Distributed data & resources models, tractability, and complexity

Author(s):
Schlude, Konrad

Publication Date:
2002

Permanent Link:

https://doi.org/10.3929/ethz-a-004484296 →

Diss. ETH No. 14794

Distributed Data & Resources: Models, Tractability, and Complexity

A dissertation submitted to the

SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZÜRICH

for the degree of

Doctor of Technical Sciences

presented by

Konrad August Schlude
Dipl.–Mathematiker, Albert–Ludwigs–Universität Freiburg

born May 24, 1968

citizen of
Germany

accepted on the recommendation of

Prof. Dr. Peter Widmayer, Examiner
Prof. Dr. Roger Wattenhofer, Co-Examiner

2002

Abstract

There are many examples for the fact that the growth of a system is limited if this system relies on important central instances for supply or control. In a growing system, the central instances turn into bottlenecks, thus making the system inefficient. Examples for this phenomenon can be found in computer science, in history, and even in the behavior of dinosaurs. Distributed systems have great potential to bypass such bottlenecks, but due to "friction" issues such as incomplete knowledge, we may not be able to exploit their potential.

In this thesis we show that some of these issues can be handled quite well in weak and realistic models. More specifically, we present and discuss three examples:

1. Distributing Rare Resources: Roman Domination and Win–Win
   Given a graph, servers (resources) should be placed on nodes to service a pair of requests that can occur at nodes. Since resources are rare, the number of used resources should be minimized. What is the computational complexity of this problems, and how much control, communication, or interaction is needed to organize the service?

   We give detailed characterization of the computational complexity of these problems. Especially, we show that these problems are NP–hard; and for Planar Roman Domination, a Polynomial Time Approximation Scheme (PTAS) is presented.

2. Distributed Data Structure
   Given a distributed system of processors/computers/servers with weak assumptions, is it possible to build and maintain a distributed data structure in this system?

   We propose a distributed dictionary that supports insert and search operations and that tolerates arbitrary single server crashes. In contrast to other proposals of distributed fault tolerant search structures, our solution works in an asynchronous and weak environmental setting.

3. Distributed Coordination: Point Formation
   There are n robots in the Euclidean plane that should move to one destination point. What is a minimal set of abilities that the robots have to have in order to complete this task?

   We define the concept of *contraction functions* and show that with this concept, the point formation problem can be solved in an asynchronous model.

The results show that these problems can be solved in weak and realistic settings. We will discuss these problems individually, but we will discuss the connections between them as well. Interestingly, we will find connections between the point formation problem and facility location as well.

Zusammenfassung

Es gibt viele Beispiele dafür, dass das Wachstum eines Systems beschränkt ist, falls dieses System auf zentralen Einheiten für Versorgung oder Kontrolle basiert. Wenn das System wächst, werden die zentralen Einheiten zu Engpässen, die das System ineffizient machen. Beispiele für dieses Phänomen finden sich in der Informatik, in der Geschichte und auch im Verhalten von Dinosauriern. Verteilte Systeme haben die Möglichkeit, solche Engpässe zu umgehen, aber auf Grund von "Reibungsverlusten", wie zum Beispiel unvollständigem Wissen, sind wir eventuell nicht in der Lage, diese Möglichkeiten auszuschöpfen.

Das Ziel dieser Arbeit ist es, zu zeigen, dass mit solchen "Reibungsverlusten" recht gut umgegangen werden kann. Dies soll an drei Beispielen gezeigt werden:

1. Verteilung rarer Ressourcen: Roman Domination und Win–Win
   Gegeben sei ein Graph, Server (Ressourcen) sollen auf Knoten des Graphs plaziert werden, um jedes Paar von Anfragen bedienen zu können, die an Knoten auftreten können. Da die Ressourcen rar sind, soll die Anzahl der verwendeten Ressourcen minimiert werden. Wie gross ist die Berechnungskomplexität, und wie viel Kontrolle, Kommunikation oder Interaktion wird benötigt, um die Bedienung der Anfrage zu organisieren?

   Wir präsentieren eine detailierte Charakterisierung der Berechnungskomplexität dieser Probleme. Insbesondere zeigen wir, dass diese Probleme NP-hart sind, und für Planares Roman Domination präsentieren wir ein Polynomielles Approximationsschema (PTAS).

2. Verteilte Datenstruktur
   Gegeben sei ein Computernetzwerk mit schwachen Modellannahmen. Ist es möglich, in einem solchen Netzwerk eine verteilte Datenstruktur zu unterhalten.

   Wir präsentieren ein verteiltes Wörterbuch, das Einfüge– und Suchoperationen ermöglicht und den Ausfall eines beliebigen Servers toleriert. Im Gegensatz zu anderen vorgeschlagenen verteilten fehlertoleranten Suchstrukturen arbeitet unser Lösungsvorschlag in einem stark asynchronen Modell.

3. Verteilte Koordination: Punkt-Formation
   Es seien n Roboter in der euklidischen Ebene verteilt. Die Roboter sollen sich an einem Punkt treffen. Was ist ein Mindestmass an Fähigkeiten, die die Roboter haben müssen, um diese Aufgabe zu bewältigen?

   Wir definieren das Konzept der *Zusammenziehfunktionen* und zeigen, dass das Punkt-Formations Problem mit diesem Konzept in einem asynchronen Modell gelöst werden kann.

Die Resultate zeigen, dass diese Probleme auch in realistischen Modellen gelöst werden können. Die Beispiele werden separat diskutiert, es werden aber auch die Zusammenhänge zwischen ihnen erklärt. Interessanterweise werden wir auch Zusammenhänge zwischen dem Punkt-Formations Problem und der Plazierungstheorie (facility location) finden.