

Universität Hannover
Fakultät für Elektrotechnik und Informatik
Institut für Praktische Informatik
Fachgebiet Software Engineering

**Analyse und Bewertung von
Softwarepaketen zur Linearen Optimierung
für ein Unternehmen der Finanzbranche**

Masterarbeit
im Studiengang Angewandte Informatik

von
Thi Phuong Thao Nguyen

Prüfer: Prof. Dr. Kurt Schneider
Zweitprüfer: Prof. Dr. Michael H. Breitner

Hannover, 14. Oktober 2005

Abstract

Die neue Eigenkapitalvereinbarung Basel II gibt Vorschriften vor, nach denen Kreditinstitute für risikobehaftete Kreditgeschäfte ein angemessenes haftendes Eigenkapital unterlegen müssen. Diese Anforderung an Eigenkapital können durch eine optimale Verteilung der Sicherheiten auf die Kreditgeschäfte minimiert werden. Eine optimale Verteilung kann durch das Lösen von Gemischt Ganzzahligen Linearen Optimierungsproblemen gefunden werden. Der Branch-And-Bound-Algorithmus, der bekannteste Algorithmus zur Gemischt Ganzzahligen Linearen Optimierung, ist in zahlreichen Softwareprodukten umgesetzt worden. Ziel der Arbeit ist, durch eine systematische Analyse eine objektive Aussage zu treffen, welches Produkt das am besten geeignete für den Einsatz im Projekt Sicherheiten-Management-System des Partnerunternehmens ist.

Bei dieser Auswahl wurde GQM (Goal-Question-Metric) als Methode eingesetzt. Zuerst wurden die Anforderungen an das Softwareprodukt bestimmt. Mit Hilfe entsprechend formulierter Fragen wurden dann die Anforderungen auf prüfbare Kriterien und messbare Metriken abgeleitet. Der Vergleich zwischen den ermittelten Größen der Metriken und Kriterien ergab eine eindeutige Aussage.

Die GQM erwies sich als eine sehr gute Methode für die Software-Auswahl. Die einmal erarbeiteten Fragen können in den künftigen Auswahlprozessen wieder verwendet werden. Die abgeleitete Kriterien, Metriken sowie das Messsystem und Auswertungswerkzeuge sind jedoch nicht übertragbar. Die Wiederverwendbarkeit der GQM-Methode liegt daher mehr in der Methodik und weniger in den Automatisierungswerkzeugen.

Inhaltsverzeichnis

Abstract	V
Abkürzungsverzeichnis	VIII
Abbildungsverzeichnis	IX
Tabellenverzeichnis	X
1 Einführung	1
2 Optimierung von Creditsicherheiten	4
2.1 Einflussfaktoren des RWA.....	5
2.2 Formulierung des Optimierungsproblems.....	11
3 Sicherheiten-Management-System der FinanzIT	14
4 Begriffe der LO und MILP	17
4.1 Definitionen.....	17
4.2 Simplex-Verfahren	18
4.3 Branch-And-Bound	22
4.4 Presolve	24
4.5 Scaling-Strategie	25
5 Systematisches Vorgehen	27
5.1 Vom Ziel zu den Metriken	29
5.1.1 Anforderungen aus bankfachlicher Sicht	31
5.1.2 Anforderung aus technischer Sicht.....	33
5.1.3 Anforderung aus betriebswirtschaftlicher Sicht	38
5.1.4 Zusammenfassung der Auswahlkriterien	40
5.2 Überblick über frei verfügbaren MILP-Software.....	40
5.2.1 BonsaiG	41
5.2.2 GLPK.....	42
5.2.3 lp_solve	42
5.2.4 MINTO.....	42
5.2.5 SYMPHONY.....	42
5.3 Code-Analyse bei den Software-Paketen des ersten Kandidatenkreises.....	43
5.3.1 GLPK.....	44
5.3.2 lp_solve	45
5.4 Testdaten	46
5.5 Messplan und Messsystem	48
5.6 Vergleichsmessungen.....	53
5.6.1 Definition des Vergleichsoperators	53
5.6.2 Auswahl der besten Konfiguration für GLPK 4.8.....	56
5.6.3 Auswahl der besten Konfiguration für lp_solve.....	57
5.6.4 Wahl des MILP-Lösers	60

5.7	Messungen auf dem Host	60
5.7.1	Anpassung des Quellcodes für den Host unter z/OS.....	62
5.7.2	Messergebnisse.....	64
6	GQM-Methode in einem Software-Auswahl-Prozess	66
7	Zusammenfassung und Ausblick	71
Anhang A	Erstellung eines Testdatenbestands	XI
Anhang B	Gesamtes Optimierungsproblem	XV
	Nomenklatur der Formelelemente.....	XV
	Nomenklatur und Zuordnung der SMS-Inputdaten.....	XIX
	Das Gemischt Ganzzahlige Lineare Optimierungsproblem.....	XXVI
Anhang C	Beiliegende CD.....	XXVIII
	Literaturverzeichnis	XXIX

1 Einführung

Kreditinstitute spielen eine besondere Rolle in der Volkswirtschaft. Im Geldkreislauf sind sie der Mittler zwischen Einlegern und Kreditnehmern. Einerseits sammeln sie Einlagen, auch solche mit kleinen Beträgen und kurzen Fristen, andererseits vergeben sie Kredite mit größeren Beträgen zu längeren Fristen um Zinsen zu verdienen. Als Kreditgeber müssen die Kreditinstitute alle mit den Kreditgeschäften verbundenen Risiken tragen. Sie müssen diese Risiken so kalkulieren, dass sie jederzeit die Pflichten gegenüber ihren Gläubigern, welche ihnen ihr Vermögen anvertrauen, erfüllen können.

Der Ertragsdruck im Kreditgeschäft stieg in den letzten Jahren erheblich. Das Bemühen nach Rentabilität kann im Gegensatz zu dem Streben nach Liquidität und Sicherheit stehen. Ein Kreditinstitut kann höhere Risiken eingehen, um höhere Zinsen zu erzielen. Die Insolvenz der BFI Banken Dresden im 2003 zeigt, dass der Konkurs eines Kreditinstituts nicht auszuschließen ist. Um die Risiken im Kreditgeschäft zu managen, wurden Basler Aufsichtsregeln für Kreditinstitute geschaffen.

Der Basler Ausschuss für Bankenaufsicht wurde durch die Präsidenten der Zentralbanken der G10 Länder gegründet und aus Vertretern der Zentralbanken und Bankaufsichtsbehörden dieser Länder zusammengesetzt. Im Jahr 1997 hat er die Eigenkapitalvereinbarung (Basel I) verabschiedet. Diese Vereinbarung konzentrierte sich auf das gesamte Eigenkapital einer Bank, welches für die Begrenzung des Insolvenzrisikos einer Bank und der möglichen Kosten für die Einleger beim Konkurs der Bank entscheidend ist. Mitte 2004 wurde die neue Eigenkapitalvereinbarung (Basel II) veröffentlicht, welche die Kapitalanforderungen an Banken stärker als bisher vom ökonomischen Risiko abhängig macht und neuere Entwicklungen an den Finanzmärkten sowie im Risikomanagement der Institute berücksichtigt.

Die Basler Anforderungen an die Eigenkapitalunterlegung von risikobehafteten Kreditgeschäften geben die Rahmenbedingungen vor, innerhalb derer Sicherheiten zur Minderung des erforderlichen Eigenkapitals eingesetzt werden können. Allerdings gibt es für den Fall, dass mehrere Sicherheiten auf eine Gruppe von Kreditgeschäften verteilt werden können, keine eindeutigen Vorschriften, wie diese Verteilung erfolgen soll. Wenn die Reihenfolge der Verteilung der Sicherheiten auf die Kredite auch nicht vertraglich zwischen dem Kreditnehmer und Kreditinstitut festgelegt wurde, hat das Kreditinstitut an dieser Stelle die Möglichkeit, seine Eigenkapitalanforderung durch eine optimale Verteilung der Sicherheiten auf die Kredite zu minimieren. Als IT-Dienstleister der S-Finanzgruppe entwickelt die FinanzIT eine Anwendung mit dem Namen Werteverteilung, welche eine optimale Verteilung der Sicherheiten auf die Kredite jede Nacht berechnet. Mit Hilfe dieser Anwendung können die Kunden der FinanzIT, die Sparkassen, die Anforderung an ihrem Eigenkapital minimieren. Die Anwendung Werteverteilung wird im Rahmen des Projekts Sicherheiten-Management-System (SMS) entwickelt. Sie besteht im Wesentlichen aus zwei Modulen: Heuristik und Lineare Optimierung. Das Heuristik-Modul steuert die Verteilung, wenn die Reihenfolge der Verteilung vertraglich festgelegt wurde. Im anderen Fall wird eine optimale Verteilung durch das Lineare-Optimierung-Modul (kurz: LO-Modul) berechnet.

Das LO-Modul bekommt bankfachliche Daten als Eingabe. Anhand dieser Daten wird ein mathematisches Optimierungsproblem formuliert. Die Lösung dieses Problems wird zurück zu bankfachlichen Daten transformiert und diese werden ausgegeben. Zum Lösen des mathematischen Problems kann man ein Software-Paket anderes Herstellers

einsetzen. Das Thema der vorliegenden Arbeit ist die Auswahl eines frei verfügbaren Software-Pakets zum Lösen des mathematischen Optimierungsproblems im LO-Modul.

Der erste Schritt im Ablauf des LO-Moduls ist die Formulierung eines mathematischen Optimierungsproblems aus den bankfachlichen Daten. Das Prinzip dieser Formulierung wird im Kapitel 2 anhand von Beispielen erläutert. Aus diesen Beispielen ist zu erkennen, dass die zu optimierenden Probleme zur Klasse „Gemischt Ganzzahlige Lineare Optimierung“ gehören. Somit werden frei verfügbare Pakete für die Lösung von Problemen dieser Klasse gesucht. In dieser Arbeit werden solche Pakete MILP-Löser¹ genannt.

Das LO-Modul soll kein eigenständiges Programm sein. Es wird von einem anderen Modul aufgerufen. Im Kapitel 3 wird erläutert, in welcher Umgebung das LO-Modul betrieben werden soll. Der MILP-Löser muss wie das LO-Modul in dieser Umgebung lauffähig sein.

Kapitel 4 geht auf die Lineare Optimierung und die Gemischt Ganzzahlige Lineare Optimierung ein. Insbesondere wird untersucht, welche Faktoren die Ergebnisse und die Effizienz der Algorithmen zur Lösung von Problem dieser Klasse beeinflussen. Während der Auswahl eines MILP-Lösers müssen diese Einflussfaktoren betrachtet werden.

Die GQM-Methode² wird angewendet um einen MILP-Löser für das LO-Modul auszuwählen. Das systematische Vorgehen wird im Kapitel 5 beschrieben. Zuerst werden die Ziele der Analyse bestimmt. Mit Hilfe von Fragen werden die Ziele dann in Metriken konkretisiert, welche messbar sind und die Grundlage für die Entscheidung bilden. Diese Ableitung ist im Abschnitt 5.1 beschrieben.

Abschnitt 5.2 gibt einen Überblick von frei verfügbaren MILP-Lösern. Die Kandidaten für die Auswahl werden im Abschnitt 5.3 näher beschrieben.

Die Qualität der Auswahl hängt sehr stark von der Qualität der Testdaten ab. Da keine Echt Daten für die Messungen vorhanden ist, müssen die Testdaten erst simuliert werden. Die simulierten Testdaten müssen so erstellt werden, dass die Ergebnisse der auf diesen Daten basierten Messungen ebenso aussagekräftig sind wie Messungen mit Echt Daten. Die Erstellung des Testdatenbestands wird im Abschnitt 5.4 beschrieben.

Nachdem der Kandidatenkreis abgegrenzt ist und die Testdaten erstellt wurden, wird das Arbeitspaket für die Auswahl im Abschnitt 5.5 erstellt. Zu diesem Arbeitspaket gehören sowohl Programmier- als auch Messaufgaben. Die Architektur des LO-Moduls und des Messsystems wird im selben Abschnitt beschrieben. Der Messplan wird ebenfalls dort vorgestellt. Zuerst werden Messungen auf einem Solaris-System durchgeführt. Mit diesen Vergleichsmessungen werden die MILP-Löser passend für die Probleme der Werteverteilung konfiguriert. Dann werden sie miteinander verglichen. Zum Schluss wird der beste MILP-Löser auf das Zielsystem, den Großrechner der FinanzIT, portiert. Hier werden Messungen ausgeführt um festzustellen, ob dieser MILP-Löser die Anforderungen der Werteverteilung erfüllen kann. Die Ergebnisse der Vergleichsmessungen sowie der Messungen auf dem Host werden in den darauf folgenden Abschnitten präsentiert.

¹ MILP: Mixed Integer Linear Programming

² GQM: Goal-Question-Metric

Software-Auswahl ist eine typische Aufgabe im IT-Bereich. Im Kapitel 6 wird diskutiert, wie weit sich die GQM-Methode, welche zur Auswahl eines MILP-Löser für die Werteverteilung angewendet wurde, in einem anderen Software-Auswahl-Prozess übertragen lässt.

Im Kapitel 7 werden die Ergebnisse dieser Arbeit zusammengefasst. Es wird u.a. eine Aussage getroffen, welcher MILP-Löser sich im LO-Modul am besten einsetzen lässt. Ferner werden verbleibende Optimierungsmöglichkeiten für das LO-Modul diskutiert.

7 Zusammenfassung und Ausblick

Die Basler Anforderungen an die Eigenkapitalunterlegung von risikobehafteten Kreditgeschäften können durch eine optimale Verteilung der Sicherheiten auf die Kreditgeschäfte minimiert werden. Im Sicherheiten-Management-System der FinanzIT steuert das Heuristik-Modul diese Verteilung, wenn die Reihenfolge der Verteilung vertraglich festgelegt wurde. Im anderen Fall wird eine optimale Verteilung durch das LO-Modul berechnet.

Das LO-Modul verarbeitet die Daten in drei Schritten: Formulieren eines mathematischen Problems aus den eingegebenen bankfachlichen Daten, Lösen dieses Problems und Transformation der Lösung zurück in die bankfachlichen Daten. In der Entwicklung des LO-Moduls wird der Ansatz verfolgt, ein frei verfügbares Software-Paket für das Lösen mathematischer Probleme einzusetzen. In der vorliegenden Arbeit wurde die Auswahl eines Software-Paketes für diese Aufgabe behandelt.

Die Anforderung an Eigenkapital eines Kreditinstituts erreicht ihr Minimum, wenn die RWA der einzelnen Sicherheitenetze dieses Instituts minimal sind. Für jedes Sicherheitenetz wird ein Optimierungsproblem formuliert, dessen Zielfunktion das RWA des Netzes ist. Die Nebenbedingungen des Optimierungsproblems entstehen aus den gesetzlichen Vorschriften. All diese Bedingungen sind linear. Einige Variablen des Optimierungsproblems dürfen nur den Wert Null oder eins annehmen. Das aus den Daten eines Netzes formulierte Problem ist somit ein MILP-Problem. Folglich wird ein Software-Paket zum Lösen MILP-Probleme gesucht.

Die an den MILP-Löser gestellten Anforderungen wurden mit Hilfe von Zielhierarchien und Verfeinerungsbögen bestimmt. Aus technischen und betriebswirtschaftlichen Gründen wurden nur Software-Pakete analysiert, deren C-Quellcode frei verfügbar war. Zu dem ersten Kandidatenkreis gehörten die Pakete, welche ohne zusätzliche Implementierung aus einem C-Programm aufrufbar sind. Hierzu gehörten zwei Produkte: GLPK 4.8 und lp_solve 5.1. Produkte des zweiten Kandidatenkreises waren Software-Pakete, für deren Einsatz in der Werteverteilung noch die Implementierungen von Schnittstellen benötigt werden. Hierzu gehörten BonsaiG und SYMPHONY. Da Entwicklungskosten gespart werden sollten, beschränkte sich die Auswahl vorerst auf den ersten Kandidatenkreis.

Ergebnisse und Effizienz eines MILP-Lösers hängen von den Einstellungen von Scaling- und Presolve-Strategien, Einstellungen von Simplextyp und Pricing-Strategien sowie Einstellungen von Branching- und Suchstrategien ab. Die Analyse des Codes von GLPK 4.8 und lp_solve 5.1 zeigte, dass es viele Möglichkeiten gibt um diese Pakete einzustellen. Für jedes Paket musste eine Einstellung gesucht werden, mit der die Aufgaben der Werteverteilung am besten und effizientesten bearbeitet werden. Der Vergleich zwischen den Alternativen wurde mit Hilfe von zwei Metriken durchgeführt: Laufzeitbedarf und Summe der Minima. Dieser Vergleich ist nur sinnvoll, wenn keine der Alternativen ausgeschlossen werden soll. Wenn für einen Anrechnungskreis keine Lösung gefunden wird, oder wenn diese Lösung die Zulässigkeitsprüfung nicht besteht, wird die eingesetzte Einstellung ausgeschlossen. Unter den bleibenden Alternativen wurden zuerst die Summen der Minima verglichen. Unter den Alternativen mit besten Minima wurde diejenige ausgewählt, welche minimale Laufzeit benötigte. Diese Alternative ist somit der beste und effizienteste Löser der aus dem Testdatenbestand formulierten MILP-Probleme.

Die Qualität der Auswahl hängt sehr stark von der Qualität der Testdaten ab. Weil keine Echtdata für die Messungen verfügbar sind, wurde der Testdatenbestand aus dem Datenbestand eines Testinstituts mit Hilfe eines Zufallsgenerators erstellt. Die Messungen mit diesem Datenbestand sollen die gleiche Aussagekraft wie die Messungen mit Echtdata haben.

Das Zielsystem, auf dem MILP-Probleme gelöst werden sollen, ist ein Großrechner unter z/OS. Weil die Portierung eines Software-Paketes in z/OS-Umgebung mit größerem Aufwand verbunden ist, sollen die Vergleichsmessungen auf einem Solaris-System durchgeführt werden. Nur das beste Software-Paket wird in die z/OS-Umgebung portiert. Dort wird geprüft, ob dieses Paket alle Anforderungen erfüllt.

Die Messungen zeigten, dass GLPK 4.8 und lp_solve 5.1 mit ihren besten Einstellungen die gleichen Summen von Minima ausgeben. Der Laufzeitbedarf von GLPK 4.8 ist deutlich geringer als der von lp_solve 5.1. Die Messungen auf dem Großrechner zeigten, dass GLPK 4.8 alle Anforderungen des Projekts erfüllt und somit zum Lösen von MILP-Problemen im LO-Modul eingesetzt werden kann.

Die Entscheidung für GLPK 4.8 wurde getroffen ohne dass die Analyse des zweiten Kandidatenkreises durchgeführt wurde. Es könnte sein, dass es in diesem Kreis noch Pakete gibt, welche für die Werteverteilung besser geeignet sind als GLPK 4.8. Allerdings ist diese Analyse nicht nur mit zusätzlichen Messungen, sondern auch mit Implementierung von Schnittstellen und Portierung eines Paketes auf den Host verbunden. GLPK 4.8 benötigt für den gesamten Datenbestand eines Kreditinstituts ca. 31 Sekunden pro Nacht. Dieser Zeitbedarf liegt weit unter der oberen Grenze von 12 Minuten. Unter Beachtung des Kosten-Nutzen-Aspekts wurde auf die Analyse von BonsaiG und SYMPHONY verzichtet.

Die Datenstruktur LpSystem wurde in das LO-Modul eingeführt, um die Bankfachlichkeit von dem Lösen mathematischer Probleme zu trennen. Mit dieser Trennung hat man den Entwicklungsaufwand bei der Auswahl eines MILP-Lösers gespart: Anstatt für jedes MILP-Paket eine Prozedur zur Transformation der bankfachlichen Daten in paketspezifischen Datenstrukturen zu implementieren, hat man das Formulieren des MILP-Problems aus den bankfachlichen Daten wiederverwendet. Das Ergebnis dieses Formulierens ist eine Instanz der Datenstruktur LpSystem. Die erzeugte Instanz von LpSystem wird in dem darauf folgenden Schritt in paketspezifischen Strukturen umgewandelt. Die Transformation der Lösung des mathematischen Problems in bankfachliche Daten geschieht ebenfalls in zwei Schritten über die Struktur LpSystem. Nachdem GLPK 4.8 zum Lösen mathematischer Probleme ausgewählt wurde, könnte man die Laufzeit und den Speicherbedarf des Moduls verbessern, indem man auf die Datenstruktur LpSystem verzichtet und die Datenstruktur von GLPK 4.8 als Ausgabe des Formulierens MILP-Probleme nutzt. Die Lösung kann dann direkt von der Datenstruktur von GLPK in bankfachliche Daten transformiert werden.

Auswahl eines Software-Paketes zum Lösen eines klassischen Problems ist eine Aufgabe, welche in der Software-Entwicklung häufig vorkommt. Die GQM erwies sich als eine sehr gute Methode für solchen Aufgaben. Mit dieser Methode werden zielorientierte Messungen und Datenauswertungen durchgeführt. Die einmal erarbeiteten Fragen können in den künftigen Auswahlprozessen wieder verwendet werden. Die abgeleitete Kriterien, Metriken sowie das Messsystem und Auswertungswerkzeuge sind jedoch nicht übertragbar. Die Wiederverwendbarkeit der GQM-Methode liegt daher mehr in der Methodik und weniger in den Automatisierungswerkzeugen.