Notes on the Communication Complexity of Multilateral Negotiation

N. Maudet

LAMSADE
Université Paris-Dauphine

28/02/05 (TFG-MARA)

Introduction

Once again...

- allocations of $|\mathcal{R}|$ resources among $|\mathcal{A}|$ agents,
- $\delta = (A, A')$ deals moving from allocation A to A',
- side-payments may enhance deals,
- local acceptability criteria (rationality),
- well-being of the society: utilitarian sw, Pareto optimality.

Lemma

$\delta = (A, A')$ rational iff $sw(A) < sw(A')$

Aspects of Complexity

- Computational complexity (see Paul's talk) can be analysed at the global level (from a designer's perspective), or at the local level (from an agent's perspective) *e.g* complexity of the decision problem "is there a sequence of 1-deals leading from A to A'"
- Communication complexity (this talk) aims at analysing the complexity of the negotiation process itself, regardless of the computational resources needed by the agent

Communication Complexity in the Literature

Two agents hold an n-bit string and their goal is to communicate in order to compute the value of a (boolean) function over these two strings. What is the minimal number of bits that need to be exchanged to do so? [Yao,1979]

- Communication complexity of a protocol
  maximal number of bits exchanged when following the protocol in the worst case
- Communication complexity of a function
  communication complexity of the best protocol that computes that function

Aspects of Communication Complexity

(1) How many deals are required to reach an optimal allocation?
   - communication complexity as number of individual deals

(2) How many dialogue moves are required to agree on one such deal?
   - affects communication complexity as number of dialogue moves

(3) How expressive a communication language do we require?
   - affects communication complexity as number of bits exchanged

Expressivness of the communication language

- Performatives of the protocol
  Minimum requirements: *propose*, *accept*, *reject*
  But we may want to add: *counter-proposal*, *justify*, ...
- Content language needed to specify the deals
  closely related to “bidding-languages” in CA
  (see also Jerome’s and Ulle’s talks)

Number of Deals (With Money, Utilitarian SW)

Upper bounds on the length of deal sequences

Theorem (Shortest path)

A single rational deal is sufficient to reach an allocation with maximal social welfare.

Proof.

Use Lemma. □

Theorem (Longest path)

A sequence of rational deals can consist of up to $|\mathcal{A}|^{|\mathcal{R}|} - 1$ deals, but not more.

Proof.

No allocation can be visited twice (same lemma) and there are $|\mathcal{A}|^{|\mathcal{R}|}$ distinct allocations $\Rightarrow$ upper bound follows ✓ □

Number of Deals (Without Money, Pareto Optimality)

Upper bounds on the length of deal sequences

Theorem (Shortest path)

A single cooperative rational deal is sufficient to reach a Pareto optimal allocation.

Theorem (Longest path)

A sequence of rational deals can consist of up to $|\mathcal{A}| \cdot (2^{|\mathcal{R}|} - 1)$ deals, but not more.

Proof.

Each deal requires at least one agent having a strict improvement. No agent can hold a bundle he held previously and changed (strict improvement). Suppose every single agent has as many improvement as possible. □

Tightness of the bounds

Are these bounds tight?
(*i.e* can we really find a scenario where that many deals would be needed to reach the optimal allocation?)

- Framework With Money: yes
  reason: it is possible to construct utility functions such that distinct allocations have disctinct social welfare
- Framework Without Money: no
  reason: each deal involves at least two agents modifying their bundle

Further Results (Restriction on Utility Functions)

(Note that the *feasibility* of reaching the optimal allocation in these cases has been proved in [AAMAS03]).

Additive Domains: number of rational one-resource deals with side payments to reach an allocation with maximal sw

- *Shortest path:* $\leq |\mathcal{R}|$
- *Longest path:* $\leq |\mathcal{R}| \cdot (|\mathcal{A}| - 1)$

0-1 Domains: number of rational one-resource deals without side payments to reach an allocation with maximal sw:

- *Shortest and longest path:* $\leq |\mathcal{R}|$

Open Questions

- tight upper bound in the framework w/o money?
- further restrictions on classes of utility functions
- number of deals in the case of non-standard sw measures?
- connections to communication complexity à la Yao?

How many dialogue moves to agree on a deal?

- Assuming our basic (propose;(accept | reject))* protocol
- Assuming a proposed and rejected deal cannot be proposed again during the same step

Upper bound related to the number of possible deals to consider at each of the negotiation process

- In general, number of allocations $|\mathcal{A}|^{|\mathcal{R}|} - 1$
- Restriction on the number of resources in a deal is an improvement, but only for very small values, e.g. one-resource-at-time $< 2.|\mathcal{R}||\mathcal{A}|^2$
- Other kinds of restriction one may think of? Nested utility functions, inspired by [Rothkopf,95]

Nested Structures

Recall the k-additive representation of utility functions...

Definition (Nested utility functions)

Nested utility functions iff no overlaping terms in k-additive form

Example

$u = 2r_1 + 3r_2 + 1r_3 + 1r_4 + 8r_3r_4$ is 2-additive and nested

Example

$u = r_1r_2 + r_2r_3$ is also 2-additive but not nested

Note that these functions can be represented as trees

Exploiting Nested Utility Functions

Definition (B-deals)

A single receiver gets an entire bundle B (possibly from many senders), as it appears in the k-additive representation

Note that there are at most $2 \times |\mathcal{R}|$ B-deals to consider!

However, allowing *any* (but only) B-deals does not allow to reach the optimal allocation any more.

Example

Let $u_1 = 2r_1$, $u_2 = 2r_2$, $u_3 = 3r_1r_2$, and $\{r_1, r_2\}$ to a_3 as initial allocation. There are no B-deal possible, still this allocation is not optimal (stuck in local optimum)

Mediating the Process

So far agents were on their own to contract deals. Now we introduce a system agent to support them in the process.

Mediated Negotiation

A *system agent* will influence the negotiation by using side-payments (similar to a bank).

- the system agent needs to know agents' utility function to compute a payment function
- the payment function can be parametrized (e.g. selfish)
- the system agent will sometimes loose money, sometimes win money (on single contracted deals)

Theorem

The system agent can be set up s.t. it globally earns money during the whole negotiation process (if it reaches an optimal allocation)

Guiding the Process: A Tree-Climbing Protocol

Incremental Protocol

Start with the smallest bundles, then allow incrementally biggest bundles.

- agents communicate their preferences to the system agent
- $s \leftarrow 1$
- repeat until $s > |\mathcal{R}|$
  - restrict deals to B-deals of size s
  - compute payment(s), contract deal(s)
  - if no more deal possible then $s \leftarrow s + 1$

Theorem

The Tree-Climbing Protocol allows to reach an allocation with maximal sw when agents use nested utility functions

Conclusion

- communication complexity (vs. computational complexity)
- can be assessed at different levels:
  - number of deals per negotiation,
  - number of dialogue moves per deal,
  - number of bits per move
- taming the complexity: it is possible to put restrictions on the type of deals, but then either
  - we are able to find domains still allowing to guarantee optimality (e.g. modular, k-separable), and agents can still negotiate autonomously, or
  - we are not and then we might help agents to avoid getting stuck in a local optimum by supporting the negotiation (system agent, tailor-made protocols)