

Malte Eckhoff und Karin Landenfeld (2019). GoMaxima - Eine performante Verwendung von STACK und Maxima in Lernmanagementsystemen.  
In: Contributions to the 1<sup>st</sup> International STACK Conference 2018. Friedrich-Alexander-Universität: Nürnberg, Germany. DOI 0.5281/zenodo.2561190

## GoMaxima - Eine performante Verwendung von STACK und Maxima in Lernmanagementsystemen

Malte Eckhoff<sup>1</sup>, Karin Landenfeld<sup>2</sup>

**Abstract:** In diesem Beitrag wird mit GoMaxima eine Alternative zu MaximaPool vorgestellt. GoMaxima wurde an der HAW Hamburg entwickelt und steigert die Performance von Moodle-Tests, die Aufgaben mit STACK und Maxima beinhalten, sehr deutlich. In dem Beitrag wird das Konzept und die Umsetzung von GoMaxima erläutert. GoMaxima wurde sowohl mit Moodle als auch mit Ilias erprobt und mit MaximaPool verglichen. Die Ergebnisse der Vergleichstests werden hier ebenfalls vorgestellt und diskutiert.

**Keywords:** Online-Test, E-Assessment, Performance, MaximaPool, GoMaxima

### 1. Ausgangslage und Problemstellung

Online-Übungsaufgaben, Online-Tests und Online-Prüfungen sind ein wichtiger Bestandteil vieler digitaler Lernangebote. Sie werden den Studierenden beispielsweise als freiwillige Ergänzung zum Selbststudium angeboten oder auch verpflichtend innerhalb von Lehrveranstaltungen genutzt. Lehrende verwenden digitale Lernumgebungen zum Beispiel im Rahmen von Veranstaltungen im Blended Learning-Format zur Bearbeitung von Lerneinheiten durch die Studierenden als Vor- oder Nachbereitung oder auch für verpflichtende Prüfungsleistungen. Insbesondere im Bereich der Mathematik ist die Unterstützung durch digitale Lernangebote und -aufgaben bereits sehr verbreitet. Zur geeigneten Umsetzung von mathematischen Online-Aufgaben wird häufig ein Computeralgebrasystem angebunden, so dass eine Überprüfung auf mathematische Äquivalenz mit einem differenzierten Feedback möglich ist.

An der Hochschule für Angewandte Wissenschaften (HAW) Hamburg werden interaktive Online-Aufgaben zur Mathematik mit einem differenzierten individuellen Feedback in verschiedenen Szenarien verwendet:

- (1) In der Online-Lernumgebung viaMINT<sup>3</sup>, in der videobasierte interaktive Online-Lernmodule zur Vorbereitung auf das Studium angeboten werden [vgl. LA18-1], werden sowohl kurze Verständnisaufgaben zwischen den Videos als auch

---

<sup>1</sup> HAW Hamburg, Fakultät Technik und Informatik, malte.eckhoff@haw-hamburg.de

<sup>2</sup> HAW Hamburg, Fakultät Technik und Informatik, karin.landenfeld@haw-hamburg.de

<sup>3</sup> <https://www.viamint.de>

Übungsaufgaben zum vertieften Lernen der vermittelten Inhalte sowie Tests zur Überprüfung der Kenntnisse eingesetzt. Die Verwendung der Online-Lernmodule findet verstärkt in einem Zeitraum von vier Wochen vor Semesterbeginn statt.

- (2) Semesterbegleitende Auffrischkurse und Modelle eines gestreckten Studieneinstiegs verwenden die viaMINT-Online-Module während des Semesters. Insbesondere werden dort die diagnostischen Online-Selbsttests mit Lernempfehlungen [vgl. LA18-2] sowie die Modultests der einzelnen mathematischen Themen innerhalb von begleiteten Tutorien eingesetzt, so dass viele Studierende gleichzeitig mit den Aufgaben und Tests arbeiten.
- (3) Online-Aufgaben werden vorlesungsbegleitend zu den Mathematik-Veranstaltungen angeboten, zum einen als wöchentliche freiwillige Aufgaben und zum anderen als verpflichtende Zwischentests zum Erreichen einer Prüfungsvorleistung. Diese Zwischentests finden gleichzeitig für alle etwa 50 Studierenden eines Studiengangs in PC-Pools statt, so dass die Performanz der Online-Lernumgebung eine sehr wichtige Anforderung beim Einsatz der Online-Aufgaben darstellt.

An der HAW Hamburg wird das Lernmanagementsystem Moodle<sup>4</sup> mit dem Plugin STACK<sup>5</sup> [SA-13] sowie dem angeschlossenen Computeralgebrasystem (CAS) Maxima<sup>6</sup> eingesetzt, welches insbesondere die Überprüfung auf mathematische Äquivalenz der Antworten ermöglicht. Mit dem Aufgabentyp STACK lassen sich offene Aufgaben mit einer Vielzahl korrekter Lösungen umsetzen, bei denen die Antwort im Hinblick auf definierbare Eigenschaften untersucht und ein detailliertes individuelles Feedback gegeben werden kann. Weiterhin erlaubt STACK die Verwendung von randomisierten Werten für Variablen und Parameter, so dass ein wiederholtes Üben mit veränderten Aufgaben zum Erwerb mathematischer Kompetenzen ermöglicht wird (vgl. Abb. 1).

Bei einer umfangreichen und auch gleichzeitigen Nutzung von Moodleaufgaben und -tests mit STACK/Maxima ist es wichtig, eine performante Durchführung zu erreichen, so dass die Studierenden ohne große Wartezeiten die Aufgaben verwenden und das Feedback bekommen können.

---

<sup>4</sup> Moodle ist eine frei verfügbare Lernplattform und ein Kursmanagementsystem <https://moodle.org/>.

<sup>5</sup> STACK (System for Teaching and Assessment using a Computer algebra Kernel) <http://www.stack.ed.ac.uk/>

<sup>6</sup> Maxima ist ein Computeralgebrasystem <http://maxima.sourceforge.net/de/>.

**Modultest: Gleichungen und Ungleichungen**  
mit 10 Aufgaben (9 STACK/ 1 Multiple Choice)

**Frage 5**  
Bisher nicht beantwortet  
Erreichbare Punkte: 1,00  
Frage markieren

Bestimmen Sie die Lösungsmenge der Ungleichung, abhängig von dem Parameter  $b$ :

$$b \cdot x + 1 \geq -32$$

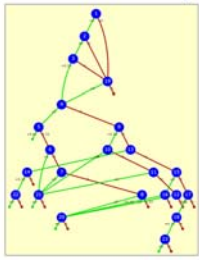
Für:

$b > \text{[ ]}$  ist  $L_{>} = \{x \in \mathbb{R}, \text{[ ]}\}$ ,

$b = \text{[ ]}$  ist  $L_{=} = \mathbb{R}$ ,

$b < \text{[ ]}$  ist  $L_{<} = \{x \in \mathbb{R}, \text{[ ]}\}$ ,

Feedback-Baum der Aufgaben



**Frage 9**  
Bisher nicht beantwortet  
Erreichbare Punkte: 1,00  
Frage markieren

Bestimmen Sie die Lösungsmenge der Ungleichung:

$$2 \cdot x^2 - 4 \cdot x - 5 < 25$$

$L = \{x \in \mathbb{R}, \text{[ ]} < x < \text{[ ]}\}$

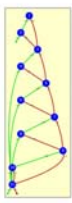


Abb. 1: Beispielaufgaben aus dem Modultest Gleichungen und Ungleichungen mit dem Rückmeldebaum für das Feedback bei der Beantwortung der Fragen

viaMINT ist eine gut frequentierte Lernumgebung und enthält über 560 Mathematikaufgaben für den Vorkursbereich auf unterschiedlichen Niveaustufen, davon sind ca. 60% STACK-Aufgaben mit zum Teil komplexen Rückmeldebäumen (vgl. Abb. 1). Der diagnostische Selbsttest mit 48 Fragen wurde innerhalb von 2 Monaten (15.8. - 15.10.2018) 1061-mal aufgerufen. Für die Mathematik 1 und 2 sind etwa 380 Aufgaben, davon 200 STACK-Aufgaben, implementiert, die neben der freiwilligen Nutzung auch in beaufsichtigten Online-Tests in PC-Poolräumen mit allen Studierenden gleichzeitig verwendet werden. Dieses ist eine besondere Herausforderung für das System, da eine gleichzeitige Nutzung von vielen STACK-Aufgaben und -Tests eine hohe Last an Anfragen auf den Server mit dem Computeralgebrasystem Maxima bringt. Weiterhin erzeugen komplexe STACK-Fragen mit einem umfangreichen Rückmeldebaum eine besonders hohe Last, da jede Verzweigung im Baum in einer Maxima-Frage resultiert. Abhängig von der System- und Serverkonfiguration kann dieses Probleme verursachen. So können sich beispielsweise durch die gleichzeitige Initialisierung vieler Moodle-STACK-Quizzes Engpässe in der Rechenkapazität ergeben, so dass sich keine weiteren Moodle-Stack-Quizzes öffnen lassen.

An der HAW Hamburg wurden diese Engpässe bei einem Test mit 30 Teilnehmern in einer Konfiguration mit mehreren virtuellen Servern beobachtet, wobei ein Server für Moodle und STACK sowie weitere Server für MaximaPool<sup>7</sup> verwendet wurden. Jeder dieser Server war dabei mit jeweils vier virtuellen Prozessorkernen ausgestattet. Konkret

<sup>7</sup> MaximaPool realisiert einen Pool von Prozessen des Computeralgebrasystems Maxima. [https://github.com/math/stack\\_util\\_maximapool](https://github.com/math/stack_util_maximapool)

sichtbar wurde, dass die CPU Auslastung auf 100% stieg, das System mit dem Neustarten der einzelnen Prozesse ausgelastet war und nicht mehr auf Anfragen geantwortet hat. Somit war keine ordnungsgemäße Durchführung des Online-Tests möglich. Eine Analyse des Problems sowie die Konzeption und Umsetzung der in diesem Beitrag beschriebenen neuentwickelten Lösung durch GoMaxima werden im nächsten Abschnitt beschrieben.

## 2. Konzept und Umsetzung

Aktuelle und zukünftige Lehr-/Lernszenarien mit einer gleichzeitigen Verwendung von Moodle-STACK-Tests, zum Beispiel für Zwischentests oder Online-Klausuren in einem PC-Pool, erfordern eine performante Konfiguration, in denen die Tests zuverlässig durchführbar sind und eine Auswertung mit Feedback zeitnah erfolgt.

Für eine performante Verwendung von STACK gibt es Abhängigkeiten von verschiedenen Konfigurationen, beispielsweise

- der Konfiguration des Servers (Hardware, Anzahl der Kerne, RAM)
- der Anzahl der STACK-Fragen im Moodle-Quiz
- der Konfiguration und Komplexität der STACK-Frage
- Optimierungen innerhalb von Maxima, Vorbereitung der STACK-Bibliotheken

Unabhängig von der in diesem Beitrag vorgestellten neu entwickelten Lösung GoMaxima, gibt es bereits einige vorhandene Verbesserungsmöglichkeiten, die hier nachfolgend kurz erläutert werden:

### (1) Verwendung eines Maxima-Antwort-Caches

STACK verwendet das Computeralgebrasystem Maxima zur Auswertung mathematischer Ausdrücke. Für die schnellere Beantwortung von Anfragen merkt sich STACK die Ergebnisse früherer Anfragen (durch den Benutzer oder Erstellung von Testfällen durch den Entwickler) in einem Cache (vgl. auch [Ka16]). Trifft die gleiche Anfrage erneut ein, wird das gespeicherte Ergebnis direkt zurückgegeben. Eine erneute, potentiell zeitaufwändige Auswertung entfällt. Liegt kein früheres Ergebnis vor, leitet STACK die Anfrage an Maxima weiter. Da die Auswertung der Anfragen rechenintensiv werden kann, wird Maxima auf einem gesonderten, über ein Netzwerk verbundenen Server ausgeführt.

### (2) Vorgenerierung der Aufgaben bei randomisierten Parametern (Deployment)

### (3) Verwendung von MaximaPool (Konfiguration dazu vgl. Abbildung 2)

Das Computeralgebrasystem Maxima ist eine Anwendung, die nur über die Ein- und Ausgabe von Text mit anderen Anwendungen kommuniziert. Die STACK-Anfragen an Maxima bestehen aus einer Menge von Maxima-Befehlen, wobei ein Befehl eine beliebig große Menge von Ausgaben produzieren kann. Um Maxima über das Netzwerk anzusprechen, wird deshalb eine spezielle Software benötigt, die Anfragen

empfängt und an Maxima weitergibt. Eine solche Software ist MaximaPool<sup>8</sup>. Um die Antwortzeiten zu minimieren, startet MaximaPool eine konfigurierbare Anzahl von parallel geöffneten Maxima-Prozessen. Dieses dient zur gleichzeitigen Bearbeitung mehrerer Anfragen. Jeder Maxima-Prozess wird zur Bearbeitung von genau einer STACK-Anfrage verwendet und danach beendet. Zur Bearbeitung der nächsten Anfrage wird sofort danach ein neuer Maxima-Prozess gestartet (vgl. Abbildung 3). Ein sogenannter Reverse-Proxy-Server sorgt für eine gleichmäßige Aufteilung der STACK-Anfragen auf mehrere MaximaPool-Instanzen. Nach erfolgter Auswertung sammelt der Server die Ergebnisse ein und gibt diese an STACK zurück.

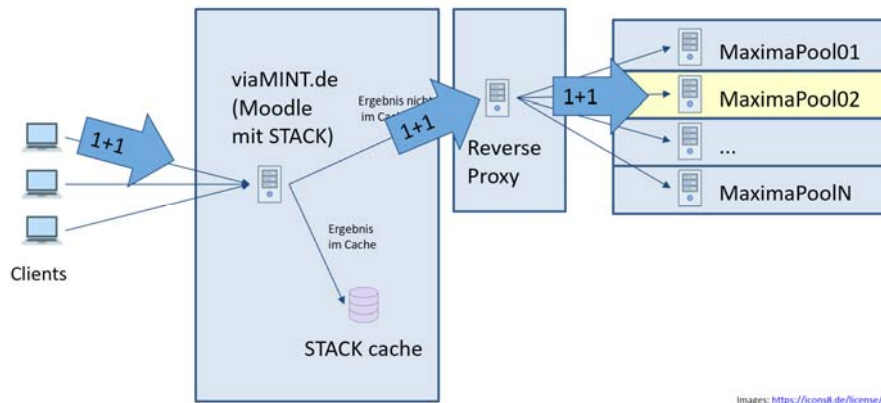


Abb. 2: Konfiguration mit drei Servern: Moodle, Reverse Proxy, MaximaPool

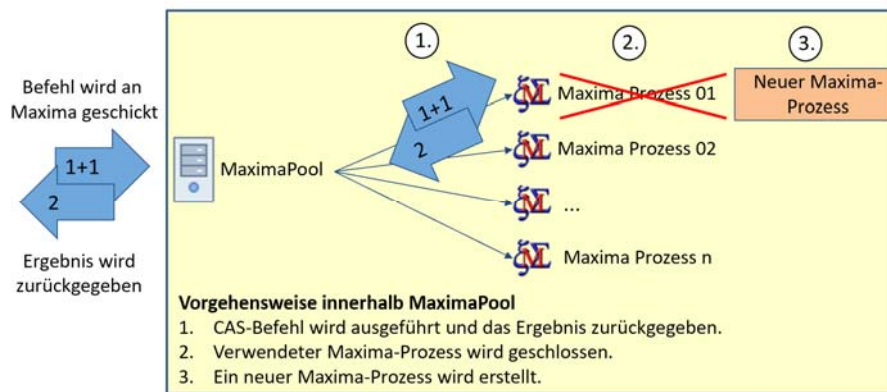


Abb. 3: Erläuterung der Arbeitsweise von MaximaPool

Die Konfiguration von STACK/Maxima mittels MaximaPool erzeugt aber ebenfalls Probleme, wenn die Anzahl der STACK-Anfragen an MaximaPool eine gewisse Anzahl<sup>9</sup> in einem kurzen Zeitraum übersteigen. Da der Start eines neuen Maxima-Prozesses relativ rechenintensiv ist, resultiert hieraus eine hohe Prozessorauslastung und damit unannehmbar lange Antwortzeiten oder sogar ein „Nichtstarten“ des Moodletests. Bei einer hohen Anzahl<sup>10</sup> an Maxima-Prozessen erzeugen auch die Wechsel zwischen den Prozessen einen zusätzlichen Zeitverlust. Die genannten Probleme treten beispielsweise bei der Ausführung vieler gleichzeitiger Moodle/STACK-Tests im Rahmen einer Veranstaltung mit einer Semestergruppe im PC-Poolraum auf.

Eine Möglichkeit diesen Problemen zu begegnen, wäre die Erweiterung der vorhandenen Rechen- und Serverkapazitäten über eine Steigerung der Prozessorkerne und des RAMs, oder die Inbetriebnahme weiterer MaximaPool-Instanzen.

Ein alternativer Lösungsansatz ist die Ersetzung von MaximaPool durch eine an der HAW Hamburg neuentwickelte Maxima-Anbindung, genannt GoMaxima. Die Zielsetzungen von GoMaxima sind zum einen die Reduzierung der Antwortzeiten, welche durch eine Minimierung der Bearbeitungszeit einer Anfrage und eine Maximierung der Anzahl der gleichzeitig bearbeiteten Anfragen erreicht wird. Zum anderen soll die definierte Schnittstelle von Moodle/STACK beibehalten werden. Damit kann MaximaPool durch GoMaxima ohne Anpassung der restlichen Infrastruktur ersetzt und die gleichmäßige Lastverteilung mittels des Reverse Proxy Servers beibehalten werden (vgl. Abbildung 4).

<sup>9</sup> Deutlich mehr Anfragen als laufende Maxima-Prozesse innerhalb weniger Sekunden.

<sup>10</sup> In Relation zu den verfügbaren CPU-Kernen.

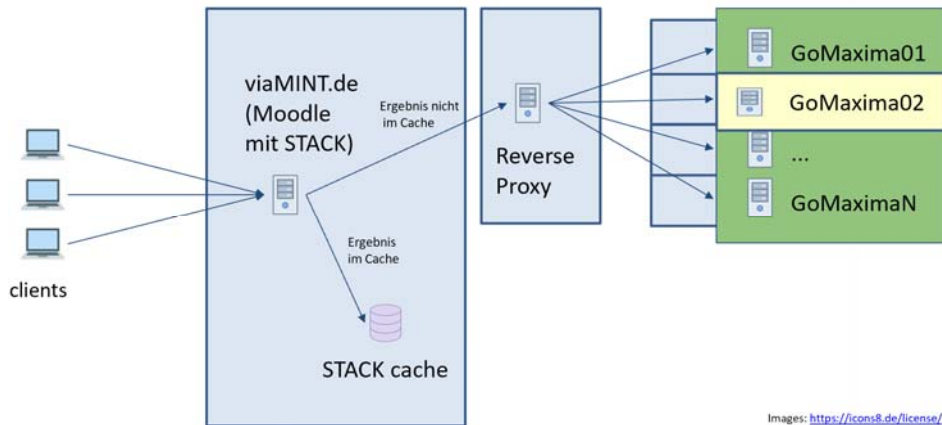


Abb. 4: Konfiguration mit drei Servern: Moodle, Reverse Proxy, GoMaxima

Als Programmiersprache wurde Golang<sup>11</sup> ausgewählt, die eine hohe Ausführungsgeschwindigkeit, vergleichbar mit C++, hat und eine effiziente Parallelisierung durch zahlreiche spezialisierte Sprachkonstrukte effektiv unterstützt. Der Hauptunterschied zwischen GoMaxima und MaximaPool ist eine Wiederverwendung der Maxima-Prozesse. Eine Instanz von GoMaxima startet eine konfigurierbare Anzahl von Maxima-Prozessen. Jedem Prozess ist eine Warteschlange vorgeschaltet, in der die STACK-Anfragen jeweils mit einem zusätzlich angehängten Steuerbefehl, der das Ende der STACK-Anfrage signalisiert, gespeichert werden. Alle Anfragen werden für eine Bearbeitung gleichmäßig auf diese Warteschlangen verteilt, parallel bearbeitet und entsprechend die Ergebnisse an STACK zurückgegeben (vgl. Abbildung 5). Aufgrund der Steuerbefehle wird erkannt, wann eine Anfrage beendet ist, so dass der Maxima-Prozess dann für die nächste Anfrage bereit ist. Ein Prozess kann mehrere Anfragen nacheinander ohne Neustart bearbeiten. Ein kontinuierliches Schließen und Neuerstellen von Maxima-Prozessen, wie bei MaximaPool, ist dadurch bei GoMaxima nicht notwendig, so dass durch diese Vorgehensweise die Performance deutlich gesteigert wird.

<sup>11</sup> Programmiersprache Golang <https://golang.org/>

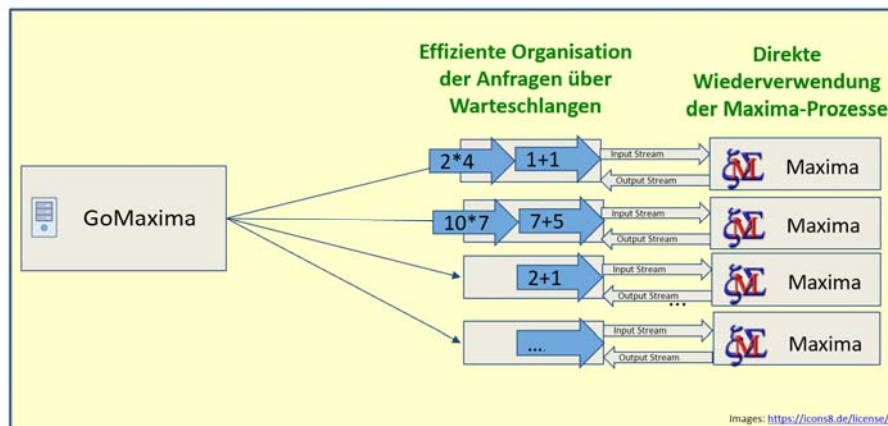


Abb. 5: Erläuterung der Arbeitsweise von GoMaxima

### 3. Einsatz und Erprobung

GoMaxima wurde an der HAW Hamburg im Rahmen des Aufbaus der Online-Lernumgebung viaMINT entwickelt und wird seit mehreren Semestern erfolgreich im Produktivbetrieb eingesetzt. Gegenüber einer Verwendung mit MaximaPool konnte im Durchschnitt eine Verbesserung der Antwortzeit um den Faktor 20 erreicht werden. Gleichzeitig konnte die Auslastung der Ressourcen signifikant reduziert werden. In der Moodle-Lernumgebung der HAW Hamburg werden aktuell alle Maxima-Anfragen von zwei GoMaxima-Instanzen<sup>12</sup> und einem Reverse Proxy<sup>13</sup> bearbeitet.

Für einen Vergleich von GoMaxima und MaximaPool wurden zwei neue virtuelle Server (Docker<sup>14</sup> Container<sup>15</sup> mit identischer Hardware) eingerichtet. Ein Server wird für GoMaxima verwendet, der andere Server für MaximaPool inklusive aller verfügbarer Optimierungen<sup>16</sup>. Beide virtuellen Server wurden auf demselben Computer<sup>17</sup> ausgeführt. In einem einfachen Vergleich wurden die Maxima-Anfragen durch ein Python-Skript mit einfachen mathematischen Ausdrücken simuliert, um den Einfluss von Maxima auf die

<sup>12</sup> virtuelle Server, zwei virtuelle CPU-Kerne, 4 GB RAM, 10 GB virtuelle SSD, Debian 9

<sup>13</sup> nginx <https://www.nginx.com/>

<sup>14</sup> Docker version 18.09.1, build 4c52b90 <https://www.docker.com/>

<sup>15</sup> GoMaxima v0.8: Offizieller Debian 9 Docker Container [https://hub.docker.com/\\_/debian/](https://hub.docker.com/_/debian/), Maxima 5.38.1-8+b1 MaximaPool: MaximaPool Docker Container der Uni-Halle <https://github.com/uni-halle/maximapool-docker>, latest (24.1.2019), Maxima 5.41.0-Linux

<sup>16</sup> Inklusive aller verfügbarer Optimierungen für MaximaPool:  
[https://github.com/math/moodle-qtype\\_stack/blob/master/doc/en/CAS/Optimising\\_Maxima.md](https://github.com/math/moodle-qtype_stack/blob/master/doc/en/CAS/Optimising_Maxima.md)

<sup>17</sup> Macbook Pro 2018, MacOS Mojave, 2,3 GHz Intel Core i5, 16 GB 2133 MHz LPDDR3



Messungen zu minimieren. Für jeden Server wurden 11 Testläufe mit 30 simulierten konkurrierenden Benutzern durchgeführt. Jeder Benutzer stellt dabei eine steigende Anzahl an Anfragen (1 bis 1001 in Schritten von 100). Es wurde die Zeit zur Bearbeitung aller Anfragen in Relation zur Anzahl der Anfragen gemessen. Die Ergebnisse zeigen, dass GoMaxima die Anfragen in diesem Test ungefähr um den Faktor fünf schneller als MaximaPool bearbeitet hat. Die Ergebnisse sind in Abbildung 6 dargestellt.

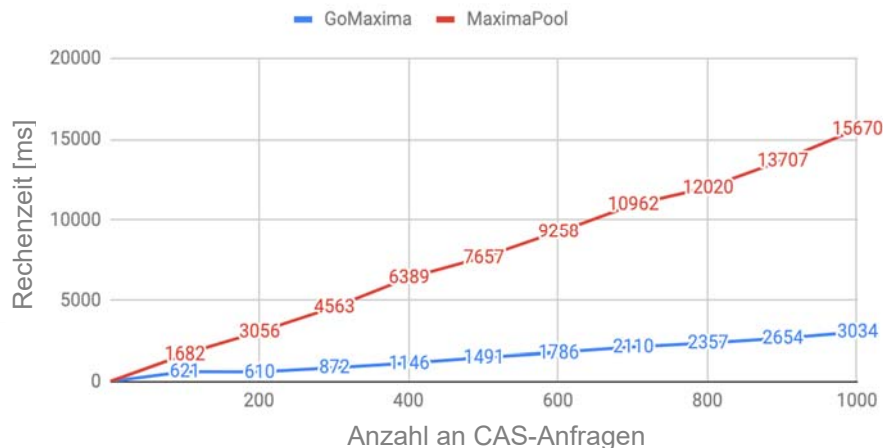


Abb. 6: Vergleich GoMaxima - MaximaPool an der HAW Hamburg mit 30 simulierten Benutzern und einfachen Maxima-Anfragen.

In einer weiteren Erprobung wurde GoMaxima an der Hochschule Karlsruhe im Zusammenwirken mit dem Lernmanagement ILIAS<sup>18</sup> anhand simulierter Benutzer getestet und mit MaximaPool verglichen. Für den Test wurde das ILIAS-Produktivsystem<sup>19</sup> und ein physikalischer Server<sup>20</sup>, auf dem GoMaxima und MaximaPool liefen, verwendet. Die Anfragen wurden mithilfe eines spezialisierten Werkzeugs zur Ausführung von Lasttests<sup>21</sup> simuliert. Der erste Testdurchlauf erfolgte unter Verwendung von MaximaPool, der zweite mit GoMaxima. Für den Test wurden mehrere Seiten von ILIAS durch die simulierten Benutzer aufgerufen und die Antwortzeiten gemessen. Es ist zu beachten, dass nicht alle diese Seitenaufrufe in Anfragen an MaximaPool oder GoMaxima resultierten. Als Ergebnis der Erprobung ist erkennbar, dass GoMaxima Anfragen bis zu 30-mal schneller bearbeitet als MaximaPool, im Durchschnitt liegt der Verbesserungsfaktor bei 20. Bei Anfragen, deren Antwortzeiten fast gleich sind, sind weder GoMaxima noch MaximaPool involviert. In der Abbildung 7 sind diese Ergebnisse

<sup>18</sup> <https://www.ilias.de/>

<sup>19</sup> System IBM x3650 M4, Intel(R) Xeon(R) CPU E5-2650, 32 Kerne, 64 GB RAM, HD 3 TB, SSD 64 GB -> Nur Datenbank, ILIAS Version 5.3.7, nginx 1.14.0, Ubuntu Server 16.04

<sup>20</sup> VMWare Virtual Machine, 2 CPU Cores (Host Machine CPU Core i5-3470 3.20GHz), 2 GB RAM, 20 GB HDD, Maxima 5.37.2, MaximaPool 1.6, GoMaxima 0.8, Ubuntu Server 16.04

<sup>21</sup> Apache JMeter <https://jmeter.apache.org>

in einer logarithmischen Skalierung dargestellt. Die in der Abbildung unten rot markierten Anfragen sind STACK/Maxima-Anfragen, wovon eine mit konkreten Zeitangaben beschrieben ist und den Faktor 30 erkennen lässt.

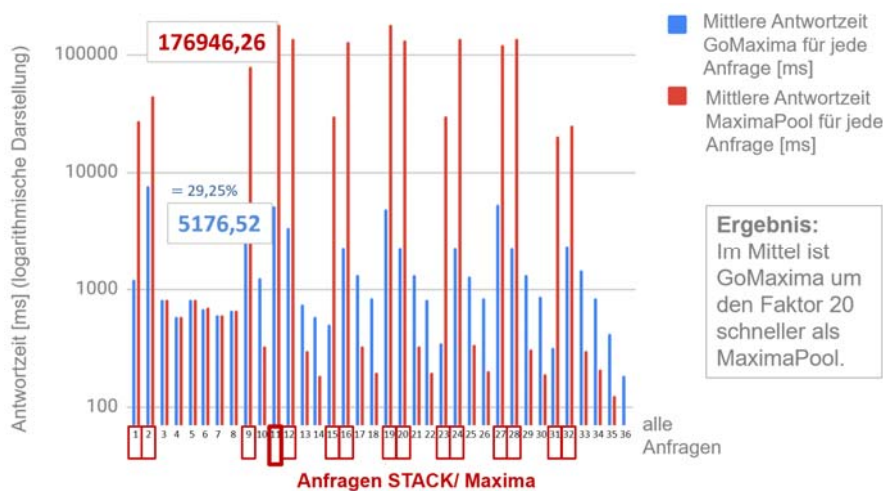


Abb. 7: Vergleich von GoMaxima und MaximaPool in einer Laborstudie an der HS Karlsruhe

#### 4. Zusammenfassung und Ausblick

In diesem Beitrag wurde die Entwicklung von GoMaxima vorgestellt, welches als Alternative zu MaximaPool eine performante Verwendung des Computeralgebrasystems Maxima zur Auswertung von STACK-Fragen ermöglicht. GoMaxima wird seit einigen Semestern sowohl in der Online-Lernumgebung viaMINT im Bereich der Vorkurse als auch für vorlesungsbegleitende Online-Zwischentests erfolgreich eingesetzt und liefert Performanceverbesserungen je nach Anfragetyp um Faktoren von 5 bis 30, im Mittel um den Faktor 20. In weiteren Moodle-STACK-Tests sowie neuen Verwendungsszenarien auch mit anderen Systemen sowie in Kooperation mit anderen Hochschulen sollen vertiefte Erfahrungen gesammelt und Evaluationen durchgeführt werden.

#### Literatur

- [Ka16] Kallweit, M.: CAS-unterstütztes Assessment von Mathematik, Computeralgebra-Rundbrief: Vol. 59. Berlin: Gesellschaft für Informatik e.V., Fachgruppe Computeralgebra, 2016. <https://dl.gi.de/bitstream/handle/20.500.12116/3094/car59-4.pdf?sequence=1&isAllowed=y>

- [La18-1] Landenfeld, K., Göbbels, M., Hintze, A., Priebe, J.: A Customized Learning Environment and Individual Learning in Mathematical Preparation Courses. In J. Silverman & V. Hoyos (Eds.): Distance Learning, E-Learning and Blended Learning in Mathematics Education - International Trends in Research and Development (S. 93-111), ICME-Monographs. Cham: Springer.
- [La18-2] Landenfeld, K., Priebe, J., Wendt, M.: Möglichkeiten zur Auffrischung der schulischen Kenntnisse durch den Einsatz einer Online-Lernumgebung. In Greifswalder Beiträge zur Hochschullehre - Erleichterung der Studieneingangsphase, Ausgabe 9 (Oktober 2018), S. 59-72. [https://www.uni-greifswald.de/fileadmin/uni-greifswald/2\\_Studium/2.1\\_Studienangebot/2.1.4\\_Qualitaet\\_in\\_Studium\\_und\\_Lehre/interStudies/UniBroschu\\_re\\_Web\\_final.pdf](https://www.uni-greifswald.de/fileadmin/uni-greifswald/2_Studium/2.1_Studienangebot/2.1.4_Qualitaet_in_Studium_und_Lehre/interStudies/UniBroschu_re_Web_final.pdf)
- [Sa13] Sangwin, C.: Computer Aided Assessment of Mathematics Using STACK. Oxford University Press, Oxford, 2013.