




Smart Cooperative Distance Learning: Ein multiagentenbasierter Ansatz zur Unterstützung des kooperativen Lernens im Distanzunterricht

Henning Gösling · Tobias Dreesbach · Jan Schulte to Brinke  · Oliver Thomas

Eingegangen: 29. Januar 2021 / Angenommen: 2. Juni 2021 / Online publiziert: 12. August 2021
© Der/die Autor(en) 2021

Zusammenfassung Lehrkräfte können auf eine Vielzahl von etablierten Unterrichtsmethoden zurückgreifen, um kooperatives Lernen in den Präsenzunterricht zu integrieren. Im Distanzunterricht während der Corona-Pandemie werden diese Methoden jedoch häufig nicht eingesetzt. Dadurch gehen das gemeinschaftliche, strukturierte Lernen und die damit einhergehenden positiven Effekte auf die Schülerinnen und Schüler verloren. Eine State-of-the-Art-Analyse zeigt, dass kaum technische Lösungen existieren, die methodenbasiertes, kooperatives Lernen in einem Distanzunterricht unterstützen – und falls doch, handelt es sich um sehr spezielle Lösungen für die Umsetzung einer konkreten Unterrichtsmethode. Wir schlagen daher das Konzept eines Smart Cooperative Distance Learning Systems vor. Dabei handelt es sich um ein Multiagentensystem in einer Cloud, welches insbesondere die methodenbasierte, kooperative Ausführung von Lerneinheiten unterstützt. Ein Szenario dient zur Veranschaulichung des Smart Cooperative Distance Learning Systems.

Schlüsselwörter Distanzunterricht · Kooperatives Lernen · Unterrichtsmethoden · Multiagentensystem

Henning Gösling · Tobias Dreesbach · Jan Schulte to Brinke (✉) · Oliver Thomas
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH, Parkstraße 40, 49080 Osnabrück,
Deutschland
E-Mail: jan.schulte_to_brinke@dfki.de

Henning Gösling
E-Mail: henning.goesling@dfki.de

Tobias Dreesbach
E-Mail: tobias.dreesbach@dfki.de

Oliver Thomas
E-Mail: oliver.thomas@dfki.de

Smart Cooperative Distance Learning: A Multi-Agent-Based Approach to Support Cooperative Learning in Distance Education

Abstract Teachers can draw on a variety of established teaching methods to integrate cooperative learning into their lessons. However, in the distance learning sessions during the Corona pandemic, these methods are often not used. As a result, method-based, cooperative learning is lost, along with the associated positive effects on students. A state-of-the-art analysis shows that hardly any technical solutions exist that support method-based, cooperative distance learning—and if they do exist, they are very specific solutions for the implementation of a concrete method. We therefore conceptualize a Smart Cooperative Distance Learning System, which we think as a multi-agent system in a cloud. A scenario is used to illustrate the workings of the Smart Cooperative Distance Learning System.

Keywords Distance Learning · Cooperative Learning · Teaching Methods · Multi-Agent System

1 Einleitung

In der Bildung zielt eine Unterrichtseinheit idealerweise auf die individuelle Förderung der Schülerinnen und Schüler¹ hinsichtlich ihrer inhaltlichen, methodischen, kommunikativen, persönlichen und sozialen Kompetenzen. Verschiedene etablierte Unterrichtsmethoden stehen Lehrkräften für die Gestaltung ihres Unterrichts zur Verfügung, wie z. B. Partnerbriefing, Lerntempoduett, World Café, Gruppenpuzzle oder Think-Pair-Share (Mattes 2016). Die genannten Methoden unterstützen das sogenannte kooperative Lernen, welches sich vom kollaborativen Lernen dadurch unterscheidet, dass bei der Kollaboration die Aufgabe gemeinsam erledigt wird während bei der Kooperation die Aufgabe aufgeteilt und die Teilaufgaben individuell gelöst und dann die Teilergebnisse zum Ergebnis zusammengefügt werden (Dillenbourg 1999). Zudem bezieht sich kollaboratives Lernen auf Situationen, in denen Schüler zwar in Gruppen zusammenarbeiten – aber ohne die Anleitung von Lehrkräften. Kooperatives Lernen setzt dagegen die Anleitung durch eine Lehrkraft voraus (Kukulska-Hulme 2004). Durch kooperatives Lernen werden Teamfähigkeit und Sozialkompetenz der Schüler gestärkt und es werden alle Schüler einbezogen, da die Mitarbeit nicht auf einer freiwilligen Meldung der Schüler beruht. Des Weiteren können stärkere Schüler ihre Mitschüler unterstützen. So können sie die schwächeren Schüler fördern und gleichzeitig ihr eigenes Wissen festigen. Auf diese Weise kann durch Methoden für das kooperative Lernen das Leistungsniveau insgesamt erhöht werden, wobei besonders schwächere Schüler gefördert werden, sodass die Leistungsunterschiede innerhalb einer Klasse verringert werden können (Mattes 2016).

¹ Zugunsten des Leseflusses nutzen wir in diesem Beitrag fortan das generische Maskulinum.

Aktuelle Zeitungsberichte (z. B. in Zeit Online oder Neue Osnabrücker Zeitung) zeigen, dass durch den Distanzunterricht² während der Corona-Pandemie das kooperative Lernen häufig ausbleibt. Stattdessen setzen Schulen auf die Übertragung einer Unterrichtsstunde per Video und die Bereitstellung von Unterrichtsmaterialien per E-Mail oder Online-Lernplattform. Eine Studie während der Corona-Pandemie ergab, dass nur sieben Prozent der befragten Schüler einen gemeinsamen Lehrunterricht per Videokonferenz mit anderen Mitschülern erlebten. Der größte Teil des Unterrichts erfolgte in Form von Einzelarbeit durch die Schüler (Goetz 2020). Das Ergebnis ist, dass die zuvor genannten Vorteile des kooperativen Lernens im Distanzunterricht nicht realisiert werden. Um dem entgegenzuwirken, stellen wir in diesem Beitrag das Konzept eines *Smart Cooperative Distance Learning Systems* (SCDLS) vor, dessen technische Umsetzung die Interaktionen zwischen Schülern in einem (möglichst synchronen) Distanzunterricht auf Basis etablierter Methoden für das kooperative Lernen fördern soll. Hierzu gehen wir zunächst, im Rahmen von Literaturrecherchen in Kap. 2, auf den State-of-the-Art für Systeme ein, die das methodenbasierte, kooperative Lernen im Distanzunterricht unterstützen. Da keine zufriedenstellenden Systeme gefunden wurden, wird in Kap. 3 das SCDLS konzipiert, welches aus einem Multiagentensystem (MAS) in einer Cloud besteht. Im MAS werden insbesondere die Lehrkräfte und Schüler einer Klasse jeweils durch einen eigenen Softwareagenten repräsentiert, der als autonomes System bestimmte Funktionen der Lehrkraft bzw. des Schülers übernimmt. Durch das Zusammenspiel der Softwareagenten im SCDLS werden Lernpartnerschaften und -gruppen entsprechend bestimmter Methoden gebildet und wieder aufgelöst. In Kap. 4 wird ein Beispiel beschrieben, wie eine konkrete Anwendung des SCDLS für einen synchron stattfindenden Distanzunterricht einer Berufsschule aussehen könnte. Im Ausblick (Kap. 5) beschreiben wir den aktuellen Stand des Prototypens und unser weiteres Vorgehen hin zu einem implementierten und evaluierten SCDLS.

2 State-of-the-art-Analyse

Im Rahmen einer Literaturrecherche wurden möglichst aktuelle technische Lösungen gesucht, welche das methodenbasierte, kooperative Lernen im Distanzunterricht unterstützen. Hierzu wurden die für die Wirtschaftsinformatik klassischen Datenbanken SpringerLink, ScienceDirect, AISel, Wiley, IEEE, ACM und JSTOR durchsucht. Für die Suche wurde der folgende Suchterm genutzt: „*Cooperative Learning*“ AND *Method* AND „*Distance Learning*“ AND *Technology*. Gefunden wurde der Ansatz von Silva et al. (2021), die das Zoom-Videokonferenz-Tool nutzten, um die Metho-

² Der Begriff Distanzunterricht wird in diesem Beitrag synonym mit den Begriffen Homeschooling und Fernunterricht verwendet.

³ Bei der Think-Pair-Share-Methode erhält jeder Schüler zunächst einen Arbeitsauftrag mit einer hierfür vorgegebenen Zeitspanne („Think“). Nach Ablauf der Zeitspanne gehen Schüler paarweise zusammen und tauschen ihre Arbeitsergebnisse aus („Pair“), sodass jeder Partner in der Lage ist, die Informationen des jeweils anderen zu erklären. Abschließend („Share“) werden einige Gruppen ausgewählt, die ihre Ergebnisse im Plenum dem Rest der Klasse vorstellen (Mattes 2016).

den Think-Pair-Share³ und Gruppenpuzzle⁴ im Rahmen eines Grundschulunterrichts einzusetzen. Parsazadeh et al. (2018) zeigten eine Mobile Application zur Umsetzung des kooperativen Lernens im Distanzunterricht anhand der Gruppenpuzzle-Methode. Es wurde besonderer Wert auf eine schnelle Reaktionszeit gelegt – Lernende, welche die Mobile Application nutzten, sollten zeitnah auf die Meldungen der anderen Lernenden reagieren können. Welche Funktionen die umgesetzte und evaluierte Mobile Application genau umfasst, bleibt jedoch unklar. Andreas et al. (2010) setzten eine kooperative Lernmethode – ebenfalls die Gruppenpuzzle-Methode – in einer Virtual Reality-Umgebung um. Dazu wurde ein virtueller Raum erschaffen, in dem die Lernenden entsprechend ihrer Rolle in der Lernmethode durch einen speziellen Avatar repräsentiert wurden und sich zusammenfinden und miteinander interagieren konnten. Als Herausforderungen zeigten sich die Hardware-Voraussetzungen für Virtual Reality und die Koordination der Diskussionen zwischen den Lernenden während der Durchführung der Lernmethode.

In einer zweiten Literaturrecherche wurde gezielt nach konkreten, innovativen Technologien für den Distanzunterricht aus den Bereichen Smart Classroom und Smart Learning Environments gesucht, um diese auf die bereits umgesetzten Funktionen hin zu überprüfen und die erste Literaturrecherche zu ergänzen. Ein Smart Learning Environment dient der digitalen Erhebung und Auswertung von Schülerdaten sowie der kontextsensitiven Bereitstellung von Lerninhalten (Safsouf et al. 2020). Ein Smart Classroom dient der digitalen Vernetzung zwischen Schülern untereinander sowie zwischen Lehrern und Schülern (Uskov et al. 2015). Beides, Datenerhebung und Vernetzung, sind notwendige Voraussetzungen für eine technische Umsetzung des methodenbasierten, kooperativen Lernens im (synchronen) Distanzunterricht. Daher wurde in der zweiten Literaturrecherche explizit nach diesen Technologien im Zusammenhang mit dem Distanzunterricht gesucht. Es wurden wiederum die oben genannten Datenbanken durchsucht. Diesmal wurde der Suchterm „Distance Learning“ AND („Smart Classroom“ OR „Smart Learning Environment“) verwendet. Im Zuge der Recherche zeigte sich, dass keine Implementierungen von Smart Classrooms oder Smart Learning Environments existieren, die für die Lösung des aufgeworfenen Problems, also für die Unterstützung des methodenbasierten, kooperativen Lernens im Distanzunterricht, verwendet werden können. Gleichwohl unterstützen die in der zweiten Literaturrecherche identifizierten Systeme andere Funktionen eines Distanzunterrichts – und zwar: die Schülerinteraktionen und den Datenaustausch (Che et al. 2003; Shi et al. 2002, 2010; Zhang et al. 2003; Li et al. 2010; Di et al. 2008; Nishantha et al. 2008; Goumopoulos et al. 2015; Stergiou et al. 2018), die Aufgabenaufbereitung (Ennouamani et al. 2020), die Aufgabenverteilung (Burguillo und Vázquez 2004; Ennouamani et al. 2020), die Lernerfolgsmessung (Burguillo und Vázquez 2004; Ennouamani et al. 2020), den Einsatz mobiler Endgeräte (Stergiou et al. 2018; Ennouamani et al. 2020) und

⁴ Bei der Gruppenpuzzle-Methode erhält zunächst jede Gruppe die gleichen Lernmaterialien, wobei die einzelnen Gruppenmitglieder unterschiedliche Teile der Materialien bearbeiten. Die Schüler, die sich mit den gleichen Lerninhalten beschäftigt haben, tauschen sich anschließend in Expertengruppen aus. Daraufhin gehen die Schüler zurück in ihre Stammgruppen, um sich gegenseitig die erarbeiteten Inhalte vorzustellen (Mattes 2016).

die Live-Integration von Maschinen und Robotern (Tan et al. 2019). Die identifizierten Systeme für den Distanzunterricht ermöglichen also insbesondere einen Datenaustausch zwischen Schülern sowie deren Interaktion über eine Chatfunktion, Videos und digitale Smart Boards. Kameras, Mikrophone und Sensoren erfassen die Positionen, Bewegungen und Wortlaute von Personen und stellen sie auf einem Computerbildschirm (Zhang et al. 2003; Goumopoulos et al. 2015) oder in einer virtuellen Umgebung (Stergiou et al. 2018) aufbereitet zur Verfügung. Ennouamani et al. (2020) präsentieren ein System für den Distanzunterricht zur Erfassung des individuellen Lernstands für die Lehrkraft und zur medialen Aufbereitung der Lerninhalte für mobile Endgeräte in Abhängigkeit von den Bedürfnissen der Schüler. Tan et al. (2019) zeigen in einem Pilotprojekt, dass im Distanzunterricht auch die Integration von Robotern in die Unterrichtseinheit möglich ist. Schüler können sich über das Internet mit Robotern in einem Smart Lab verbinden und diese steuern.

3 Konzeption des Smart Cooperative Distance Learning Systems

Oberstes Ziel des in diesem Paper aufgestellten Konzepts und damit die erste und wichtigste funktionale Anforderung an das SCDLS ist die Erschließung etablierter Unterrichtsmethoden (Partnerbriefing, Lerntempoduett, World Café, Think-Pair-Share, Gruppenpuzzle usw.) für das kooperative Lernen im Distanzunterricht. Hierbei ist zu beachten, dass die positiven Effekte des kooperativen Lernens bereits in der Vor-Corona-Zeit für den Distanzunterricht relevant waren, die Corona-Pandemie aber aus unserer Sicht den „Tipping Point“ darstellen kann, um diesen Ansatz nachhaltig in digitale Lernplattformen zu etablieren.

Die weiteren funktionalen Anforderungen an das SCDLS entsprechen den in der zweiten Literaturrecherche identifizierten Funktionen von bestehenden Systemen für den Distanzunterricht: Schülerinteraktionen, Datenaustausch, Aufgabenaufbereitung und -verteilung, Lernerfolgsmessung sowie die Integration von mobilen Endgeräten, Maschinen und Robotern. Um diese Funktionen umsetzen zu können, muss das SCDLS die verteilten Daten der Schüler erfassen und auswerten sowie die Schüler untereinander sowie Schüler und Lehrkräfte miteinander vernetzen. Das SCDLS kann somit als ein Konzept basierend auf den beiden Ansätzen Smart Learning Environment und Smart Classroom betrachtet werden. Als weitere funktionale Anforderung gilt, dass mit dem System auch der synchrone Distanzunterricht adressiert werden soll.

Wir schlagen für das SCDLS eine technische Umsetzung vor, welche die Ausstattung eines jeden Akteurs mit einem PC oder Tablet, die Repräsentation eines jeden Akteurs durch einen Softwareagenten in einer Cloud sowie eine Vernetzung aller Softwareagenten in der Cloud über eine horizontale Rundruf-Kommunikation erfordert. Mit autonomen Softwareagenten als modulare Komponenten, die dynamisch auf Veränderungen in ihrer Umgebung reagieren und sich auf bestimmte Funktionen spezialisieren, wird das Zielsystem kontextsensitiv, robust und anpassungsfähig sein (Franklin und Graesser 1997). Ein Softwareagent, der eine Lehrkraft repräsentiert, wird als „Lehrkraftagent“ und ein Softwareagent, der einen Schüler repräsentiert, wird entsprechend als „Schüleragent“ bezeichnet. Um dem Konzept der zunehmen-

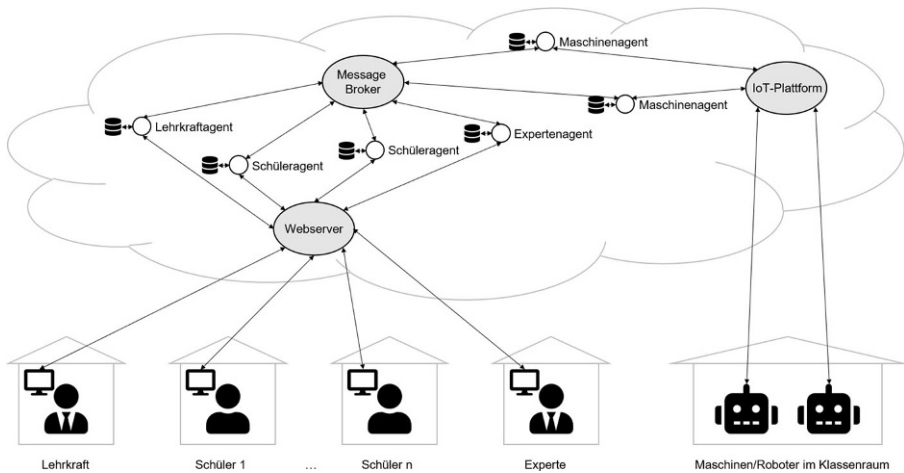


Abb. 1 Die SCDLS-Architektur

den Vernetzung von Geräten sowohl in Unternehmen als auch in der Lehre gerecht zu werden, können Maschinen und Roboter über einen „Maschinen-/Roboteragenten“ in das SCDLS integriert werden. Zudem ist als weitere Funktion die Einbindung externer Experten über einen „Expertenagenten“ vorgesehen. Experten können Ausbilder in Ausbildungswerkstätten, Praktiker in Betrieben oder Wissenschaftler von Universitäten, Hochschulen oder Forschungszentren sein. Im Zuge der Corona-Pandemie werden diese Experten vermehrt im Homeoffice anzutreffen sein.

Abb. 1 zeigt die technische Architektur des SCDLS, welches insbesondere aus einem MAS besteht. Wie darin zu erkennen ist, verfügt jeder der Softwareagenten über eine eigene Datenbank. Die Rundruf-Kommunikation zwischen allen in der Cloud befindlichen Softwareagenten läuft über einen Message Broker nach dem Publish-Subscribe-Prinzip. Die Kommunikation zwischen den Softwareagenten und den jeweiligen Lehrkräften, Schülern und Experten erfolgt über eine Webanwendung, die auf einem Webserver in der Cloud gehostet wird. Die Maschinen und Roboter werden über eine Internet-of-Things-Plattform (IoT-Plattform) angebunden, über welche die Maschinen-/Roboteragenten auf die Steuerungen der jeweiligen Maschinen und Roboter zugreifen können.

Der *Lehrkraftagent* hat die folgenden Funktionen:

- *Autonome Vorschläge von Aufgaben und Unterrichtsmethoden (L1)*: Der Lehrkraftagent schlägt Aufgaben und dazugehörige Unterrichtsmethoden vor, die das kooperative Lernen fördern und die Aufgabe strukturieren. Die vorgeschlagenen Aufgaben und Unterrichtsmethoden werden über die Webanwendung der Lehrkraft angezeigt. Die Vorschläge werden durch den Lehrkraftagenten auf die inhaltlichen, methodischen, kommunikativen und persönlichen Kompetenzen der aktuell anwesenden Schüler zugeschnitten. Für die Generierung der Vorschläge wird ein Recommender System eingesetzt.

- *Freigabe von Aufgabe, Unterrichtsmethode und Lernzielkontrolle (L2)*: Die Freigabe einer Aufgabe erfolgt durch die Lehrkraft in der Webanwendung. Die Aufgabe kann zuvor auf Basis einer ausgewählten Unterrichtsmethode für das kooperative Lernen strukturiert werden. Beispielsweise besteht eine Aufgabe, die kooperativ auf Basis der Think-Pair-Share-Methode bearbeitet werden soll, aus den drei separat freizugebenden Schritten „Think“, „Pair“ und „Share“. Eine Aufgabe basierend auf der World-Café-Methode⁵ besteht aus so vielen Schritten wie es Themen-Tische gibt und eine Aufgabe basierend auf der Gruppenpuzzle-Methode besteht wiederum aus drei separaten Schritten für Einzelarbeit, Arbeit in der Expertengruppe und Vorstellung der Ergebnisse. Damit also eine Aufgabe entsprechend einer bestimmten Unterrichtsmethode freigegeben werden kann, muss diese in die notwendigen Schritte untergliedert worden sein und pro Einzelschritt zumindest der konkrete Lerninhalt, die Anzahl der notwendigen Schüler und die maximale Dauer spezifiziert werden. Die Aufgabenstrukturierung je nach Unterrichtsmethode passiert durch die Lehrkraft in einer Modellierungsumgebung, die als ein Teil der Webanwendung zu verstehen ist. Außerdem kann einer Aufgabe in der Modellierungsumgebung eine bestimmte Art der Lernzielkontrolle zugewiesen werden (z. B. Multiple-Choice oder Single-Choice).
- *Autonome Erfassung der aktuellen Zustände aller Schüleragenten (L3)*: Über die Webanwendung für die Lehrkraft wird ein Gantt-Diagramm angezeigt, in dem für jeden Schüler dargestellt wird, welche Aufgabe – und falls die Aufgabe in Schritte unterteilt ist, welcher Schritt – derzeit bearbeitet wird und wie lange diese Bearbeitung bereits andauert. Je nach Dauer der aktuellen Bearbeitung verlängert sich der Balken im Gantt-Diagramm. Zudem wird über das Gantt-Diagramm unmittelbar angezeigt, welche Art von Unterrichtsmethode und Aufgabe durch welche Schüler angewandt werden. Falls einer Aufgabe Lernzielkontrollen zugeordnet wurden, wird deren Ergebnis ebenfalls im Gantt-Diagramm angezeigt.
- *Autonome Verarbeitung aktueller Unterstützungsanfragen der Schüler (L4)*: Schüler können über den Schüleragenten Unterstützungsanfragen absenden (siehe Schüleragent). Der Lehrkraftagent verarbeitet diese Anfragen sowie die kontinuierlichen Zustandsmeldungen der Schüleragenten und schlägt der Lehrkraft über die Webanwendung eine möglichst optimale Abfolge von Lehrkraft-Aktivitäten vor. Die Reihenfolge der Lehrkraft-Aktivitäten wird kontinuierlich mithilfe von Heuristiken an die sich ändernden Rahmenbedingungen der Lehrkraft angepasst. Die Lehrkraft kann über die Webanwendung bestimmte Unterstützungsanfragen einzelner Schüler für andere Schüler freigeben, sodass das Peer-Teaching zwischen den Schülern gefördert wird.
- *Autonome Auswertung der Ergebnisse einer Unterrichtsstunde (L5)*: Auf Basis der Erfassung der Zustände und der Ergebnisse aus den Lernzielkontrollen aller Schüleragenten erstellt der Lehrkraftagent am Ende der Unterrichtsstunde einen

⁵ Für die World Café-Methode werden mehrere Gruppen gebildet. In jeder Gruppe wird ein Gastgeber bestimmt, während die übrigen Gruppenmitglieder die Gäste am Tisch sind. In den einzelnen Gruppen werden ausgewählte Themenbereiche des Stundenthemas diskutiert. Die wichtigsten Stichpunkte werden dabei auf einem Plakat notiert. Nach einer vorgegebenen Zeit wechseln die Gäste zu einem anderen Gastgeber. Dieser führt die neue Gruppe in die bisherigen Ergebnisse ein, um das Thema anschließend weiter zu diskutieren (Mattes 2016).

individuellen Bericht, der von der Lehrkraft für die Lernerfolgsmessung und Diagnostik verwendet werden kann. Zudem dient dieser Bericht den Lehrkraftagenten als Datenbasis für das Recommender System (siehe *L1*).

- *Hinzuschalten von Experten in bestimmten Unterrichtssituationen (L6)*: Der Lehrkraftagent schlägt auf Basis der inhaltlichen und methodischen Kompetenzen der aktuell aktiven Expertenagenten bestimmte Experten vor, die über die Webanwendung der Lehrkraft angefragt und ggf. in den Unterricht hinzugeschaltet werden können (siehe Expertenagent).

Der *Schüleragent* besitzt die folgenden Funktionen:

- *Autonome Zuordnung der freigegebenen Aufgabenschritte (S1)*: Sobald neue Aufgaben und deren einzelne Schritte für den Schüleragenten zur Verfügung stehen, werden diese von dem Softwareagenten automatisch auf Basis der inhaltlichen, methodischen, kommunikativen und persönlichen Kompetenzen des Schülers in die interne Aufgabenreihenfolge eingeplant. Diese Reihenfolgeplanung kann durch eine Heuristik übernommen werden, welche die bevorzugte Aufgabenreihenfolge kontinuierlich an die dynamischen Rahmenbedingungen des Schülers anpasst.
- *Autonome Zuordnung der notwendigen Gruppenpartner für einen Aufgabenschritt entsprechend der festgelegten Unterrichtsmethode (S2)*: Sobald dem Schüler ein Aufgabenschritt zugeordnet wurde, wird überprüft, ob für die Bearbeitung weitere Schüler notwendig sind: beim „Think“-Schritt der Think-Pair-Share-Methode wäre das beispielsweise nicht der Fall, beim „Pair“-Schritt dagegen schon. Wenn andere Schüler für die Bearbeitung des Aufgabenschritts notwendig sind, werden alle anderen Schüleragenten angefragt. Die Anfrage sollte spätestens nach der Aktivierung eines Aufgabenschrittes durch den jeweiligen Schüler an die anderen Schüleragenten verschickt werden.
- *Autonome Verarbeitung von Gruppenpartner-Anfragen anderer Schüleragenten (S3)*: Ein Schüleragent stimmt sich automatisch über eine Schwarmintelligenz mit den anderen Schüleragenten ab, ob und ggf. wann eine Lerngemeinschaft initiiert werden soll. Nachdem eine Gruppenpartner-Anfrage eingegangen ist, berechnet jeder Schüleragent den Grenznutzen für den jeweiligen Schüler, der sich ergeben würde, wenn dieser Schüler Teil der angefragten Lerngemeinschaft werden würde. Die Berechnung des Grenznutzens erfolgt auf Basis eines Prognosemodells, welches ermittelt, inwieweit sich die inhaltlichen, methodischen, kommunikativen und persönlichen Kompetenzen durch die Lerngemeinschaft erhöhen könnten. Anschließend versenden alle Schüleragenten ihren Grenznutzen per Rundruf. Der Schüleragent bzw. die Schüleragenten mit dem/den höchsten Grenznutzen ordnen sich der Gruppenaufgabe zu.
- *Autonome Anzeige der zugeordneten Aufgabenschritte, Unterrichtsmethoden sowie Gruppenpartner (S4)*: Über die Webanwendung werden einem Schüler die Reihenfolge und die Planzeiten der autonom zugeordneten Aufgabenschritte angezeigt. Außerdem wird angezeigt, welcher Schritt einer Unterrichtsmethode bei welcher Aufgabe angewandt wird und ob dazu bereits bestimmte Gruppen gebildet wurden.

- *Aktivierung und Unterbrechung von Aufgaben (S5)*: Schüler können über die Webanwendung Aufgaben starten. Sobald eine Aufgabe gestartet wurde, wird dieser Zustand durch den jeweiligen Schüleragenten an alle anderen Softwareagenten (inkl. dem Lehrkraftagenten) gesendet. Zudem können Schüler über die Webanwendung Aufgaben unterbrechen und eine Begründung für die Unterbrechung angeben.
- *Versendung von Unterstützungsanfragen (S6)*: Schüler können über die Webanwendung Unterstützung anfordern. Eine Unterstützungsanfrage geht per Broadcast durch den entsprechenden Schüleragenten sowohl an den Lehrkraftagenten als auch an die anderen Schüleragenten.
- *Autonome Verarbeitung von Unterstützungsanfragen anderer Schüler (S7)*: Die Lehrkraft kann bestimmte Unterstützungsanfragen anderer Schüler freigeben. Ein Schüleragent stimmt sich automatisch über eine Schwarmintelligenz mit den anderen Schüleragenten ab, ob und ggf. wann die Unterstützungsanfrage von dem repräsentierten Schüler bearbeitet wird. Die optimale Einordnung der Unterstützungsanfragen in die aktuelle Aufgabenliste eines Schülers erfordert ebenfalls die Lösung eines Reihenfolgeproblems, denn die Einordnung einer Aufgabe in die aktuelle Liste eines Schüleragenten verursacht potenziell eine Erhöhung der inhaltlichen, kommunikativen, methodischen und persönlichen Kompetenzen und somit einen bestimmten Grenznutzen. Eine Schwarmintelligenz entscheidet auf Basis der Grenznutzen aller Schüleragenten über die Zuordnung. Anschließend wird dem ausgewählten Schüler die Anfrage angezeigt, die entweder angenommen oder abgelehnt werden kann.
- *Autonome Vorschläge für Maschinen- und Robotereinbindungen in bestimmten Unterrichtssituationen (S8)*: Der Schüleragent schlägt auf Basis der aktuellen Aufgabe und Unterrichtsmethode bestimmte Maschinen und Roboter vor, die durch einen Schüler remote genutzt werden können (siehe Maschinen-/Roboteragent). Für die Generierung dieser Vorschläge ist ein Recommender System vorgesehen.
- *Versendung von Buchungsanfragen an Maschinen- und Roboteragenten (S9)*: Über die Webanwendung eines Schülers können Buchungsanfragen an Maschinen und Roboter versendet werden. Eine Buchungsanfrage wird durch den jeweiligen Schüleragenten per Broadcast allen anderen Softwareagenten gemeldet.

Der *Expertenagent* umfasst die folgenden Funktionen:

- *Autonome Verarbeitung der Anfragen (E1)*: Sobald neue Anfragen für Experten durch die Lehrkraft freigegeben werden, werden diese von den jeweiligen Expertenagenten automatisch auf Basis der hinterlegten inhaltlichen Kompetenzen des repräsentierten Experten auf Übereinstimmung hin überprüft.
- *Autonome Anzeige der aktuellen Anfragen von Lehrkräften (E2)*: Die Webanwendung des Experten zeigt die potenziell relevanten Anfragen inkl. Inhalt und Zeitraum an.
- *Zustimmung und Ablehnung von Anfragen (E3)*: Der Experte kann über die Webanwendung bestimmten Anfragen zustimmen oder diese ablehnen.
- *Autonome Verteilung der Anfragen (E4)*: Ein Expertenagent stimmt sich automatisch über eine Schwarmintelligenz mit den anderen Expertenagenten ab, falls

Tab. 1 Übersicht über die (autonomen) Funktionen der vier SCDLS-Softwareagenten

Agent	Funktion	Autonomie
Lehrkraftagent	<i>L1</i> : Vorschläge für Aufgaben und Unterrichtsmethoden	Ja
	<i>L2</i> : Freigabe von Aufgaben, Unterrichtsmethoden und Lernzielkontrollen	Nein
	<i>L3</i> : Erfassung der aktuellen Zustände aller Schüleragenten	Ja
	<i>L4</i> : Verarbeitung von Unterstützungsanfragen der Schüler	Ja
	<i>L5</i> : Auswertung der Ergebnisse einer Unterrichtsstunde	Ja
	<i>L6</i> : Hinzuschalten von Experten in bestimmten Unterrichtssituationen	Nein
Schüleragent	<i>S1</i> : Zuordnung der freigegebenen Aufgaben	Ja
	<i>S2</i> : Zuordnung der notwendigen Gruppenpartner für einen Aufgabenschritt entsprechend der festgelegten Unterrichtsmethode	Ja
	<i>S3</i> : Verarbeitung von Gruppenpartner-Anfragen anderer Schüleragenten	Ja
	<i>S4</i> : Anzeige der zugeordneten Aufgabenschritte, Unterrichtsmethoden und Gruppenpartner	Ja
	<i>S5</i> : Aktivierung und Unterbrechung von Aufgaben	Nein
	<i>S6</i> : Versendung von Unterstützungsanfragen	Nein
	<i>S7</i> : Verarbeitung von Unterstützungsanfragen anderer Schüler	Ja
	<i>S8</i> : Vorschläge für Maschinen und Roboter in bestimmten Unterrichtssituationen	Ja
	<i>S9</i> : Versendung von Anfragen an Maschinen- und Roboteragenten	Nein
Expertenagent	<i>E1</i> : Verarbeitung der Anfragen	Ja
	<i>E2</i> : Anzeige der aktuellen Anfragen von Lehrkräften	Ja
	<i>E3</i> : Zustimmung und Ablehnung von Anfragen	Nein
	<i>E4</i> : Verteilung der Anfragen	Ja
Maschinen-/Roboteragent	<i>M1</i> : Verarbeitung der Anfragen	Ja
	<i>M2</i> : Verteilung der Anfragen	Ja

mehrere Experten einer Anfrage zugestimmt haben. Der Experte, für den der höchste Grenznutzen aus Sicht der Schüler prognostiziert wurde, darf der Anfrage nachkommen.

Der *Maschinen-/Roboteragent* hat die folgenden Funktionen:

- *Autonome Verarbeitung der Anfragen (M1)*: Nachdem eine neue Buchungsanfrage für einen Maschinen-/Roboteragenten durch einen Schüler ausgelöst wurde, wird diese von den jeweiligen Maschinen-/Roboteragenten automatisch auf Basis der hinterlegten technischen Funktionen der Maschine bzw. des Roboters auf Übereinstimmung hin überprüft.
- *Autonome Verteilung der Buchungsanfragen (M2)*: Ein Maschinen-/Roboteragent stimmt sich automatisch über eine Schwarmintelligenz mit den anderen Maschinen-/Roboteragenten ab, falls mehrere Maschinen bzw. Roboter die gleichen Funktionen aufweisen. Derjenige Maschinen-/Roboteragent, der den höchsten Grenznutzen für eine Anfrage aus Sicht der Schüler aufweist, ordnet sich dieser Anfrage zu.

In Tab. 1 werden die Funktionen der vier Softwareagententypen eines SCDLS zusammengefasst, die benötigt werden, um den zu Beginn des Kapitels aufgewor-

fenen Anforderungen gerecht zu werden. In der Tab. 1 wird auch angezeigt, ob eine Funktion autonom durch einen Softwareagenten ausgeführt werden soll oder nicht.

4 Szenario zur Veranschaulichung des Smart Cooperative Distance Learning Systems

Eine Mechatronik-Lehrkraft einer Berufsschule unterrichtet im Rahmen eines synchronen Distanzunterrichts eine Klasse mit 20 Schülern und ist durch einen Lehrkraftagenten repräsentiert. Zudem ist jeder der Schüler durch einen Schüleragenten repräsentiert. Die erste Hälfte der Unterrichtsstunde ist bereits vorbei. Auf Basis der aktuellen Kompetenzen der Schüler und des fortgeschrittenen Stands der bisherigen Aufgabe wird der Lehrkraft die Aufgabe „Konfiguration einer Speicherprogrammierbaren Steuerung (SPS)“ und Think-Pair-Share als Unterrichtsmethode vom Lehrkraftagenten bzw. dem darin enthaltenen Recommender System vorgeschlagen (Funktion *L1*).

Die Lehrkraft legt Lerninhalte, Gruppengröße und geplante Bearbeitungsdauer der Aufgabenschritte entsprechend der Think-Share-Share-Methode in der Modellierungsumgebung an und gibt nun diese Aufgabe für alle Schüler frei (*L2*). Alle Schüleragenten reagieren, indem sie den „Think“-Schritt der internen Aufgabenreihenfolge hinzufügen (*S1*). Gleichzeitig senden sie eine Anfrage an die anderen Schüleragenten nach einem passenden Aufgabenpartner für den „Pair“-Schritt (*S2*, *S3*). Sobald ein Schüler den „Think“-Schritt abgeschlossen hat, kann es einen zeitlichen Versatz geben, bis der zugewiesene Aufgabenpartner den eigenen „Think“-Schritt abgeschlossen hat. Muss ein Schüler auf seinen Lernpartner warten, kann durch den entsprechenden Schüleragenten eine durch die Lehrkraft freigegebene Zusatzaufgabe ausgewählt (*S1*) und über die Webanwendung angezeigt (*S4*) werden. Die Zusatzaufgaben gehen über den notwendigen Lerninhalt hinaus und sollen so lange bearbeitet werden, bis die ausgewählten Partner für den „Pair“-Schritt mit ihrem jeweiligen „Think“-Schritt abgeschlossen haben. Sind beide Schüler bereit für die Konfiguration der SPS im „Pair“-Schritt, bekommen sie durch einen Videochat oder Screen-Sharing die Möglichkeit, gemeinsam ein Programm zu schreiben. Da es Schwierigkeiten bei der Konfiguration der SPS gibt, sendet einer der Lernpartner eine Unterstützungsanfrage (*S6*). Die Lehrkraft ist zu der Zeit in einem Videochat zur Unterstützung einer anderen Lerngruppe und gibt die Unterstützungsanfrage daher an alle Schüleragenten weiter (*L4*), die über eine Schwarmintelligenz die Zuordnung der Unterstützungsanfrage regeln. Einem Schüler mit einem weiten Fortschritt aus einer anderen Lerngruppe wird die Anfrage angezeigt. Bei Interesse kann dieser die Anfrage annehmen (*S7*).

Nachdem die Aufgabenpartner gemeinsam ein SPS-Programm geschrieben haben, sollen sie es live an einem Lehrgerät testen. In dem physischen Klassenraum sind fünf Lehrgeräte mit einer integrierten SPS verfügbar, die von den Schülern für die Aufgabebearbeitung genutzt werden können. Die Lehrgeräte haben einen nahezu identischen Funktionsumfang und werden jeweils durch einen Maschinen-/Roboteragenten repräsentiert. Gemäß dem Gedanken eines Smart Labs sind die Lehrgeräte über eine IoT-Plattform angeschlossen und lassen sich über das Inter-

net steuern. Der Schüleragent zeigt den Schülern in der Webanwendung an, dass sie für die nächste anstehende Aufgabe ein Lehrgerät benötigen (S8). Einer der Schüler bestätigt daher noch während der Programmierung eines SPS Programms in der Webanwendung, dass die beiden in Kürze ein Gerät benötigen, weshalb der Schüleragent eine Buchungsanfrage an alle Maschinen-/Roboteragenten sendet (S9). Diese wird von den Maschinen-/Roboteragenten verarbeitet (M1). Anschließend ordnet sich genau ein Maschinen-/Roboteragent über eine Schwarmintelligenz der Buchungsanfrage zu (M2). Die Schüleragenten erhalten die ID der jeweiligen SPS und zeigen diese der Lerngemeinschaft an. Anschließend können die Schüler die Konfiguration über die IoT-Plattform auf die SPS laden. Am Ende der Aufgabe erfolgt eine Lernerfolgsmessung durch eine von der Lehrkraft erstellte Zuordnungsaufgabe. Die von den Schülern erzielten Ergebnisse werden an den Lehrkraftagenten gesendet und als Datenbasis für das Recommender System genutzt (L5).

Für die letzten 10 min der Unterrichtseinheit schlägt der Lehrkraftagent vor, Experten anzufragen, die mit der Programmierung von SPS bereits praktische Erfahrungen haben. Nachdem die Lehrkraft den Auftrag freigegeben hat (L6), überprüfen die Expertenagenten, ob sie die Anfrage inhaltlich bedienen können und ob der Zeitraum der Unterrichtsstunde als verfügbar angegeben wurde (E1). Treffen die Kriterien zu, bekommen die Experten eine Benachrichtigung (E2). Ein Experte gibt für den Auftrag eine positive Rückmeldung und wird daher für die Aufgabe ausgewählt (E3). Es ist ein weiterer Lehrer, der angeboten hat, in den Kursen von seinen praktischen Erfahrungen zu erzählen. Der Experte kann sich am Ende der Unterrichtseinheit über einen Videochat einwählen. Alle Schüleragenten fügen den Vortrag des Experten ihrer Aufgabenliste hinzu (S1).

5 Zusammenfassung und Ausblick

Das in diesem Paper vorgestellte SCDLS soll das kooperative Lernen im Distanzunterricht während der Corona-Pandemie und darüber hinaus unterstützen. Mit diesem Paper wurden die funktionalen Anforderungen für ein multiagentenbasiertes SCDLS identifiziert. Genauer gesagt wurden die Funktionen der einzelnen Softwareagenten definiert, welche in ihrem Zusammenspiel kooperatives Lernen in einem möglichst synchronen Distanzunterricht realisieren sollen. Auffällig ist, dass einzelne Funktionen (z.B. Schülerinteraktion, Aufgabenverteilung und Maschinen/Roboter-Integration) unabhängig voneinander bereits für den Distanzunterricht implementiert wurden. Ein wie in diesem Beitrag vorgestelltes Konzept, welches die bereits bestehenden Funktionen aufnimmt und um die Perspektive des kooperativen Lernens ergänzt, konnten wir in der bestehenden Literatur jedoch nicht identifizieren.

Die Umsetzung des Konzepts erfordert seitens der Anwender keine hardwaretechnischen Voraussetzungen außer einem Internetanschluss sowie einem PC oder einem Tablet. Das beschriebene Konzept wurde auf Basis der Zwischenergebnisse aus dem BMBF-geförderten Projekt AdePT heraus entwickelt und befindet sich derzeit in der prototypischen Umsetzung. Der aktuelle Prototyp besteht aus einem MAS, welches auf einem Standard-PC läuft. Hinsichtlich der vorgesehenen Funktionalitäten (siehe Tab. 1) wurden eine Funktionalität des Lehreragenten (L2) sowie

drei Funktionalitäten des Schüleragenten (S1, S6, S7) bereits umgesetzt. Im bestehenden Prototyp schreibt jeder Schüleragent seinen absolvierten Lernpfad in eine Neo4j-Graphdatenbank. Als Message Broker wurde ein MQTT-Server eingesetzt. Als Ausgabemedium des Schüleragenten dient – aufgrund des Augmented Reality-Fokus des AdEPT-Projektes – eine Microsoft HoloLens2. Langfristiges Ziel ist es, für einen Proof-of-Concept, die übrigen Funktionen der Softwareagenten des vorgeschlagenen SCDLS zu implementieren und zu evaluieren und damit einen signifikanten Beitrag für die Verbesserung des Distanzunterrichts, auch über die Corona-Pandemie hinaus, zu leisten.

Danksagung Dieser Beitrag ist im Rahmen des Projekts AdEPT – Augmented Reality in Bildungsprozessen der technischen Ausbildung (FKZ: 01PV18008E) entstanden. Das AdEPT-Projekt wird durch das Bundesministerium für Bildung und Forschung (BMBF) gefördert. Die Verantwortung für den Inhalt dieses Beitrags liegt bei den Autoren.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access Dieser Artikel wird unter der Creative Commons Namensnennung 4.0 International Lizenz veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Artikel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.

Weitere Details zur Lizenz entnehmen Sie bitte der Lizenzinformation auf <http://creativecommons.org/licenses/by/4.0/deed.de>.

Literatur

- Andreas K, Tsiatsos T, Terzidou T, Pomportsis A (2010) Fostering collaborative learning in second life: metaphors and affordances. *Comput Educ* 55:603–615. <https://doi.org/10.1016/j.compedu.2010.02.021>
- Burguillo JC, Vázquez E (2004) X-Learn: an intelligent educational system oriented towards the net. In: Conejo R, Urrutavizcaya M, Pérez-de-la-Cruz JL (Hrsg) *Current topics in artificial intelligence*. Springer, Berlin, Heidelberg, S 628–637
- Che Y, Shi R, Shi Y, Xu G (2003) Sameview: a large-scale real-time interactive E-learning system based on TORM and AMTP. In: Zhou W, Nicholson P, Corbitt B, Fong J (Hrsg) *Advances in web-based learning*. Springer, Berlin, Heidelberg, S 122–133
- Di C, Gang Z, Juhong X (2008) A new system for interactive demonstration in distance education—the research and design of the smart classroom based on blending reality. *IEEE International Conference on Granular Computing*, S 121–124 <https://doi.org/10.1109/GRC.2008.4664659>
- Dillenbourg P (1999) Introduction: What do you mean by “collaborative learning”? In: Dillenbourg P (Hrsg) *Collaborative learning: cognitive and computational approaches*. Elsevier, Oxford, S 1–19
- Ennouamani S, Mahani Z, Akharraz L (2020) A context-aware mobile learning system for adapting learning content and format of presentation: design, validation and evaluation. *Educ Inf Technol* 25:3919–3955. <https://doi.org/10.1007/s10639-020-10149-9>
- Franklin S, Graesser A (1997) Is it an agent, or just a program?: A taxonomy for autonomous agents. In: Müller JP, Wooldridge MJ, Jennings NR (Hrsg) *Intelligent agents III agent theories, architectures, and languages*. ATAL 1996. *Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence)*, vol 1193. Springer, Berlin, Heidelberg. <https://doi.org/10.1007/BFb0013570>

- Goetz M (2020) Distance Learning in der COVID-19 Krise: Ein Praxischeck. Medienimpulse. <https://doi.org/10.21243/mi-02-20-19>
- Goumopoulos C, Kokkos N, Karachristos C, Kameas A (2015) Bridging the gap between virtual and physical classrooms to enhance distance learning. International Conference on Intelligent Environments. <https://doi.org/10.1109/IE.2015.23>
- Kukulska-Hulme A (2004) Do Online collaborative groups need leaders? In: Roberts TS (Hrsg) Online collaborative learning: theory and practice. IGI Global, London, S 262–280
- Li X, Li M, Zeng L (2010) Virtual classrooms supporting a two-way synchronized video and audio interaction. In: Zhang X, Zhong S, Pan Z et al (Hrsg) Entertainment for education. Digital techniques and systems. Springer, Berlin, Heidelberg, S 446–455
- Mattes W (2016) Methoden für den Unterricht: kompakte Übersichten für Lehrende und Lernende. Schöningh, Paderborn
- Nishantha GGD, Pishva D, Hayashida Y (2008) Smart classrooms: architectural requirements and deployment issues. IEEE Region 10 and the Third international Conference on Industrial and Information Systems, S 1–6 <https://doi.org/10.1109/ICIINFS.2008.4798437>
- Parsazadeh N, Ali R, Rezaei M (2018) A framework for cooperative and interactive mobile learning to improve online information evaluation skills. Comput Educ 120:75–89. <https://doi.org/10.1016/j.compedu.2018.01.010>
- Safsouf Y, Mansouri K, Poirier F (2020) Smart learning environment, measure online student satisfaction: a case study in the context of higher education in Morocco. International Conference on Electrical and Information Technologies (ICEIT), S 1–5 <https://doi.org/10.1109/ICEIT48248.2020.9113189>
- Shi Y, Xie W, Xu G (2002) Smart remote classroom: creating a revolutionary real-time interactive distance learning system. In: Fong J, Cheung CT, Leong HV, Li Q (Hrsg) Advances in web-based learning learning. Springer, Berlin, Heidelberg, S 130–141
- Shi Y, Qin W, Suo Y, Xiao X (2010) Smart classroom: bringing pervasive computing into distance learning. In: Nakashima H, Aghajan H, Augusto JC (Hrsg) Handbook of ambient intelligence and smart environments. Springer US, Boston, S 881–910
- Silva H, Lopes J, Morais E, Dominguez C (2021) Cooperative learning and critical thinking in face to face and online environments. In: Reis A, Barroso J, Lopes JB, Mikropoulos T, Fan CW (Hrsg) Technology and innovation in learning, teaching and education. TECH-EDU 2020. Communications in computer and information science, Bd. 1384. Springer, Cham https://doi.org/10.1007/978-3-030-73988-1_12
- Stergiou C, Plageras AP, Psannis KE et al. (2018) Proposed high level architecture of a smart interconnected interactive classroom. South-eastern European design automation, computer engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM), S 1–6 <https://doi.org/10.23919/SEEDA-CECNSM.2018.8544922>
- Tan Q, Denojean-Mairet M, Wang H et al (2019) Toward a telepresence robot empowered smart lab. Smart Learn Environ. <https://doi.org/10.1186/s40561-019-0084-3>
- Uskov VL, Bakken JP, Pandey A (2015) The ontology of next generation smart classrooms. In: Uskov L, Howlett RJ, Jain LC (Hrsg) Smart education and smart e-learning. Springer, Cham, S 3–14
- Zhang D, Chen E, Shi Y, Xu G (2003) A kind of smart space for remote real-time interactive learning based on pervasive computing mode. In: Zhou W, Nicholson P, Corbitt B, Fong J (Hrsg) Advances in Web-Based Learning. Springer, Berlin, Heidelberg, S 297–307