Multi-Agent Systems

Sept. 2000, Bahia Blanca

University Nacional del Sur

- **Last two weeks in September**.
- **Tentative Dates:** Tuesday, Sept. 19th, Thursday, Sept. 21st, Friday, Sept. 22nd, Tuesday, Sept. 26th, Thursday, Sept. 28th, Friday, Sept. 29th.
- **Time:** From 4–6 pm, unless otherwise indicated.
- Lecture Course is on theoretical issues, emphasis on mathematical-logical foundations.

Overview

1. Introduction, Terminology
2. Three Basic Architectures
3. Logic Based Architectures
4. Distributed Decision Making
5. Contract Nets, Coalition Formation

Chapter 5. Contract Nets, Coalition Formation

5.1 General Contract Nets

5.2 OCSM-Nets

5.3 Abstract Coalition Formation

5.4 Payoff Division

5 Contract Nets, Coalition Formation

119-1

5.1 General Contract Nets

How to distribute tasks?

- Global Market Mechanisms. Implementations use a single centralized mediator .
- **Announce, bid, award** -cycle. Distributed Negotiation .

We need the following:

1. **Define a task allocation problem in precise terms.**
2. **Define a formal model for making bidding and awarding decisions.**

Definition 5.1 (Task-Allocation Problem)
A task allocation problem is given by

1. *a set of tasks* T,
2. *a set of agents* A,
3. *a cost function* $\mathbf{cost}_{\mathsf{i}} : 2^T \longrightarrow \mathbb{R} \cup \{\infty\}$ *(stating the costs that agent* i *incurs by handling some tasks), and*
4. *the initial allocation of tasks*

$$\langle T_{\mathsf{1}}^{init}, \ldots, T_{|\mathsf{A}|}^{init} \rangle,$$

where $T = \bigcup_{\mathsf{i} \in \mathsf{A}} T_{\mathsf{i}}^{init}$, $T_{\mathsf{i}}^{init} \cap T_{\mathsf{j}}^{init} = \emptyset$ *for* $\mathsf{i} \neq \mathsf{j}$.

Definition 5.2 (Accepting Contracts and Allocating Tasks)
A contractee $\mathbf{q}$ *accepts a contract if it gets paid more than the marginal cost of handling the tasks of the contract*

$$MC^{add}(T^{contract}|T_{\mathbf{q}}) =_{def} \mathbf{cost}_{\mathbf{q}}(T^{contract} \cup T_{\mathbf{q}}) - \mathbf{cost}_{\mathbf{q}}(T_{\mathbf{q}}).$$

A contractor $\mathbf{r}$ *is willing to allocate the tasks* $T^{contract}$ *from its current task set* $T_{\mathbf{r}}$ *to a contractee, if it has to pay less than it saves by handling them itself:*

$$MC^{remove}(T^{contract}|T_{\mathbf{r}}) =_{def} \mathbf{cost}_{\mathbf{r}}(T_{\mathbf{r}}) - \mathbf{cost}_{\mathbf{r}}(T_{\mathbf{r}} - T^{contract}).$$

Definition 5.3 (The Protocol)
Agents suggest contracts to others and make their decisions according to the above MC^{add} and MC^{remove} sets.

Agents can be both contractors and contractees. Tasks can be recontracted.

- The protocol is domain independent.
- Can only improve at each step: **Hill-climbing in the space of all task allocations**. Maximum is social welfare: $-\sum_{i \in A} \mathbf{cost}_i(T_i)$.
- Anytime algorithm!

5.2 4 Types of Nets

Definition 5.4 (O-, C-, S-, M- Nets)
A contract is called of type

O (Original): *if only one task is moved,*

C (Cluster): *if a set of tasks is moved,*

S (Swap): *if a pair of agents swaps a pair of tasks,*

M (Multi): *if more than two agents are involved in an atomic exchange of tasks.*

Problem: local maxima.

A contract may be individually rational but the task allocation is not globally optimal.

Theorem 5.1 (Each Type Avoids Local Optima of the Others)
For each of the 4 types there exist task allocations where no IR contract with the remaining 3 types is possible, but an IR contract with the fourth type is.

Theorem 5.2 (O-, C-, S-, M- Nets do not reach Global Optima)
There are instances of the task allocation problem where no IR sequence from the initial task allocation to the optimal one exists using O-, C-, S-, and M- contracts.

Definition 5.5 (OCSM Nets)
A OCSM-contract is a pair $\langle \boldsymbol{T}, \boldsymbol{\rho} \rangle$ *of* $|A| \times |A|$ *matrices. An element* $T_{i,j}$ *stands for the set of tasks that agent* i *gives to agent* j*.* $\rho_{i,j}$ *is the amount that* i *pays to* j*.*

Theorem 5.3 (OCSM-Nets Suffice)
Let $|\mathbf{A}|$ and $|T|$ be finite. If a protocol allows OCSM-contracts, any hill-climbing algorithm finds the globally optimal task allocation in a finite number of steps without backtracking.

Theorem 5.4 (OCSM-Nets are Neccessary)
If a protocol does not allow a certain OCSM contract, then there are instances of the task allocation problem where no IR-sequence exists from the initial allocation to the optimal one.

5.3 Coalition Formation

Idea:

Consider a protocol (to build coalitions) as a game and consider Nash-equilibrium.

Problem: Nash-Eq is too weak!

Definition 5.6 (Strong Nash Equilibrium)
A profile is in strong Nash-Eq if there is no subgroup that can deviate by changing strategies jointly in a manner that increases the payoff of all its members, given that nonmembers stick to their original choice.

This is often too strong and does not exist.

Definition 5.7 (Characteristic Function Game (CFG))
In a CFG the value of a coalition S is given by a characteristic function v_S.

Thus it is independent of the nonmembers. But:

1. **Positive Externalities:** Caused by overlapping goals. Nonmembers perform actions and move the world closer to the coalition's goal state.
2. **Negative Externalities:** Caused by shared resources. Nonmembers may use the resources so that not enough is left.

Definition 5.8 (Coalition Formation in CFG's)
Coalition Formation in CFG's consists of the following three steps

Forming $\mathcal{CS}$: *formation of coalitions such that within each coalition agents coordinate their activities. This partitioning is called coalition structure* $\mathcal{CS}$.

Solving Optimazation Problem: *For each coalition the tasks and resources of the agents have to be pooled. Maximize monetary value.*

Payoff Division: *Divide the value of the generated solution among agents.*

An interesting property.

Definition 5.9 (Super-additive Games)
A game is called super-additive, if

$$v_{S \cup T} \geq v_S + v_T,$$

where $S, T \subseteq \mathcal{A}$ *and* $S \cap T = \emptyset$.

Lemma 5.1
Coalition formation for super-additive games is trivial.

Conjecture 5.1
All games are super-additive.

The conjecture is wrong, because the coalition process is not for free: **communication costs, penalties, time limits**.

Maximize the social welfare of the agents $\mathcal{A}$ by finding a coalition structure

$$\mathcal{CS}^* = \arg\max_{\mathcal{CS} \in part(\mathcal{A})} \mathbf{Val}(\mathcal{CS}),$$

where

$$\mathbf{Val}(\mathcal{CS}) := \sum_{S \in \mathcal{CS}} v_S.$$

How many coalition structures are there?

Too many: $\Omega(|\mathcal{A}|^{\frac{|\mathcal{A}|}{2}})$. Enumerating is only feasible if $|\mathcal{A}| < 15$.

How can we approximate $\mathbf{Val}(\mathcal{CS})$?

Choose set $\mathcal{N}$ (a subset of all partitions of $\mathbf{A}$) and pick the best coalition seen so far:

$$\mathcal{CS}^*_{\mathcal{N}} = \arg\max_{\mathcal{CS} \in \mathcal{N}} \mathbf{Val}(\mathcal{CS}).$$

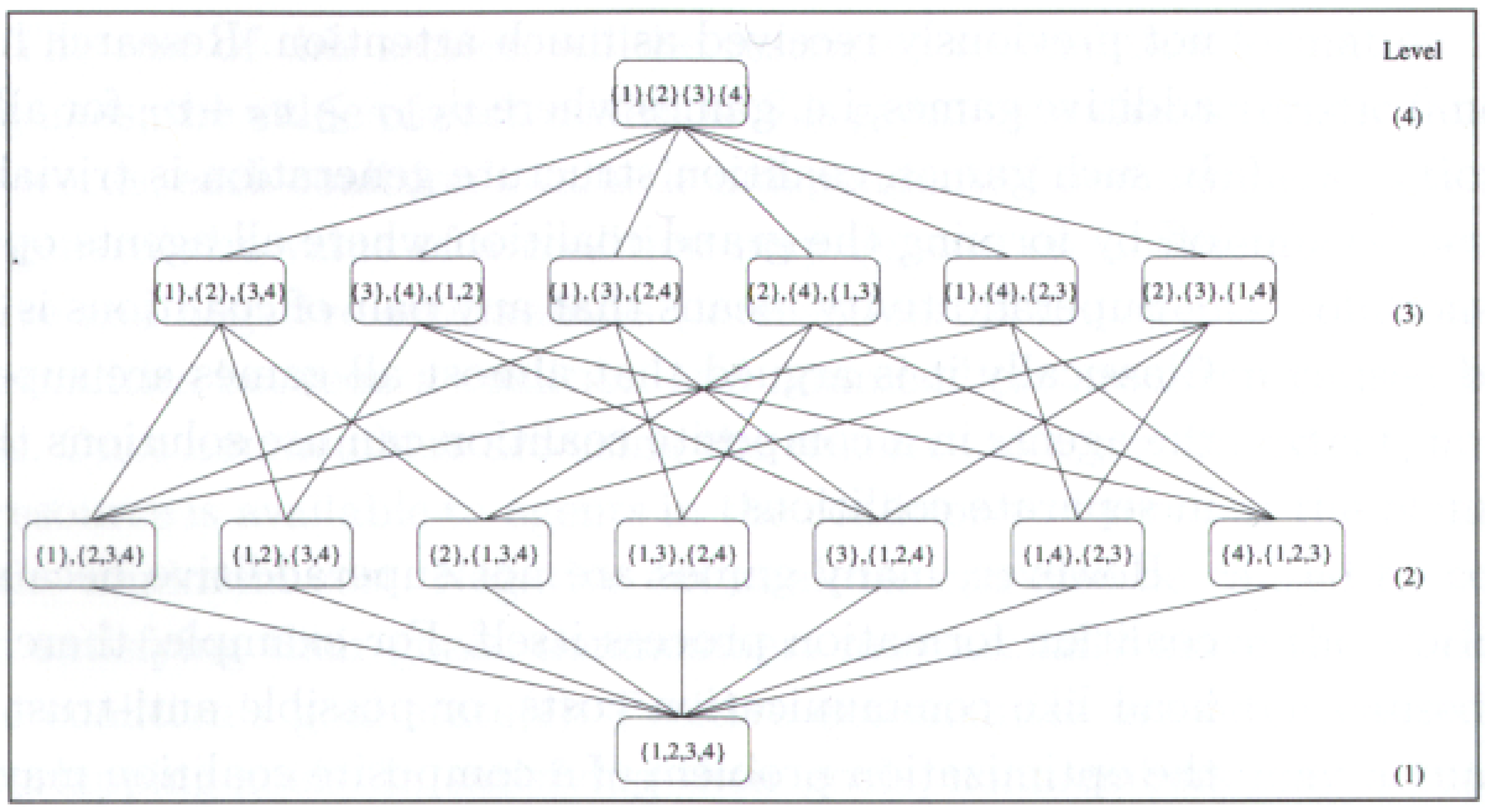

Figure 5.1: Coalition Structure Graph.

We want our approximation as good as possible. That means:

$$\frac{\mathbf{Val}(\mathcal{CS}^*)}{\mathbf{Val}(\mathcal{CS}^*_{\mathcal{N}})} \leq k,$$

where k is as small as possible.

Theorem 5.5 (Minimal Search to get a bound)
To bound k, it suffices to search the lowest two levels of the $\mathcal{CS}$-graph. Using this search, the bound $k = |\mathbf{A}|$ can be taken. This bound is tight and the number of nodes searched is $2^{|\mathbf{A}|-1}$.

No other search algorithm can establish the bound k while searching through less than $2^{|\mathbf{A}|-1}$ nodes.

What exactly means the last theorem? Let n_{min} be the smallest size of $\mathcal{N}$ such that a bound k can be established.

Positive result: $\frac{n_{min}}{\text{partitions of } \mathcal{A}}$ approaches 0 for $|\mathcal{A}| \longrightarrow \infty$.

Negative result: To determine a bound k, one needs to search through exponentially many coalition structures.

Algorithm 5.1 ($\mathcal{CS}$-Search-1)
The algorithm comes in 3 steps:

1. *Search the bottom two levels of the $\mathcal{CS}$-graph.*
2. *Do a breadth-first search from the top of the graph.*
3. *Return the $\mathcal{CS}$ with the highest value.*

This is an **anytime algorithm**.

Theorem 5.6 ($\mathcal{CS}$-Search-1 up to Layer l)
*With the algorithm $\mathcal{CS}$**-Search-1** we get the following bound for k after searching through layer l:*

$$\begin{cases} \lceil \frac{|\mathbb{A}|}{h} \rceil & \textit{if } |\mathbb{A}| \equiv h-1 \mod h \textit{ and } |\mathbb{A}| \equiv l \mod 2, \\ \lfloor \frac{|\mathbb{A}|}{h} \rfloor & \textit{otherwise.} \end{cases}$$

where $h =_{def} \lfloor \frac{|\mathbb{A}|-l}{2} \rfloor + 2$.

Thus, for $l = |\mathbb{A}|$ (check the top node), k switches from $|\mathbb{A}|$ to $\frac{|\mathbb{A}|}{2}$.

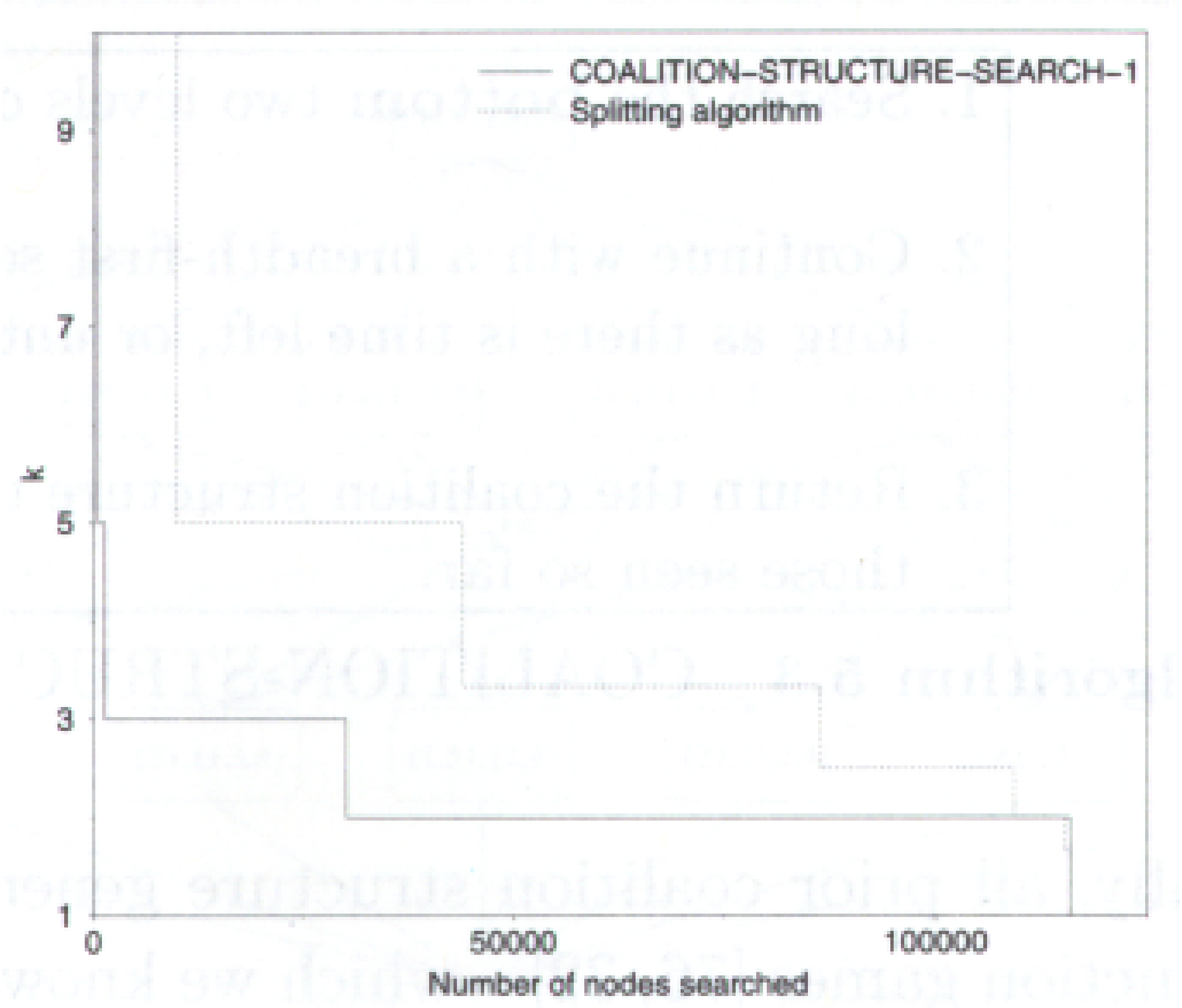

Figure 5.2: Comparing $\mathcal{CS}$-Search-1 with another algorithm.

1. Is $\mathcal{CS}$**-Search-1** the best anytime algorithm?
2. The search for best k for $n' > n$ is perhaps not the same search to get best k for n.
3. $\mathcal{CS}$**-Search-1** does not use any information while searching. Perhaps k can be made smaller by not only considering $\mathbf{Val}(\mathcal{CS})$ but also v_S in the searched $\mathcal{CS}'$.

5.4 Payoff Division

The payoff division should be fair between the agents, otherwise they leave the coalition.

Definition 5.10 (Dummies, Interchangeable)
Agent $\mathtt{i}$ is called a **dummy***, if*

$$\text{for all coalitions } S \text{ with } \mathtt{i} \notin S\text{: } v_{S\cup\{\mathtt{i}\}} - v_S = v_{\{\mathtt{i}\}}.$$

Agents $\mathtt{i}$ and $\mathtt{j}$ are called interchangeable, if

$$\text{for all coalitions } S \text{ with } \mathtt{i} \in S \text{ and } \mathtt{j} \notin S\text{: } v_{S\setminus\{\mathtt{i}\}\cup\{\mathtt{j}\}} = v_S$$

Three axioms:

Symmetry: If i and j are interchangeable, then $x_i = x_j$.

Dummies: For all dummies i: $x_i = v_{\{i\}}$.

Additivity: For any two games v, w:

$$x_i^{v \oplus w} = x_i^v + x_i^w,$$

where $v \oplus w$ denotes the game defined by $(v \oplus w)_S = v_S + w_S$.

Theorem 5.7 (Shapley-Value)
There is only one payoff division satisfying the above 3 axioms. It is called the Shapley value of agent i *and is defined by*

$$x_i = \sum_{S \subseteq A} \frac{(|A| - |S|)!(|S| - 1)!}{|A|!}(v_S - v_{S\setminus\{i\}}).$$

Note:

- $(|A| - S)!$ is the number of all possible joining orders of the agents (to form a coalition).
- The Shapley value sums up the marginal contributions of agent i averaged over all joining orders.
- An **expected gain** can be computed by taking a random joining order and computing the Shapley value.

References

Arisha, K., F. Ozcan, R. Ross, V. S. Subrahmanian, T. Eiter, and S. Kraus (1999, March/April). IMPACT: A Platform for Collaborating Agents. *IEEE Intelligent Systems 14*, 64–72.

Bratman, M., D. Israel, and M. Pollack (1988). Plans and Resource-Bounded Practical Reasoning. *Computational Intelligence 4*(4), 349–355.

Dix, J., S. Kraus, and V. Subrahmanian (2001). Temporal agent reasoning. *Artificial Intelligence to appear*.

Dix, J., M. Nanni, and V. S. Subrahmanian (2000). Probabilistic agent reasoning. *Transactions of Computational Logic 1*(2).

Dix, J., V. S. Subrahmanian, and G. Pick (2000). Meta Agent Programs. *Journal of Logic Programming 46*(1-2), 1–60.

Eiter, T., V. Subrahmanian, and G. Pick (1999). Heterogeneous Active Agents, I: Semantics. *Artificial Intelligence 108*(1-2), 179–255.

Eiter, T., V. Subrahmanian, and T. J. Rogers (2000). Heterogeneous Active Agents, III: Polynomially Implementable Agents. *Artificial Intelligence 117*(1), 107–167.

Eiter, T. and V. S. Subrahmanian (1999). Heterogeneous Active Agents, II: Algorithms and Complexity. *Artificial Intelligence 108*(1-2), 257–307.

Georgeff, M. and A. Lansky (1987). Reactive Reasoning and Planning. In *Proceedings of the Conference of the American Association of Artificial Intelligence*, Seattle, WA, pp. 677–682.

Rao, A. S. (1995). Decision Procedures for Propositional Linear-Time Belief-Desire-Intention Logics. In M. Wooldridge, J. Müller, and M. Tambe (Eds.), *Intelligent Agents II – Proceedings of the 1995 Workshop on Agent Theories, Architectures and Languages (ATAL-95)*, Volume 890 of *LNAI*, pp. 1–39. Berlin, Germany: Springer-Verlag.

Rao, A. S. and M. Georgeff (1991). Modeling Rational Agents within a BDI-Architecture. In J. F. Allen, R. Fikes, and E. Sandewall (Eds.), *Proceedings of the International Conference on Knowledge Representation and Reasoning*, Cambridge, MA, pp. 473–484. Morgan Kaufmann.

Rao, A. S. and M. Georgeff (1995, June). Formal models and decision procedures for multi-agent systems. Technical Report 61, Australian Artificial Intelligence Institute, Melbourne.

Subrahmanian, V., P. Bonatti, J. Dix, T. Eiter, S. Kraus, F. Özcan, and R. Ross (2000). *Heterogenous Active Agents*. MIT-Press.

Weiss, G. (Ed.) (1999). *Multiagent Systems*. MIT-Press.