

9 Ausblick

Diese Arbeit konnte auf dem Weg zu einem umfassenden Gesamtkonzept für das Schnittstellensystem von Solvern für lineare mathematische Optimierungsprobleme eine Reihe von grundlegenden Aspekten behandeln, und sie zeigt auch anhand der beiden größeren Implementationskomplexe „Komponentenbibliothek“ und „MOPS Studio“ deren tatsächliches Funktionieren in realen Anwendungen.

Nun ist es notwendig, die aus Aufwandsgründen zunächst etwas simpler gehaltene prototypische Implementation dem umrissenen Konzept anzugleichen, sowie weitere Schnittstellen zu implementieren. Aus Praxis-Sicht wäre vor allem das häufig benötigte .NET-Interface wünschenswert. Weiterhin müssten Anbindungen für weitere Optimierer, insbesondere für CPLEX und Open-Source-Solver, geschaffen werden.

Visionär wäre die Möglichkeit, bestimmte, häufig vorkommende Modellierungssachverhalte, wie z.B. stückweise Linearisierung oder logische „Wenn-Dann“-Weichen, mithilfe entsprechender Klassen oder Komponenten zu modellieren. Dies könnte ein wichtiger Schritt sein, um Modellgeneratoren, die in herkömmlichen Programmiersprachen geschrieben sind, kürzer, intuitiver und stabiler zu machen.

Ein anderer Bereich, der in der Arbeit bereits angeschnitten wurde (Kapitel 6.8), der aber in zukünftigen Architekturen Entscheidungsunterstützender Systeme eine bedeutende Rolle spielen dürfte, ist die verteilte Optimierung im Rahmen Service Orientierter Architekturen (SOA). Die hier vorgenommenen Implementationen berücksichtigen diesen Aspekt nur ansatzweise, u.a indem die Hauptkomponente der COM-Komponentenbibliothek DCOM-fähig ist und so Optimierungsläufe auf entfernten Rechnern erlaubt. Wichtig wäre aber darüber hinaus die weitere Ausarbeitung der hier angerissenen Konzepte zur verteilten Optimierung mit Webservices sowie deren Umsetzung und Erprobung im betrieblichen Umfeld.

Auch in Zukunft ist zu erwarten, dass die verbreiteten mathematischen Optimierungssysteme Performance und Lösungsqualität noch weiter steigern können. Die Leistungsfähigkeit der Schnittstellen und die einfache Einbettbarkeit in betriebliche Anwendungssysteme dürften aber zukünftig angesichts schnellerer Entwicklungszyklen und begrenzter Entwicklerressourcen immer wichtiger werden. Auch wenn Performance das primäre Qualitätskriterium eines Solvers bleiben wird, so entscheiden über den kommerziellen Erfolg eines Optimierers immer mehr auch Schnittstellenqualität sowie verfügbare Tools, Modellierungssprachen und Entwicklungsumgebungen.

