 1 1 0 0 0 0]]
    [[0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 1 0 0 1 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [2 0 1 0 0 1 0 0 0 0 0]]
    [[4 0 0 0 0 4 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [2
1 0 0 0 0 1 0 0 0 0] [3 0 0 0 0 1 2 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [3 0 0 0 0
1 2 0 0 0 0] [3 1 0 0 0 1 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [2 0 0 0 0 1 1 0 0
0 0] [1 0 0 0 0 0 1 0 0 0 0] [4 1 1 0 0 1 1 0 0 0 0]]
```

```
    [[3 0 0 0 0 0 0 0 3 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 1 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 1 0 0] [2 1 0 0 0 1 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [2 1 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [3 1 1 0 0
1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 1 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 1 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0]]
    [[0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 1 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0]]
    [[2 0 1 0 0 0 0 0 1 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 1 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 0 0 0 0 0 1] [1 0 0 0 0 1 0 0 0 0 0] [2 1 0 0 0 0 0 0 0 0 1] [0 0 0 0 0
0 0 0 0 0 0] [2 1 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 0 1] [1 0 0 0 0 0 1 0 0
0 0] [2 1 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 0 1]]
    [[2 0 0 0 0 2 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 0 1] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1
0 0 0 0 0 0 0 0 0 1] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 0 1] [0 0 0 0 0
0 0 0 0 0 0] [3 0 0 0 0 1 1 0 0 0 1] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0
0 1] [3 1 0 0 0 1 1 0 0 0 0] [1 0 0 0 0 0 0 0 0 0 1]]
    [[1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 1 0] [1 0 0 0 0
1 0 0 0 0 0] [2 0 0 0 0 0 1 0 0 1 0] [1 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0
1 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [1 0 0 0 0 0 0 0 0 0 0]]
    [[11 0 7 0 2 0 0 0 0 0 2] [2 0 2 0 0 0 0 0 0 0 0] [2 0 1 0 0 0 0 0 0 0 1]
[1 0 1 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0 0] [3 0 2
0 0 0 0 0 0 0 1] [1 0 1 0 0 0 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0
0 0 0 0] [3 1 1 0 0 0 0 0 0 0 1] [5 4 1 0 0 0 0 0 0 0 0] [4 3 1 0 0 0 0 0 0 0
0] [6 2 1 0 0 2 0 0 0 0 1] [6 3 1 0 0 1 0 0 0 1 0] [6 2 1 0 0 1 0 0 0 1 1] [7 2
1 0 0 2 0 0 0 2 0] [7 3 1 0 0 1 0 0 0 1 1] [8 2 1 0 1 2 1 0 0 1 0] [7 3 1 0 0 1
0 0 0 1 1] [7 2 1 0 0 1 1 0 0 2 0] [7 2 1 0 0 2 1 0 0 1 0]]
    [[1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [4 2 0 0 0 1 1 0 0 0 0] [2 1 1 0 0
0 0 0 0 0 0] [4 2 1 0 0 1 0 0 0 0 0] [4 1 0 0 0 1 1 0 0 1 0] [3 1 1 0 0 1 0 0 0
0 0] [4 2 1 0 0 1 0 0 0 0 0] [2 1 0 0 0 0 1 0 0 0 0]]
    [[10 0 0 0 0 0 0 0 3 7 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0
0 0 1 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [2 1 0 0 0 1 0
0 0 0 0] [2 2 0 0 0 0 0 0 0 0 0] [3 2 0 0 0 1 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0
0] [4 2 0 0 0 1 1 0 0 0 0] [3 1 1 0 0 0 0 0 0 1 0] [2 1 0 0 0 1 0 0 0 0 0] [1 1
0 0 0 0 0 0 0 0 0] [5 1 1 0 0 1 1 0 0 1 0] [1 1 0 0 0 0 0 0 0 0 0] [3 1 1 0 0 1
0 0 0 0 0] [1 1 0 0 0 0 0 0 0 0 0] [4 1 1 0 0 0 1 0 0 1 0]]
    [[4 0 0 0 0 2 0 0 2 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0] [2
0 0 0 0 1 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0] [1 0 0 0 0
0 0 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [1 0 0 0 0 0 0 0 0
1 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0] [3 0 0 0 0 1 1 0 0 1 0] [2
0 0 0 0 0 0 0 2 0] [3 0 0 0 0 1 1 0 0 1 0] [1 0 0 0 0 0 0 0 0 1 0] [4 0 0 0 0
1 1 0 0 2 0] [2 0 0 0 0 1 0 0 0 1 0] [2 0 0 0 0 0 1 0 0 1 0] [2 0 0 0 0 0 1 0 0
1 0] [3 0 0 0 0 0 1 0 0 2 0] [2 0 0 0 0 1 0 0 0 1 0]]
    [[7 0 0 0 0 1 0 0 1 5 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2
0 0 0 0 1 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0
1 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [0
0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [1 0 0 0 0 0 0 0 0 1 0] [2 0 0 0 0
1 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [3 0 0 0 0 1 0 0 1
1 0] [2 0 0 0 0 0 1 0 0 1 0] [1 0 0 0 0 1 0 0 0 0 0]]
    [[2 0 0 0 2 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [2 0 0 0 0 1 1 0 0 0 0] [2 0 0 0 0 1 1 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 0 0 0 0 0 1 0 0 0 0] [3 0 1 0 0 1 1 0 0 0 0]]
    [[1 0 1 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 1 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0 0] [1
0 0 0 1 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 1 0 0 1 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [1 1 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [3 1 1 0 0 1 0 0 0
0 0] [2 1 0 0 0 1 0 0 0 0 0] [2 1 1 0 0 0 0 0 0 0 0]]
```

```
    [[4 0 0 0 0 1 0 0 3 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
0 0 0 0 1 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [2 1 0 0 0 0 0 0 0 1 0] [2 0 0 0 0 1 0 0 1 0 0] [1 1 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [3 1 0 0 0 1 0 0 0 1 0]]
    [[0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
0 1 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [1 0 0 0 0 0 1 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 0 1 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 1 1 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 0 1 0 0
0 0] [2 0 0 0 0 1 1 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0]]
    [[3 0 0 0 0 0 0 0 0 2 1] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0 1 0] [0 0 0 0 0
0 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 1 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 0 0 0
1 0] [2 0 0 0 0 1 0 0 0 0 1] [1 0 0 0 0 0 0 0 0 1 0] [3 0 0 0 0 2 0 0 0 1 0] [1
0 0 0 0 0 0 0 0 1 0] [2 0 0 0 0 1 0 0 0 1 0] [4 0 0 0 0 1 0 0 0 2 1] [2 0 0 0 0
1 0 0 0 1 0] [2 0 0 0 0 1 0 0 0 1 0] [3 0 0 0 0 1 0 0 0 2 0] [3 0 0 0 0 1 0 0 0
1 1] [3 1 0 0 0 1 0 0 0 1 0] [2 0 0 0 0 1 0 0 0 1 0]]
    [[1 0 0 0 0 0 0 0 1 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0
0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0
0 1 0 0 1 0] [1 0 0 0 0 0 0 0 0 1 0] [1 0 0 0 0 0 0 0 0 1 0] [6 2 1 0 0 0 1 0 0
2 0] [1 0 0 0 0 0 0 0 0 1 0] [3 1 0 0 0 0 1 0 0 1 0]]
    [[36 0 0 0 0 10 0 0 4 22 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0]
[2 0 0 0 0 1 0 0 0 1 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [2 0 0
0 0 2 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 1 0
0 0 1 0] [1 0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 1 0 0 0 0 1] [4 0 0 0 0 3 0 0 0 1
0] [4 0 0 0 0 3 0 0 0 0 1] [4 0 0 0 0 3 0 0 0 1 0] [5 0 0 0 0 3 0 0 0 1 1] [6 3
0 0 0 2 0 0 0 0 1] [6 2 0 0 0 3 0 0 0 1 0] [8 3 0 0 0 3 0 0 0 1 1] [7 2 0 0 0 3
2 0 0 0 0] [9 3 0 0 0 3 1 0 0 1 1] [9 3 0 0 0 2 2 0 0 1 1]]
    [[5 0 0 0 0 5 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2
0 0 0 0 2 0 0 0 0 0] [4 0 0 0 0 2 2 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [3 0 0 0 0
2 1 0 0 0 0] [6 3 0 0 0 2 1 0 0 0 0] [6 4 0 0 0 1 1 0 0 0 0] [6 3 0 0 0 2 1 0 0
0 0] [6 3 0 0 0 2 1 0 0 0 0] [7 4 0 0 0 2 1 0 0 0 0]]
    [[4 0 0 0 1 2 0 0 1 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1
0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [4 2 0 0 0 2 0 0 0 0 0] [2 1 0 0 0 1 0 0 0 0 0] [4
2 0 0 0 2 0 0 0 0 0] [6 2 0 0 0 2 2 0 0 0 0] [3 2 0 0 0 1 0 0 0 0 0] [7 2 1 0 0
2 2 0 0 0 0] [4 2 0 0 0 2 0 0 0 0 0] [6 2 1 0 0 1 2 0 0 0 0] [5 2 0 0 0 2 1 0 0
0 0] [6 2 1 0 0 2 1 0 0 0 0] [5 1 1 0 0 2 1 0 0 0 0]]
  ]

  set neue_schulen_districts [
    [[4 0 0 0 0 4 0 0 0 0 0] [4 0 0 0 0 4 0 0 0 0 0] [5 0 0 0 0 5 0 0 0 0 0] [5
0 0 0 0 5 0 0 0 0 0] [5 0 0 0 0 5 0 0 0 0 0] [5 0 0 0 0 5 0 0 0 0 0] [6 0 0 0 0
6 0 0 0 0 0] [7 1 0 0 0 6 0 0 0 0 0] [7 1 0 0 0 6 0 0 0 0 0] [7 1 0 0 0 6 0 0 0
0 0] [8 1 0 0 0 7 0 0 0 0 0] [8 1 0 0 0 7 0 0 0 0 0] [9 1 0 0 0 8 0 0 0 0 0]
[11 2 0 0 0 8 1 0 0 0 0] [14 2 0 0 0 9 3 0 0 0 0] [14 2 0 0 0 9 3 0 0 0 0] [17
2 0 0 0 10 5 0 0 0 0] [20 3 0 0 0 11 6 0 0 0 0] [21 3 0 0 0 11 7 0 0 0 0] [23 3
0 0 0 12 8 0 0 0 0] [24 3 0 0 0 12 9 0 0 0 0] [28 4 1 0 0 13 10 0 0 0 0]]
    [[0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0
0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [4
0 0 0 0 4 0 0 0 0 0] [4 0 0 0 0 4 0 0 0 0 0] [5 0 0 0 0 5 0 0 0 0 0] [6 0 0 0 0
6 0 0 0 0 0] [7 0 0 0 0 7 0 0 0 0 0] [7 0 0 0 0 7 0 0 0 0 0] [9 0 1 0 0 8 0 0 0
0 0] [10 0 1 0 0 9 0 0 0 0 0] [12 0 2 0 0 10 0 0 0 0 0]]
    [[4 0 0 0 0 4 0 0 0 0 0] [4 0 0 0 0 4 0 0 0 0 0] [5 0 0 0 0 5 0 0 0 0 0] [5
0 0 0 0 5 0 0 0 0 0] [6 0 0 0 0 6 0 0 0 0 0] [6 0 0 0 0 6 0 0 0 0 0] [7 0 0 0 0
7 0 0 0 0 0] [7 0 0 0 0 7 0 0 0 0 0] [8 0 0 0 0 8 0 0 0 0 0] [8 0 0 0 0 8 0 0 0
0 0] [9 0 0 0 0 9 0 0 0 0 0] [10 0 0 0 0 10 0 0 0 0 0]
[11 0 0 0 0 11 0 0 0 0 0] [12 0 0 0 0 12 0 0 0 0 0] [14 0 0 0 0 13 1 0 0 0 0]
[15 0 0 0 0 14 1 0 0 0 0] [19 0 0 0 0 15 2 0 0 2 0] [23 0 0 0 0 17 3 0 0 3 0]
[25 0 0 0 0 17 4 0 0 4 0] [29 0 0 0 0 19 5 0 0 5 0] [33 0 0 0 0 20 6 0 0 7 0]]
    [[3 0 0 0 0 0 0 0 3 0 0] [3 0 0 0 0 0 0 0 3 0 0] [3 0 0 0 0 0 0 0 3 0 0] [3
0 0 0 0 0 0 0 3 0 0] [3 0 0 0 0 0 0 0 3 0 0] [4 0 0 0 0 1 0 0 3 0 0] [4 0 0 0 0
1 0 0 3 0 0] [4 0 0 0 0 1 0 0 3 0 0] [4 0 0 0 0 1 0 0 3 0 0] [5 0 0 0 0 2 0 0 3
0 0] [6 1 0 0 0 2 0 0 3 0 0] [7 1 0 0 0 2 0 0 4 0 0] [9 2 0 0 0 3 0 0 4 0 0] [9
2 0 0 0 3 0 0 4 0 0] [11 3 0 0 0 4 0 0 4 0 0] [11 3 0 0 0 4 0 0 4 0 0] [14 4 1
0 0 5 0 0 4 0 0] [15 4 1 0 0 6 0 0 4 0 0] [16 5 1 0 0 6 0 0 4 0 0] [17 5 1 0 0
7 0 0 4 0 0] [18 6 1 0 0 7 0 0 4 0 0] [19 6 1 0 0 8 0 0 4 0 0]]
    [[0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0
0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0
1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0
0 0] [1 0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2
0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [3 0 0 0 0
3 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [4 0 0 0 0 4 0 0 0 0 0] [5 1 0 0 0 4 0 0 0
0 0] [5 1 0 0 0 4 0 0 0 0 0] [6 1 0 0 0 5 0 0 0 0 0]]
    [[2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2
0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0
0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0
0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [2 0 1 0 0 0 0 0 1 0 0] [3
0 2 0 0 0 0 0 1 0 0] [3 0 2 0 0 0 0 0 1 0 0] [4 0 2 0 0 1 0 0 1 0 0] [4 0 2 0 0
```

1 0 0 1 0 0] [4 0 2 0 0 1 0 0 1 0 0] [5 0 2 0 0 2 0 0 1 0 0] [5 0 2 0 0 2 0 0 1 0 0] [6 0 2 0 0 3 0 0 1 0 0] [6 0 2 0 0 3 0 0 1 0 0]]

[[1 0 0 0 0 0 0 0 0 0 1] [1 0 0 0 0 0 0 0 0 0 1] [2 0 0 0 0 0 0 0 0 0 2] [2 0 0 0 0 0 0 0 0 0 2] [2 0 0 0 0 0 0 0 0 0 2] [2 0 0 0 0 0 0 0 0 0 2] [2 0 0 0 0 0 0 0 0 0 2] [2 0 0 0 0 0 0 0 0 0 2] [3 0 0 0 0 0 0 0 0 0 3] [3 0 0 0 0 0 0 0 0 0 3] [3 0 0 0 0 0 0 0 0 0 3] [3 0 0 0 0 0 0 0 0 0 3] [4 0 0 0 0 1 0 0 0 0 3] [5 0 0 0 0 1 0 0 0 0 4] [6 0 0 0 0 2 0 0 0 0 4] [8 1 0 0 0 2 0 0 0 0 5] [8 1 0 0 0 2 0 0 0 0 5] [10 2 0 0 0 3 0 0 0 0 5] [11 2 0 0 0 3 0 0 0 0 6] [12 2 0 0 0 3 1 0 0 0 6] [14 3 0 0 0 4 1 0 0 0 6] [15 3 0 0 0 4 1 0 0 0 7]]

[[2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [4 0 0 0 0 3 0 0 0 0 1] [4 0 0 0 0 3 0 0 0 0 1] [4 0 0 0 0 3 0 0 0 0 1] [4 0 0 0 0 3 0 0 0 0 1] [4 0 0 0 0 3 0 0 0 0 1] [5 0 0 0 0 4 0 0 0 0 1] [6 0 0 0 0 4 0 0 0 0 2] [6 0 0 0 0 4 0 0 0 0 2] [8 0 0 0 0 5 0 0 0 0 3] [8 0 0 0 0 5 0 0 0 0 3] [11 0 0 0 0 6 1 0 0 0 4] [11 0 0 0 0 6 1 0 0 0 4] [12 0 0 0 0 6 1 0 0 0 5] [15 1 0 0 0 7 2 0 0 0 5] [16 1 0 0 0 7 2 0 0 0 6]]

[[1 0 0 0 0 1 0 0 0 0 0] [1 0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [4 0 0 0 0 4 0 0 0 0 0] [4 0 0 0 0 4 0 0 0 0 0] [5 0 0 0 0 4 0 0 0 1 0] [6 0 0 0 0 5 0 0 0 1 0] [8 0 0 0 0 5 1 0 0 2 0] [9 0 0 0 0 5 1 0 0 3 0] [11 0 0 0 0 6 1 0 0 4 0] [11 0 0 0 0 6 1 0 0 4 0] [13 0 0 0 0 6 2 0 0 5 0]]

[[11 0 7 0 2 0 0 0 0 0 2] [13 0 9 0 2 0 0 0 0 0 2] [15 0 10 0 2 0 0 0 0 0 3] [16 0 11 0 2 0 0 0 0 0 3] [18 0 13 0 2 0 0 0 0 0 3] [19 0 14 0 2 0 0 0 0 0 3] [22 0 16 0 2 0 0 0 0 0 4] [23 0 17 0 2 0 0 0 0 0 4] [24 0 18 0 2 0 0 0 0 0 4] [26 0 20 0 2 0 0 0 0 0 4] [29 1 21 0 2 0 0 0 0 0 5] [34 5 22 0 2 0 0 0 0 0 5] [38 8 23 0 2 0 0 0 0 0 5] [44 10 24 0 2 2 0 0 0 0 6] [50 13 25 0 2 3 0 0 0 1 6] [56 15 26 0 2 4 0 0 0 2 7] [63 17 27 0 2 6 0 0 0 4 7] [70 20 28 0 2 7 0 0 0 5 8] [78 22 29 0 3 9 1 0 0 6 8] [85 25 30 0 3 10 1 0 0 7 9] [92 27 31 0 3 11 2 0 0 9 9] [99 29 32 0 3 13 3 0 0 10 9]]

[[1 0 0 0 0 1 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [2 0 0 0 0 2 0 0 0 0 0] [3 0 0 0 0 3 0 0 0 0 0] [5 0 0 0 0 4 0 0 0 1 0] [5 0 0 0 0 4 0 0 0 1 0] [6 0 0 0 0 5 0 0 0 1 0] [7 0 0 0 0 6 0 0 0 1 0] [8 0 0 0 0 7 0 0 0 1 0] [8 0 0 0 0 7 0 0 0 1 0] [9 0 0 0 0 8 0 0 0 1 0] [10 0 0 0 0 9 0 0 0 1 0] [10 0 0 0 0 9 0 0 0 1 0] [11 0 0 0 0 10 0 0 0 1 0] [12 0 0 0 0 11 0 0 0 1 0] [16 2 0 0 0 12 1 0 0 1 0] [18 3 1 0 0 12 1 0 0 1 0] [22 5 2 0 0 13 1 0 0 1 0] [26 6 2 0 0 14 2 0 0 2 0] [29 7 3 0 0 15 2 0 0 2 0] [33 9 4 0 0 16 2 0 0 2 0] [35 10 4 0 0 16 3 0 0 2 0]]

[[10 0 0 0 0 0 0 0 3 7 0] [10 0 0 0 0 0 0 0 3 7 0] [10 0 0 0 0 0 0 0 3 7 0] [10 0 0 0 0 0 0 0 3 7 0] [11 0 0 0 0 1 0 0 3 7 0] [11 0 0 0 0 1 0 0 3 7 0] [13 0 0 0 0 2 0 0 3 8 0] [13 0 0 0 0 2 0 0 3 8 0] [14 0 0 0 0 2 1 0 3 8 0] [16 1 0 0 0 3 1 0 3 8 0] [18 3 0 0 0 3 1 0 3 8 0] [21 5 0 0 0 4 1 0 3 8 0] [22 5 1 0 0 4 1 0 3 8 0] [26 7 1 0 0 5 2 0 3 8 0] [29 8 2 0 0 5 2 0 3 9 0] [31 9 2 0 0 6 2 0 3 9 0] [32 10 2 0 0 6 2 0 3 9 0] [37 11 3 0 0 7 3 0 3 10 0] [38 12 3 0 0 7 3 0 3 10 0] [41 13 4 0 0 8 3 0 3 10 0] [42 14 4 0 0 8 3 0 3 10 0] [46 15 5 0 0 8 4 0 3 11 0]]

[[4 0 0 0 0 2 0 0 2 0 0] [4 0 0 0 0 2 0 0 2 0 0] [5 0 0 0 0 2 0 0 2 1 0] [7 0 0 0 0 3 0 0 2 2 0] [7 0 0 0 0 3 0 0 2 2 0] [8 0 0 0 0 3 0 0 2 3 0] [9 0 0 0 0 3 0 0 2 4 0] [9 0 0 0 0 3 0 0 2 4 0] [11 0 0 0 0 4 0 0 2 5 0] [12 0 0 0 0 4 0 0 2 6 0] [12 0 0 0 0 4 0 0 2 6 0] [13 0 0 0 0 4 0 0 2 7 0] [16 0 0 0 0 5 1 0 2 8 0] [18 0 0 0 0 5 1 0 2 10 0] [21 0 0 0 0 6 2 0 2 11 0] [22 0 0 0 0 6 2 0 2 12 0] [26 0 0 0 0 7 3 0 2 14 0] [28 0 0 0 0 8 3 0 2 15 0] [30 0 0 0 0 8 4 0 2 16 0] [32 0 0 0 0 9 4 0 2 17 0] [35 0 0 0 0 9 5 0 2 19 0] [37 0 0 0 0 10 5 0 2 20 0]]

[[7 0 0 0 0 1 0 0 1 5 0] [7 0 0 0 0 1 0 0 1 5 0] [7 0 0 0 0 1 0 0 1 5 0] [9 0 0 0 0 2 0 0 1 6 0] [9 0 0 0 0 2 0 0 1 6 0] [9 0 0 0 0 2 0 0 1 6 0] [10 0 0 0 0 3 0 0 1 6 0] [10 0 0 0 0 3 0 0 1 6 0] [10 0 0 0 0 3 0 0 1 6 0] [11 0 0 0 0 3 0 0 1 7 0] [12 0 0 0 0 4 0 0 1 7 0] [12 0 0 0 0 4 0 0 1 7 0] [14 0 0 0 0 5 0 0 1 8 0] [14 0 0 0 0 5 0 0 1 8 0] [16 0 0 0 0 6 0 0 1 9 0] [17 0 0 0 0 6 0 0 1 10 0] [19 0 0 0 0 7 0 0 1 11 0] [19 0 0 0 0 7 0 0 1 11 0] [21 0 0 0 0 8 0 0 1 12 0] [24 0 0 0 0 9 0 0 2 13 0] [26 0 0 0 0 9 1 0 2 14 0] [27 0 0 0 0 10 1 0 2 14 0]]

[[2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [2 0 0 0 2 0 0 0 0 0 0] [3 0 0 0 2 0 1 0 0 0 0] [3 0 0 0 2 0 1 0 0 0 0] [4 0 0 0 2 0 2 0 0 0 0] [5 0 0 0 2 0 3 0 0 0 0] [5 0 0 0 2 0 3 0 0 0 0] [7 0 0 0 2 1 4 0 0 0 0] [9 0 0 0 2 2 5 0 0 0 0] [10 0 0 0 2 3 5 0 0 0 0] [11 0 0 0 2 3 6 0 0 0 0] [14 0 1 0 2 4 7 0 0 0 0]]

[[1 0 1 0 0 0 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0 0] [1 0 1 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0 0 0 0 0] [2 0 2 0 0 0 0 0 0 0 0] [3 0 3 0 0 0 0 0 0 0 0] [3 0 3 0 0 0 0 0 0 0 0] [3 0 3 0 0 0 0 0 0 0 0] [3 0 3 0 0 0 0 0 0 0 0] [4 0 4 0 0 0 0 0 0 0 0] [5 0 4 0 0 1 0 0 0 0 0] [5 0 4 0 0 1 0 0 0 0 0] [7 0 5 0 0 2 0 0 0 0 0] [7 0 5 0 0 2 0 0 0 0 0] [8 1 5 0 0 2 0 0 0 0 0] [11 2 6 0 0 3 0 0 0 0 0] [12 2 6 0 0 3 0 0 1 0 0] [14 3 6 0 0 4 0 0 1 0 0] [16 4 7 0 0 4 0 0 1 0 0]]

[[4 0 0 0 0 1 0 0 3 0 0] [4 0 0 0 0 1 0 0 3 0 0] [4 0 0 0 0 1 0 0 3 0 0] [5 0 0 0 0 2 0 0 3 0 0] [5 0 0 0 0 2 0 0 3 0 0] [5 0 0 0 0 2 0 0 3 0 0] [6 0 0 0 0 2 0 0 3 1 0] [7 0 0 0 0 3 0 0 3 1 0] [7 0 0 0 0 3 0 0 3 1 0] [7 0 0 0 0 3 0 0 3 1 0] [8 0 0 0 0 4 0 0 3 1 0] [8 0 0 0 0 4 0 0 3 1 0] [8 0 0 0 0 4 0 0 3 1 0] [9 0 0 0 0 5 0 0 3 1 0] [10 0 0 0 0 5 0 0 3 2 0] [10 0 0 0 0 5 0 0 3 2 0] [11 0 0 0 0 6 0 0 3 2 0] [13 1 0 0 0 6 0 0 3 3 0] [15 1 0 0 0 7 0 0 4 3 0] [16 2 0 0 0 7 0 0 4 3 0] [16 2 0 0 0 7 0 0 4 3 0] [19 3 0 0 0 8 0 0 4 4 0]]

[[0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [0 0 0 0 0 0 0 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [1 0 0 0 0 0 1 0 0 0 0] [2 0 0 0 0 0 2 0 0 0 0] [2 0 0 0 0 0 2 0 0 0 0] [2 0 0 0 0 0 2 0 0 0 0] [3 0 0 0 0 0 3 0 0 0 0] [3 0 0 0 0 0 3 0 0 0 0] [5 0 0 0 0 1 4 0 0 0 0] [6 0 0 0 0 2 4 0 0 0 0] [7 0 0 0 0 2 5 0 0 0 0] [8 0 0 0 0 3 5 0 0 0 0] [9 0 0 0 0 3 6 0 0 0 0] [11 0 0 0 0 4 7 0 0 0 0] [12 0 0 0 0 5 7 0 0 0 0]]

[[3 0 0 0 0 0 0 0 0 2 1] [3 0 0 0 0 0 0 0 0 2 1] [4 0 0 0 0 0 0 0 0 3 1] [4 0 0 0 0 0 0 0 0 3 1] [4 0 0 0 0 0 0 0 0 3 1] [5 0 0 0 0 0 0 0 0 4 1] [5 0 0 0 0

```
0 0 0 0 4 1] [7 0 0 0 0 1 0 0 0 5 1] [7 0 0 0 0 1 0 0 0 5 1] [8 0 0 0 0 0 1 0 0 0
6 1] [10 0 0 0 0 0 2 0 0 0 6 2] [11 0 0 0 0 2 0 0 0 7 2] [14 0 0 0 0 4 0 0 0 8 2]
[15 0 0 0 0 4 0 0 0 9 2] [17 0 0 0 0 5 0 0 0 10 2] [21 0 0 0 0 6 0 0 0 12 3]
[23 0 0 0 0 7 0 0 0 13 3] [25 0 0 0 0 8 0 0 0 14 3] [28 0 0 0 0 9 0 0 0 16 3]
[31 0 0 0 0 10 0 0 0 17 4] [34 1 0 0 0 11 0 0 0 18 4] [36 1 0 0 0 12 0 0 0 19
4]]
    [[1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1
0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0
0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1
0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1
0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [1 0 0 0 0 0 0 0 1 0 0] [3 0 0 0 0
0 1 0 1 1 0] [4 0 0 0 0 0 1 0 1 2 0] [5 0 0 0 0 0 1 0 1 3 0] [11 2 1 0 0 0 2 0
1 5 0] [12 2 1 0 0 0 2 0 1 6 0] [15 3 1 0 0 0 3 0 1 7 0]]
    [[36 0 0 0 0 10 0 0 4 22 0] [36 0 0 0 0 10 0 0 4 22 0] [38 0 0 0 0 12 0 0 4
22 0] [40 0 0 0 0 13 0 0 4 23 0] [41 0 0 0 0 14 0 0 4 23 0] [42 0 0 0 0 15 0 0
4 23 0] [44 0 0 0 0 17 0 0 4 23 0] [44 0 0 0 0 17 0 0 4 23 0] [46 0 0 0 0 19 0
0 4 23 0] [48 0 0 0 0 20 0 0 4 24 0] [49 0 0 0 0 21 0 0 4 24 0] [51 0 0 0 0 22
0 0 4 24 1] [55 0 0 0 0 25 0 0 4 25 1] [59 0 0 0 0 28 0 0 4 25 2] [63 0 0 0 0
31 0 0 4 26 2] [68 0 0 0 0 34 0 0 4 27 3] [74 3 0 0 0 36 0 0 4 27 4] [80 5 0 0
0 39 0 0 4 28 4] [88 8 0 0 0 42 0 0 4 29 5] [95 10 0 0 0 45 2 0 4 29 5] [104 13
0 0 0 48 3 0 4 30 6] [113 16 0 0 0 50 5 0 4 31 7]]
    [[5 0 0 0 0 5 0 0 0 0 0] [6 0 0 0 0 6 0 0 0 0 0] [6 0 0 0 0 6 0 0 0 0 0] [7
0 0 0 0 7 0 0 0 0 0] [8 0 0 0 0 8 0 0 0 0 0] [8 0 0 0 0 8 0 0 0 0 0] [9 0 0 0 0
9 0 0 0 0 0] [10 0 0 0 0 10 0 0 0 0 0] [10 0 0 0 0 10 0 0 0 0 0] [11 0 0 0 0 11
0 0 0 0 0] [12 0 0 0 0 12 0 0 0 0 0] [12 0 0 0 0 12 0 0 0 0 0] [14 0 0 0 0 14 0
0 0 0 0] [16 0 0 0 0 16 0 0 0 0 0] [20 0 0 0 0 18 2 0 0 0 0] [22 0 0 0 0 20 2 0
0 0 0] [25 0 0 0 0 22 3 0 0 0 0] [31 3 0 0 0 24 4 0 0 0 0] [37 7 0 0 0 25 5 0 0
0 0] [43 10 0 0 0 27 6 0 0 0 0] [49 13 0 0 0 29 7 0 0 0 0] [56 17 0 0 0 31 8 0
0 0 0]]
    [[4 0 0 0 1 2 0 0 1 0 0] [5 0 0 0 1 3 0 0 1 0 0] [6 0 0 0 1 4 0 0 1 0 0] [7
0 0 0 1 5 0 0 1 0 0] [9 0 0 0 1 7 0 0 1 0 0] [10 0 0 0 1 8 0 0 1 0 0] [11 0 0 0
1 9 0 0 1 0 0] [13 0 0 0 1 11 0 0 1 0 0] [14 0 0 0 1 12 0 0 1 0 0] [15 0 0 0 1
13 0 0 1 0 0] [16 0 0 0 1 14 0 0 1 0 0] [20 2 0 0 1 16 0 0 1 0 0] [22 3 0 0 1
17 0 0 1 0 0] [26 5 0 0 1 19 0 0 1 0 0] [32 7 0 0 1 21 2 0 1 0 0] [35 9 0 0 1
22 2 0 1 0 0] [42 11 1 0 1 24 4 0 1 0 0] [46 13 1 0 1 26 4 0 1 0 0] [52 15 2 0
1 27 6 0 1 0 0] [57 17 2 0 1 29 7 0 1 0 0] [63 19 3 0 1 31 8 0 1 0 0] [68 20 4
0 1 33 9 0 1 0 0]]
  ]

  set schüler/klasse_districts [
    [[24.1 24.2 24.3 0 0 24.9 23.9 0 23.8 19.6 0] [24.2 24.3 24.3 0 0 25 23.9 0
23.6 19.6 0] [24.3 24.5 24.3 0 0 24.7 24 0 23.5 19.6 0] [24.3 24.6 24.1 0 0
24.8 24.1 0 23.3 19.6 0] [24.4 24.7 24.1 0 0 24.9 24.2 0 23.1 19.6 0] [24.5
24.9 24.1 0 0 25 24.2 0 22.9 19.6 0] [24.6 25 24.1 0 0 24.7 24.3 0 22.7 19.7 0]
[24.6 24.3 24.1 0 0 24.8 24.4 0 22.5 19.7 0] [24.7 24.5 24.1 0 0 24.9 24.4 0
22.4 19.7 0] [24.8 24.6 24 0 0 25 24.5 0 22.2 19.7 0] [24.8 24.7 24 0 0 24.7
24.6 0 22 19.7 0] [24.9 24.8 24 0 0 24.8 24.7 0 21.8 19.7 0] [24.9 25 24.1 0 0
24.6 25 0 21.6 19.8 0] [24.9 24.4 24.3 0 0 24.9 25 0 21.4 19.8 0] [25 24.6 24.4
0 0 24.7 24.7 0 21.2 19.9 0] [25 24.8 24.5 0 0 25 25 0 21 20 0] [25 25 24.6 0 0
24.8 24.8 0 20.8 20.1 0] [25 24.4 24.6 0 0 24.6 24.8 0 20.5 20.2 0] [25 24.6
24.8 0 0 24.9 24.8 0 20.4 20.3 0] [25 24.7 24.9 0 0 24.7 24.8 0 20.2 20.4 0]
[24.9 24.9 25 0 0 25 24.8 0 19.9 20.4 0] [24.9 24.4 22.3 0 0 24.8 24.9 0 19.7
20.5 0]]
    [[20 20.8 22.1 6.5 0 23.1 20.4 0 19.8 18.3 0] [20.2 21 22.2 6.5 0 23.4 20.6
0 19.7 18.3 0] [20.4 21.3 22.4 6.5 0 23.6 20.8 0 19.6 18.3 0] [20.5 21.5 22.6
6.6 0 23.8 20.9 0 19.5 18.3 0] [20.7 21.7 22.7 6.6 0 24.1 21 0 19.4 18.3 0]
[20.9 21.9 22.9 6.6 0 24.3 21.2 0 19.2 18.3 0] [21.1 22.1 23.1 6.6 0 24.5 21.4
0 19.2 18.3 0] [21.2 22.4 23.3 6.6 0 24.8 21.5 0 19 18.3 0] [21.4 22.6 23.4 6.7
0 25 21.7 0 18.9 18.3 0] [21.6 22.8 23.6 6.7 0 24.8 21.8 0 18.8 18.3 0] [21.7
23 23.8 6.7 0 25 22 0 18.7 18.3 0] [21.9 23.2 23.9 6.7 0 24.9 22.1 0 18.5 18.3
0] [22.1 23.4 24.1 6.8 0 24.8 22.4 0 18.4 18.4 0] [22.3 23.5 24.2 6.9 0 24.7
22.7 0 18.2 18.4 0] [22.4 23.7 24.4 6.9 0 25 23 0 18 18.5 0] [22.6 23.9 24.5 7
0 24.9 23.3 0 17.8 18.6 0] [22.8 24 24.7 7.1 0 24.8 23.6 0 17.6 18.8 0] [22.9
24.2 24.8 7.1 0 24.8 23.8 0 17.5 18.8 0] [23.1 24.4 25 7.2 0 25 24.1 0 17.2
18.9 0] [23.3 24.5 24.8 7.2 0 24.9 24.4 0 17.1 19 0] [23.5 24.7 25 7.3 0 24.9
24.7 0 16.8 19 0] [23.6 24.9 24.8 7.3 0 24.8 25 0 16.7 19.1 0]]
    [[22 21.5 22.8 6.9 22.9 24.8 23.3 0 22.1 22.6 0] [22.1 21.7 22.8 7 22.9 25
23.4 0 22.1 22.7 0] [22.2 21.8 22.8 7 22.9 24.8 23.4 0 22 22.7 0] [22.3 21.9
22.8 7.1 22.8 24.9 23.5 0 22 22.8 0] [22.3 22.1 22.8 7.1 22.8 24.8 23.5 0 22
22.9 0] [22.4 22.2 22.8 7.2 22.7 24.9 23.6 0 22 23 0] [22.5 22.3 22.8 7.2 22.7
25 23.7 0 22 23.1 0] [22.6 22.5 22.8 7.3 22.7 24.9 23.7 0 22 23.2 0] [22.7 22.6
22.8 7.3 22.7 25 23.8 0 22 23.3 0] [22.7 22.8 22.8 7.4 22.6 24.9 23.8 0 22 23.4
0] [22.8 22.9 22.7 7.4 22.6 25 23.9 0 22 23.5 0] [22.9 23 22.7 7.5 22.5 24.8 24
0 22 23.6 0] [23.1 23.2 22.8 7.5 22.6 24.9 24.2 0 22.1 23.9 0] [23.3 23.3 22.9
7.6 22.7 24.9 24.5 0 22.1 24.2 0] [23.5 23.5 23.1 7.7 22.7 24.9 24.8 0 22.2
24.4 0] [23.7 23.7 23.2 7.8 22.8 24.9 24.8 0 22.2 24.7 0] [23.9 23.8 23.3 7.9
22.9 25 25 0 22.3 25 0] [24.1 24 23.4 8 22.9 25 25 0 22.4 24.8 0] [24.3 24.1
23.5 8.1 22.9 24.8 25 0 22.4 24.9 0] [24.5 24.3 23.6 8.1 23 25 24.9 0 22.5 24.9
0] [24.7 24.4 23.7 8.2 23.1 24.8 24.9 0 22.6 25 0] [24.9 24.6 23.8 8.3 23.1
24.8 24.9 0 22.6 24.9 0]]
    [[23.3 23.1 23.9 0 15.7 24.4 21.2 0 24.5 0 0] [23.4 23.3 24 0 15.7 24.5
21.3 0 24.6 0 0] [23.6 23.5 24 0 15.6 24.7 21.3 0 24.6 0 0] [23.7 23.7 24 0
15.6 24.8 21.4 0 24.7 0 0] [23.8 23.9 24.1 0 15.6 25 21.4 0 24.8 0 0] [23.9
24.1 24.1 0 15.4 24.6 21.4 0 24.8 0 0] [24 24.3 24.2 0 15.4 24.7 21.5 0 24.8 0
0] [24.2 24.5 24.3 0 15.3 24.8 21.5 0 24.8 0 0] [24.3 24.7 24.3 0 15.3 25 21.6
0 24.9 0 0] [24.4 25 24.4 0 15.3 24.6 21.6 0 25 0 0] [24.5 24.8 24.4 0 15.1
24.7 21.6 0 25 0 0] [24.6 25 24.5 0 15.1 24.9 21.7 0 23.3 0 0] [24.9 24.8 24.6
0 15.1 24.7 22 0 23.4 0 0] [25 25 24.7 0 15.1 24.9 22.3 0 23.5 0 0] [24.9 24.7
24.9 0 15.3 24.7 22.5 0 23.6 0 0] [25 24.9 25 0 15.3 25 22.8 0 23.7 0 0] [25
24.7 24 0 15.3 24.8 23.1 0 23.9 0 0] [24.9 24.9 24.1 0 15.3 24.6 23.4 0 23.9 0
0] [24.9 24.7 24.2 0 15.3 24.9 23.7 0 24.1 0 0] [25 24.9 24.3 0 15.4 24.7 23.9
```

0 24.1 0 0] [24.9 24.7 24.4 0 15.4 25 24.2 0 24.3 0 0] [24.9 24.9 24.5 0 15.4 24.8 24.5 0 24.4 0 0]]

[[20.9 21.7 23.9 12.3 0 24.1 21 15.9 21.8 20.9 0] [21 21.9 23.9 12.3 0 24.3 21 15.8 21.7 20.9 0] [21 22.1 23.8 12.3 0 24.4 21.1 15.8 21.5 21 0] [21.1 22.3 23.8 12.3 0 24.6 21.2 15.8 21.4 21.1 0] [21.1 22.5 23.7 12.3 0 24.8 21.3 15.8 21.3 21.1 0] [21.2 22.7 23.7 12.3 0 24.9 21.4 15.8 21.1 21.2 0] [21.3 22.9 23.6 12.7 0 24.2 21.5 15.8 21 21.2 0] [21.3 23.1 23.6 12.7 0 24.4 21.6 15.8 20.9 21.3 0] [21.4 23.3 23.6 12.7 0 24.6 21.7 15.8 20.8 21.4 0] [21.5 23.4 23.5 12.7 0 24.7 21.7 15.7 20.7 21.4 0] [21.5 23.7 23.5 12.7 0 24.9 21.8 15.7 20.5 21.5 0] [21.6 23.8 23.4 12.7 0 24.2 21.9 15.7 20.4 21.5 0] [21.8 24 23.6 12.7 0 24.5 22.1 15.8 20.3 21.8 0] [21.9 24.2 23.7 12.7 0 24.7 22.4 15.9 20.3 22.1 0] [22.1 24.3 23.9 12.7 0 25 22.7 16 20.2 22.4 0] [22.3 24.5 24 12.7 0 24.4 23 16.1 20.1 22.7 0] [22.5 24.7 24.1 13 0 24.7 23.2 16.2 20.1 23 0] [22.7 24.8 24.2 13 0 24.9 23.4 16.3 20 23.3 0] [22.9 25 24.4 13 0 24.4 23.7 16.4 20 23.5 0] [23.1 24.6 24.5 13 0 24.6 24 16.5 19.9 23.8 0] [23.3 24.8 24.6 13 0 24.8 24.2 16.5 19.8 24.1 0] [23.4 25 24.7 13 0 24.3 24.5 16.6 19.7 24.4 0]]

[[21.9 21.6 24.9 12.2 0 23.2 21.4 21.6 18 0 0] [21.9 21.8 24.9 12.2 0 23.3 21.4 21.6 18 0 0] [21.9 21.9 24.9 12.3 0 23.4 21.4 21.5 18 0 0] [21.9 22 24.9 12.3 0 23.4 21.5 21.5 18 0 0] [21.9 22.1 24.9 12.5 0 23.5 21.5 21.5 18 0 0] [22 22.3 24.9 12.5 0 23.6 21.6 21.5 18 0 0] [22 22.4 24.9 12.7 0 23.7 21.6 21.5 18 0 0] [22 22.5 24.9 12.7 0 23.8 21.6 21.5 18 0 0] [22 22.7 24.9 12.8 0 23.8 21.7 21.5 18 0 0] [22 22.8 24.9 12.8 0 23.9 21.7 21.5 18 0 0] [22.1 22.9 24.9 13 0 24 21.8 21.5 18 0 0] [22.1 23.1 24.9 13 0 24.1 21.8 21.4 18 0 0] [22.2 23.2 25 13.2 0 24.3 22.1 21.6 18 0 0] [22.4 23.4 24 13.2 0 24.6 22.3 21.7 18 0 0] [22.6 23.6 24.1 13.3 0 24.8 22.6 21.9 18 0 0] [22.7 23.7 24.2 13.3 0 24.5 22.9 22 18 0 0] [22.9 23.9 24.3 13.5 0 24.8 23.1 22.1 18.3 0 0] [23.1 24.1 24.4 13.7 0 25 23.4 22.3 18.3 0 0] [23.2 24.2 24.5 13.7 0 24.7 23.7 22.4 18.3 0 0] [23.4 24.4 24.6 13.8 0 25 24 22.5 18.3 0 0] [23.6 24.6 24.8 13.8 0 24.7 24.2 22.7 18.3 0 0] [23.7 24.8 24.9 14 0 24.9 24.5 22.8 18.3 0 0]]

[[20 22.8 22.6 0 23 24.1 23.2 9.1 0 0 24.6] [20 22.9 22.6 0 23 24.1 23.2 9.1 0 0 24.8] [20.1 23.1 22.6 0 23 24.2 23.2 9.1 0 0 23.6] [20.2 23.2 22.5 0 23 24.3 23.3 9.1 0 0 23.9] [20.2 23.4 22.5 0 23 24.4 23.3 9.1 0 0 24.1] [20.3 23.5 22.5 0 23 24.5 23.3 9.1 0 0 24.4] [20.4 23.7 22.5 0 23 24.6 23.3 9.1 0 0 24.6] [20.5 23.8 22.4 0 23 24.7 23.3 9.1 0 0 24.9] [20.5 23.9 22.4 0 23 24.8 23.4 9.1 0 0 23.8] [20.6 24.1 22.4 0 23 24.8 23.4 9.1 0 0 24] [20.7 24.2 22.4 0 23 24.9 23.4 9.1 0 0 24.3] [20.7 24.4 22.3 0 23 25 23.4 9.1 0 0 24.5] [20.9 24.6 22.4 0 23 24.6 23.7 9.2 0 0 25] [21.1 24.7 22.5 0 23 24.9 23.9 9.2 0 0 24.2] [21.3 24.9 22.6 0 23 24.5 24.1 9.3 0 0 24.7] [21.5 24.7 22.7 0 23 24.7 24.3 9.3 0 0 24] [21.6 24.9 22.7 0 23 24.9 24.6 9.4 0 0 24.4] [21.8 24.6 22.8 0 23 24.6 24.8 9.4 0 0 24.9] [22 24.8 22.9 0 23 24.8 25 9.5 0 0 24.1] [22.2 25 23 0 23 25 24.5 9.5 0 0 24.6] [22.4 24.7 23.1 0 23 24.7 24.7 9.6 0 0 25] [22.5 24.9 23.1 0 23 24.9 24.9 9.6 0 0 24.3]]

[[23.3 21.7 23.4 0 0 24.5 23.6 0 24.5 21.6 24] [23.4 21.9 23.4 0 0 24.6 23.6 0 24.4 21.6 24.2] [23.5 22.1 23.4 0 0 24.7 23.7 0 24.2 21.6 24.3] [23.5 22.3 23.4 0 0 24.8 23.7 0 24 21.6 24.5] [23.6 22.4 23.4 0 0 24.9 23.7 0 23.8 21.6 24.6] [23.7 22.6 23.4 0 0 25 23.7 0 23.6 21.6 24.8] [23.7 22.8 23.3 0 0 25 23.7 0 23.5 21.6 25] [23.8 23 23.3 0 0 24.5 23.7 0 23.3 21.6 24] [23.9 23.1 23.3 0 0 24.6 23.7 0 23.1 21.6 24.1] [23.9 23.3 23.3 0 0 24.7 23.7 0 22.9 21.6 24.2] [24 23.5 23.3 0 0 24.8 23.7 0 22.7 21.6 24.4] [24.1 23.7 23.3 0 0 24.9 23.7 0 22.5 21.6 24.5] [24.3 23.8 23.3 0 0 24.6 24 0 22.3 21.7 25] [24.5 24 23.4 0 0 24.8 24.3 0 22 21.8 24.5] [24.6 24.1 23.4 0 0 25 24.5 0 21.7 21.9 25] [24.8 24.3 23.6 0 0 24.6 24.8 0 21.5 22.1 24.5] [25 24.5 23.6 0 0 24.9 25 0 21.2 22.2 25] [25 24.6 23.6 0 0 24.5 24.4 0 20.8 22.3 24.5] [24.9 24.8 23.7 0 0 24.7 24.6 0 20.5 22.4 25] [25 24.9 23.7 0 0 25 24.9 0 20.3 22.5 24.5] [24.9 24.5 23.9 0 0 24.6 24.2 0 20 22.6 24.9] [24.9 24.6 23.9 0 0 24.8 24.5 0 19.7 22.7 24.4]]

[[22.6 21.1 22.7 0 0 24.8 23 0 22.3 21.8 0] [22.7 21.2 22.7 0 0 25 23.1 0 22.2 21.9 0] [22.8 21.4 22.6 0 0 24.2 23.1 0 22.2 22 0] [22.9 21.5 22.6 0 0 24.3 23.2 0 22.2 22.1 0] [23 21.7 22.5 0 0 24.5 23.2 0 22.1 22.2 0] [23 21.8 22.5 0 0 24.6 23.3 0 22.1 22.4 0] [23.1 22 22.5 0 0 24.7 23.3 0 22.1 22.5 0] [23.2 22.1 22.5 0 0 24.8 23.4 0 22 22.6 0] [23.3 22.3 22.4 0 0 25 23.4 0 21.9 22.7 0] [23.4 22.5 22.4 0 0 24.3 23.5 0 21.9 22.8 0] [23.5 22.6 22.3 0 0 24.4 23.5 0 21.9 22.9 0] [23.6 22.8 22.3 0 0 24.5 23.6 0 21.8 23.1 0] [23.9 22.9 22.4 0 0 24.8 23.9 0 21.8 23.6 0] [24.1 23.1 22.5 0 0 24.3 24.1 0 21.8 24.1 0] [24.4 23.2 22.5 0 0 24.5 24.5 0 21.8 24.7 0] [24.7 23.4 22.6 0 0 24.8 24.8 0 21.8 24.4 0] [24.9 23.6 22.7 0 0 24.3 25 0 21.8 25 0] [24.8 23.7 22.8 0 0 24.5 24.2 0 21.8 24.8 0] [24.9 23.9 22.9 0 0 24.8 24.5 0 21.8 24.6 0] [24.9 24 23 0 0 24.3 24.8 0 21.8 24.4 0] [25 24.2 23.1 0 0 24.6 25 0 21.8 24.8 0] [24.9 24.4 23.2 0 0 24.8 24.2 0 21.8 24.7 0]]

[[22.1 22 25 13.9 24.3 21.2 20.7 16.1 20.5 22.4 24.3] [22.3 22.3 24.9 14.2 24.3 21.5 20.9 16.1 20.7 22.6 24.7] [22.6 22.6 24.9 14.3 24.3 21.8 21.1 16.1 20.8 22.7 23.8] [22.8 22.9 25 14.6 24.3 22.1 21.3 16.1 20.9 22.9 24.2] [23 23.2 24.9 14.8 24.3 22.5 21.5 16.2 21 23 24.5] [23.3 23.5 24.9 15 24.4 22.8 21.7 16.2 21.1 23.2 24.9] [23.5 23.8 24.8 15.3 24.4 23.1 21.9 16.2 21.2 23.4 24] [23.8 24.1 24.9 15.4 24.4 23.4 22.1 16.2 21.3 23.5 24.3] [24 24.4 25 15.7 24.4 23.7 22.3 16.2 21.4 23.7 24.7] [24.2 24.8 24.9 15.9 24.4 24 22.5 16.2 21.5 23.8 25] [24.5 25 24.9 16.1 24.5 24.3 22.7 16.2 21.6 24 24.2] [24.7 25 25 16.3 24.5 24.6 22.9 16.2 21.7 24.2 24.5] [25 24.9 25 16.5 24.6 24.9 23.3 16.3 22 24.5 25] [25 25 25 16.7 24.7 24.8 23.6 16.4 22.2 24.8 24.4] [25 24.9 25 16.9 24.8 24.9 23.9 16.5 22.5 24.9 24.9] [25 25 25 17.1 24.8 25 24.3 16.6 22.7 24.9 24.3] [25 25 25 17.3 24.9 24.8 24.6 16.7 23 24.8 24.8] [25 24.9 25 17.4 25 24.9 24.9 16.8 23.2 24.8 24.2] [25 25 25 17.6 23.6 24.8 24.7 16.9 23.5 24.9 24.7] [25 24.9 25 17.8 23.6 24.9 25 17 23.7 25 24.2] [25 25 25 18 23.8 25 24.8 17.1 24 24.8 24.6] [25 25 25 18.2 23.8 24.8 24.6 17.2 24.2 24.9 25]]

[[22.4 21.4 22.3 12 0 24.8 22.3 0 25 24.7 0] [22.7 21.7 22.5 12 0 24.7 22.5 0 25 24.8 0] [22.9 22 22.7 12 0 25 22.6 0 24.7 24.9 0] [23.1 22.3 22.9 12.5 0 24.9 22.7 0 24.7 25 0] [23.4 22.5 23 12.5 0 24.8 22.9 0 24.7 23.4 0] [23.6 22.8 23.2 12.5 0 25 23 0 24.3 23.5 0] [23.8 23.1 23.4 12.5 0 24.9 23.2 0 24.3 23.6 0] [24.1 23.3 23.6 12.5 0 24.8 23.3 0 24 23.7 0] [24.3 23.6 23.8 12.5 0 24.7 23.5 0 24 23.7 0] [24.5 23.9 24 13 0 25 23.6 0 24 23.8 0] [24.8 24.1 24.2 13 0 24.8 23.8 0 23.7 23.9 0] [25 24.4 24.4 13 0 24.7 23.9 0 23.7 24 0] [24.9 24.6 24.5 13.5 0 25 24.2 0 23.3 24.1 0] [25 24.8 24.7 13.5 0 24.9 24.5 0 23 24.3 0]

```
[25 25 24.9 14 0 24.8 24.8 0 23 24.5 0] [25 24.9 25 14 0 24.8 24.3 0 22.7 24.7 0] [25 25 25 14.5 0 25 24.6 0 22.3 24.8 0] [24.9 24.9 24.9 14.5 0 25 24.9 0 22 24.9 0] [25 24.9 25 15 0 24.9 24.4 0 21.7 23.6 0] [25 25 25 15 0 24.8 24.7 0 21.7 23.7 0] [25 24.9 24.9 15.5 0 24.7 24.9 0 21.3 23.9 0] [25 24.9 25 15.5 0 25 24.5 0 21 24 0]]
    [[23.1 22 22.8 13.3 0 23.8 23.2 22.1 23.6 24.9 0] [23.3 22.3 23 13.3 0 24.1 23.5 22.1 23.4 24.9 0] [23.4 22.7 23.2 13.3 0 24.5 23.7 22.1 23.2 24.9 0] [23.6 23 23.4 13.3 0 24.8 24 22.1 23 25 0] [23.7 23.3 23.6 13.3 0 24.4 24.2 22.1 22.8 25 0] [23.9 23.7 23.8 13.3 0 24.8 24.5 22.1 22.6 25 0] [24 24 24 13.3 0 24.4 24.7 22.1 22.4 24.7 0] [24.1 24.4 24.2 13.3 0 24.7 24.9 22.1 22.2 24.7 0] [24.3 24.7 24.3 13.3 0 25 23.8 22.1 22 24.8 0] [24.4 24.9 24.5 13.3 0 24.7 24.1 22.1 21.8 24.8 0] [24.6 24.9 24.7 13.3 0 25 24.3 22.1 21.6 24.8 0] [24.7 24.8 24.9 13.3 0 24.6 24.5 22.1 21.4 24.8 0] [24.9 25 24.7 13.3 0 24.9 24.8 22.2 21.1 24.9 0] [25 24.9 24.9 13.7 0 24.6 23.8 22.4 20.8 25 0] [25 24.9 24.7 13.7 0 24.9 24.2 22.5 20.5 24.8 0] [25 24.9 24.9 13.7 0 24.5 24.5 22.7 20.3 24.9 0] [25 24.9 25 14 0 24.8 24.8 22.8 19.9 25 0] [25 24.9 24.9 14 0 24.5 23.9 22.9 19.6 24.7 0] [25 24.9 25 14 0 24.8 24.2 23.1 19.4 24.8 0] [25 25 24.8 14 0 24.5 24.5 23.2 19.1 24.9 0] [25 25 25 14.3 0 24.7 24.8 23.4 18.8 25 0] [25 25 24.8 14.3 0 25 23.9 23.5 18.5 24.8 0]]
    [[22.6 21.2 21.7 6.3 0 24.8 23.9 21 24.5 24.7 0] [22.7 21.3 21.7 6.3 0 24.9 24 21 24.5 24.9 0] [22.8 21.4 21.7 6.3 0 25 24.1 21 24.3 24.8 0] [22.9 21.5 21.7 6.3 0 24.6 24.2 21.1 24.2 24.7 0] [23 21.6 21.7 6.3 0 24.7 24.3 21.1 24.1 25 0] [23.1 21.7 21.7 6.3 0 24.8 24.4 21.1 24 24.8 0] [23.2 21.8 21.7 6.4 0 24.9 24.5 21.1 23.8 24.8 0] [23.3 21.9 21.7 6.4 0 25 24.6 21.1 23.8 25 0] [23.4 22 21.7 6.4 0 24.6 24.7 21.2 23.6 24.9 0] [23.5 22.1 21.7 6.4 0 24.7 24.8 21.2 23.5 24.8 0] [23.6 22.2 21.7 6.4 0 24.8 24.9 21.2 23.4 25 0] [23.7 22.3 21.7 6.4 0 24.8 25 21.2 23.3 24.9 0] [23.9 22.5 21.8 6.4 0 24.6 24.7 21.4 23.1 25 0] [24.2 22.7 22 6.5 0 24.9 25 21.5 22.8 24.8 0] [24.4 22.9 22 6.5 0 24.7 24.7 21.6 22.7 24.9 0] [24.7 23.1 22.1 6.6 0 25 25 21.7 22.5 25 0] [24.9 23.3 22.3 6.6 0 24.8 24.7 21.9 22.2 24.7 0] [25 23.5 22.4 6.6 0 24.6 25 22 22 24.8 0] [24.9 23.7 22.5 6.7 0 24.9 24.7 22.1 21.8 24.9 0] [25 23.9 22.5 6.7 0 24.7 24.9 22.2 21.6 25 0] [25 24.1 22.7 6.8 0 24.9 24.6 22.4 21.4 24.8 0] [25 24.3 22.8 6.8 0 24.8 24.9 22.5 21.2 24.9 0]]
    [[22.1 21.4 21.6 7.8 0 24.6 21.7 23 23.6 24.8 0] [22.2 21.5 21.6 7.8 0 24.8 21.8 23 23.6 24.9 0] [22.3 21.7 21.7 7.9 0 25 21.9 22.9 23.6 25 0] [22.4 21.9 21.7 7.9 0 24.6 22 22.9 23.6 24.5 0] [22.5 22 21.7 7.9 0 24.8 22.2 22.9 23.6 24.6 0] [22.6 22.2 21.8 8 0 24.9 22.3 22.9 23.6 24.7 0] [22.7 22.4 21.8 8 0 24.6 22.3 22.8 23.7 24.8 0] [22.8 22.5 21.8 8 0 24.7 22.5 22.8 23.7 24.9 0] [22.9 22.7 21.9 8.1 0 24.9 22.6 22.8 23.7 25 0] [23 22.9 21.9 8.1 0 25 22.7 22.8 23.7 24.5 0] [23.1 23 21.9 8.1 0 24.7 22.8 22.8 23.7 24.6 0] [23.2 23.2 22 8.2 0 24.8 22.9 22.7 23.7 24.7 0] [23.4 23.4 22.1 8.3 0 24.6 23.2 22.9 23.9 24.6 0] [23.6 23.5 22.2 8.4 0 24.9 23.4 23 24.1 25 0] [23.8 23.7 22.3 8.5 0 24.7 23.7 23.1 24.3 24.8 0] [24 23.9 22.4 8.5 0 25 24 23.3 24.4 24.7 0] [24.2 24.1 22.5 8.6 0 24.7 24.2 23.4 24.7 24.6 0] [24.5 24.2 22.6 8.7 0 25 24.5 23.5 24.9 25 0] [24.7 24.4 22.8 8.8 0 24.8 24.8 23.7 25 24.8 0] [24.9 24.6 22.9 8.9 0 24.6 25 23.8 22 24.7 0] [25 24.8 23 9 0 24.9 24.4 23.9 22.3 24.6 0] [25 24.9 23.1 9 0 24.7 24.6 24.1 22.4 24.9 0]]
    [[20.5 20.8 24.2 7.5 23.7 22.1 24.5 18.2 24.9 20.4 0] [20.6 20.9 24.2 7.5 23.6 22.3 24.6 18.2 24.8 20.4 0] [20.6 21 24.2 7.5 23.5 22.4 24.6 18.2 24.7 20.4 0] [20.6 21.2 24.1 7.5 23.4 22.6 24.6 18.2 24.5 20.4 0] [20.7 21.3 24.1 7.5 23.3 22.7 24.7 18.2 24.4 20.4 0] [20.7 21.4 24.1 7.5 23.1 22.9 24.7 18.2 24.3 20.4 0] [20.7 21.5 24.1 7.5 23.1 23.1 24.7 18.2 24.1 20.4 0] [20.8 21.7 24 7.5 22.9 23.2 24.8 18.2 23.9 20.4 0] [20.8 21.8 24 7.5 22.8 23.4 24.8 18.2 23.8 20.4 0] [20.8 21.9 23.9 7.6 22.7 23.5 24.9 18.2 23.7 20.4 0] [20.9 22.1 23.9 7.6 22.6 23.7 24.9 18.2 23.5 20.4 0] [20.9 22.2 23.9 7.6 22.5 23.8 24.9 18.2 23.4 20.4 0] [21 22.3 24 7.6 22.6 24.1 24.8 18.3 23.3 20.4 0] [21.2 22.5 24.1 7.7 22.6 24.3 25 18.4 23.1 20.4 0] [21.3 22.6 24.3 7.8 22.6 24.6 24.9 18.5 23 20.4 0] [21.4 22.8 24.4 7.8 22.7 24.8 24.8 18.6 22.9 20.4 0] [21.6 22.9 24.5 7.9 22.8 25 25 18.7 22.7 20.4 0] [21.7 23 24.6 8 22.8 24.9 24.9 18.8 22.5 20.4 0] [21.8 23.2 24.8 8.1 22.9 24.8 24.8 18.9 22.4 20.4 0] [22 23.3 24.9 8.1 22.9 24.8 25 19 22.3 20.4 0] [22.1 23.5 25 8.2 22.9 25 24.9 19.1 22.1 20.4 0] [22.2 23.6 24.7 8.3 23 24.9 24.8 19.3 22 20.4 0]]
    [[21 21.5 24.8 12.3 0 22.7 20 16.6 22.7 20.8 0] [21.1 21.8 24.9 12.3 0 22.9 20.1 16.6 22.7 20.9 0] [21.2 22 25 12.4 0 23.1 20.1 16.6 22.7 21 0] [21.3 22.3 24.7 12.4 0 23.2 20.2 16.6 22.8 21 0] [21.4 22.5 24.8 12.5 0 23.4 20.3 16.6 22.9 21.1 0] [21.5 22.8 24.9 12.5 0 23.6 20.4 16.6 22.9 21.2 0] [21.7 23 24.9 12.6 0 23.8 20.4 16.5 22.9 21.3 0] [21.8 23.2 25 12.6 0 24 20.5 16.5 23 21.3 0] [21.9 23.5 24.7 12.7 0 24.2 20.6 16.5 23.1 21.4 0] [22 23.7 24.8 12.7 0 24.3 20.7 16.5 23.1 21.5 0] [22.1 24 24.9 12.8 0 24.5 20.7 16.5 23.1 21.5 0] [22.2 24.2 25 12.8 0 24.7 20.8 16.5 23.2 21.6 0] [22.4 24.4 24.8 12.9 0 24.9 21 16.6 23.5 21.9 0] [22.6 24.5 24.9 13.1 0 24.6 21.3 16.7 23.7 22.3 0] [22.8 24.7 25 13.2 0 24.8 21.5 16.8 23.9 22.6 0] [23 24.8 24.8 13.4 0 24.5 21.7 16.9 24.2 22.9 0] [23.2 25 24.9 13.5 0 24.8 21.9 16.9 24.5 23.2 0] [23.3 24.9 25 13.6 0 25 22.1 17 24.7 23.6 0] [23.5 24.8 24.8 13.8 0 24.7 22.4 17.1 25 23.9 0] [23.7 25 24.9 13.9 0 24.9 22.6 17.2 23.6 24.2 0] [23.9 24.9 25 14.1 0 24.6 22.8 17.3 23.9 24.5 0] [24.1 24.9 24.8 14.2 0 24.8 23 17.3 24.1 24.9 0]]
    [[21.5 21.4 20.8 8.6 0 24.5 20.6 0 24 24.8 0] [21.6 21.6 20.9 8.6 0 24.7 20.7 0 24 24.9 0] [21.7 21.9 21 8.7 0 24.9 20.7 0 24 24.9 0] [21.9 22.1 21.1 8.8 0 24.4 20.8 0 24 24.9 0] [22 22.4 21.2 8.8 0 24.6 20.9 0 24 25 0] [22.2 22.7 21.3 8.9 0 24.8 21 0 24 25 0] [22.3 22.9 21.4 8.9 0 25 21.1 0 24.1 24.4 0] [22.4 23.2 21.5 9 0 24.5 21.1 0 24.1 24.4 0] [22.6 23.4 21.6 9 0 24.7 21.2 0 24.1 24.5 0] [22.7 23.7 21.7 9.1 0 24.9 21.3 0 24.1 24.5 0] [22.9 23.9 21.7 9.2 0 24.4 21.4 0 24.1 24.5 0] [23 24.2 21.8 9.2 0 24.6 21.4 0 24.1 24.6 0] [23.2 24.4 21.9 9.3 0 24.9 21.7 0 24.3 24.8 0] [23.4 24.5 22 9.4 0 24.5 22 0 24.4 25 0] [23.6 24.7 22.1 9.5 0 24.7 22.2 0 24.5 24.5 0] [23.7 24.8 22.2 9.6 0 25 22.5 0 24.7 24.7 0] [23.9 25 22.3 9.7 0 24.6 22.7 0 24.8 24.9 0] [24.1 24.8 22.4 9.8 0 24.9 23 0 25 24.4 0] [24.3 25 22.5 9.8 0 24.5 23.3 0 24.1 24.6 0] [24.5 24.8 22.6 10 0 24.7 23.5 0 24.2 24.7 0] [24.7 25 22.7 10 0 24.9 23.8 0 24.4 24.9 0] [24.8 24.8 22.8 10.1 0 24.6 24 0 24.5 24.4 0]]
    [[20 20.3 22.1 8.2 22.9 22.4 24.4 21 0 0 0] [20.2 20.5 22.1 8.2 22.8 22.6 24.5 21 0 0 0] [20.3 20.7 22.2 8.3 22.8 22.8 24.7 21 0 0 0] [20.4 20.8 22.3 8.3
```

```
22.7 22.9 24.8 21 0 0 0] [20.5 21 22.3 8.3 22.6 23.1 24.9 21 0 0 0] [20.6 21.2
22.4 8.3 22.6 23.3 25 21 0 0 0] [20.8 21.4 22.4 8.4 22.4 23.4 24.6 20.7 0 0 0]
[20.9 21.5 22.5 8.4 22.4 23.6 24.7 20.7 0 0 0] [21 21.7 22.5 8.4 22.3 23.8 24.8
20.7 0 0 0] [21.1 21.9 22.6 8.5 22.3 24 24.9 20.7 0 0 0] [21.2 22.1 22.6 8.5
22.2 24.1 24.5 20.7 0 0 0] [21.3 22.2 22.7 8.5 22.1 24.3 24.6 20.7 0 0 0] [21.5
22.4 22.8 8.7 22.2 24.5 24.9 20.7 0 0 0] [21.7 22.6 22.9 8.8 22.2 24.8 24.7
20.7 0 0 0] [21.9 22.7 22.9 8.9 22.3 25 25 20.7 0 0 0] [22 22.9 23 9 22.3 24.9
24.7 20.7 0 0 0] [22.2 23.1 23.1 9.1 22.3 24.7 25 20.7 0 0 0] [22.4 23.2 23.2
9.2 22.3 25 24.8 20.3 0 0 0] [22.6 23.4 23.3 9.3 22.4 24.8 25 20.3 0 0 0] [22.7
23.6 23.4 9.4 22.4 25 24.8 20.3 0 0 0] [22.9 23.7 23.5 9.5 22.4 24.9 24.6 20.3
0 0 0] [23.1 23.9 23.6 9.7 22.4 24.7 24.8 20.3 0 0 0]]
    [[21.9 22.1 21.3 6.4 24 24.1 21 0 18.7 24.7 23.1] [22 22.2 21.3 6.4 24 24.3
21.1 0 18.5 24.9 23.3] [22.1 22.3 21.2 6.4 24 24.4 21.2 0 18.3 24.6 23.5] [22.3
22.5 21.2 6.4 24 24.5 21.3 0 18.1 24.8 23.7] [22.4 22.6 21.2 6.4 24 24.7 21.4 0
17.9 25 23.9] [22.5 22.7 21.2 6.4 24 24.8 21.5 0 17.7 24.8 24.1] [22.6 22.8
21.2 6.4 24 24.9 21.6 0 17.6 25 24.3] [22.7 23 21.2 6.4 24 24.8 21.7 0 17.4
24.7 24.5] [22.8 23.1 21.2 6.4 24 24.9 21.8 0 17.2 24.9 24.8] [22.9 23.2 21.2
6.4 24 25 21.9 0 17 24.7 24.9] [23 23.3 21.1 6.4 24 24.9 22.1 0 16.8 24.9 23.2]
[23.2 23.5 21.1 6.4 24 25 22.2 0 16.6 24.6 23.4] [23.4 23.6 21.2 6.4 24 24.7
22.4 0 16.3 24.8 23.8] [23.7 23.8 21.3 6.5 24 25 22.7 0 16 24.9 24.4] [23.9 24
21.4 6.6 24 25 23 0 15.7 25 24.8] [24.2 24.2 21.4 6.7 24 25 23.2 0 15.4 24.7
23.6] [24.5 24.4 21.5 6.7 24 24.9 23.5 0 15.2 24.8 24] [24.7 24.5 21.6 6.8 24
24.9 23.8 0 14.9 24.9 24.4] [25 24.7 21.7 6.9 24 24.9 24 0 14.6 24.7 24.9] [25
24.9 21.8 6.9 24 24.9 24.3 0 14.3 24.8 23.7] [25 24.8 21.8 7 24 24.9 24.5 0 14
24.9 24.1] [25 25 21.9 7.1 24 24.9 24.8 0 13.7 25 24.5]]
    [[21.9 21.5 23.3 11.3 22.6 20.1 22.4 0 20.8 23 0] [22.1 21.7 23.4 11.3 22.6
20.3 22.5 0 20.8 23 0] [22.2 21.9 23.5 11.3 22.6 20.4 22.6 0 20.8 23.1 0] [22.3
22.2 23.6 11.3 22.6 20.6 22.7 0 21 23.2 0] [22.5 22.4 23.6 11.4 22.6 20.7 22.8
0 21 23.3 0] [22.6 22.6 23.7 11.4 22.6 20.9 22.9 0 21 23.3 0] [22.7 22.8 23.8
11.5 22.7 21 23.1 0 21 23.4 0] [22.9 23.1 23.9 11.5 22.7 21.2 23.2 0 21 23.5 0]
[23 23.3 24 11.6 22.7 21.3 23.3 0 21 23.6 0] [23.1 23.5 24.1 11.6 22.7 21.5
23.4 0 21.3 23.6 0] [23.3 23.7 24.1 11.7 22.7 21.6 23.5 0 21.3 23.7 0] [23.4
23.9 24.2 11.7 22.7 21.8 23.6 0 21.3 23.8 0] [23.6 24.1 24.3 11.8 22.8 22 23.9
0 21.3 24.1 0] [23.8 24.3 24.4 11.9 22.8 22.2 24.2 0 21.5 24.3 0] [24 24.4 24.6
12.1 22.8 22.4 24.5 0 21.5 24.6 0] [24.2 24.6 24.7 12.2 22.8 22.7 24.8 0 21.8
24.9 0] [24.4 24.7 24.8 12.3 22.9 22.9 24.4 0 21.8 24.9 0] [24.6 24.9 24.9 12.4
22.9 23.1 24.7 0 21.8 25 0] [24.8 25 25 12.6 23 23.3 25 0 22 25 0] [25 24.8
24.7 12.7 23 23.6 24.5 0 22 24.8 0] [24.9 25 24.8 12.8 23.1 23.8 24.8 0 22.3
24.8 0] [24.9 25 24.9 12.9 23.1 24 24.4 0 22.3 24.9 0]]
    [[22.5 21.7 23.1 6.6 22.6 24.9 22 17.9 24.5 24.9 22.9] [22.6 21.9 23.1 6.6
22.6 25 22 17.9 24.3 25 23.1] [22.7 22.1 23.1 6.6 22.4 24.9 22.1 17.9 24 25
23.3] [22.8 22.3 23.1 6.7 22.4 25 22.2 17.9 23.8 24.6 23.5] [22.9 22.5 23.1 6.7
22.3 25 22.2 17.9 23.6 24.7 23.8] [23.1 22.7 23.1 6.7 22.2 25 22.3 17.9 23.4
24.8 24] [23.2 22.9 23.1 6.8 22.1 24.9 22.4 17.9 23.1 24.8 24.2] [23.3 23.2
23.1 6.8 22 25 22.5 17.9 22.9 24.9 24.4] [23.4 23.4 23.1 6.9 21.9 24.9 22.5
17.9 22.6 25 24.6] [23.5 23.6 23.1 6.9 21.9 25 22.6 17.9 22.4 24.6 24.8] [23.6
23.8 23.1 6.9 21.7 25 22.7 17.9 22.1 24.6 25] [23.7 24 23.1 7 21.6 25 22.7 17.9
21.9 24.7 24.5] [24 24.3 23.3 7.1 21.7 25 23.1 18.1 21.8 24.6 24.9] [24.3 24.5
23.5 7.2 21.8 24.9 23.4 18.2 21.6 24.9 24.6] [24.6 24.7 23.7 7.3 21.9 24.9 23.7
18.3 21.5 24.7 25] [24.8 25 23.9 7.4 22 24.9 24 18.4 21.4 24.6 24.7] [25 24.9
24.1 7.5 22.1 25 24.4 18.6 21.2 24.9 24.4] [25 25 24.3 7.6 22.1 25 24.7 18.7
21.1 24.8 24.9] [25 24.9 24.5 7.8 22.2 24.9 25 18.8 20.9 24.6 24.6] [25 25 24.6
7.9 22.4 24.9 24.8 19 20.8 24.9 25] [25 25 24.8 8 22.4 24.9 24.9 19.1 20.6 24.8
24.7] [25 24.9 25 8.1 22.5 25 24.8 19.2 20.5 24.6 24.4]]
    [[20.7 21.4 20.6 9.8 22.6 24.9 23 17.8 21.8 15.9 0] [20.8 21.6 20.6 9.8
22.5 24.9 23.2 17.8 21.7 15.9 0] [20.9 21.8 20.6 9.9 22.5 25 23.3 17.7 21.6
15.9 0] [21 22 20.6 10 22.4 24.9 23.4 17.7 21.5 16 0] [21.1 22.2 20.6 10 22.4
24.9 23.5 17.7 21.4 16 0] [21.2 22.4 20.6 10 22.4 25 23.6 17.6 21.3 16.1 0]
[21.3 22.6 20.6 10.1 22.3 24.9 23.7 17.6 21.2 16.1 0] [21.4 22.8 20.6 10.1 22.3
24.9 23.9 17.6 21 16.1 0] [21.5 23 20.6 10.2 22.3 25 24 17.6 20.9 16.2 0] [21.6
23.2 20.6 10.3 22.2 24.9 24.1 17.5 20.8 16.2 0] [21.7 23.4 20.6 10.3 22.2 24.9
24.2 17.5 20.7 16.2 0] [21.8 23.6 20.6 10.3 22.1 25 24.3 17.5 20.6 16.3 0]
[22.1 23.9 20.8 10.5 22.2 25 24.7 17.6 20.6 16.5 0] [22.3 24.2 21 10.7 22.3
24.9 25 17.6 20.7 16.6 0] [22.5 24.4 21.2 10.9 22.4 24.9 24.7 17.7 20.7 16.8 0]
[22.8 24.7 21.4 11 22.5 24.9 25 17.8 20.8 17 0] [23 24.9 21.6 11.2 22.6 24.9 25
17.9 20.8 17.2 0] [23.3 25 21.8 11.4 22.6 24.8 25 18 20.8 17.3 0] [23.5 24.9 22
11.6 22.8 25 24.9 18.1 20.9 17.5 0] [23.7 25 22.2 11.7 22.8 25 24.9 18.1 20.9
17.7 0] [24 25 22.4 11.9 22.9 25 24.9 18.2 21 17.9 0] [24.2 25 22.6 12.1 23
24.9 24.9 18.3 21 18.1 0]]
    [[22.2 22.4 22.7 7.9 23.5 24.8 22.1 0 22 18.8 0] [22.4 22.7 22.8 8 23.5
24.9 22.3 0 21.8 19 0] [22.6 22.9 22.9 8.1 23.6 24.9 22.5 0 21.6 19 0] [22.9
23.2 23.1 8.1 23.7 25 22.7 0 21.4 19.2 0] [23.1 23.4 23.2 8.2 23.8 24.8 22.9 0
21.2 19.2 0] [23.3 23.7 23.3 8.3 23.8 24.9 23.1 0 21 19.4 0] [23.5 23.9 23.5
8.4 23.9 24.9 23.3 0 20.6 19.4 0] [23.7 24.2 23.6 8.6 24 24.8 23.6 0 20.4 19.6
0] [24 24.4 23.8 8.7 24.1 24.8 23.8 0 20.2 19.6 0] [24.2 24.7 23.9 8.7 24.2
24.9 24 0 20 19.8 0] [24.4 24.9 24 8.8 24.2 25 24.2 0 19.8 19.8 0] [24.6 24.9
24.2 8.9 24.3 24.8 24.4 0 19.6 20 0] [24.9 25 24.3 9 24.4 24.9 24.7 0 19.2 20.4
0] [25 25 24.5 9.1 24.5 24.9 25 0 19 21 0] [25 25 24.7 9.3 24.5 24.8 24.7 0
18.6 21.4 0] [25 24.9 24.9 9.4 24.6 25 25 0 18.4 22 0] [25 24.9 24.8 9.5 24.7
24.9 24.8 0 18 22.4 0] [25 24.9 25 9.6 24.7 24.8 25 0 17.6 22.8 0] [25 24.9
24.9 9.7 24.8 25 24.8 0 17.4 23.4 0] [25 24.9 25 9.8 24.8 24.9 24.8 0 17 23.8
0] [25 24.9 24.9 9.9 24.9 24.9 24.8 0 16.8 24.4 0] [25 25 24.8 10 25 24.8 24.8
0 16.4 24.8 0]]
  ]

  set colors_schultypen (list
    [0 0 0]
    [215 50 41 200]
    [241 106 21 200]
    [157 110 72 200]
    [237 237 49 200]
    [89 176 60 200]
```

```
    [44 209 59 200]
    [29 159 120 200]
    [84 196 196 200]
    [45 141 190 200]
    [52 93 169 200]
    )

  set centers_districts [
    [-18.382608695652173 -48.36521739130435]
    [99.53236976814213 -53.79110207768744]
    [39.93133047210301 -98.69098712446352]
    [-15.506849315068493 -102.28767123287672]
    [-49.83545770567787 -118.0034762456547]
    [-72 -90.7560975609756]
    [-77.04123711340206 -59.79381443298969]
    [-75.10576923076923 -32.40384615384615]
    [-51.869565217391305 10.652173913043478]
    [21.094606785644245 -238.38023141792507]
    [154.14276264357449 -195.72445727343796]
    [-133.06447638603697 -176.66755646817248]
    [-315.97816446054367 -153.13551575882732]
    [-322.5536507936508 -2.6008216619981326]
    [-128.78571428571428 -84.10204081632654]
    [-191.27747070528412 -13.322794605350431]
    [-206.41021719180176 49.096359743040686]
    [-143.77333333333334 56.68]
    [-107.82026631021542 133.9968639144517]
    [-2.220601237842617 67.5893015030946]
    [74.76490398978788 210.94606045791576]
    [266.6092517360291 0.7923850914072464]
    [-203.39740847387907 -270.9986425339367]
  ]

  set schulbesuche_wien []
  set neue_schulen []
  repeat (length bezirke_wien - 1)[
    set neue_schulen lput 0 neue_schulen
  ]
end

to setup-statistics
  ;setup-prognose
end

to setup-patches
  import-pcolors-rgb "Wien_Bezirke_sw2.jpg"
end

to setup-districts
  let i 0
  while [i < (length bezirke_wien - 1)][
    create-districts 1 [
      set color [0 0 0 50]
      set size 25
      set shape "circle"
      set zahl (i + 1)
      set label (word zahl ".")
      set label-color black
      setxy (item 0 item i centers_districts) (item 1 item i centers_districts)
      set schüler/klasse_district_list item i schüler/klasse_districts
      set neue_schule_list item i neue_schulen_districts
      set neue_klassen_list item i neue_klassen_districts
    ]
    set i (i + 1)
  ]
end

to setup-cursor
  create-cursors 5 [
    setxy 0 0
    set color black
    set size 15
    set shape "x"
    if who > 0 [
      set color [255 255 255 0]
    ]
    set label-color black
    set hidden? true
  ]
end

to go
  ask schools [die]
  ifelse Schultyp = "Alle" [
    update-districts_alle_schultypen
    color-coding_alle
    build-schools_alle
  ][
  update-districts
  color-coding
  build-schools
  ]
```

```
  update-neue_schulen_insgesamt
  ifelse slider? = false [
    if ticks != 0 [
      set jahre (jahre + 1)
    ]
    if jahre = 2030 [
      ask districts [
        set color [0 0 0 50]
      ]
      export-csv
      stop
    ]
  ][
  set jahre Welches_Jahr
  ]
  tick
end

to update-districts_alle_schultypen
  ; schüler/klasse für bezirke berechnen
  ifelse slider? = false [
    ask districts [
      set neue_klassen item ticks neue_klassen_list
      set schüler/klasse_district item ticks schüler/klasse_district_list
      set neue_schule item ticks neue_schule_list

      ; neue_schulen erhöhen
      set neue_schulen replace-item (zahl - 1) neue_schulen neue_schule
    ]
  ][
  ask districts [
    set neue_klassen item (jahre - 2009) neue_klassen_list
    set       schüler/klasse_district       item       (jahre       -       2009)
schüler/klasse_district_list
    set neue_schule item (jahre - 2009) neue_schule_list

    ; neue_schulen erhöhen
    set neue_schulen replace-item (zahl - 1) neue_schulen neue_schule
  ]
  ]
end

to color-coding_alle
  ; normalize anhand anzahl neue_klassen
  let districts_sort sort-on [sum neue_klassen] districts
  let factor 255 / [sum neue_klassen] of last districts_sort
  ; scale-color anhand anzahl neue_klassen
  ask districts [
    ifelse sum neue_klassen > 0 [
      ; color mit transparenz
      set color (list 255 (255 - (sum neue_klassen * factor)) 0 255)
    ][
    set color [0 0 0 50]
    ]
  ]
end

to build-schools_alle
  let i 1
  let j 0
  let districts_list sort-on [zahl] districts
  while [j < length districts_list][
    while [i < length [neue_schule] of item j districts_list] [
      create-schools 1 [
        setxy  ([xcor]  of  item  j  districts_list)  ([ycor]  of  item  j
districts_list)
        set color item i colors_schultypen
        set label-color black
        set klassen ([item i neue_schule] of item j districts_list)
        set size (klassen * size_schools)
        set shape "pentagon"
        set mein_bezirk [zahl] of item j districts_list
        set mein_schultyp item i schultypen_wien_prognose
        set heading 0
        right i * 360 / 10
        fd 30
      ]
    set i (i + 1)
    ]
    set i 1
    set j (j + 1)
  ]
end

to update-neue_schulen_insgesamt
  ifelse slider? = false [
    set neue_schulen_insgesamt item ticks neue_schulen_insgesamt_list
  ][
  set neue_schulen_insgesamt item (jahre - 2009) neue_schulen_insgesamt_list
  ]
end
```

```
to update-districts
  ; gefragter schultyp?
  let i 0
  set schultyp_nummer 0
  while [i < length schultypen_wien_prognose] [
    if Schultyp = item i schultypen_wien_prognose [
      set schultyp_nummer i
    ]
    set i (i + 1)
  ]

  ; schüler/klasse für bezirke berechnen
  ifelse slider? = false [
    ask districts [
      set neue_klassen item schultyp_nummer item ticks neue_klassen_list
      set    schüler/klasse_district    item    schultyp_nummer    item    ticks
schüler/klasse_district_list
      set neue_schule item schultyp_nummer item ticks neue_schule_list

      ; neue_schulen erhöhen
      set neue_schulen replace-item (zahl - 1) neue_schulen neue_schule
    ]
  ][
  ask districts [
    set neue_klassen item schultyp_nummer item (jahre - 2009) neue_klassen_list
    set  schüler/klasse_district  item  schultyp_nummer  item  (jahre  -  2009)
schüler/klasse_district_list
    set neue_schule item schultyp_nummer item (jahre - 2009) neue_schule_list

    ; neue_schulen erhöhen
    set neue_schulen replace-item (zahl - 1) neue_schulen neue_schule
  ]
  ]
end

to color-coding
  if sum [neue_klassen] of districts != 0 [
    ; normalize anhand anzahl neue_klassen
    let districts_sort sort-on [neue_klassen] districts
    let factor 255 / [neue_klassen] of last districts_sort

    ; scale-color anhand anzahl neue_klassen
    ask districts [
      ifelse neue_klassen > 0 [
        ; color mit transparenz
        set color (list 255 (255 - (neue_klassen * factor)) 0 255)
      ][
      set color [0 0 0 50]
      ]
    ]
  ]
end

to build-schools
  let i 1
  let j 0
  let districts_list sort-on [zahl] districts
  while [j < length districts_list][
    create-schools 1 [
      setxy ([xcor] of item j districts_list) ([ycor] of item j districts_list)
      set color item schultyp_nummer colors_schultypen
      set label-color black
      set klassen ([neue_schule] of item j districts_list)
      set size (klassen * size_schools)
      set shape "pentagon"
      set mein_bezirk [zahl] of item j districts_list
      set mein_schultyp item i schultypen_wien_prognose
      set heading 0
      right i * 360 / 10
      fd 30
    ]
    set j (j + 1)
  ]
end

to statistics-from-mouse
  ask schools with [klassen = 0] [
    die
  ]
  ask districts [
    set color [0 0 0 50]
  ]

  let j 0
  let cursors_list sort cursors
  if mouse-inside? = true [
    while [j < count cursors] [
      ask item j cursors_list [
        setxy (mouse-xcor) (mouse-ycor + (j * 30))
        ;set hidden? false
      ]
```

```
      set j (j + 1)
    ]

    ; auf welchem bezirk befindet sich der cursor?
    let this_district ([zahl] of min-one-of districts [distance item 0
cursors_list])

    ; label von turtles
    set j 0
    let label_list reverse (list
      (sentence (word this_district ".") "Bezirk")
      (sentence "Neue Klassen nach Schultypen" ([neue_schule] of districts with
[zahl = this_district]))
      (sentence "Schüler/Klasse" ([schüler/klasse_district] of districts with
[zahl = this_district]))
      ""
      )
    while [j < count cursors - 1] [
      ask item j cursors_list [
        set label item j label_list
      ]
      set j (j + 1)
    ]

    ; schulen des bezirks größer anzeigen
    let i 0
    ask schools with [mein_bezirk = this_district] [
      set hidden? false
      if size_alt = 0 [

        back 30
        set heading i * 360 / (count schools with [mein_bezirk =
this_district])

        fd 80
        set size_alt size
        set size (size_alt * 2.5)
        set label (sentence mein_schultyp klassen "Klassen")
      ]
      set i (i + 1)
    ]

    ; andere schulen zurücksetzen
    ask schools with [mein_bezirk != this_district] [
      set hidden? true
      if size_alt != 0 [
        back 50
        set size (size_alt)
        set size_alt 0
        set label ""
      ]
    ]
  ]

  ;reset districts
  if mouse-down? = true [
    ask schools [
      set hidden? false
      if size_alt != 0 [
        back 50
        set size (size_alt)
        set size_alt 0
        set label ""
      ]
    ]
    stop
  ]

  display
end

to export-csv
  file-close-all
  setup-prognose

  export-schülerzahlen
  export-neue_schulen
  export-neue_klassen
  export-schüler/klasse
  export-schüler/klasse_exkl
  export-neue_schulen_jahre
  export-schüler/klasse_jahre
  export-schüler/klasse_jahre_exkl
end

to export-schülerzahlen
  let filename "schüler_2009-30.csv"
  ;let headers ["" 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020
2021 2022 2023 2024 2025 2026 2027 2028 2029 2030]

  if file-exists? filename [file-delete filename]
  file-open filename
```

```
  show filename

  ;;output insgesamt
  print-row (sentence "Wien" schultypen_wien_prognose)
  let i 0
  while [i < length item 0 neue_schulen_districts] [
    print-row (sentence (2009 + i) item 0 item i schulbesuche_wien)
  set i (i + 1)
  ]

  file-print ""


  ; klassenzahlen

  ;;output insgesamt
  print-row (sentence "Wien" schultypen_wien_prognose)
  set i 0
  while [i < length item 0 neue_schulen_districts] [
    print-row (sentence (2009 + i) (map + first klassen_wien_2009 item i
neue_schulen_insgesamt_list))
  set i (i + 1)
  ]

  file-close
end

to export-neue_schulen
  let filename "Neue_Schulen_Prognose.csv"
  ;let headers ["" 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020
2021 2022 2023 2024 2025 2026 2027 2028 2029 2030]

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename

  ;;output insgesamt
  print-row (sentence "Wien" schultypen_wien_prognose)
  let i 0
  while [i < length item 0 neue_schulen_districts] [
    print-row (sentence (2009 + i) item i neue_schulen_insgesamt_list)
  set i (i + 1)
  ]

  file-print ""

  ;; output bezirke
  let j 0
  while [j < length neue_schulen_districts] [
    ;;output bezirk & schultypen
    print-row (sentence item (j + 1) bezirke_wien schultypen_wien_prognose)

  ;; output jahre & neue-schulen
  set i 0
  while [i < length item 0 neue_schulen_districts] [
    print-row (sentence (2009 + i) item i item j neue_schulen_districts)
      set i (i + 1)
  ]
  file-print ""
  set i 0
  set j (j + 1)
  ]

  file-close
end

to export-neue_klassen
  let filename "Neue_Klassen_Prognose.csv"
  ;let headers ["" 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020
2021 2022 2023 2024 2025 2026 2027 2028 2029 2030]

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename

  ;;output insgesamt
  print-row (sentence "Wien" schultypen_wien_prognose)
  let i 0
  while [i < length item 0 neue_schulen_districts] [
    print-row (sentence (2009 + i) item i neue_klassen_insgesamt_list)
  set i (i + 1)
  ]

  file-print ""

  ;; output bezirke
  let j 0
  while [j < length neue_klassen_districts] [
    ;;output bezirk & schultypen
    print-row (sentence item (j + 1) bezirke_wien schultypen_wien_prognose)

  ;; output jahre & neue-schulen
```

```
  set i 0
  while [i < length item 0 neue_klassen_districts] [
    print-row (sentence (2009 + i) item i item j neue_klassen_districts)
      set i (i + 1)
  ]
  ;; output insgesamt
  print-row (sentence ("Insgesamt") last item j neue_schulen_districts)

  file-print ""
  set i 0
  set j (j + 1)
  ]

  file-close
end

to export-schüler/klasse
  setup-prognose

  let filename "SchülerproKlasse_Prognose.csv"

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename


  ;; output insgesamt
  print-row (sentence "Wien" schultypen_wien_prognose)

  let i 0
  while [i < length schulbesuche_wien] [
    let klassen_wien_jahr []

    let j 0
    ;   schulbesuche_wien   insgesamt   (item   0)   /   klassen_wien_2009   +
neue_schulen_insgsamt

    while [j < length item 0 klassen_wien_2009] [
      set   klassen_wien_jahr   lput   (precision   ((item   j   item   0   item   i
schulbesuche_wien)   /   ((item   j   item   0   klassen_wien)   +   (item   j   item   i
neue_schulen_insgesamt_list))) 1) klassen_wien_jahr
      set j (j + 1)
    ]

    print-row (sentence (2009 + i) klassen_wien_jahr)
    set i (i + 1)
  ]

  file-print ""

  ;; output bezirke
  let j 0
  while [j < length schüler/klasse_districts] [
    ;;output bezirk & schultypen
    print-row (sentence item (j + 1) bezirke_wien schultypen_wien_prognose)

    ;; output jahre & schüler/klasse
    set i 0
    while [i < length item 0 schüler/klasse_districts] [
      print-row (sentence (2009 + i) item i item j schüler/klasse_districts)
      set i (i + 1)
    ]
    file-print ""
    set i 0
    set j (j + 1)
  ]

  file-close
end

to export-schüler/klasse_exkl
  let filename "SchülerproKlasse_Prognose_exkl.csv"

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename

  ;; output insgesamt
  print-row (sentence "Wien" schultypen_wien_prognose)

  let i 0
  while [i < length schüler/klasse] [
    print-row (sentence (2009 + i) item 0 item i schüler/klasse)
  set i (i + 1)
  ]
  file-print ""


  ;; output bezirke
  let j 1
  while [j < length schüler/klasse_districts] [
    ;;output bezirk & schultypen
```

```
    print-row (sentence item j bezirke_wien schultypen_wien_prognose)

  ;; output jahre & schüler/klasse
  set i 0
  while [i < length schüler/klasse] [
    print-row (sentence (2009 + i) item j item i schüler/klasse)
      set i (i + 1)
  ]
  file-print ""
  set i 0
  set j (j + 1)
  ]

  file-close
end

to export-neue_schulen_jahre
  let filename "Neue_Schulen_Jahre_Prognose.csv"

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename

  ;;output insgesamt
  print-row (sentence "2009 - 30" schultypen_wien_prognose)
  print-row (sentence item 0 bezirke_wien neue_schulen_insgesamt)
  let i 0
  while [i < length neue_klassen_districts] [
    let j 0
    let insgesamt (item 0 item i neue_klassen_districts)
    while [j < (length item i neue_klassen_districts) - 1] [
      set insgesamt (map + insgesamt (item (j + 1) item i
neue_klassen_districts))
      set j (j + 1)
    ]

    print-row (sentence item (i + 1) bezirke_wien insgesamt)
    set i (i + 1)
  ]

  file-print ""

    ;;output 2020
  print-row (sentence "2009 - 20" schultypen_wien_prognose)
  print-row (sentence item 0 bezirke_wien item 11 neue_schulen_insgesamt_list)
  set i 0
  while [i < length neue_klassen_districts] [
    let j 0
    let insgesamt (item 0 item i neue_klassen_districts)
    while [j < (length item i neue_klassen_districts) - 11] [
      set insgesamt (map + insgesamt (item (j + 1) item i
neue_klassen_districts))
      set j (j + 1)
    ]

    print-row (sentence item (i + 1) bezirke_wien insgesamt)
    set i (i + 1)
  ]

  file-print ""


  ;; output jahre
  set i 0
  while [i < length item i neue_klassen_districts] [
    ;;output jahr & schultypen
    print-row (sentence (2009 + i) schultypen_wien_prognose)

    ;; output bezirk & neue-schulen
    ;alle bezirke
    let j 0
    let insgesamt (item i item 0 neue_klassen_districts)
    while [j < length item i neue_klassen_districts] [
      set insgesamt (map + insgesamt (item i item (j + 1)
neue_klassen_districts))
      set j (j + 1)
    ]
    print-row (sentence item 0 bezirke_wien insgesamt)

    ;bezirke einzeln
    set j 0
    while [j < length neue_klassen_districts] [
      print-row (sentence item (j + 1) bezirke_wien item i item j
neue_klassen_districts)
      set j (j + 1)
    ]
    file-print ""
    set i (i + 1)
  ]

  file-close
end
```

```
to export-schüler/klasse_jahre
  let filename "SchülerproKlasse_Jahre_Prognose.csv"

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename

  ;; output jahre
  let i 0
  while [i < length item i schüler/klasse_districts] [
    ;;output jahr & schultypen
    print-row (sentence (2009 + i) schultypen_wien_prognose)

  ;; output bezirk & neue-schulen
  ;alle bezirke
  let j 0
  let klassen_wien_jahr []
  ; klassen_insgesamt für jeweiliges jahr
  let   klassen_jahr_i   (map   +   (first   klassen_wien_2009)   (item   i
neue_schulen_insgesamt_list))
  while [j < length klassen_jahr_i] [
    ifelse item j klassen_jahr_i != 0 [
      set   klassen_wien_jahr   lput   (precision   ((item   j   first   item   i
schulbesuche_wien) / (item j klassen_jahr_i)) 1) klassen_wien_jahr
    ][
    set klassen_wien_jahr lput 0 klassen_wien_jahr
    ]
    set j (j + 1)
  ]
  print-row (sentence item 0 bezirke_wien klassen_wien_jahr)

  ;bezirke einzeln
  set j 0
  while [j < length schüler/klasse_districts] [
    print-row   (sentence   item   (j   +   1)   bezirke_wien   item   i   item   j
schüler/klasse_districts)
    set j (j + 1)
  ]
  file-print ""
  set i (i + 1)
  ]

  file-close
end
```



```
to export-schüler/klasse_jahre_exkl
  let filename "SchülerproKlasse_Jahre_Prognose_exkl.csv"

  if file-exists? filename [file-delete filename]
  file-open filename
  show filename

  ;; output jahre
  let i 0
  while [i < length item i schüler/klasse_districts] [
    ;;output jahr & schultypen
    print-row (sentence (2009 + i) schultypen_wien_prognose)

  ;; output bezirk & neue-schulen
  ;alle bezirke
  let j 0
  let klassen_wien_jahr []
  ; klassen_insgesamt für jeweiliges jahr
  let klassen_jahr_i (first klassen_wien_2009)
  while [j < length item 0 item 0 schulbesuche_wien] [
    ifelse item j klassen_jahr_i != 0 [
      set   klassen_wien_jahr   lput   (precision   ((item   j   first   item   i
schulbesuche_wien) / (item j klassen_jahr_i)) 1) klassen_wien_jahr
    ][
    set klassen_wien_jahr lput 0 klassen_wien_jahr
    ]
    set j (j + 1)
  ]
  print-row (sentence item 0 bezirke_wien klassen_wien_jahr)

  ;bezirke einzeln
  set j 0
  while [j < length neue_schulen_districts] [

    ; klassen_insgesamt für jeweiligen bezirk und jeweiliges jahr
    set klassen_jahr_i (item (j + 1) klassen_wien_2009)

    set klassen_wien_jahr []
    let k 0
    while [k < length klassen_jahr_i] [
      ifelse item k klassen_jahr_i != 0 [
        set klassen_wien_jahr lput (precision ((item k item (j + 1) item i
schulbesuche_wien) / (item k klassen_jahr_i)) 1) klassen_wien_jahr
      ][
      set klassen_wien_jahr lput 0 klassen_wien_jahr
      ]
```

```
      set k (k + 1)
    ]

    print-row (sentence item (j + 1) bezirke_wien klassen_wien_jahr)
    set j (j + 1)
  ]
  file-print ""
  set i (i + 1)
  ]

  file-close
end

to print-row [ #items ]
  ;; write all but the last item to theif   currently open file, inserting
commas
  ;; uses file-write to produce output, strings are quoted.
  foreach but-last #items
    [ file-write ?
      file-type ";"
    ]
  ;; write the last item
  file-write last #items
  ;; write end-of-line
  file-print ""
end
```

schulbesuche_prognose.nls

https://gist.github.com/anonymous/852d74a51403f43b479fce20ee7969e7

```
globals [
  schultypen_prognose
  bezirke_wien
  klassen_wien_2009
]


to setup-prognose
  setup-schulbesuche
  calculate-schulbesuche_2030
  schüler/klasse_2030
end


to setup-schulbesuche
  set schultypen_prognose (list
    "Insgesamt"
    "Volksschulen"
    "Hauptschulen"
    "Sonderschulen"
    "Polytechn Schulen"
    "AHS-Unterstufe"
    "AHS-Oberstufe"
    "Berufsschulen"
    "BMS"
    "BHS"
    "LHS"
    )
  let files (list "schülerinnen_2009.csv" "schülerinnen_2020.csv"
"schülerinnen_2030.csv")
  read-csv files
  set schulbesuche_wien output
end

to read-csv [files]
  let i 0
  set output []
  while [i < length files] [
    file-open item i files
    let result []
    while[not file-at-end?] [
      let row csv:from-row file-read-line
      set result lput row result
    ]
    file-close
    set i (i + 1)
    set output lput result output
  ]
  ;show output
end

to calculate-schulbesuche_2030
  let i 0
  let j 0
  let k 0
  let l 0
  let schulbesuche_wien_neu []
  set schulbesuche_wien_neu lput first schulbesuche_wien schulbesuche_wien_neu
  let jahre_dazwischen (list 10 9)
  while [k < 2] [
    while [j < item k jahre_dazwischen] [
      let jahr []
      while [i < length item k schulbesuche_wien] [
        let bezirk []
        while [l < length item l item k schulbesuche_wien] [
          set bezirk lput round (item l item i item k schulbesuche_wien +
(((item l item i item (k + 1) schulbesuche_wien) - (item l item i item k
schulbesuche_wien)) / (item k jahre_dazwischen + 1) * (j + 1))) bezirk
          set l (l + 1)
        ]
        set jahr lput bezirk jahr
        set l 0
        set i (i + 1)
      ]
      set schulbesuche_wien_neu lput jahr schulbesuche_wien_neu
      set i 0
      set j (j + 1)
    ]
    set schulbesuche_wien_neu lput item (k + 1) schulbesuche_wien
schulbesuche_wien_neu
    set j 0
    set k (k + 1)
  ]
  set schulbesuche_wien schulbesuche_wien_neu

  ;show schulbesuche_wien
  ;show length schulbesuche_wien
```

```
end

to schüler/klasse_2030
  ; klassenanzahlen nach schultypen
  let files (list "klassen_2009.csv")
  read-csv files
  set klassen_wien_2009 item 0 output
  set klassen_wien klassen_wien_2009

  ; durchschnittliche schüler pro klassen nach schultypen
  let i 0
  let j 0
  let k 0
  set schüler/klasse []

  while [i < length schulbesuche_wien] [
    let jahr []
    while [k < length item i schulbesuche_wien] [
      let bezirk  []
      while [j < length item k item i schulbesuche_wien] [
        ifelse item j item k item i schulbesuche_wien = 0 or item j item k
klassen_wien_2009 = 0 [
          set bezirk lput 0 bezirk
        ][
        set bezirk lput (precision ((item j item k item i schulbesuche_wien) /
(item j item k klassen_wien_2009)) 1) bezirk
        ]
        set j (j + 1)
      ]
      set jahr lput bezirk jahr
      set j 0
      set k (k + 1)
    ]
    set schüler/klasse lput jahr schüler/klasse
    set j 0
    set k 0
    set i (i + 1)
  ]
  ;show schüler/klasse
  ;show length schüler/klasse

  ;wenn schüler/klasse > 25, dann erhöhe anzahl der klassen um 1
end
```

7.3.2. Schulgenerator

7.3.2.1. Simulation der Bewegungsabläufe

https://gist.github.com/anonymous/8c9f27d94498a91397996eb7387cb8cf

```
extensions [nw]

__includes ["bildungscampus_nordbahnhof.nls" "statistics.nls" "export2csv.nls"]

globals [
  funktionen
  bereiche_xy
  verbindungen
  verbindungen_insgesamt
  sort_bereiche
  sort_unterbereiche
  fächer_biber
  schultypen
  anzahlen_klassen
  schultypen+klassen
  stop?
  who_global_wcc
  who_local_wcc
  who_links
  through-put_list
  most-neighbors_list
  ]

breed [bereiche bereich]
breed [unterbereiche unterbereich]
breed [bibere biber]
breed [küchen küche]
breed [kreativbereiche kreativbereich]
breed [cursors cursor]
breed [classes class]

bereiche-own [fläche höhe funktion gefordert fächer schultyp mein_bereich
besucht auslastung weighted_centrality global_wcc local_wcc]
unterbereiche-own [art fläche höhe gefordert mein_bereich meine_biber fächer
#raum besucht auslastung weighted_centrality global_wcc local_wcc funktion]
classes-own [schultyp #biber klasse jahrgang meine_targets stundentafel
noch_stunden fach_gefordert home_bereich home_biber target target_alt
target_bereich target_bereich_alt weg_zurückgelegt noch_stunden_in_bereich
doppelstunde stundenplan]
links-own [through-put through-put_inverse]

to setup
  clear-all
  reset-ticks
  if seed != 0 [
    random-seed seed
  ]
  setup-globals
  ;setup-patches
  setup-bildungscampus_nordbahnhof
  setup-bereiche
  setup-unterbereiche
  ifelse anzahl_klassen_test != 0 [
    setup-classes-test
  ][
  setup-classes
  ]
  setup-verbindungen
  setup-links
  setup-layout
  setup-observer
  ;raumgrößen

end

to setup_2030
  clear-all
  if seed != 0 [
    random-seed seed
  ]
  setup-globals
  ;setup-patches
  setup-bildungscampus_nordbahnhof
  setup-bereiche
  setup-unterbereiche
  ifelse anzahl_klassen_test != 0 [
    setup-classes-test
  ][
  setup-classes
  ]
  setup-verbindungen
  setup-links
  setup-layout
  setup-observer
  ;raumgrößen
```

```
  reset-ticks
end

to setup-observer
  if ceiling __oxcor = 0 and ceiling __oycor = 0 and ceiling __ozcor = 222 [
    setxyz max-pxcor max-pycor max-pzcor
    zoom (max-pxcor * -.5)
  ]
  facexyz (mean [xcor] of bereiche) (mean [ycor] of bereiche) (mean [zcor] of
bereiche)
end

to setup-globals
  set funktionen (list "Gymnastiksäle" "3-Fach-Turnhalle" "Veranstaltungssaal"
"VS + KDG" "NMS / FMS" "Musikschule")
  set fächer_list (list (list "sport") (list "sport") (list 0) (list 0) (list
0) (list 0))
  set fächer_biber (list "ethik" "sachunterricht" "deutsch" "fremdsprache"
"geschichte" "geographie" "mathematik" "ernaehrung" "berufsorientierung")
  set verbindungen []
  set verbindungen_insgesamt []
  ; ausschreibung s.20
  set schultypen (list "KKG" "KDG" "GTVS" "SOPÄD" "GTNMS" "FMS" "MS")
  set anzahlen_klassen (list 5 11 22 2 20 2 12)
  set schultypen+klassen []
  let i 0
  while [i < length schultypen][
    let schultyp+klasse []
    set schultyp+klasse lput item i schultypen schultyp+klasse
    set schultyp+klasse lput item i anzahlen_klassen schultyp+klasse
    set schultypen+klassen lput schultyp+klasse schultypen+klassen
    set i (i + 1)
  ]
  set who_global_wcc []
  set who_local_wcc []
  set who_links []
  set through-put_list []
end

to setup-patches
  import-pcolors-rgb "1.jpg"
  ask patches [
    if pcolor != [254 0 0] [
      set pcolor 0
    ]
  ]
  ask patches [
    if pcolor != 0 [
      ask patch ([pxcor] of self) ([pycor] of self) min-pzcor [
        set pcolor blue
      ]
    ]
  ]
  ask patches [
    if pcolor != blue [
      set pcolor 0
    ]
  ]

end

to setup-bereiche
  let i 0
  let j 0
  let schultypen_list (list (list 0) (list 0) (list 0) (list "KKG" "KDG" "GTVS"
"SOPÄD") (list "GTNMS" "FMS") (list "MS"))
  let colors (list [192 192 192 150] [192 192 192 150] [234 102 28 150] [255
255 255 150] [255 255 255 150] [0 157 224 150])
  while [i < length funktionen] [
    set j 0
    create-bereiche 1 [
      ;setxyz random-xcor random-ycor random-zcor
      set color item i colors
      set fläche item i item 0 flächen
      set höhe item i item 0 höhen
      set size sqrt (fläche / pi) * 2
      set shape "circle"
      set schultyp []
      while [j < length item i schultypen_list] [
        set schultyp lput item j item i schultypen_list schultyp
        set j (j + 1)
      ]
      set funktion item i funktionen
      set fächer item i fächer_list
      set label funktion
      set mein_bereich self
      set label-color white
    ]
    set i (i + 1)
  ]
end
```

```
to setup-unterbereiche
  ; setup biber
  let anzahlen_biber (list 0 0 0 6 4 1)
  let i 0
  let j 0
  let k 1
  let l 0
  while [l < count bereiche] [
    ask bereich l [
      while [i < item j anzahlen_biber] [
        hatch-unterbereiche 1 [
          set mein_bereich myself
          set art "biber"
          ifelse count unterbereiche with [art = "biber"] < 7 [
            set color [240 154 191 150]
          ][
          set color [253 204 26 150]
          ]
          set fläche item (count unterbereiche with [art = "biber"] - 1) item 1
flächen
          set höhe item (count unterbereiche with [art = "biber"] - 1) item 1
höhen
          set size sqrt (fläche / pi) * 2
          set shape "circle"
          ifelse count unterbereiche with [art = "biber"] != 6 [
            set #raum k
          ][
          set #raum "KKG"
          set k (k - 1)
          ]
          set fächer fächer_biber
          if l = 3 [
            set fächer (sentence fächer "essen")
          ]
          set label #raum
        ]
        set i (i + 1)
        set k (k + 1)
      ]
      set i 0
      set j (j + 1)
    ]
    set l (l + 1)
  ]

  ;setup küchen
  set i 0
  set j 1
  while [i < 4] [
    create-unterbereiche 1 [
      set art "küche"
      set color [18 128 129 150]
      set fläche item (count unterbereiche with [art = "küche"] - 1) item 2
flächen
      set höhe item (count unterbereiche with [art = "küche"] - 1) item 2 höhen
      set size sqrt (fläche / pi) * 2
      set shape "circle"
      set meine_biber []
      ifelse i < 3 [
        set meine_biber lput unterbereiche with [art = "biber" and #raum = j or
#raum = (j + 1)] meine_biber
      ][
      set meine_biber lput unterbereiche with [art = "biber" and #raum = j or
#raum = (j + 1) or #raum = (j + 2) or #raum = (j + 3) or #raum = (j + 4)]
meine_biber
      ]
      if i = 2 [
        set meine_biber []
        set meine_biber lput unterbereiche with [art = "biber" and #raum = j or
#raum = "KKG"] meine_biber
        set j (j - 1)
      ]
      set mein_bereich [mein_bereich] of item 0 meine_biber
      set mein_bereich remove-duplicates mein_bereich
      set mein_bereich item 0 mein_bereich
      ifelse i = 3 [
        set fächer (list "essen")
      ][
      set fächer (list 0)
      ]
      set #raum (i + 1)
      if i = 3 [
        set #raum "Speisesaal"
      ]
      set label #raum
      set meine_biber (turtle-set meine_biber)
      setxyz ([xcor] of mein_bereich) ([ycor] of mein_bereich) ([zcor] of
mein_bereich)
    ]
    set i (i + 1)
    set j (j + 2)
  ]
```

```
  ;setup kreativbereiche
  set i 0
  set j 1
  let biber_list (list 6 10 11)
  while [i < 3] [
    create-unterbereiche 1 [
      set art "kreativbereich"
      set color [234 102 28 150]
      set fläche item (count unterbereiche with [art = "kreativbereich"] - 1)
item 3 flächen
      set höhe item (count unterbereiche with [art = "kreativbereich"] - 1)
item 3 höhen
      set size sqrt (fläche / pi) * 2
      set shape "circle"
      set #raum (i + 1)
      set label #raum
      set meine_biber []
      while [j < item i biber_list] [
        set meine_biber lput unterbereiche with [art = "biber" and #raum = j]
meine_biber
        set j (j + 1)
      ]
      if i = 0 [
        set meine_biber lput unterbereiche with [art = "biber" and #raum =
"KKG"] meine_biber
      ]
      set mein_bereich [mein_bereich] of item 0 meine_biber
      set mein_bereich remove-duplicates mein_bereich
      set mein_bereich item 0 mein_bereich
      set meine_biber (turtle-set meine_biber)
      set fächer (list "biologie" "chemie" "physik" "musik" "be" "werken")
    ]
    set i (i + 1)
  ]

  ;setup administrative bereiche
  set i 0
  while [i < 2] [
    create-unterbereiche 1 [
      set art "koll. führung"
      set color [238 131 59 150]
      set fläche 0
      set fläche item (count unterbereiche with [art = "koll. führung"] - 1)
item 4 flächen
      set höhe item (count unterbereiche with [art = "koll. führung"] - 1) item
4 höhen
      set size sqrt (fläche / pi) * 2
      set shape "circle"
      set #raum (i + 1)
      set label art
      set mein_bereich bereich (i + 3)
    ]
    set i (i + 1)
  ]

  ; biber stellen links mit anderen bibern desselben bereichs her
  ask unterbereiche with [art = "biber"] [
    set i 0
    ;create-link-with mein_bereich
    let link_partner other unterbereiche with [art = [art] of myself and
mein_bereich = [mein_bereich] of myself]
    set link_partner sort link_partner
    while [i < length link_partner] [
      if link (who) ([who] of item i link_partner) = nobody [
        create-link-with item i link_partner [
          set color [255 255 255 150]
          ;set hidden? true
        ]
      ]
      set i (i + 1)
    ]
  ]

  ; küchen stellen links mit ihren bibern her
  ask unterbereiche with [art = "küche" and #raum != "Speisesaal"] [
    let biber_set (turtle-set meine_biber)
    create-links-with biber_set [
      set color [255 255 255 150]
      ;set hidden? true
    ]
  ]

  ; kreativbereiche stellen links mit ihren bibern (& küche???) her
  ask unterbereiche with [art = "kreativbereich"] [
    let biber_set []
    ifelse self != unterbereich 23 [
      set biber_set (turtle-set meine_biber unterbereiche with [art = "küche"
and #raum != "Speisesaal" and mein_bereich = [mein_bereich] of myself])
    ][
```

```
    set biber_set (turtle-set meine_biber unterbereiche with [art = "küche" and
mein_bereich = bereich 4])
    ]
    create-links-with biber_set [
      set color [255 255 255 150]
      ;set hidden? true
    ]
  ]

  ; koll_führung stellt links mit bibern desselben bereichs her
  ask unterbereiche with [art = "koll. führung"] [
    create-links-with other unterbereiche with [mein_bereich = [mein_bereich]
of myself and art = "biber" or art = "küche" or art = "kreativbereich"] [
      set color [255 255 255 150]
      ;set hidden? true
    ]
  ]

  ask unterbereiche [
    set funktion (word art #raum)
  ]
end

to raumgrößen
  let räume (list bereiche unterbereiche)

  let i 0
  ask item i räume [
    let fläche_scale sqrt (fläche / global_scale)
    ;show fläche_scale
    let höhe_scale höhe / global_scale
    ;show höhe_scale

    ask patches with [(pxcor < round [xcor] of myself + fläche_scale)
      and (pxcor > round [xcor] of myself - fläche_scale)
      and (pycor < round [ycor] of myself + fläche_scale)
      and (pycor > round [ycor] of myself - fläche_scale)
      and (pzcor = round [zcor] of myself)] [
    set pcolor [color] of myself
      ]
    set i (i + 1)
  ]
end

to setup-classes-test
  let i 0
while [i < anzahl_klassen_test][
  create-classes 1 [
    setxyz 0 0 0
    set size 5
    set shape "person"
    set schultyp "MS"
    set #biber 10
    set home_bereich bereich 5
    set home_biber unterbereich 16
    set klasse 0
    set jahrgang 1
    set stundentafel item 0 item 6 stundentafeln
    let l 1
    while [l < length stundentafel] [
      set noch_stunden (noch_stunden + item l stundentafel)
      set l (l + 2)
    ]
    set color white
    set noch_stunden_in_bereich stunden_in_bereich
  ]
  set i (i + 1)
]

; find meine_targets
ask classes [
  set meine_targets []
  set meine_targets lput home_biber meine_targets
  set i 0
  let j 0
  let räume (list "küche" "kreativbereich")
  while [j < length räume] [
    ask unterbereiche with [art = item j räume] [
      let meine_biber_list sort meine_biber
      while [i < length meine_biber_list] [
        if item i meine_biber_list = [home_biber] of myself [
          ask myself [
            set meine_targets lput myself meine_targets
          ]
        ]
        set i (i + 1)
      ]
      set i 0
    ]
    set j (j + 1)
  ]
```

```
    set meine_targets lput bereiche with [funktion = "3-Fach-Turnhalle" or
funktion = "Gymnastiksäle"] meine_targets
    set meine_targets (turtle-set meine_targets)
  ]
end

to setup-classes-test_gtnms
  let i 0
  while [i < anzahl_klassen_test][
    create-classes 1 [
      setxyz 0 0 0
      set size 5
      set shape "person"
      set schultyp "GTNMS"
      set #biber 6
      set home_bereich bereich 4
      set home_biber unterbereich 12
      set klasse 0
      set jahrgang 1
      set stundentafel item 0 item 4 stundentafeln
      let l 1
      while [l < length stundentafel] [
        set noch_stunden (noch_stunden + item l stundentafel)
        set l (l + 2)
      ]
      set color white
      set noch_stunden_in_bereich stunden_in_bereich
    ]
    set i (i + 1)
  ]

  ; find meine_targets
  ask classes [
    set meine_targets []
    set meine_targets lput home_biber meine_targets
    set i 0
    let j 0
    let räume (list "küche" "kreativbereich")
    while [j < length räume] [
      ask unterbereiche with [art = item j räume] [
        let meine_biber_list sort meine_biber
        while [i < length meine_biber_list] [
          if item i meine_biber_list = [home_biber] of myself [
            ask myself [
              set meine_targets lput myself meine_targets
            ]
            set i (i + 1)
          ]
          set i 0
        ]
        set j (j + 1)
      ]
    set meine_targets lput bereiche with [funktion = "3-Fach-Turnhalle" or
funktion = "Gymnastiksäle"] meine_targets
    set meine_targets (turtle-set meine_targets)
  ]
end

to setup-classes
  set schultypen (list "KKG" "KDG" "GTVS" "SOPÄD" "GTNMS" "FMS" "MS")
  let biber_list 0
  ifelse jahr? = "2016" [
    set anzahlen_klassen (list 5 11 22 2 20 2 12)
    ; ausschreibung s.30
    set biber_list (list (list "KKG" "KKG" "KKG" "KKG" "KKG") (list 1 1 2 2 3 3
4 4 5 5 5) (list 1 1 1 1 1 2 2 2 2 2 3 3 3 3 4 4 4 4 5 5 5 5 5) (list 3 4)  (list
6 6 6 6 6 7 7 7 7 7 8 8 8 8 8 9 9 9 9 9) (list 8 9) (list 10 10 10 10 10 10 10
10 10 10 10))
  ][
  set anzahlen_klassen (list 5 11 24.4 2.2 22.2 1.8 13.3)
  set biber_list (list (list "KKG" "KKG" "KKG" "KKG" "KKG") (list 1 1 1 2 2 2 3
3 3 4 4 5 5 5) (list 1 1 1 1 1 1 2 2 2 2 2 2 3 3 3 3 3 4 4 4 4 5 5 5 5) (list 3
4 4)  (list 6 6 6 6 6 6 7 7 7 7 7 7 8 8 8 8 8 8 9 9 9 9 9) (list 8 9) (list 10
10 10 10 10 10 10 10 10 10 10 10 10))
  ]

  let colors (list white red blue green yellow orange grey)
  ;let jahrgänge (list 1 2 3 4)
  let i 0
  let j 0
  let k 1
  while [j < length anzahlen_klassen] [
    while [i < ceiling item j anzahlen_klassen] [
      create-classes 1 [
        setxyz random-xcor random-ycor random-zcor
        set size 2
        set shape "person"
        set schultyp item j schultypen
        set #biber item i item j biber_list
        set home_biber one-of unterbereiche with [art = "biber" and #raum =
[#biber] of myself]
```

```
        set klasse 0
        ifelse k <= length item j stundentafeln and length item j stundentafeln
!= 0[
          set jahrgang k
          set k (k + 1)
        ][
        set k 1
        set jahrgang k
        ]
        ;stundentafel abhängig von jahrgang und schultyp
        ifelse length item j stundentafeln != 0 [
          set stundentafel item (jahrgang - 1) item j stundentafeln
        ][
        set stundentafel (list )
        ]
        let l 1
        while [l < length stundentafel] [
          set noch_stunden (noch_stunden + item l stundentafel)
          set l (l + 2)
        ]
        set color item j colors
        set noch_stunden_in_bereich stunden_in_bereich
        set doppelstunde 2
        set stundenplan []
      ]
      set i (i + 1)
    ]
    set i 0
    set j (j + 1)
  ]

  ; find home_bereich
  set i 0
  ask classes [
    ask bereiche [
      while [i < length schultyp] [
        if item i schultyp = [schultyp] of myself [
          ask myself [
            set home_bereich myself
          ]
        ]
        set i (i + 1)
      ]
      set i 0
    ]
  ]

  ; find meine_targets
  ask classes [
    set meine_targets []
    set meine_targets lput home_biber meine_targets
    set i 0
    set j 0
    let räume (list "küche" "kreativbereich")
    while [j < length räume] [
      ask unterbereiche with [art = item j räume] [
        let meine_biber_list sort meine_biber
        while [i < length meine_biber_list] [
          if item i meine_biber_list = [home_biber] of myself [
            ask myself [
              set meine_targets lput myself meine_targets
            ]
          ]
          set i (i + 1)
        ]
        set i 0
      ]
      set j (j + 1)
    ]
    set meine_targets lput bereiche with [funktion = "3-Fach-Turnhalle" or
funktion = "Gymnastiksäle"] meine_targets
    if home_bereich = bereich 5 [
       set meine_targets lput unterbereiche with [art = "küche" and #raum =
"Speisesaal"] meine_targets
    ]
    set meine_targets (turtle-set meine_targets)
  ]
end

to setup-verbindungen
  ; alle möglichen verbindungen der bereiche miteineander, auch mit sich selbst
  let i 0
  let j 0
  let verbindung []
  let alle_bereiche sort-on [who] bereiche
  while [i < length alle_bereiche] [
    while [j < length alle_bereiche] [
      set verbindung lput item i alle_bereiche verbindung
      set verbindung lput item j alle_bereiche verbindung
      ;set verbindung lput 0 verbindung
      set verbindungen lput verbindung verbindungen
      set verbindung []
```

```
      set j (j + 1)
    ]
    set j 0
    set i (i + 1)
  ]
  set i 0
  while [i < length verbindungen] [
    let neu item i verbindungen
    set neu sort neu
    set neu lput 0 neu
    ;ifelse item 0 neu = item 1 neu [
    ;  set verbindungen remove-item i verbindungen
    ;][
    set verbindungen replace-item i verbindungen neu
    ;]
    set i (i + 1)
  ]
  set verbindungen remove-duplicates verbindungen

  set  i 0
  while [i < length verbindungen] [
    let neu item i verbindungen
    if item 0 neu = item 1 neu [
      set verbindungen remove-item i verbindungen
    ]
    set i (i + 1)
  ]

  ;setup-verbindungen_insgesamt
  set i 0
  while [i < length verbindungen] [
    set verbindungen_insgesamt lput 0 verbindungen_insgesamt
    set i (i + 1)
  ]
end

to setup-links
  let i 0
  while [i < length verbindungen] [
    ask item 0 item i verbindungen [
      create-link-with item 1 item i verbindungen [
        set color [255 0 0 150]
        set through-put 0
        set thickness through-put
        set tie-mode "none"
        ;set hidden? true
      ]
      set i (i + 1)
    ]
  ]
end

to setup-layout
  bereiche-anordnen

  let i 0
  let unterbereiche_list (list "biber" "küche" "kreativbereich" "koll.
führung")
  while [i < length unterbereiche_list] [
    ask unterbereiche with [art = item i unterbereiche_list] [
      setxyz ([xcor] of mein_bereich) ([ycor] of mein_bereich) ([zcor] of
mein_bereich)
    ]
    set i (i + 1)
  ]

  unterbereiche-anordnen
  ; küchen von vs zwischen ihre biber
  ask unterbereiche with [art = "küche"] [
    if [funktion] of mein_bereich = "VS + KDG" [
      set meine_biber (turtle-set meine_biber)
      setxyz (mean [xcor] of meine_biber) (mean [ycor] of meine_biber) (mean
[zcor] of meine_biber)
    ]
  ]
  ; speisesaal zischen nms und ms
  ask unterbereiche with [art = "küche" and #raum = "Speisesaal"] [
    setxyz (([xcor] of bereich 4 + [xcor] of bereich 5) / 2) (([ycor] of
bereich 4 + [ycor] of bereich 5) / 2) (([zcor] of bereich 4 + [zcor] of bereich
5) / 2)
  ]
  ask bereiche with [funktion = "VS + KDG" or funktion = "NMS / FMS" or
funktion = "Musikschule"] [
    ;set size (size / 3)
    ;set color [255 255 255 1]
    set hidden? true
  ]

  ask classes [
    if target != 0 [
      setxyz ([xcor] of target) ([ycor] of target) ([zcor] of target)
    ]
```

```
  ]

  ask links [die]
end

to bereiche-anordnen
  ;layout-spring turtle-set link-set spring-constant spring-length repulsion-
constant
  let turtle_set bereiche
  let link_set links with [color = [255 0 0 150]]
  repeat 10 [
    layout-spring turtle_set link_set 1 50 .1
  ]
end

to unterbereiche-anordnen
  ; biber, küchen und kreativbereiche und koll. führung um bereich anordnen
  let i 0
  while [i < count bereiche] [
    let link_biber unterbereiche with [art = "biber" and mein_bereich = bereich
i]
    let link_küchen unterbereiche with [art = "küche" and mein_bereich =
bereich i]
    let link_kreativbereiche unterbereiche with [art = "kreativbereich" and
mein_bereich = bereich i]
    let link_führung unterbereiche with [art = "koll. führung" and mein_bereich
= bereich i]
    let link_bereich bereich i
    let turtle_set (turtle-set link_biber link_küchen link_kreativbereiche
link_führung link_bereich)
    ;let link_set links with [color = [255 255 255 150]]
    let link_set links with [[mein_bereich] of end1 = bereich i and
[mein_bereich] of end2 = bereich i]
    repeat 50 [
      layout-spring turtle_set link_set 1 25 .1
    ]
    set i (i + 1)
  ]
end

to go
  find-target
  move-classes-to-target
  update-stundentafel
  ;bereiche-anordnen
  ;unterbereiche-anordnen
  invert-through_put
  update-network
  setup-observer
  tick
  update-auslastung
  update-through-put_list
  if count classes with [hidden? = false] = 0 [
    set stop? true
    experiment-vorbereiten
    stop
  ]
end

to find-target
    ask classes with [hidden? = false] [
      ifelse ticks = 0 or noch_stunden_in_bereich = 0 or target = unterbereich
20[
        find-new-target
      ][
      stay-at-target
      ]
      set noch_stunden_in_bereich (noch_stunden_in_bereich - 1)
      set stundenplan lput (sentence fach_gefordert target) stundenplan
    ]
end

to find-new-target
  set noch_stunden_in_bereich stunden_in_bereich

  ask turtles with [breed = bereiche or breed = unterbereiche] [
    set gefordert false
  ]
  set target_bereich_alt target_bereich
  set target_alt target

  if target = 0 [
    setxyz ([xcor] of home_bereich) ([ycor] of home_bereich) ([zcor] of
home_bereich)
  ]

  ; überprüfe, welche meine_targets fächer besitzen, die die klasse noch nicht
absolviert hat
  let i 0
  let j 0
  if is-list? stundentafel [
    while [i < length stundentafel] [
```

```
      ask meine_targets [
        while [j < length fächer] [
          if ticks = 22 [
          ]
          if item j fächer = [item i stundentafel] of myself [
            set gefordert true
            set j length fächer
          ]
          set j (j + 1)
        ]
        set j 0
      ]
      set i (i + 2)
    ]
    set i 0
  ]

  ; geforderte meine_targets können nach bestimmten kriterien sortiert werden
  set target one-of meine_targets with [gefordert = true]
  ;set  target  min-one-of  meine_targets  with  [gefordert  =  true]  [distance
myself]

  ; set fach_gefordert
  set i 0
  set j 0
  while [i < length stundentafel] [
    while [j < length [fächer] of target] [
      if item i stundentafel = [item j fächer] of target [
        set fach_gefordert item i stundentafel
      ]
      set j (j + 1)
    ]
    set j 0
    set i (i + 1)
  ]

  ifelse target != nobody [
    set target_bereich [mein_bereich] of target
    if  target  =  one-of  unterbereiche  with  [#raum  =  "Speisesaal"]  and
[home_bereich] of self = bereich 5 [
      set target_bereich [home_bereich] of self
    ]
  ][
  set target_bereich [mein_bereich] of home_biber
  ]
end
```

```
to stay-at-target
  ifelse doppelstunden? = false or doppelstunde = 0 [
    ; set neues fach_gefordert
    set fach_gefordert 0
    let i 0
    let j 0
    while [i < length stundentafel] [
      while [j < length [fächer] of target] [
        if item i stundentafel = [item j fächer] of target [
          set fach_gefordert item i stundentafel
        ]
        set j (j + 1)
      ]
      set j 0
      set i (i + 1)
    ]
  ][
  set doppelstunde (doppelstunde - 1)
  ]


  if fach_gefordert = 0 [
    find-new-target
  ]
end

to move-classes-to-target
  ask classes [
    ifelse target != nobody [
      ifelse pen-down? = true [pd][pu]

      ; zum target bewegen
      facexyz ([xcor] of target) ([ycor] of target) ([zcor] of target)
      while [distance target > 1] [
        forward 1
        set weg_zurückgelegt (weg_zurückgelegt + 1)
      ]
      ; besucht von target u bereich +1 setzen
      ask target [
        set besucht (besucht + 1)
      ]
      ask [mein_bereich] of target [
        set besucht (besucht + 1)
      ]
```

```
      ; erreichtes target soll eine verbindung zum letzten erreichten der
klasse herstellen
      update-verbindungen
      update-link
    ][
    set target home_biber
    facexyz ([xcor] of target) ([ycor] of target) ([zcor] of target)
      while [distance target > 1] [
        forward 1
        set weg_zurückgelegt (weg_zurückgelegt + 1)
      ]
      ask target [
        set besucht (besucht + 1)
      ]
    ]
  ]
end

to update-stundentafel
  ask classes [
    if is-list? stundentafel [
      let i 0
      while [i < length stundentafel] [
        if fach_gefordert = item i stundentafel [
          ifelse item (i + 1) stundentafel = 1 [
            repeat 2 [
              set stundentafel remove-item i stundentafel
            ]
            set doppelstunde 0
          ][
          set stundentafel replace-item (i + 1) stundentafel ((item (i + 1)
stundentafel) - 1)
          ]
        ]
        set i (i + 2)
      ]
    ]
    if length stundentafel = 0 [
      set hidden? true
    ]
    set noch_stunden (noch_stunden - 1)
  ]
end

to update-auslastung
    let i 0
  let räume (list bereiche bibere kreativbereiche küchen)
  while [i < length räume] [
    ask item i räume [
      set auslastung round ((besucht / ticks) * 100)
    ]
    set i (i + 1)
  ]
end

to update-verbindungen
  let i 0
  let j 0
  if target_bereich_alt != 0 [
    while [i < length verbindungen][
      if item 0 item i verbindungen = target_bereich and item 1 item i
verbindungen = [target_bereich_alt] of self [
        let neu item i verbindungen
        set neu replace-item 2 neu (item 2 neu + 1)
        set verbindungen replace-item i verbindungen neu
      ]
      if item 0 item i verbindungen = [target_bereich_alt] of self and item 1
item i verbindungen = target_bereich [
        let neu item i verbindungen
        set neu replace-item 2 neu (item 2 neu + 1)
        set verbindungen replace-item i verbindungen neu
      ]
      set i (i + 1)
    ]
  ]
  ;verbindungen_insgesamt updaten
  set i 0
  while [i < length verbindungen_insgesamt] [
    set verbindungen_insgesamt replace-item i verbindungen_insgesamt item 2
item i verbindungen
    set i (i + 1)
  ]
end

to update-link
  ; link zwischen bereichen updaten
  if target_alt != 0 [
    ifelse target_bereich != target_bereich_alt[
      ;link zwischen bereichen
      ifelse link ([who] of target_bereich) ([who] of target_bereich_alt) =
nobody [
        ask target_bereich [
```

```
      create-link-with [target_bereich_alt] of myself [
        set color [255 0 0 75]
        set through-put 1
        set thickness ((sum verbindungen_insgesamt) / through-put) * 1
      ]
    ]
  ][
  ask link ([who] of target_bereich) ([who] of target_bereich_alt) [
    set through-put (through-put + 1)
    set thickness ((sum verbindungen_insgesamt) / through-put) * 1
  ]
  ]
  ;link zwischen bereich und unterbereich
  ifelse link ([who] of target) ([who] of target_alt) = nobody [
    ask target [
      create-link-with [target_alt] of myself [
        set color [0 0 255 75]
        set through-put 1
        set thickness ((sum verbindungen_insgesamt) / through-put) * .1
      ]
    ]
  ][
  ask link ([who] of target) ([who] of target_alt) [
    set through-put (through-put + 1)
    set thickness ((sum verbindungen_insgesamt) / through-put) * .1
  ]
  ]
][

; link zwischen unterbereichen updaten
if target != target_alt [
  ifelse link ([who] of target) ([who] of target_alt) = nobody [
    ask target [
      create-link-with [target_alt] of myself [
        set color [255 255 255 75]
        set through-put 1
        set thickness ((sum verbindungen_insgesamt) / through-put) * .1
      ]
    ]
  ][
  ask link ([who] of target) ([who] of target_alt) [
    set through-put (through-put + 1)
    set thickness ((sum verbindungen_insgesamt) / through-put) * .1
  ]
  ]
]
    ]
  ]
  ifelse links_down? = false [
    ask links [
      set hidden? true
    ]
  ][
  ask links [
    set hidden? false
  ]
  ]
end

to invert-through_put
  ask links [
    set through-put_inverse (1 / through-put)
  ]
end

to update-network
  ; festlegen, welche turtles und links teil das netzwerk bilden
  ;let turtle_set bereiche
  let turtle_set turtles
  ;let link_set links with [[breed] of end1 = bereiche and [breed] of end2 =
bereiche]
  let link_set links
  nw:set-context turtle_set link_set
  ;show nw:get-context

  ;show nw:mean-path-length
  ;show nw:mean-weighted-path-length "through-put"

  ask bereiche [
    ;show nw:eigenvector-centrality
    ;show nw:closeness-centrality
    set    weighted_centrality    nw:weighted-closeness-centrality    "through-
put_inverse"
    ;set global_wcc nw:weighted-closeness-centrality "through-put_inverse"
  ]

  ask turtles with [breed != classes and breed != bereiche] [
    ;show nw:eigenvector-centrality
    ;show nw:closeness-centrality
    set    weighted_centrality    nw:weighted-closeness-centrality    "through-
put_inverse"
    ;set global_wcc nw:weighted-closeness-centrality "through-put_inverse"
```

```
  ]

  ;show nw:bicomponent-clusters
  ;show nw:maximal-cliques
  if ticks != 0 [
    normalize
  ]
end

to normalize
    normalize_wcc
    show-wcc
end

to normalize_wcc
  ; faktor für local_wcc der überbereiche
  let weighted_centrality_list sort [weighted_centrality] of bereiche
  if last weighted_centrality_list != 0 [
    let üb_factor (100 / last weighted_centrality_list)

    ; für alle gültiger faltor_global_wcc
    set weighted_centrality_list sort [weighted_centrality] of turtles with
[breed = bereiche or breed = unterbereiche]
    let global_factor (100 / last weighted_centrality_list)

    ask bereiche [
      set global_wcc (weighted_centrality * global_factor)
      set local_wcc (weighted_centrality * üb_factor)
    ]

    ;normalize unterbereiche pro bereich
    let i 0
    while [i < count bereiche] [
      set weighted_centrality_list sort [weighted_centrality] of unterbereiche
with [mein_bereich = bereich i]
      if length weighted_centrality_list != 0 and last weighted_centrality_list
!= 0 [
        let factor (100 / last weighted_centrality_list)

        ask unterbereiche with [mein_bereich = bereich i] [
          set global_wcc (weighted_centrality * global_factor)
          set local_wcc (weighted_centrality * factor)
        ]
      ]
      set i (i + 1)
    ]
    ; bereiche sort-on wc
    set sort_bereiche reverse sort-on [global_wcc] bereiche
    set i 0
    while [i < length sort_bereiche][
      ifelse global_local? = "global" [
        set sort_bereiche replace-item i sort_bereiche (sentence [funktion] of
item i sort_bereiche round [global_wcc] of item i sort_bereiche)
      ][
      set sort_bereiche replace-item i sort_bereiche (sentence [funktion] of
item i sort_bereiche round [local_wcc] of item i sort_bereiche)
      ]
      set i (i + 1)
    ]

    ; unterbereiche sort-on wc
    set i 0
    let j 0
    set sort_unterbereiche []
    while [j < count bereiche] [
      let item_j []
      ifelse global_local? = "global" [
        set item_j reverse sort-on [global_wcc] unterbereiche with
[mein_bereich = bereich j]
      ][
      set item_j reverse sort-on [local_wcc] unterbereiche with [mein_bereich =
bereich j]
      ]
      while [i < length item_j][
        ifelse global_local? = "global" [
          set item_j replace-item i item_j (sentence [art] of item i item_j
[#raum] of item i item_j round [global_wcc] of item i item_j)
        ][
        set item_j replace-item i item_j (sentence [art] of item i item_j
[#raum] of item i item_j round [local_wcc] of item i item_j)
        ]
        set i (i + 1)
      ]
      set sort_unterbereiche lput item_j sort_unterbereiche
      set i 0
      set j (j + 1)
    ]
  ]
end

to show-wcc
```

```
  if weighted_centrality? = true [
    ifelse global_local? = "global" [
      ;nach global_wcc einfärben
      ask turtles with [breed != classes] [
        set label []
        ifelse breed = bereiche [
          set label lput funktion label
        ][
        set label lput #raum label
        ]
        set label lput round global_wcc label
        ; color mit transparenz
        set color (list 255 (255 - (global_wcc * (255 / 100))) 0 150)
      ]
    ][
    ;nach local_wcc einfärben
    ask turtles with [breed != classes] [
      set label []
      ifelse breed = bereiche [
        set label lput funktion label
        ; color mit transparenz
        set color (list 0 (255 - (local_wcc * (255 / 100))) 255 150)
      ][
      set label lput #raum label
      set label lput round local_wcc label
      ; color mit transparenz
      set color (list 255 (255 - (local_wcc * (255 / 100))) 0 150)
      ]
    ]
    ]
  ]
end

to layout-on-weighted_centrality
  ask bereiche [
    set hidden? false
    set size 10
  ]
  let abstand 5

  let i 0
  while [i < length sort_bereiche] [
    ask item i sort_bereiche [
      setxyz (0 + i * abstand) (0 + i * abstand) (0 + i * abstand)
    ]
    set i (i + 1)
  ]
end

to update-through-put_list
  set through-put_list []
  let links_sort reverse sort-on [through-put] links
  let i 0
  while [i < length links_sort] [
    ask item i links_sort [
      let verbindung []
      set verbindung lput (sentence [funktion] of end1 [funktion] of end2)
verbindung
      set verbindung lput through-put verbindung
      set through-put_list lput verbindung through-put_list
    ]
    set i (i + 1)
  ]

  set most-neighbors_list []
  let turtles_sort reverse sort-on [count link-neighbors] turtles with [breed
!= classes]
  set i 0
  while [i < length turtles_sort] [
    ask item i turtles_sort [
      let verbindung []
      set verbindung lput funktion verbindung
      set verbindung lput count link-neighbors verbindung
      set most-neighbors_list lput verbindung most-neighbors_list
    ]
    set i (i + 1)
  ]
end

to experiment-vorbereiten
  let bereiche_who sort-on [who] turtles with [breed != classes]
  let i 0
  while [i < length bereiche_who] [
    ask item i bereiche_who [
      set who_global_wcc lput round [global_wcc] of self who_global_wcc
      set who_local_wcc lput round [local_wcc] of self who_local_wcc
      set who_links lput count [link-neighbors] of self who_links
    ]
    set i (i + 1)
  ]
end
```

bildungscampus_nordbahnhof.nls

https://gist.github.com/anonymous/aab21cdad86ab39fc82ac80f74488817

```
globals [
  flächen
  höhen
  grundstück
  fächer_list
  stundentafeln
]

to setup-bildungscampus_nordbahnhof
  setup-stundentafeln
  setup-funktionsdiagramm
  ;setup-bauplatz
end

to setup-stundentafeln
  let fächer_KKG (list
    (list 0)
    )

  let fächer_KDG (list
    (list 0)
    )

  let fächer_GTVS (list
    (list "ethik" 2 2 2 2)
    (list "sachunterricht" 3 3 3 3)
    (list "deutsch" 7 7 7 7)
    (list "mathematik" 4 4 4 4)
    (list "musik" 1 1 1 1)
    (list "be" 1 1 1 1)
    (list "werken" 1 1 2 2)
    (list "sport" 3 3 2 2)
    (list "essen" 5 5 5 5)
    )

  let fächer_SOPÄD (list
    (list 0)
    )

  let fächer_GTNMS (list
    (list "ethik" 2 2 2 2)
    (list "deutsch" 4 4 4 4)
    (list "fremdsprache" 4 4 3 3)
    (list "geschichte" 0 2 2 2)
    (list "geographie" 2 1 2 2)
    (list "mathematik" 4 4 4 3)
    (list "ernaehrung" 0 1 0 0)
    (list "berufsorientierung" 0 0 1 0)
    (list "biologie" 2 2 2 2)
    (list "chemie" 0 0 0 2)
    (list "physik" 0 1 2 2)
    (list "musik" 2 2 2 1)
    (list "be" 2 2 2 2)
    (list "werken" 2 2 2 2)
    (list "sport" 4 3 3 4)
    (list "essen" 5 5 5 5)
    )


  let fächer_FMS (list
    (list 0)
    )

  let fächer_MS (list
    (list "ethik" 2 2 2 2)
    (list "deutsch" 4 4 4 4)
    (list "fremdsprache" 4 4 3 3)
    (list "geschichte" 0 2 2 2)
    (list "geographie" 2 1 2 2)
    (list "mathematik" 4 4 4 3)
    (list "ernaehrung" 0 1 0 0)
    (list "berufsorientierung" 0 0 1 0)
    (list "biologie" 2 2 1 1)
    (list "chemie" 0 0 0 2)
    (list "physik" 0 1 2 2)
    (list "musik" 2 2 3 3)
    (list "be" 2 2 2 2)
    (list "werken" 2 2 2 2)
    (list "sport" 4 3 3 3)
    (list "essen" 5 5 5 5)
    )

  let stundentafel_alt (list fächer_KKG fächer_KDG fächer_GTVS fächer_SOPÄD
fächer_GTNMS fächer_FMS fächer_MS)
  set stundentafeln []
  let i 0
```

```
  let j 0
  let k 0
  while [k < length stundentafel_alt] [
    let stundentafel_j []
    while [j < (length item i item k stundentafel_alt - 1)] [
      let item_i []
      while [i < length item k stundentafel_alt] [
        set item_i lput item 0 item i item k stundentafel_alt item_i
        set item_i lput item (j + 1) item i item k stundentafel_alt item_i
        set i (i + 1)
      ]

      ; fächer mit 0 stunden löschen
      let l 1
      while [l < length item_i] [
        ifelse item l item_i = 0 [
          repeat 2 [
            set item_i remove-item (l - 1) item_i
          ]
        ][
        set l (l + 2)
        ]
      ]

      set stundentafel_j lput item_i stundentafel_j
      set i 0
      set j (j + 1)
    ]
    set stundentafeln lput stundentafel_j stundentafeln
    set j 0
    set k (k + 1)
  ]
end

to setup-funktionsdiagramm
  set flächen (list
    (list 892 1619 283 7200 4901 824) ; bereiche_flächen
    (list 1117 1117 1121 1121 1115 622 896 896 914 914 500)  ; bibere_flächen
    (list 63 63 63 649) ; küchen_flächen
    (list 267 1185 300) ; kreativbereiche_flächen
    (list 168 96) ; koll. führung_flächen
    )
  set höhen (list
    (list 5.5 7.5 4 3 3 3)
    (list 3 3 3 3 3 3 3 3 3 3 3)
    (list 3 3 3 3)
    (list 3 3 3)
    (list 3 3)
    )
    let i 0
  while [i < length flächen] [
    set flächen replace-item i flächen (map [? / global_scale] item i flächen)
    set i (i + 1)
  ]
end

to setup-bauplatz
  setup-grundstück
end

to setup-grundstück
  ; grundstücksgrenzen
  let seitenlängen (list 187.34 143.85 82.64 142.9 58.15)
  set seitenlängen map [? / (sqrt global_scale)] seitenlängen
  ask patches [
    let pxcor_pos pxcor
    let pycor_pos pycor
    if pxcor_pos < 0 [set pxcor_pos pxcor * -1]
    if pycor_pos < 0 [set pycor_pos pycor * -1]
    if pxcor_pos <= (item 0 seitenlängen) / 2 and pycor_pos <= (item 1
seitenlängen) / 2 and pzcor = min-pzcor [
      set pcolor black
    ]
  ]

  create-turtles 1 [
    setxyz (- item 0 seitenlängen / 2) (- item 1 seitenlängen / 2) (min-pzcor)
    set heading 90
    set size 5
    set color white
  ]

  ask turtle 0 [
    let i 0
    set pcolor blue
    while [i < item 0 seitenlängen] [
      forward 1
      set pcolor blue
      set i (i + 1)
    ]
    left 90
    set i 0
```

```
  while [i < item 2 seitenlängen] [
    forward 1
    set pcolor blue
    set i (i + 1)
  ]
  left 65
  set i 0
  while [i < item 3 seitenlängen] [
    forward 1
    set pcolor blue
    set i (i + 1)
  ]
  left 25
  set i 0
  while [i < item 4 seitenlängen] [
    forward 1
    set pcolor blue
    set i (i + 1)
  ]
  left 90
  set i 0
  while [i < item 1 seitenlängen] [
    forward 1
    set pcolor blue
    set i (i + 1)
  ]
   ; grundstücks ausfüllen
  back 1
  left 90
  fd 1
  einfärben

  let höchste_ycor sort-on [pycor] patches with [pcolor = blue]
  set höchste_ycor reverse höchste_ycor

  set i 0
  while [ycor < ([pycor] of item 0 höchste_ycor - 1)] [
    left 90
    fd 1
    left 90
    fd 1
    while [pcolor = blue] [
      fd 1
      set i ( i + 1)
    ]
    if i > 0 [
      back 1
      set pcolor [255 255 255 20]
      fd 1
    ]
    set i 0
    einfärben

    right 90
    fd 1
    right 90
    fd 1
    while [pcolor = blue] [
      fd 1
      set i (i + 1)
    ]
    if i > 0 [
      back 1
      set pcolor [255 255 255 20]
      fd 1
    ]
    set i 0
    einfärben
   ]
  ]
  ct
   ask patches with [pcolor = blue] [
    set pcolor 0
  ]
end

to einfärben
  while [pcolor != blue] [
    set pcolor [255 255 255 20]
    forward 1
  ]
end

to setup-gebäudehöhe
  ask patches with [pcolor != 0] [
    ask patches with [pxcor = [pxcor] of myself and pycor = [pycor] of myself
and pzcor < (min-pzcor + 23)] [
      set pcolor [pcolor] of myself
    ]
   display
  ]
end
```

statistics.nls

https://gist.github.com/anonymous/4ca6faf874d9c3064288ba016d3ee2b3

```
extensions [csv]

globals [
  birthrate      ;relative birth in percent
  mortalityrate  ;relative mortality in percent
  migration      ;absolute migration in persons
  schultypen_wien
  schulbesuche_wien   ;absolute schulbesuche, in persons
  schulbesuche_wien2   ;absolute schulbesuche, in persons
  factors
]


to setup-statistics
  setup-schulbesuche
  calculate-schulbesuche
  calculate-classes-2030
end

to read-csv
  let     files     (list     "schülerinnen_2009.csv"     "schülerinnen_2020.csv"
"schülerinnen_2030.csv")
  let i 0
  while [i < length files] [
    file-open item i files
    let result []
    while[not file-at-end?] [
      let row csv:from-row file-read-line
      set result lput row result
    ]
    file-close
    set i (i + 1)
    show result
  ]

end

to setup-schulbesuche
  set schultypen_wien (list
    "Insgesamt"
    "Volksschulen"
    "Hauptschulen"
    "Sonderschulen"
    "Polytechn Schulen"
    "AHS-Unterstufe"
    "AHS-Oberstufe"
    "Berufsschulen"
    "BMS"
    "BHS"
    "LHS"
    )

  set schulbesuche_wien (list
    ;Prognose der Schülerinnen- und Schülerzahl für die Schuljahre 2020/21 und
2030/31 nach Politischen Bezirken
    (list  ;2009-10
      213148
      62475
      30151
      2879
      2911
      33921
      24742
      23786
      7277
      22636
      2370
      )

    (list ;2020-21
      226345
      69329
      31316
      3056
      2865
      36775
      25916
      23727
      7053
      23678
      2630
      )

    (list ;2030-31
      247066
      74992
```

```
      33277
      3412
      2942
      41330
      29039
      25153
      7022
      26733
      3166
      )
    )

    set schulbesuche_wien2 (list
    ;Prognose der Schülerinnen- und Schülerzahl für die Schuljahre 2020/21 und
2030/31 nach Politischen Bezirken
    (list  ;2009-10
      8514
      3479
      2007
      246
      0
      1458
      920
      0
      258
      146
      0
      )

    (list ;2020-21
      9313
      3878
      2179
      256
      0
      1618
      995
      0
      241
      146
      0
      )

    (list ;2030-31
      10043
      4150
      2311
      279
      0
      1809
      1124
      0
      217
      153
      0
      )
    )
end

to calculate-schulbesuche
  let i 0
  let j 0
  let k 0
  let schulbesuche_wien_neu []
  while [k < 2] [
    while [j < 10] [
      let jahr []
      while [i < length item k schulbesuche_wien2] [
        set jahr lput round (item i item k schulbesuche_wien2 + (((item i item
(k + 1) schulbesuche_wien2) - (item i item k schulbesuche_wien2)) / 11 * (j +
1))) jahr
        set i (i + 1)
      ]
      set schulbesuche_wien_neu lput jahr schulbesuche_wien_neu
      set i 0
      set j (j + 1)
    ]
    set j 0
    ifelse k = 0 [
      set     schulbesuche_wien_neu     fput     item     k     schulbesuche_wien2
schulbesuche_wien_neu
      set   schulbesuche_wien_neu   lput   item   (k   +   1)   schulbesuche_wien2
schulbesuche_wien_neu
    ][
    set     schulbesuche_wien_neu     remove     last     schulbesuche_wien_neu
schulbesuche_wien_neu
    ]
    set k (k + 1)
  ]
  set     schulbesuche_wien_neu     lput     item     k     schulbesuche_wien2
schulbesuche_wien_neu
  set schulbesuche_wien2 schulbesuche_wien_neu
```

```
end

to calculate-classes-2030
  let classes_pro_schultyp_2016_list (list (list 22 2 20 2 12) (list 22) (list
20 12) (list 2) 0 0 0 0 (list 2) 0 0)
  let classes_pro_schultyp_2016 (list 58 22 32 2 0 0 0 0 2 0 0)
  let students_pro_schultyp_2016 map [? * 25] classes_pro_schultyp_2016

  ; welchen anteil aller schüler des 2. bezirks nimmt der neue bildungscampus
im jahr 2016 auf?
  let i 0
  set factors []
  while [i < length item 7 schulbesuche_wien2] [
    ifelse item i students_pro_schultyp_2016 != 0 [
      set factors lput (precision ((item i item 7 schulbesuche_wien2) / (item i
students_pro_schultyp_2016)) 1) factors
      set schultypen_wien replace-item i schultypen_wien (sentence item i
schultypen_wien item i classes_pro_schultyp_2016 (precision ((item i item 7
schulbesuche_wien2) / (item i students_pro_schultyp_2016)) 1))
    ][
    set factors lput 0 factors
    set schultypen_wien replace-item i schultypen_wien (sentence item i
schultypen_wien 0)
    ]
    set i (i + 1)
  ]

  ; wie viele klassen müsste der bildungscampus bei gleichem anteil im jahr
2030 aufnehmen?
  set i 0
  let anzahlen_students_2030 []
  while [i < length last schulbesuche_wien2] [
    ifelse item i students_pro_schultyp_2016 != 0 [
      set anzahlen_students_2030 lput round ((item i last schulbesuche_wien2) /
(item i factors)) anzahlen_students_2030
    ][
    set anzahlen_students_2030 lput 0 anzahlen_students_2030
    ]
    set i (i + 1)
  ]
  let classes_pro_schultyp_2030 map [? / 25] anzahlen_students_2030

  show students_pro_schultyp_2016
  show anzahlen_students_2030

  set i 0
  while [i < length schultypen_wien] [
    ifelse item i classes_pro_schultyp_2030 != 0 [
      set schultypen_wien replace-item i schultypen_wien (sentence item i
schultypen_wien item i classes_pro_schultyp_2030)
    ][
    set schultypen_wien replace-item i schultypen_wien (sentence item i
schultypen_wien 0)
    ]
    set i (i + 1)
  ]

  ;show classes_pro_schultyp_2016
  ;show classes_pro_schultyp_2030

  ; classes_pro_schultyp_2030 auf wettbewerb zurückrechnen
  set i 0
  let classes_pro_schultyp_2030_schultypen classes_pro_schultyp_2030
  while [i < length classes_pro_schultyp_2016_list] [
    let j 0
    let item_i [] ; item i classes_pro_schultyp_2030_list
    if item i classes_pro_schultyp_2016_list != 0 [
      while [j < length item i classes_pro_schultyp_2016_list] [
        let factor (item j item i classes_pro_schultyp_2016_list) / (sum item i
classes_pro_schultyp_2016_list)
        set item_i lput (precision (item i classes_pro_schultyp_2030 * factor)
1) item_i
        set j (j + 1)
      ]
    ]
    set classes_pro_schultyp_2030_schultypen replace-item i
classes_pro_schultyp_2030_schultypen item_i
    set i (i + 1)
  ]
  show classes_pro_schultyp_2016_list
  show classes_pro_schultyp_2030_schultypen
end
```

export2csv.nls

https://gist.github.com/anonymous/249d180a74114afeb5a60bc5a0adf816

```
to export2csv
  file-close-all
  let filename "export-to-csv_neu.csv"
  ;let headers ["Bereich" "Koordinaten" "Fläche" "Color"]

  if file-exists? filename [file-delete filename]
  file-open filename

  ;print-row headers

  ;; output grundstück
  ;print-row bauplatz
  let i 0
  let j 0
  let räume (list bereiche unterbereiche with [art = "biber"] unterbereiche
with [art = "küche"] unterbereiche with [art = "kreativbereich"] unterbereiche
with [art = "koll. führung"])
  while [j < length räume] [
    let räume_sort sort-on [who] item j räume
    while [i < count item j räume] [
      ask item i räume_sort [
        let raum []
        ;; output funktion
        ifelse j = 0 [
          set raum lput funktion raum
        ][
        set raum lput (word art #raum) raum
        ]
        ;; output xyz
        let x xcor
        let y ycor
        let z zcor
        let xyz (sentence x y z)
        set raum lput xyz raum

        ;; output fläche & höhe
        set raum lput (sentence (fläche * global_scale) höhe) raum

        ;; output color
        set raum lput color raum

        ;; output mein_bereich
        set raum lput [who] of mein_bereich raum

        ;; output link partner
        ;let link_partner []
        let link_partner [who] of link-neighbors
        set raum lput link_partner raum

        ;; output through-put
        let through-puts []
        if length link_partner > 0 [
          let k 0
          while [k < length link_partner] [
            set through-puts lput [through-put] of link (who) (item k
link_partner) through-puts
            set k (k + 1)
          ]
        ]
        set raum lput through-puts raum

        ; output wcc
        set raum lput weighted_centrality raum

        ; output global_wcc
        set raum lput global_wcc raum

        show raum
        print-row-csv raum
      ]
      set i (i + 1)
    ]
    set i 0
    set j (j + 1)
  ]

  let output []
  ;; output global_scale
  set output lput global_scale output

  ;; output grundstück
  let grundstück_ []
  ask patches with [pcolor != 0] [
    set grundstück_ lput (sentence pxcor pycor pzcor) grundstück_
  ]
  set output lput grundstück_ output
  print-row-csv output
```

```
  file-close
end

to print-row-csv [ #items ]
   ;; write all but the last item to theif   currently open file, inserting
commas
   ;; uses file-write to produce output, strings are quoted.
   foreach but-last #items
     [ file-write ?
       ;file-type ","
     ]
   ;; write the last item
   file-write last #items
   ;; write end-of-line
   file-write "end"
   file-print ""
end
```

7.3.2.2. Verteilalgorithmus

https://gist.github.com/anonymous/13e4df915e1374bae42718a6a3c22875

```
extensions [nw]

__includes ["import-from-csv.nls" "export-to-csv.nls"]

globals [
  grundstücksfläche
  geschoßfläche
  anzahl_geschoße
  geschoß_patches
  auslastung_geschoße
  geschoße_sort
  geschoße_boolean
  hauptgeschoß_sort
  sum_wcc
  bereiche/geschoß
  bereiche/erschließungen
  bereich_verschoben
  neighbors_sort
  verhältnis
  counter
  erschließungen_sort
  stop?
  übereiche_wcc
  ubereiche_wcc
  ]

breed [geschoße geschoß]
breed [haupteingänge haupteingang]
breed [erschließungen erschließung]

patches-own [zahl bereich_here neue_erschließung?]
geschoße-own   [#geschoß   fläche   fläche_belegt   fläche_rest   prozent_belegt
prozent_farben meine_bereiche]
erschließungen-own    [#erschließung    #geschoß    meine_bereiche    anzahl_ubs
auslastung]
to setup
  ca
  if seed != 0 [
    random-seed seed
  ]
  import-from-csv
  import-durchschnittwerte
  setup-globals
  setup-patches
  setup-geschoße
  setup-observer
  reset-ticks

  ask turtles with [breed != geschoße] [
    set hidden? true
  ]
  ask links [
    set hidden? true
  ]
end

to reset
  ct
  cp
  if seed != 0 [
    random-seed seed
  ]
  import-from-csv
  ; setup-globals
  ;  verkleinerung  der  geschoßfläche  abhängig  von  auslastung  des  obersten
geschoßes und anzahl_geschoße
  ifelse  last  auslastung_geschoße  =  0  or  (precision  ((10  /  (last
auslastung_geschoße    *    .05))    *    (5    /    anzahl_geschoße))    1)    >
geschoßfläche_reduzieren%  [
    set geschoßfläche round (geschoßfläche * ((100 - geschoßfläche_reduzieren%)
* 0.01))
    show (word "verkleinerung der geschoßfläche um " geschoßfläche_reduzieren%
"%")
  ][
  set   geschoßfläche   round   (geschoßfläche   *   (100   -   ((10   /   (last
auslastung_geschoße * .05)) * (5 / anzahl_geschoße))) * .01)
  show (word "verkleinerung der geschoßfläche um " (precision ((10 / (last
auslastung_geschoße * .05)) * (5 / anzahl_geschoße)) 1) "%")
  ]

  set bereich_verschoben []
  set neighbors_sort []
  set auslastung_geschoße []
  repeat anzahl_geschoße [
    set bereich_verschoben lput 0 bereich_verschoben
    set auslastung_geschoße lput 0 auslastung_geschoße
    ;set neighbors_sort lput 0 neighbors_sort
```

```
  ]
  setup-patches
  setup-geschoße
  setup-observer
  ask patches with [zahl != 0 and pcolor != white] [
    set pcolor white
  ]
  reset-ticks
    ask turtles with [breed != geschoße] [
    set hidden? true
  ]

  ask links [
    set hidden? true
  ]
end

to import-durchschnittwerte
  let durchschnittswerte [
    [96.712  96.712]
    [96.132  96.132]
    [0  0]
    [76.34  76.34]
    [81.35  81.35]
    [73.636  73.636]
    [48.595  62.646]
    [48.467  62.492]
    [44.764  57.718]
    [43.17  55.691]
    [43.352  55.899]
    [0  0]
    [66.618  73.289]
    [66.646  73.328]
    [66.683  73.375]
    [66.534  73.178]
    [72.567  94.326]
    [0  0]
    [0  0]
    [0  0]
    [92.015  99.637]
    [77.706  100]
    [85.96  94.58]
    [71.41  92.849]
    [0  0]
    [0  0]
  ]
  let turtles_sort sort-on [who] turtles
  let i 0
  while [i < length turtles_sort] [
    ask item i turtles_sort [
      set global_wcc item 0 item i durchschnittswerte
      set local_wcc item 1 item i durchschnittswerte
    ]
    set i (i + 1)
  ]

  set übereiche_wcc reverse sort-on [global_wcc] bereiche
  set i 0
  while [i < length übereiche_wcc] [
    set übereiche_wcc replace-item i übereiche_wcc (sentence [funktion] of item
i übereiche_wcc [global_wcc] of item i übereiche_wcc)
    set i (i + 1)
  ]

  set ubereiche_wcc []
  let j 0
  while [j < count bereiche] [
    let ubereiche_wcc_üb reverse sort-on [local_wcc] unterbereiche with
[mein_bereich = bereich j]
    set i 0
    while [i < length ubereiche_wcc_üb] [
      set ubereiche_wcc_üb replace-item i ubereiche_wcc_üb (sentence [funktion]
of item i ubereiche_wcc_üb [local_wcc] of item i ubereiche_wcc_üb)
      set i (i + 1)
    ]
     set ubereiche_wcc lput ubereiche_wcc_üb ubereiche_wcc
    set j (j + 1)
  ]
end

to setup-globals
  ; grundstücksfläche 22990 m² (S.21)
  ; bebaute fläche max. 10500 m² (S.21)
  ; zulässige gebäudehöhe (5 geschoße) soll ausgenützt werden, oberstes
fluchtniveau unter 22m (S. 27)
  set grundstücksfläche 22990
  set geschoßfläche 10500
  if geschoßfläche_override != 0 [
    set geschoßfläche geschoßfläche_override
  ]
  set anzahl_geschoße geschoße_anzahl
  set geschoße_sort []
```

```
  set auslastung_geschoße []
  set bereiche/geschoß []
  set bereich_verschoben []
  set neighbors_sort []
  set geschoße_boolean []
  set hauptgeschoß_sort []
  repeat anzahl_geschoße [
    set bereich_verschoben lput 0 bereich_verschoben
    ;set neighbors_sort lput 0 neighbors_sort
    set geschoße_boolean lput false geschoße_boolean
  ]
  repeat anzahl_geschoße [
    set auslastung_geschoße lput 0 auslastung_geschoße
    set bereiche/geschoß lput 0 bereiche/geschoß
  ]

  if global_scale_neu != 0 [
    set global_scale global_scale_neu
  ]
end

to setup-patches
  ; setup grundstücksfläche
  let seitenlänge ((sqrt grundstücksfläche) / global_scale)
  let seitenlänge_bebaut ((sqrt geschoßfläche) / global_scale)
  ask patches with [pzcor = min-pzcor and (pxcor > (- seitenlänge / 2)) and
(pxcor < (seitenlänge / 2)) and (pycor > (- seitenlänge / 2)) and (pycor <
(seitenlänge / 2))] [
    ifelse (pxcor > (- seitenlänge_bebaut / 2)) and (pxcor <
(seitenlänge_bebaut / 2)) and (pycor > (- seitenlänge_bebaut / 2)) and (pycor <
(seitenlänge_bebaut / 2)) [
      set pcolor grey
    ][
    set pcolor green
    ]
  ]

  ; setup bauvolumen
  set seitenlänge ((sqrt geschoßfläche) / global_scale)
  let i 0
  while [i < anzahl_geschoße] [
    ask patches with [(pzcor = round (min-pzcor + ((i + 1) * (abstand_geschoße
/ (anzahl_geschoße / 5))))) and (pxcor > (- seitenlänge / 2)) and (pxcor <
(seitenlänge / 2)) and (pycor > (- seitenlänge / 2)) and (pycor < (seitenlänge
/ 2))] [
      set pcolor white
      set zahl (i + 1)
    ]
    set i (i + 1)
  ]

  ; patches nach geschoßen & koordinaten in liste speichern
  set geschoß_patches []
  set i 0
  while [i < anzahl_geschoße] [
    let patches_y remove-duplicates sort [pycor] of patches with [zahl = (i +
1)]
    let patches_x 0
    let patches_xy []
    let j 0
    while [j < length patches_y] [
      set patches_x sort-on [pxcor] patches with [zahl = (i + 1) and pycor =
item j patches_y]

      let k 0
      while [k < length patches_x] [
        set patches_xy lput item k patches_x patches_xy
        set k (k + 1)
      ]

      set j (j + 1)
    ]
    set geschoß_patches lput patches_xy geschoß_patches
    set j 0
    set i (i + 1)
  ]
  ;show geschoß_patches
end

to geschoß_patches_diagonal
  ; patches nach geschoßen & koordinaten in liste speichern
  let i 0
  while [i < anzahl_geschoße] [
    let patches_i sort-on [pxcor + pycor] patches with [zahl = (i + 1)]
    set geschoß_patches lput patches_i geschoß_patches
    set i (i + 1)
  ]
end

to setup-geschoße
  let seitenlänge ((sqrt geschoßfläche) / global_scale)
  let i 0
```

```
  while [i < anzahl_geschoße] [
    create-geschoße 1 [
      setxyz  0  0  round  (min-pzcor  +  (i  +  1)  *  (abstand_geschoße  /
(anzahl_geschoße / 5)))
      set shape "circle"
      set size 2.5
      set color red
      set #geschoß i
      set fläche geschoßfläche
      set fläche_belegt 0
      set fläche_rest fläche
    ]
    set i (i + 1)
  ]

   set geschoße_sort sort-on [zahl] geschoße
end

to setup-observer
  if ceiling __oxcor = 0 and ceiling __oycor = 0 and ceiling __ozcor = 162 [
    setxyz min-pxcor max-pycor (max-pzcor - 30)
    zoom (max-pxcor * -.3)
    orbit-right 10
    facexyz (mean [pxcor] of geschoße) (mean [pycor] of geschoße) (mean [pzcor]
of geschoße - 4.5 / (anzahl_geschoße / 6))
  ]
end

to go
  ; veranstaltungssaal auf eg
  ifelse ticks = 0 [
    ask bereich 2 [
      set mein_geschoß (first geschoße_sort)
      setxyz  ([xcor]  of  mein_geschoß)  ([ycor]  of  mein_geschoß)  ([zcor]  of
mein_geschoß)
      set hidden? false
    ]
  ][
  ifelse neighbors_sort = [] [
    ifelse count bereiche with [mein_geschoß = 0] = 0 [
    ][
    bereich-auf-neues-geschoß
    ]
  ][
  unterbereiche-anordnen
  ]
  if neighbors_sort != [] [
    hauptgeschoß-herausfinden
  ]
  ]

  bereiche-auf-geschoß-verteilen
  update-geschoße
  if patches_einfärben? = true [
    update-patches
  ]

  tick

  let auslastung_sort sort auslastung_geschoße
  if count bereiche with [mein_geschoß = 0] = 0 and last auslastung_sort <= 100
- restfläche% [
    ifelse restart? = true and last auslastung_geschoße < ((100 - restfläche%)
- 10) [
      reset
      go
    ][
    if patches_einfärben? = false [
      update-patches
    ]
    ;setup-räumliche-anordnung
    set stop? true
    stop
    ]
  ]
end

to bereich-auf-neues-geschoß
  ; einen bereich (nach wcc sortiert) eine nach oben/unten verschieben
  ; bereiche auf geschoß nach wcc sortieren
  let bereiche_sort reverse sort-on [weighted_centrality] turtles with [breed =
bereiche and mein_geschoß = 0]

  ; einen bereich auf ein neues geschoß verschieben
  ask first bereiche_sort [
    ; bereich auf oberstes geschoß verschieben, auf dem etwas frei ist
    ifelse (item 1 auslastung_geschoße = 0) [
      set mein_geschoß (first geschoße_sort)
    ][
    let i 0
    while [i < length auslastung_geschoße] [
```

```
      if (item i reverse auslastung_geschoße) > 0 and (item i reverse
auslastung_geschoße) <= (100 - restfläche%) and (item (i + 1) reverse
auslastung_geschoße) > 0 [
        set mein_geschoß (item i reverse geschoße_sort)
        set i length auslastung_geschoße
      ]
      set i (i + 1)
    ]
    ]

    setxyz ([xcor] of mein_geschoß) ([ycor] of mein_geschoß) ([zcor] of
mein_geschoß)
    set bereich_verschoben self

    if who < 3 [
      set hidden? false
    ]

    ; unterbereiche des verschobenen bereichs nach wcc sortieren
    set neighbors_sort (sort-on [weighted_centrality] unterbereiche with
[mein_bereich = myself])
  ]

  ; alle unterbereiche mitverschieben
  ask unterbereiche with [mein_bereich = bereich_verschoben] [
    set mein_geschoß [mein_geschoß] of bereich_verschoben
    setxyz ([xcor] of mein_geschoß) ([ycor] of mein_geschoß) ([zcor] of
mein_geschoß)
    set hidden? false
  ]
end

to unterbereiche-anordnen
  ; wenn mein_geschoß das unterste ist
  ifelse [mein_geschoß] of bereich_verschoben = first geschoße_sort [
    ; wenn mein_geschoß voll ist, unterbereich nach oben schieben, sonst
nächsten bereich verschieben
    ifelse [prozent_belegt] of [mein_geschoß] of bereich_verschoben > (100 -
restfläche%) [
      ; unterbereich auf +1
      ask first neighbors_sort [
        set mein_geschoß geschoß ([who] of mein_geschoß + 1)
      ]
      ; unterbereich aus neighbors_sort löschen
      set neighbors_sort remove-item 0 neighbors_sort
    ][
    ; nächsten bereich verschieben
    set neighbors_sort []
    go
    ]

  ][

  ; wenn mein_geschoß nicht das unterste ist
  ; wenn geschoß -1 (inkl. first neighbors_sort) voll ist, testen ob geschoß 0
ausgelastet ist
  ifelse round ((([fläche_belegt] of geschoß ([who] of [mein_geschoß] of
bereich_verschoben - 1)) + ([fläche] of first neighbors_sort)) / ([fläche] of
geschoß ([who] of [mein_geschoß] of bereich_verschoben - 1) / 100)) < (100 -
restfläche%) [
    ; unterbereich auf -1
    ask first neighbors_sort [
      set mein_geschoß geschoß ([who] of mein_geschoß - 1)
    ]
    ; unterbereich aus neighbors_sort löschen
    set neighbors_sort remove-item 0 neighbors_sort
  ][
  ; wenn mein_geschoß voll ist, unterbereich nach unten/oben schieben
  ifelse [prozent_belegt] of [mein_geschoß] of bereich_verschoben > (100 -
restfläche%) [

    ; wenn geschoß +1 voll ist, dieses zu neuem hauptgeschoß machen
    if geschoß ([who] of [mein_geschoß] of bereich_verschoben + 1) = nobody [
      user-message ("geschoßfläche zu klein")
    ]
    ifelse round ((([fläche_belegt] of geschoß ([who] of [mein_geschoß] of
bereich_verschoben + 1)) + ([fläche] of first neighbors_sort)) / ([fläche] of
geschoß ([who] of [mein_geschoß] of bereich_verschoben + 1) / 100)) > (100 -
restfläche%) [
      bereich-auf-neues-hauptgeschoß geschoß ([who] of [mein_geschoß] of
bereich_verschoben + 1)
    ][

    ; unterbereich auf +1
    ask first neighbors_sort [
      set mein_geschoß geschoß ([who] of mein_geschoß + 1)
    ]
    ; unterbereich aus neighbors_sort löschen
    set neighbors_sort remove-item 0 neighbors_sort
    ]

  ][
```

```
  ; nächsten bereich verschieben
  set neighbors_sort []
  go
  ]
  ]

  ]
end

to hauptgeschoß-herausfinden
  ; auf welchen geschoßen befinden sich unterbereiche des bereichs?
  set hauptgeschoß_sort sort remove-duplicates [mein_geschoß] of unterbereiche
with [mein_bereich = bereich_verschoben]

  ; wcc von unterbereichen auf gleichen geschoßen summieren
  let j 0
  while [j < length hauptgeschoß_sort] [
    let geschoß_wcc []
    set geschoß_wcc lput item j hauptgeschoß_sort geschoß_wcc
    set geschoß_wcc lput round sum [weighted_centrality] of unterbereiche with
[mein_bereich = bereich_verschoben and mein_geschoß = [item j
hauptgeschoß_sort] of bereich_verschoben] geschoß_wcc
    set hauptgeschoß_sort replace-item j hauptgeschoß_sort geschoß_wcc
    set j (j + 1)
  ]

  ; hauptgeschoß_sort nach item 1 sortieren
  set hauptgeschoß_sort sort-with task [ item 1 ? ] hauptgeschoß_sort

  ; befindet sich bereich auf hauptgeschoß?
  if first first hauptgeschoß_sort != [mein_geschoß] of bereich_verschoben [
    bereich-auf-neues-hauptgeschoß first first hauptgeschoß_sort
  ]

end

to bereich-auf-neues-hauptgeschoß [hauptgeschoß]
  ; bereich_verschoben auf neues hauptgeschoß
  ask bereich_verschoben [
    set mein_geschoß hauptgeschoß
    setxyz ([xcor] of mein_geschoß) ([ycor] of mein_geschoß) ([zcor] of
mein_geschoß)

    ; link_neighbors des verschobenen bereichs nach wcc sortieren
    set neighbors_sort (sort-on [weighted_centrality] unterbereiche with
[mein_bereich = myself])
  ]

  ; unterbereiche auch auf neues hauptgeschoß
  ask unterbereiche with [mein_bereich = bereich_verschoben] [
    set mein_geschoß [mein_geschoß] of bereich_verschoben
    setxyz ([xcor] of mein_geschoß) ([ycor] of mein_geschoß) ([zcor] of
mein_geschoß)
  ]
end

to bereiche-auf-geschoß-verteilen
  let i 0
  while [i < anzahl_geschoße] [
    let j 0
    let anzahl (count turtles with [breed != geschoße and hidden? = false and
mein_geschoß = one-of geschoße with [#geschoß = i]])
    let bereiche_sort sort turtles with [breed != geschoße and hidden? = false
and mein_geschoß = one-of geschoße with [#geschoß = i]]
    while [j < length bereiche_sort] [
      ask item j bereiche_sort [
        setxyz ([xcor] of mein_geschoß) ([ycor] of mein_geschoß) ([zcor] of
mein_geschoß)
        set heading 0 + (j * 360 / anzahl)
        fd 15
      ]
      set j (j + 1)
    ]
    set i (i + 1)
  ]
end

to update-geschoße
  ; prozent_belegt von geschoßen berechnen
  let geschoße_sort2 sort-on [#geschoß] geschoße
  let i 0
  while [i < length geschoße_sort2] [
    ask item i geschoße_sort2[
      set fläche_belegt (sum [fläche] of turtles with [breed != geschoße and
hidden? = false and mein_geschoß = myself])
      set fläche_rest (fläche - fläche_belegt)
      set color scale-color red fläche_belegt geschoßfläche 0
      set prozent_belegt round (fläche_belegt / (fläche / 100))
      set label prozent_belegt
      set meine_bereiche sort turtles with [breed != geschoße and hidden? =
false and mein_geschoß = myself]
```

```
      ; auslastung_geschoße updaten
      set auslastung_geschoße replace-item i auslastung_geschoße prozent_belegt

      ; bereiche/geschoß updaten
      set bereiche/geschoß replace-item i bereiche/geschoß length
meine_bereiche
    ]
    set i (i + 1)
  ]
end

to update-patches
  ask patches with [zahl != 0 and pcolor != white] [
    set pcolor white
  ]

  ; welche farbe nimmt wieviel fläche ein
  let farben_flächen []
  let farben_flächen_prozent []
  let i 0
  let j 0
  while [i < anzahl_geschoße] [
    let farben_geschoß remove-duplicates [color] of turtles with [breed !=
geschoße and hidden? = false and mein_geschoß = one-of geschoße with [#geschoß
= i]]
    set j 0
    while [j < length farben_geschoß] [
      let farbe_fläche_geschoß []
      set farbe_fläche_geschoß lput item j farben_geschoß farbe_fläche_geschoß
      set farbe_fläche_geschoß lput sum [fläche] of turtles with [breed !=
geschoße and hidden? = false and mein_geschoß = one-of geschoße with [#geschoß
= i] and color = item j farben_geschoß] farbe_fläche_geschoß
      set farben_geschoß replace-item j farben_geschoß farbe_fläche_geschoß
      set j (j + 1)
    ]

    ; farben_geschoß nach item 1 sortieren
    set farben_geschoß sort-with task [ item 1 ? ] farben_geschoß

    set farben_flächen lput farben_geschoß farben_flächen

    ; welche farbe nimmt wieviel prozent des geschoßes ein
    set j 0
    let farbe_fläche_geschoß []
    while [j < length item i farben_flächen][
      let farbe_fläche (item j item i farben_flächen)
      set farbe_fläche replace-item 1 farbe_fläche (precision (item 1
farbe_fläche / ([fläche] of one-of geschoße with [#geschoß = i] / 100)) 1)
      set farbe_fläche_geschoß lput farbe_fläche farbe_fläche_geschoß
      set j (j + 1)
    ]
    set farben_flächen_prozent lput farbe_fläche_geschoß farben_flächen_prozent

    ; patches verhältnismäßig nach prozent_belegt einfärben
    let k 0
    while [k < length item i farben_flächen_prozent] [
      let patches_belegt (count patches with [zahl = (i + 1)] * (item 1 item k
item i farben_flächen_prozent * 0.01))
      set j 0
      let l 0
      while [j < length item i geschoß_patches] [
        ask item j item i geschoß_patches [
          ifelse pcolor = white [
            if l < patches_belegt [
              let farbe remove-item 3 item 0 item k item i
farben_flächen_prozent
              set pcolor farbe
            ]
            set l (l + 1)
          ][
          ]
          set j (j + 1)
        ]
      ]
      set k (k + 1)
    ]

    set i (i + 1)
  ]

  ask geschoße [
    set prozent_farben item #geschoß farben_flächen_prozent
  ]
end

to-report sort-with [ key lst ]
  report sort-by [(runresult key ?1) > (runresult key ?2)] lst
end

to setup-räumliche-anordnung
  setup-haupteingang
  setup-erschließung
```

```
  setup-globals2
  setup-new-links
end

to reset2
  setup-globals2
  ask erschließungen [
    set meine_bereiche []
  ]
end

to setup-globals2
  set counter 0
  set bereiche/erschließungen []
  repeat count erschließungen with [#geschoß = 0] [
    set bereiche/erschließungen lput [] bereiche/erschließungen
  ]
end

to setup-haupteingang
  ask haupteingänge [
    die
  ]

  let patch_haupt one-of patches with [zahl = 1 and count neighbors with [pzcor
= [pzcor] of myself and pcolor = 0] != 0]
  create-haupteingänge 1 [
      setxyz ([pxcor] of patch_haupt) ([pycor] of patch_haupt) ([pzcor] of
patch_haupt)
        set heading 0
        set shape "square"
        set color [0 0 255 150]
        set size 2
  ]
  display
end

to setup-erschließung
  ; verhältnis fläche/patches der geschoße
  set verhältnis (geschoßfläche / count patches with [zahl = 1])

  ask erschließungen [
    die
  ]
  ; wo liegen erschließungskerne? 1 patch = 1m x 1m
  ; patches sortieren?
  ask patches with [zahl = 1] [
    ; wenn die distanz zur nächsten erschließung größer als 40 Meter ist,
erstelle eine neue
    if count patches with [neue_erschließung? = true] = 0 or distance min-one-
of patches with [neue_erschließung? = true] [distance myself] > (40 /
global_scale_neu)[
      set neue_erschließung? true
    ]
  ]

  ; neue erschließungen bauen
  let patches_erschließungen sort patches with [neue_erschließung? = true]
  let i 0
  while [i < length patches_erschließungen] [
    let j 0
    while [j < anzahl_geschoße] [
      create-erschließungen 1 [
        setxyz ([pxcor] of item i patches_erschließungen) ([pycor] of item i
patches_erschließungen) ceiling (([pzcor] of item i patches_erschließungen) +
(j * (abstand_geschoße / (anzahl_geschoße / 5))))
        set heading 0
        set shape "square"
        set color [0 0 0 150]
        set size 2
        set #erschließung i
        set #geschoß j
        set meine_bereiche []
      ]
      set j (j + 1)
    ]
    set j 0
    set i (i + 1)
  ]
end

to setup-new-links
  ; unterbereiche stellen links mit allen bereiche her, die link-partner ihres
bereichs sind
  ask unterbereiche with [link_partner != []] [
    create-links-with [link-neighbors] of mein_bereich [
      set color [255 0 0 50]
    ]
  ]
end

to geschoß-updaten
```

```
  if welches_geschoß? < length geschoße_sort [
    set counter welches_geschoß?
  ]
  ask erschließungen [
    set meine_bereiche []
  ]

  bereiche-an-erschließung counter
  ; bereiche/erschließungen nach jeden geschoß updaten
  update-bereiche/erschließungen counter

  tick
  set counter (counter + 1)

  if counter >= length geschoße_sort [
    set counter 0
    ;update-links
    stop
  ]
end

to bereiche-an-erschließung [i]
  ask patches with [zahl = (counter + 1) and pcolor != white] [
    set pcolor white
  ]

  ; bereiche auf geschoß i sortiert nach absteigender wcc ihres bereich und
innerhalb ihres bereichs
  let bereiche_sort reverse sort-on [weighted_centrality] bereiche
  let bereiche_auf_geschoß []
  let j 0
  while [j < length bereiche_sort] [
    let bereiche_geschoß reverse sort-on [weighted_centrality] turtles with
[breed = bereiche or breed = unterbereiche and hidden? = false and mein_bereich
= item j bereiche_sort and mein_geschoß = item i geschoße_sort]
    set bereiche_auf_geschoß (sentence bereiche_auf_geschoß bereiche_geschoß)
    set j (j + 1)
  ]

  ; bereiche suchen sich der reihe nach einen freien platz auf ihrem geschoß
  set j 0
  while [j < length bereiche_auf_geschoß] [
    ask item j bereiche_auf_geschoß [

      ;erschließungen sortieren
      sort-erschließungen i j

      ; patches sortieren & einfärben
      color-patches i j

      ; erschließung & bereiche/erschließungen updaten
      update-erschließung

      ; bereich auf gescchoß setzen
      move-bereich
    ]
    set j (j + 1)
  ]
end

to sort-erschließungen [i j]
  ; erschließungen sortieren
  set erschließungen_sort []
  ifelse i = 0 [
    ; bei eg abstand vom haupteingang?
    ifelse j = 0 [
      set erschließungen_sort sort-on [distance one-of haupteingänge]
erschließungen with [#geschoß = i]
    ][
    set erschließungen_sort sort-on [auslastung] erschließungen with [#geschoß
= i]
    ]
  ][
  ; erschließungen danach sortieren, wieviele unterbereiche meines bereichs sie
benutzen
  let k 0
  let anzahlen []
  while [k < length bereiche/erschließungen] [
    let erschließung_k filter [[breed] of ? = unterbereiche or [breed] of ? =
bereiche] item k bereiche/erschließungen
    set erschließung_k filter [[mein_bereich] of ? = [mein_bereich] of self]
erschließung_k
    set anzahlen lput length erschließung_k anzahlen
    ask erschließungen with [#erschließung = k and #geschoß = i] [
      set anzahl_ubs length erschließung_k
    ]
    set k (k + 1)
  ]
  show anzahlen

  ;wenn sich gleich viele unterbereiche bei jeder erschließung befinden, nach
auslastung sortieren
```

```
  let anzahlen_ubs remove-duplicates [anzahl_ubs] of erschließungen with
[#geschoß = i]
  ifelse length anzahlen_ubs = 1 [
    set erschließungen_sort sort-on [auslastung] erschließungen with [#geschoß
= i]
  ][
  set erschließungen_sort reverse sort-on [anzahl_ubs] erschließungen with
[#geschoß = i]
  ]
  ]
end

to color-patches [i j]
  ; wieviele patches belegt bereich
  let meine_patches round (fläche / verhältnis)

  ; patches nach abstand von gewählter erschließung sortieren
  let patches_sort sort-on [distance first erschließungen_sort] patches with
[zahl = (i + 1) and pcolor = white]
  ; patches des bereichs einfärben
  let color_bereich remove-item 3 color
  let k 0
  while [k < meine_patches] [
    ask item k patches_sort [
      set pcolor color_bereich
      set bereich_here myself
    ]
    set k (k + 1)
  ]
end

to update-erschließung
  ; erschließung updaten
  ask first erschließungen_sort [
    set auslastung (auslastung + [size] of myself)
    set meine_bereiche lput myself meine_bereiche
  ]
  ; bereich updaten
  set meine_erschließung first erschließungen_sort

  ; bereiche/erschließungen nach jeden bereich/ub updaten
  ;let     bereiche/erschließung     item     ([#erschließung]     of     first
erschließungen_sort) bereiche/erschließungen
  ;set bereiche/erschließung lput self bereiche/erschließung
  ;set   bereiche/erschließungen   replace-item   ([#erschließung]   of   first
erschließungen_sort) bereiche/erschließungen bereiche/erschließung
end

to move-bereich
  ; bereich auf seine patches setzen
  ;setxyz (mean [pxcor] of patches with [bereich_here = myself]) (mean [pycor]
of patches with [bereich_here = myself]) (mean [pzcor] of patches with
[bereich_here = myself])
end

to update-bereiche/erschließungen [i]
  ; bereiche/erschließung updaten
  let erschließung_sort sort erschließungen with [#geschoß = i]
  let j 0
  while [j < length erschließung_sort] [
    let bereiche/erschließung [meine_bereiche] of item j erschließung_sort
    set  bereiche/erschließungen  replace-item  ([#erschließung]  of  item  j
erschließung_sort) bereiche/erschließungen (sentence (item ([#erschließung] of
item j erschließung_sort) bereiche/erschließungen) bereiche/erschließung)
    set j (j + 1)
  ]
end

to bereiche-an-erschließung_40m
  ; erschließungen aus erschließungen_sort löschen, falls meine_patches mehr
als 40m von ihr entfernt wären
  ;let patches_sort sort-on [distance first erschließungen_sort] patches with
[zahl = (i + 1) and pcolor = white]
  ;ask item meine_patches patches_sort [
  ;  while [distance first erschließungen_sort > (40 / global_scale_neu)] [
  ;    set patches_sort remove-item 0 patches_sort
  ;  ]
  ;]
end

to update-links
  ; update distanz
  ask links with [hidden? = false] [
    ; end1 bis erschließung + end2 bis erschließung + höhenüberwindung
    let dist_hor 0
    let geschoße_ends []
    let a end1
    let b end2
    let patch1 patch (mean [pxcor] of patches with [bereich_here = a]) (mean
[pycor] of patches with [bereich_here = a]) (mean [pzcor] of patches with
[bereich_here = a])
```

```
    let patch2 patch (mean [pxcor] of patches with [bereich_here = b]) (mean
[pycor] of patches with [bereich_here = b]) (mean [pzcor] of patches with
[bereich_here = b])
    ask patch1 [
      set dist_hor distance [meine_erschließung] of a
      set geschoße_ends lput [mein_geschoß] of a geschoße_ends
    ]
    ask patch2 [
      set dist_hor (dist_hor + distance [meine_erschließung] of b)
      set geschoße_ends lput [mein_geschoß] of b geschoße_ends
    ]
    let dist_vert ([#geschoß] of first geschoße_ends - [#geschoß] of first
geschoße_ends)
    if dist_vert < 0 [
      set dist_vert (dist_vert * -1)
    ]

    set distanz (dist_hor + (dist_vert * 7 / global_scale_neu)) *
global_scale_neu
  ]

  ask turtles with [breed = bereiche or breed = unterbereiche and link_partner
!= []] [
    show (precision mean [distanz] of links with [end1 = myself or end2 =
myself] 1)
  ]
end

to update-links_bereiche
  ; update distanz
  ask links with [hidden? = false] [
    ; end1 bis erschließung + end2 bis erschließung + höhenüberwindung
    let dist_hor 0
    let geschoße_ends []
    ; mean [px-, py-, pzcor] of patches with [bereich_here = myself]
    ask end1 [
      set dist_hor distance meine_erschließung
      set geschoße_ends lput mein_geschoß geschoße_ends
    ]
    ask end2 [
      set dist_hor (dist_hor + distance meine_erschließung)
      set geschoße_ends lput mein_geschoß geschoße_ends
    ]
    let dist_vert ([#geschoß] of first geschoße_ends - [#geschoß] of first
geschoße_ends)
    if dist_vert < 0 [
      set dist_vert (dist_vert * -1)
    ]

    set distanz (dist_hor + (dist_vert * 7 / global_scale_neu)) *
global_scale_neu
    show distanz
  ]

  ask turtles with [breed = unterbereiche and link_partner != []] [
    show mean [distanz] of links with [end1 = myself or end2 = myself]
  ]
end
```

export-to-csv.nls

https://gist.github.com/anonymous/4cdb8f241ac2b44fc84879a8079484fb

```
to export2csv
  file-close-all
  let filename (word anzahl_geschoße "geschoße_anordnung.csv")
  let headers ["Geschoss" "Funktion" "Breed" "Mein Bereich" "Farbe" "Flaeche" "Hoehe" "Proportion" "global_wcc" "local_wcc" "Link-Neighbors auf selbem Geschoss" "alle Link-Neighbors"]

  if file-exists? filename [file-delete filename]
  file-open filename

  print-row (list (word anzahl_geschoße " " "Geschosse") geschoßfläche restfläche%)

  print-row headers

  let j 0
  while [j < length geschoße_sort] [
    ; bereiche auf geschoß sortiert nach absteigender wcc ihres bereich und innerhalb ihres bereichs
    let bereiche_sort reverse sort-on [weighted_centrality] bereiche
    let bereiche_auf_geschoß []
    let k 0
    while [k < length bereiche_sort] [
      let bereiche_geschoß reverse sort-on [weighted_centrality] turtles with [breed = bereiche or breed = unterbereiche and hidden? = false and mein_bereich = item k bereiche_sort and mein_geschoß = item j geschoße_sort]
      set bereiche_auf_geschoß (sentence bereiche_auf_geschoß bereiche_geschoß)
      set k (k + 1)
    ]

    ; bereiche auf geschoß sortiert nach global_wcc
    let räume_auf_geschoß reverse sort-on [global_wcc] turtles with [breed = bereiche or breed = unterbereiche and hidden? = false and mein_geschoß = item j geschoße_sort]

    let i 0
    while [i < length bereiche_auf_geschoß] [
      ask item i bereiche_auf_geschoß [
        let raum []

        ;; output geschoss
        set raum lput j raum

        ;; output funktion
        set raum lput label raum

        ;; output kategorie
        set raum lput breed raum

        ;; output mein_bereich
        set raum lput [funktion] of mein_bereich raum

        ;; output color
        set raum lput color raum

        ;; output fläche
        set raum lput fläche raum

        ;; output höhe
        set raum lput höhe raum

        ;; output proportion
        set raum lput 0 raum

        ;; output global_wcc
        set raum lput round global_wcc raum

        ;; output local_wcc
        set raum lput round weighted_centrality raum

        ;; output link-neighbors und through-put AUF GESCHOß
        let link-partner sort link-neighbors with [mein_geschoß = [mein_geschoß] of myself and hidden? = false]
        show link-partner
        if length link-partner != 0 [
          set k 0
          while [k < length link-partner] [
            set link-partner replace-item k link-partner (sentence ([funktion] of item k link-partner) ([through-put] of link (who) ([who] of item k link-partner)))
            set k (k + 1)
          ]
        ]
        ; link-partner nach item 1 sortieren
        set link-partner sort-with task [ item 1 ? ] link-partner
        set raum lput link-partner raum
```

```
        ;; output link-neighbors of link-neighbors AUF GESCHOß
        set k 0
        while [k < length link-partner] [
          ask item 0 item k link-partner [
            let link-partner2 [link-neighbors] of myself
          ]
        ]


        ;; output link-neighbors und through-put ALLE
        set link-partner sort link-neighbors with [hidden? = false]
        if length link-partner != 0 [
          set k 0
          while [k < length link-partner] [
            set link-partner replace-item k link-partner (sentence ([funktion]
of item k link-partner) ([through-put] of link (who) ([who] of item k link-
partner)))
            set k (k + 1)
          ]
        ]
        ; link-partner nach item 1 sortieren
        set link-partner sort-with task [ item 1 ? ] link-partner
        set raum lput link-partner raum

        show raum
        print-row raum
      ]
      set i (i + 1)
    ]
    set j (j + 1)
  ]

  file-close
end

to print-row [ #items ]
   ;; write all but the last item to theif   currently open file, inserting
commas
   ;; uses file-write to produce output, strings are quoted.
   foreach but-last #items
     [ file-write ?
       file-type ";"
     ]
   ;; write the last item
   file-write last #items
   ;; write end-of-line
   file-print ""
end

to export2csv_normalized_through-put
  file-close-all
  let filename (word anzahl_geschoße "geschoße_anordnung.csv")
  let headers ["Geschoss" "Funktion" "Breed" "Mein Bereich" "Farbe" "Flaeche"
"Hoehe"  "Proportion"  "Link-Neighbors  auf  selbem  Geschoss"  "alle  Link-
Neighbors"]

  if file-exists? filename [file-delete filename]
  file-open filename

  print-row  (list  (word  anzahl_geschoße  "  "  "Geschosse")  geschoßfläche
restfläche%)

  print-row headers

  let j 0
  while [j < length geschoße_sort] [
    ; bereiche auf geschoß sortiert nach absteigender wcc ihres bereich und
innerhalb ihres bereichs
    let bereiche_sort reverse sort-on [weighted_centrality] bereiche
    let bereiche_auf_geschoß []
    let k 0
    while [k < length bereiche_sort] [
      let bereiche_geschoß reverse sort-on [weighted_centrality] turtles with
[breed = bereiche or breed = unterbereiche and hidden? = false and mein_bereich
= item k bereiche_sort and mein_geschoß = item j geschoße_sort]
      set bereiche_auf_geschoß (sentence bereiche_auf_geschoß bereiche_geschoß)
      set k (k + 1)
    ]
    ;let räume_auf_geschoß sort turtles with [breed = bereiche or breed =
unterbereiche and hidden? = false and mein_geschoß = item j geschoße_sort]

    let i 0
    while [i < length bereiche_auf_geschoß] [
      ask item i bereiche_auf_geschoß [
        let raum []

        ;; output gesch0ss
        set raum lput j raum

        ;; output funktion
        set raum lput label raum
```

```
        ;; output kategorie
        set raum lput breed raum

        ;; output mein_bereich
        set raum lput [funktion] of mein_bereich raum

        ;; output color
        set raum lput color raum

        ;; output fläche
        set raum lput fläche raum

        ;; output höhe
        set raum lput höhe raum

        ;; output proportion
        set raum lput 0 raum

        ;; output link-neighbors und through-put AUF GESCHOß
        let   link-partner   sort   link-neighbors   with   [mein_geschoß   =
[mein_geschoß] of myself]
        if length link-partner != 0 [
          let through_put_list []
          set k 0
          while [k < length link-partner] [
            set through_put_list lput [through-put] of link (who) ([who] of
item k link-partner) through_put_list
            set k (k + 1)
          ]
          ; normalize through_put_list
          let through_put_list_sort sort through_put_list
          let factor (1 / last through_put_list_sort)

          set k 0
          while [k < length link-partner] [
            set link-partner replace-item k link-partner (sentence ([funktion]
of item k link-partner) (precision (item k through_put_list * factor) 2))
            set k (k + 1)
          ]
        ]
        ; link-partner nach item 1 sortieren
        set link-partner sort-with task [ item 1 ? ] link-partner
        set raum lput link-partner raum

        ;; output link-neighbors und through-put ALLE
        set link-partner sort link-neighbors
        if length link-partner != 0 [
          let through_put_list []
          set k 0
          while [k < length link-partner] [
            set through_put_list lput [through-put] of link (who) ([who] of
item k link-partner) through_put_list
            set k (k + 1)
          ]
          ; normalize through_put_list
          let through_put_list_sort sort through_put_list
          let factor (1 / last through_put_list_sort)

          set k 0
          while [k < length link-partner] [
            set link-partner replace-item k link-partner (sentence ([funktion]
of item k link-partner) (precision (item k through_put_list * factor) 2))
            set k (k + 1)
          ]
        ]
        ; link-partner nach item 1 sortieren
        set link-partner sort-with task [ item 1 ? ] link-partner
        set raum lput link-partner raum

        show raum
        print-row raum
      ]
      set i (i + 1)
    ]
    set j (j + 1)
  ]

  file-close
end
```

import-from-csv.nls

https://gist.github.com/anonymous/e3ef8459e0750317607cf2f4246904f2

```
extensions [csv]

globals [
  input
  global_scale
  ]

breed [bereiche bereich]
breed [unterbereiche unterbereich]

bereiche-own [funktion fläche höhe mein_geschoß link_partner mein_bereich
through-put_list weighted_centrality global_wcc local_wcc geschoße_bereich
meine_erschließung]
unterbereiche-own [funktion fläche höhe mein_geschoß mein_bereich link_partner
through-put_list weighted_centrality global_wcc local_wcc meine_erschließung]
links-own [through-put distanz]

to import-from-csv
  file-close-all
  import-data
  ;setup-grundstück
  setup-bereiche
  setup-links
end

to import-data
  ; alle daten importieren
  file-open "export-to-csv_neu.csv"
  set input []
  while [not file-at-end?] [
    set input lput file-read input
  ]
  file-close

  ; daten sortieren
  let i 0
  let input_sort []
  let row []
  while [i < length input] [
    ifelse item i input != "end" [
      set row lput item i input row
    ][
    ;show row
    set input_sort lput row input_sort
    set row []
    ]
    set i (i + 1)
  ]
  set input input_sort
  ;show input
end


to setup-grundstück
  ; global_scale
  set global_scale first last input

  ; grundfläche
  let i 0
  while [i < length last last input] [
    ask patch (item 0 item i last last input) (item 1 item i last last input)
(item 2 item i last last input) [
      set pcolor white ;[255 255 255 50]
    ]
    set i (i + 1)
  ]
end

to setup-bereiche
  let i 0
  while [i < (length input - 1)] [
    ifelse i < 6 [
      create-bereiche 1 [
        set shape "circle"
        set funktion (item 0 item i input)
        setxyz (item 0 item 1 item i input) (item 1 item 1 item i input) (item
2 item 1 item i input)
        set fläche (item 0 item 2 item i input)
        set höhe (item 1 item 2 item i input)
        set color (item 3 item i input)
        set link_partner (item 5 item i input)
        set through-put_list (item 6 item i input)
        set weighted_centrality (item 7 item i input)
        set global_wcc (item 8 item i input)
        set size ((sqrt (fläche / pi) * 2) / global_scale_neu)
        set mein_geschoß 0
```

```
        set mein_bereich self
        set label funktion
      ]
    ][
    create-unterbereiche 1 [
      set shape "circle"
      set funktion (item 0 item i input)
      setxyz (item 0 item 1 item i input) (item 1 item 1 item i input) (item 2
item 1 item i input)
      set fläche (item 0 item 2 item i input)
      set höhe (item 1 item 2 item i input)
      set color (item 3 item i input)
      set mein_bereich bereich (item 4 item i input)
      set link_partner (item 5 item i input)
      set through-put_list (item 6 item i input)
      set weighted_centrality (item 7 item i input)
      set global_wcc (item 8 item i input)
      set size ((sqrt (fläche / pi) * 2) / global_scale_neu)
      set mein_geschoß 0
      set label funktion
    ]
    ]
    set i (i + 1)
  ]
  ask bereiche with [who = 3 or who = 4 or who = 5] [
    set hidden? true
  ]
end

to setup-links
  let breeds (list bereiche unterbereiche)
  let colors (list [255 0 0 50] [255 255 255 50] [0 0 255 50])
  let j 0
  while [j < length breeds] [
    ask item j breeds [
      if length link_partner != 0 [
        let i 0
        while [i < length link_partner] [
          if link (who) (item i link_partner) = nobody [
            create-links-with turtles with [who = [item i link_partner] of
myself] [
              set through-put [item i through-put_list] of myself
              set thickness (through-put * .15)
              ifelse [breed] of end1 = bereiche and [breed] of end2 =
unterbereiche [
                set color item 2 colors
                ][
                set color item j colors
                ]
              ]
            ]
            set i (i + 1)
          ]
        ]
      ]
      set j ( j + 1)
    ]

    ask links with [color = [255 0 0 50]] [
      set hidden? true
    ]
end
```

7.4. Excel-Tabellen: Klassenzuwachs in Wien bis 2030

Disclaimer:

Die nachfolgenden Tabellen basieren auf prognostizierten Daten - Der Verfasser dieser Diplomarbeit übernimmt keine Gewährleistung für deren Richtigkeit

Klassenzuwachs nach Jahren

Wien	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
2009	106	0	9	0	5	33	0	0	19	36	4
2010	5	0	2	0	0	3	0	0	0	0	0
2011	11	0	1	0	0	6	0	0	0	2	2
2012	12	0	2	0	0	7	0	0	0	3	0
2013	10	0	2	0	0	7	0	0	0	1	0
2014	6	0	1	0	0	3	0	0	0	2	0
2015	16	0	2	0	0	9	1	0	0	3	1
2016	12	1	1	0	0	8	0	0	0	1	1
2017	10	0	2	0	0	5	1	0	0	1	1
2018	15	1	2	0	0	8	0	0	0	4	0
2019	16	4	1	0	0	8	1	0	0	0	2
2020	21	8	1	0	0	8	0	0	1	2	1
2021	30	5	3	0	0	16	2	0	0	4	0
2022	35	7	2	0	0	16	3	0	0	3	4
2023	40	7	2	0	0	16	8	0	0	7	0
2024	43	8	2	0	0	17	4	0	0	7	5
2025	47	10	4	0	0	18	7	0	0	7	1
2026	59	17	3	0	0	19	8	0	0	10	2
2027	59	15	3	0	1	18	9	0	1	10	2
2028	61	16	5	0	0	18	9	0	2	8	3
2029	60	18	3	0	0	18	11	0	0	9	1
2030	67	16	7	0	0	19	12	0	0	10	3

Klassenzuwachs insgesamt nach Jahren

Wien	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
2009	106	0	9	0	5	33	0	0	19	36	4
2010	111	0	11	0	5	36	0	0	19	36	4
2011	122	0	12	0	5	42	0	0	19	38	6
2012	134	0	14	0	5	49	0	0	19	41	6
2013	144	0	16	0	5	56	0	0	19	42	6
2014	150	0	17	0	5	59	0	0	19	44	6
2015	166	0	19	0	5	68	1	0	19	47	7
2016	178	1	20	0	5	76	1	0	19	48	8
2017	188	1	22	0	5	81	2	0	19	49	9
2018	203	2	24	0	5	89	2	0	19	53	9
2019	219	6	25	0	5	97	3	0	19	53	11
2020	240	14	26	0	5	105	3	0	20	55	12
2021	270	19	29	0	5	121	5	0	20	59	12
2022	305	26	31	0	5	137	8	0	20	62	16
2023	345	33	33	0	5	153	16	0	20	69	16
2024	388	41	35	0	5	170	20	0	20	76	21
2025	435	51	39	0	5	188	27	0	20	83	22
2026	494	68	42	0	5	207	35	0	20	93	24
2027	553	83	45	0	6	225	44	0	21	103	26
2028	614	99	50	0	6	243	53	0	23	111	29
2029	674	117	53	0	6	261	64	0	23	120	30
2030	741	133	60	0	6	280	76	0	23	130	33

Klassenzuwachs nach Bezirken

2009 - 30	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
Wien	741	133	60	0	6	280	76	0	23	130	33
1. Bezirk	28	4	1	0	0	13	10	0	0	0	0
2. Bezirk	12	0	2	0	0	10	0	0	0	0	0
3. Bezirk	33	0	0	0	0	20	6	0	0	7	0
4. Bezirk	19	6	1	0	0	8	0	0	4	0	0
5. Bezirk	6	1	0	0	0	5	0	0	0	0	0
6. Bezirk	6	0	2	0	0	3	0	0	1	0	0
7. Bezirk	15	3	0	0	0	4	1	0	0	0	7
8. Bezirk	16	1	0	0	0	7	2	0	0	0	6
9. Bezirk	13	0	0	0	0	6	2	0	0	5	0
10. Bezirk	99	29	32	0	3	13	3	0	0	10	9
11. Bezirk	35	10	4	0	0	16	3	0	0	2	0
12. Bezirk	46	15	5	0	0	8	4	0	3	11	0
13. Bezirk	37	0	0	0	0	10	5	0	2	20	0
14. Bezirk	27	0	0	0	0	10	1	0	2	14	0
15. Bezirk	14	0	1	0	2	4	7	0	0	0	0
16. Bezirk	16	4	7	0	0	4	0	0	1	0	0
17. Bezirk	19	3	0	0	0	8	0	0	4	4	0
18. Bezirk	12	0	0	0	0	5	7	0	0	0	0
19. Bezirk	36	1	0	0	0	12	0	0	0	19	4
20. Bezirk	15	3	1	0	0	0	3	0	1	7	0
21. Bezirk	113	16	0	0	0	50	5	0	4	31	7
22. Bezirk	56	17	0	0	0	31	8	0	0	0	0
23. Bezirk	68	20	4	0	1	33	9	0	1	0	0

Prognostizierte Entwicklung der SchülerInnenzahlen

Wien	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
2009	213148	62475	30151	2879	2911	33921	24742	23786	7277	22636	2370
2010	214348	63098	30257	2895	2907	34180	24849	23781	7257	22731	2394
2011	215547	63721	30363	2911	2903	34440	24955	23775	7236	22825	2417
2012	216747	64344	30469	2927	2898	34699	25062	23770	7216	22920	2441
2013	217947	64967	30575	2943	2894	34959	25169	23765	7196	23015	2465
2014	219147	65590	30681	2959	2890	35218	25276	23759	7175	23110	2488
2015	220346	66214	30786	2976	2886	35478	25382	23754	7155	23204	2512
2016	221546	66837	30892	2992	2882	35737	25489	23748	7134	23299	2535
2017	222746	67460	30998	3008	2878	35997	25596	23743	7114	23394	2559
2018	223946	68083	31104	3024	2873	36256	25703	23738	7094	23489	2583
2019	225145	68706	31210	3040	2869	36516	25809	23732	7073	23583	2606
2020	226345	69329	31316	3056	2865	36775	25916	23727	7053	23678	2630
2021	228417	69895	31512	3092	2873	37231	26228	23870	7050	23984	2684
2022	230489	70462	31708	3127	2880	37686	26541	24012	7047	24289	2737
2023	232561	71028	31904	3163	2888	38142	26853	24155	7044	24595	2791
2024	234633	71594	32100	3198	2896	38597	27165	24297	7041	24900	2844
2025	236706	72161	32297	3234	2904	39053	27478	24440	7038	25206	2898
2026	238778	72727	32493	3270	2911	39508	27790	24583	7034	25511	2952
2027	240850	73293	32689	3305	2919	39964	28102	24725	7031	25817	3005
2028	242922	73859	32885	3341	2927	40419	28414	24868	7028	26122	3059
2029	244994	74426	33081	3376	2934	40875	28727	25010	7025	26428	3112
2030	247066	74992	33277	3412	2942	41330	29039	25153	7022	26733	3166

Prognostizierte Entwicklung der SchülerInnen/Klasse exklusive neuer Klassen

2030	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
Wien	25.2	25.9	25.4	9.7	23.9	29.9	26.4	20.1	23.1	27.1	33
1. Bezirk	27.3	27.6	25.1	0	0	31.8	27.9	0	19.7	20.5	0
2. Bezirk	23.6	24.9	25.4	7.3	0	28.7	25	0	16.7	19.1	0
3. Bezirk	24.9	24.6	23.8	8.3	23.1	30.7	26.7	0	22.6	26.4	0
4. Bezirk	26.9	27.4	25.7	0	15.4	29	24.5	0	34.1	0	0
5. Bezirk	23.4	25.5	24.7	13	0	28.6	24.5	16.6	19.7	24.4	0
6. Bezirk	23.7	24.8	27.4	14	0	26.6	24.5	22.8	27.5	0	0
7. Bezirk	22.5	26.2	23.1	0	23	27.6	25.6	9.6	0	0	35.7
8. Bezirk	26	25.2	23.9	0	0	29.3	26.3	0	19.7	22.7	31.4
9. Bezirk	26.2	24.4	23.2	0	0	30.3	26.5	0	21.8	28.4	0
10. Bezirk	27	27.3	30.6	18.2	28.9	27.9	26.2	17.2	24.2	27.4	38.2
11. Bezirk	27.2	26.5	26.1	15.5	0	31.9	26.8	0	21	27.4	0
12. Bezirk	26.4	27.8	26.6	14.3	0	30.7	29.5	23.5	22.8	29.3	0
13. Bezirk	26.1	24.3	22.8	6.8	0	29.9	28	22.5	25	32.2	0
14. Bezirk	25.3	24.9	23.1	9	0	30.3	25.6	24.1	29.8	34.6	0
15. Bezirk	22.2	23.6	25.2	8.3	26.3	26.2	26.6	19.3	22	20.4	0
16. Bezirk	24.1	25.7	27.4	14.2	0	27.1	23	17.3	25.7	24.9	0
17. Bezirk	24.8	25.8	22.8	10.1	0	30.4	24	0	29.2	27.3	0
18. Bezirk	23.1	23.9	23.6	9.7	22.4	26.8	28.7	20.3	0	0	0
19. Bezirk	25.8	25.3	21.9	7.1	24	28.4	24.8	0	13.7	34.4	33.5
20. Bezirk	25.3	25.5	25.3	12.9	23.1	24	26.7	0	29.7	26.6	0
21. Bezirk	26.5	26.4	25	8.1	22.5	33.3	26	19.2	28.7	61	29.8
22. Bezirk	24.2	26.2	22.6	12.1	23	31.8	28	18.3	21	18.1	0
23. Bezirk	27.3	27.7	26	10	27.1	33.2	27.5	0	20.5	24.8	0

Prognostizierte Entwicklung der SchülerInnen/Klasse inklusive neuer Klassen

2030	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
Wien	23.4	24.8	24.3	9.7	22.8	24.8	24.7	20.1	21.5	24	24.5
1. Bezirk	24.9	24.4	22.3	0	0	24.8	24.9	0	19.7	20.5	0
2. Bezirk	23.6	24.9	24.8	7.3	0	24.8	25	0	16.7	19.1	0
3. Bezirk	24.9	24.6	23.8	8.3	23.1	24.8	24.9	0	22.6	24.9	0
4. Bezirk	24.9	24.9	24.5	0	15.4	24.8	24.5	0	24.4	0	0
5. Bezirk	23.4	25	24.7	13	0	24.3	24.5	16.6	19.7	24.4	0
6. Bezirk	23.7	24.8	24.9	14	0	24.9	24.5	22.8	18.3	0	0
7. Bezirk	22.5	24.9	23.1	0	23	24.9	24.9	9.6	0	0	24.3
8. Bezirk	24.9	24.6	23.9	0	0	24.8	24.5	0	19.7	22.7	24.4
9. Bezirk	24.9	24.4	23.2	0	0	24.8	24.2	0	21.8	24.7	0
10. Bezirk	25	25	25	18.2	23.8	24.8	24.6	17.2	24.2	24.9	25
11. Bezirk	25	24.9	25	15.5	0	25	24.5	0	21	24	0
12. Bezirk	25	25	24.8	14.3	0	25	23.9	23.5	18.5	24.8	0
13. Bezirk	25	24.3	22.8	6.8	0	24.8	24.9	22.5	21.2	24.9	0
14. Bezirk	25	24.9	23.1	9	0	24.7	24.6	24.1	22.4	24.9	0
15. Bezirk	22.2	23.6	24.7	8.3	23	24.9	24.8	19.3	22	20.4	0
16. Bezirk	24.1	24.9	24.8	14.2	0	24.8	23	17.3	24.1	24.9	0
17. Bezirk	24.8	24.8	22.8	10.1	0	24.6	24	0	24.5	24.4	0
18. Bezirk	23.1	23.9	23.6	9.7	22.4	24.7	24.8	20.3	0	0	0
19. Bezirk	25	25	21.9	7.1	24	24.9	24.8	0	13.7	25	24.5
20. Bezirk	24.9	25	24.9	12.9	23.1	24	24.4	0	22.3	24.9	0
21. Bezirk	25	24.9	25	8.1	22.5	25	24.8	19.2	20.5	24.6	24.4
22. Bezirk	24.2	25	22.6	12.1	23	24.9	24.9	18.3	21	18.1	0
23. Bezirk	25	25	24.8	10	25	24.8	24.8	0	16.4	24.8	0

Prognostizierte Entwicklung der SchülerInnen/Klasse exklusive neuer Klassen

Wien	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
2009	21.7	21.6	23	8.2	23.7	24.5	22.5	19	23.9	23	24.7
2010	21.9	21.8	23.1	8.2	23.6	24.7	22.6	19	23.9	23.1	24.9
2011	22	22	23.2	8.2	23.6	24.9	22.7	19	23.8	23.1	25.2
2012	22.1	22.3	23.3	8.3	23.6	25.1	22.8	19	23.7	23.2	25.4
2013	22.2	22.5	23.3	8.3	23.5	25.3	22.9	19	23.7	23.3	25.7
2014	22.4	22.7	23.4	8.4	23.5	25.4	23	18.9	23.6	23.4	25.9
2015	22.5	22.9	23.5	8.4	23.5	25.6	23.1	18.9	23.5	23.5	26.2
2016	22.6	23.1	23.6	8.5	23.4	25.8	23.2	18.9	23.5	23.6	26.4
2017	22.7	23.3	23.7	8.5	23.4	26	23.2	18.9	23.4	23.7	26.7
2018	22.8	23.5	23.7	8.6	23.4	26.2	23.3	18.9	23.3	23.8	26.9
2019	23	23.8	23.8	8.6	23.3	26.4	23.4	18.9	23.3	23.9	27.1
2020	23.1	24	23.9	8.7	23.3	26.6	23.5	18.9	23.2	24	27.4
2021	23.3	24.2	24.1	8.8	23.4	26.9	23.8	19	23.2	24.3	28
2022	23.5	24.4	24.2	8.9	23.4	27.2	24.1	19.1	23.2	24.6	28.5
2023	23.7	24.6	24.4	9	23.5	27.6	24.4	19.3	23.2	24.9	29.1
2024	23.9	24.8	24.5	9.1	23.5	27.9	24.7	19.4	23.2	25.3	29.6
2025	24.1	25	24.7	9.2	23.6	28.2	25	19.5	23.2	25.6	30.2
2026	24.4	25.2	24.8	9.3	23.7	28.5	25.2	19.6	23.1	25.9	30.8
2027	24.6	25.4	25	9.4	23.7	28.9	25.5	19.7	23.1	26.2	31.3
2028	24.8	25.5	25.1	9.5	23.8	29.2	25.8	19.8	23.1	26.5	31.9
2029	25	25.7	25.3	9.6	23.9	29.5	26.1	19.9	23.1	26.8	32.4
2030	25.2	25.9	25.4	9.7	23.9	29.9	26.4	20.1	23.1	27.1	33

Prognostizierte Entwicklung der SchülerInnen/Klasse inklusive neuer Klassen

Wien	Insgesamt	Volks-schulen	Haupt-schulen	Sonder-schulen	Polytechn. Schulen	AHS-Unterstufe	AHS-Oberstufe	Berufs-schulen	Berufsbildende mittlere Schulen	Berufsbildende hoehere Schulen	Lehrerbildende hoehere Schulen
2009	21.5	21.6	22.9	8.2	22.7	23.9	22.5	19	22.5	22.1	23.7
2010	21.6	21.8	22.9	8.2	22.7	24.1	22.6	19	22.5	22.2	23.9
2011	21.7	22	23	8.2	22.7	24.2	22.7	19	22.4	22.3	23.7
2012	21.8	22.3	23	8.3	22.6	24.2	22.8	19	22.3	22.3	23.9
2013	21.9	22.5	23.1	8.3	22.6	24.3	22.9	19	22.3	22.4	24.2
2014	22	22.7	23.1	8.4	22.6	24.4	23	18.9	22.2	22.4	24.4
2015	22.1	22.9	23.2	8.4	22.5	24.4	23	18.9	22.2	22.5	24.4
2016	22.2	23.1	23.2	8.5	22.5	24.5	23.1	18.9	22.1	22.5	24.4
2017	22.3	23.3	23.3	8.5	22.5	24.6	23.2	18.9	22	22.6	24.4
2018	22.4	23.5	23.3	8.6	22.4	24.6	23.3	18.9	22	22.6	24.6
2019	22.5	23.7	23.4	8.6	22.4	24.7	23.4	18.9	21.9	22.7	24.4
2020	22.5	23.9	23.4	8.7	22.4	24.7	23.5	18.9	21.8	22.7	24.4
2021	22.7	24	23.5	8.8	22.4	24.7	23.7	19	21.8	23	24.9
2022	22.8	24.2	23.6	8.9	22.5	24.8	23.9	19.1	21.8	23.2	24.4
2023	22.9	24.3	23.8	9	22.6	24.8	24	19.3	21.7	23.3	24.9
2024	23	24.4	23.9	9.1	22.6	24.8	24.2	19.4	21.7	23.4	24.3
2025	23.1	24.5	23.9	9.2	22.7	24.8	24.4	19.5	21.7	23.6	24.6
2026	23.2	24.6	24	9.3	22.7	24.8	24.5	19.6	21.7	23.6	24.6
2027	23.3	24.6	24.1	9.4	22.6	24.8	24.5	19.7	21.6	23.7	24.6
2028	23.3	24.7	24.2	9.5	22.7	24.8	24.6	19.8	21.5	23.8	24.5
2029	23.4	24.7	24.3	9.6	22.7	24.8	24.7	19.9	21.5	23.9	24.7
2030	23.4	24.8	24.3	9.7	22.8	24.8	24.7	20.1	21.5	24	24.5