Coordination without Communication: Experimental Validation of Focal Point Techniques

Maier Fenster **Sarit Kraus***
Department of Mathematics and Computer Science, Bar Ilan University, Ramat Gan, Israel
{fenster,sarit}@bimacs.cs.biu.ac.il

Jeffrey S. Rosenschein
Institute of Computer Science, Hebrew University, Givat Ram, Jerusalem, Israel
jeff@cs.huji.ac.il

Abstract

Coordination is a central theme of Distributed Artificial Intelligence. Much work in this field can be seen as a search for mechanisms that allow agents with differing knowledge and goals to coordinate their actions for mutual benefit. Additionally, one cornerstone assumption of the field is that communication is expensive relative to local computation. Thus, coordination techniques that minimize communication are of particular importance.

This paper considers how automated agents could use a coordination technique common to communication-free human interactions, namely *focal points*. Given a problem and a set of possible solutions from which the agents need to choose one, focal points are prominent solutions of the problem to which agents are drawn. Theoretical work on this subject includes (Schelling 1963; Kraus & Rosenschein 1992).

The purpose of the current research is to consider the practical use of focal point techniques in various domains. We present simulations over randomly generated domains; these simulations strongly suggest that focal points can act as an effective heuristic for coordination in real-world environments.

Keywords: Coordination, Distributed AI

Introduction

Coordination is a central theme of Distributed Artificial Intelligence (DAI). Much of the work in this field can be seen as a search for mechanisms that will allow agents with differing views of the world, and possibly with different goals, to coordinate their actions for mutual benefit.

One of the cornerstone assumptions of DAI is that communication is expensive relative to computation (Bond & Gasser 1988) (i.e., DAI agents are loosely coupled). Thus, work in DAI has actively explored coordination techniques that require little or no communication. Researchers in this area may allow some limited communication in their models, especially insofar as it is required to establish problem constraints. So, for example, in (Genesereth, Ginsberg, & Rosenschein 1986) agents are assumed to jointly perceive an interaction (the joint perception could conceivably involve communication), then proceed without further communication. Similarly, there have been attempts to get multiple agents to interact effectively with little *explicit* communication, while allowing the implicit communication of sensing other agents' external actions or condition (e.g., location) (Levy & Rosenschein 1992).

Another motivation for studying communication-impoverished interactions, other than the expense of communication, has been that communication is sometimes impossible or inconsistent with the environment (communication has been cut off, or is inadvisable in the presence of hostile forces). There has also been a deep-seated intuition that humans are sometimes capable of sophisticated interaction with little explicit communication, and that it ought to be possible for automated agents to emulate this.

This paper explores a coordination technique common to communication-free human interactions, namely *focal points*. Given a problem and a set of possible solutions from which the agents need to choose one, focal points are prominent solutions of the problem to which agents are drawn.[1] Theoretical work on this subject includes (Schelling 1963; Kraus & Rosenschein 1992).

The purpose of the current research is to carry out simulations of focal point techniques. We describe in this paper the application of an algorithm based on focal points to an abstraction of a real world situation. We assume that there are objects in the world with various properties, and we want two agents to choose one of the objects (i.e., the same one) without communicating. If the two agents choose the same object, we have a "meeting" and success. We make no assumptions in our simulations about the properties of the objects, the way they are ordered, or any other special

*Kraus is also affiliated with the Institute for Advanced Computer Studies, University of Maryland, College Park, Maryland.

[1]The Focal Point concept is discussed below in the section "Focal Points".

characteristics of the domain.

We present a domain-independent algorithm and test it in simulations of various instances of the abstract world. The power of focal points in coordinating common choices among agents was highly evident in our simulations. We found that in most randomly generated situations there is more than a 90% probability that the agents will make a common choice, and in many circumstances the probability rises to 100%.

The Coordination Problem

Two agents are trying to choose the same object out of a set of objects. The following examples might occur in an environment where communication is difficult (radio frequency disturbance, or secrecy demands during a battle, or the simple inability to communicate because a specific frequency has been jammed), and therefore an attempt must be made to come to an agreement without any communication. There are various scenarios that might require this kind of communication-poor interaction. For example, two agents that are out of touch with one another must agree on the same plan of action, out of a set of several equally reasonable plans. Another example is of agents that are unable to communicate, but must choose one of several "safe houses" where they will meet and communicate. Another possibility is when agents may need to actually reestablish communication, by choosing a radio frequency to use for future messages.

The worlds we examine have the following characteristics:

- There is a group of objects (denoted by $\mathcal{T}erm$) out of which the agents must choose one (ideally the same one).
- There is a set of predicates $\mathcal{P}red$. Each of the predicates $P \in \mathcal{P}red$ has two arguments: an object in $\mathcal{T}erm$ and a value from the set $\mathcal{V}alue_P$, so that each object has a value for each predicate. For example, $Color(1, red)$ might mean that object 1 is red, while $Height(1, 4)$ might mean that the height of object 1 is 4.
- Any characteristic of an object can be encoded in the values that the predicates can take. We assume, without loss of generality, that the predicates in $\mathcal{P}red$ are ordered and numbered by 1, 2, 3, and so on.[2]

We make the following additional assumptions:

1. The agents observe the same objects and properties in the world. They have the sets $\mathcal{T}erm$, $\mathcal{P}red$ and $\mathcal{V}alue_P$'s as described above.
2. The agents have great flexibility regarding their internal representations of the world, and these internal representations of the different agents need not match one another. For example, they may have different predicates and may represent the value of the predicates differently.
3. Utility is only attached to success at choosing the same object, not to the selection of any specific object (i.e., the agents are indifferent among objects).

In game theory the above problem can be described using a game matrix. One agent needs to choose a row and the other agent needs to choose a column. The payoff for each agent is written in the cell specified by their respective choices.

		K	
		a	*b*
J	*a*	1 / 1	0 / 0
	b	0 / 0	1 / 1

For example, the game matrix shown here can be used to describe situations where two agents need to choose the same object; if both agents choose the same object (e.g., "a") their payoffs are 1 to each of them. Otherwise, they both get 0.

Since the matrix is symmetric, there is no guarantee, using game theory techniques, that the agents will both choose the same object[3] and their chance of choosing the same object is only 50%.

Focal Points

Originally introduced by Schelling (Schelling 1963; Rasmusen 1989), focal points refer to prominent solutions of an interaction, solutions to which agents are drawn. His work on this subject explored a number of simple games where, despite surface equivalence among many solutions, human players were predictably drawn to a particular solution by using contextual information. The following "toy example" illustrates the concept clearly.

Consider a TV game show, where two people have each been asked to divide 100 identical objects into two arbitrarily-sized piles. Their only concern in deciding how much goes into each pile is to match the other person's behavior. If the two agents match one another, they each win $40,000, otherwise they get nothing. Schelling found that most people, presented with this scenario, choose an even division of 50 objects per pile. The players reason that, since at one level of analysis all choices are equivalent, they must focus on any uniqueness that distinguishes a particular option (such as symmetry), and rely on the other person's doing likewise. A similar problem has each person asked to choose any positive number, with their only concern to match the other person's choice. Most people

[2] The predicates' numbers are used only for the presentation of the paper. As is explained below, we allow flexibility regarding the agents' internal representation. In particular, we do not assume that the agents assign the same names to the predicates.

[3] In these situations there are multiple equilibria, and there is a problem in choosing among them. Detailed discussion can be found in (Kraus & Rosenschein 1992).

seem to choose the number 1, it being the only positive number without a predecessor.

There are a number of intuitive properties that seem to qualify a given agreement as a focal point. Among these properties are uniqueness, symmetry, and extremeness. Even when we consider these special properties, more must be done to identify focal points. In the TV game show example above, there is another fairly strong contender for a solution in the choice of 0 objects in one pile, and 100 objects in the second pile (or vice versa). Of course, it is precisely the "vice versa" aspect of this solution that makes it appear less appealing in comparison to the 50 50 split.

In adapting the idea of focal points from human behavior patterns to automated agents, one major difference must be considered. Focal points are based on the naturalness and intuitiveness of certain objects (or solutions) in the world. Automated agents do not have the cultural background (common sense) needed to judge naturalness and intuitiveness. However, their designers can endow them with an algorithm capable of identifying focal points to which they will adhere.

The Focal Point Algorithm

We devise a mathematical formula that specifies the prominence of an object in the world. This formula is based on intuitive properties such as rarity and extremeness. The premise of the work is that in any random world some objects will have a predicate-value vector that is different from those of other objects, and so the object itself will be marked as special. Several agents examining the same world, using the same formula, will see the same "special" objects. Focal point algorithms provide a technique to choose one of these "special objects" uniquely.

The algorithm described below is useful in situations where a single designer builds both agents,[4] and sends them to an environment about which s/he does not have advance information. If the designer suspects that the agents may lose communication and need to get back in touch, s/he might choose to provide them with a mechanism as described below. Since s/he doesn't know the exact details of their environment, the coordination policy can't make use of instructions like "go to the highest building," since there may not be a unique building that satisfies the criterion.

The important point here is that the designer wants to use as little prior information as possible to aid the agents' coordination, but some prior information (e.g., the existence of certain predicates) might still be required by a focal point algorithm. This is not an unreasonable demand. For example, the fact that the agents have certain sensors to which they have access mirrors the prior existence of predicates that can be used in a focal point algorithm.

[4] The algorithm is also useful, if the agents were designed by *different* designers, and they agreed by prior communication to use this algorithm.

Algorithm 1 Joint selection of an object using focal points

1. *Calculate the focal point value for all objects $i \in Term$ using the following equation:*

$$F(i) = \sum_{P \in Pred} R_i^P + 0.5 * E_i^{P,\leq,>} \tag{1}$$

where R_i^P is the rarity of i with respect to predicate P, i.e., how rare is the value of i relative to the other objects, and E_i^P is the extremeness of i with respect to predicate P, i.e., how close (relative to the other objects) is the value of i to one of the extreme values that predicate P takes in this particular world.[5] Formally, assume $P(i,x)$; then,

$$R_i^P = \frac{100}{|\{i' | P(i',x) \text{ is true in this world}\}|} \tag{2}$$

Suppose we have $P(i,x)$, the order on $Value_P$ denoted by $\leq$ and $>$, and let $MAX(i,P)$ be the largest of the following numbers: (1) number of objects that have the value x or less for predicate P; (2) number of objects that have a value greater than x for predicate P. Then we have

$$E_i^{P,\leq,>} = \frac{100 MAX(i,P)}{|Term|} \tag{3}$$

2. *Choose the object c with the largest value that is unique in having that value. Formally, let $UFP = \{i | i \in Term, \forall i' \in Term, \text{ if } i' \neq i \text{ then } F(i) \neq F(i')\}$. If $UFP \neq \emptyset$ then $c = argmax_{i \in UFP} F(i)$.*

There are several aspects of the algorithm that were chosen arbitrarily.[6] To normalize the values calculated, an arbitrary factor of 100 was chosen. The extremeness property was given a lower weight (0.5 in Equation 1), because it seemed to be intuitively weaker than the rarity property. Most importantly, the definitions of the rarity and extremeness properties are arbitrary.

Another problem we faced in creating this algorithm was the relative weight of the different predicates. Since we chose to assume the maximum possible flexibility in the internal representation of the agents, we couldn't assume that agents would identify the predicates in a similar manner. The solution, as mirrored in the formula, was to give equal weight to all the predicates.

[5] E_i^P is only calculated if there is an order on the values of the predicate P.

[6] It is important to emphasize that there is no unique "focal point algorithm"; rather, there are many algorithms that might make use of the basic focal point idea, as presented above (the identification and use of "prominent" objects to heuristically aid coordination).

Example

Suppose there are five objects in the world and three predicates: Type, Size and Color. The values of the predicates with respect to the objects are given in the following table.

object	type	size	color
1	1 (=bridge)	1 (=small)	1 (=red)
2	1 (=bridge)	2 (=big)	2 (=blue)
3	2 (=house)	1 (=small)	1 (=red)
4	2 (=house)	2 (=big)	3 (=brown)
5	2 (=house)	3 (=huge)	3 (=brown)

Some examples of how one would calculate the extremeness and rarity values:

E_1^{type} = Not relevant $\quad R_1^{type} = \frac{100}{2} = 50$

$E_1^{size,\leq,>} = 100 * \frac{3}{5} = 60 \quad R_1^{size} = \frac{100}{2} = 50$

The general formula for calculating the focal point value in this example is: $F(i) = R_i^{type} + .5 * E_i^{type,\leq,>} + R_i^{size} + .5 * E_i^{size,\leq,>} + R_i^{color} + .5 * E_i^{color,\leq,>}$. See Figure 1.

Thus, the agents choose the big blue bridge, i.e., Object 2, which has the largest unique focal point number.[7]

Properties of the Algorithm

The focal point algorithm described above has the following properties:

Success Rate:
The high success rate of the algorithm is demonstrated in the section "Results" below.

Front End:
If the focal point algorithm succeeds (i.e., $UFP \neq \emptyset$), the agents will definitely meet. That is, both agents, when choosing the object according to the algorithm, *know* that the other agent will choose the same object, too. That is the simplest case. In the rare cases that the focal point algorithm fails, both agents know that it failed (also common knowledge), and so this algorithm can be used as a front end for any other coordination algorithm.

Simplicity and Intuitiveness
The algorithm is simple to describe, which is important in case it needs to be transmitted in a noisy environment (e.g., just before communication cut-off). In addition, the algorithm resembles human thought processes, which can help in communicating between man and machine.

Complexity of the Algorithm:
One of the advantages of the focal point algorithm is its low complexity:

Lemma 1 *Given a set of objects $Term$ and a set of predicates $Pred$, the complexity of Algorithm 1 is* $O(|Term| * Max(|Pred|, Log(|Term|)))$.

[7] Note that in this case all the objects belong to the the set UFP that is defined in Algorithm 1, however, Object 2 is the one that has the largest value.

Domain Independence:
The algorithm is applicable in any domain where there are objects, predicates, and the need to choose one of the objects.

Independence of Agents' Internal Representations:
All agents must have sets of objects, predicates, and values for the predicates. However, the agents may have different names for objects, predicates and values. For example, one agent might see a big house and a little house, i.e., $Size(1, big)$ and $Size(2, little)$ while, the other agent sees a medium sized house $Size(1, medium)$ and a small house respectively $Size(2, small)$. Furthermore, agents may have different names for the houses, i.e., the first agent may denote the big house by 1 and the small house by 2, and the second agent may call them 2 and 1 respectively. They may also use different terminology internally; the first agent may use the concept of "house" and the second the concept of "building."

Description of the Simulations

A *configuration* of the world included the number of objects, and the number of predicates, in the world. In each run of the simulation, a new random world was created, by giving new values to the predicates. First, the number of values that a predicate could take was randomly chosen from the range 2–20. Second, the values were generated for each predicate/object pair. The third step was to calculate the focal point value for each object, as described by Algorithm 1. Finally, if there was an object with a unique focal point value, the run was considered a success; otherwise, it was a failure.

To make the simulations computationally tractable, we assumed that the world contained up to 19 objects, that there were up to 9 orthogonal predicates, and that each predicate had up to 20 different possible values.

For each configuration, 500 runs over random worlds were simulated, giving a calculated accuracy of 5% (with 95% probability (Devore 1991) [Section 7]). The final output of 500 runs for each configuration was a number between 0 and 100 that represented the probability of finding an object with a unique focal point value in a random world with the parameters described above. That is, the number represented the probability that the agents will be in a world such that they *definitely* meet when they use the focal point mechanism.[8]

We conducted many sets of simulations. In each set of simulations, we varied some aspect of the world (such as the distribution of the values of predicates and the homogeneity of the world) so as to cover a variety of situations in which agents might find themselves.

In this paper we present in detail only two sets of simulations. These simulations, while quite abstract,

[8] This in no way depends on the number of agents in the world.

Object	R_i^{type}	E_i^{type}	R_i^{size}	$E_i^{size,\leq,>}$	R_i^{color}	$E_i^{color,\leq,>}$	F(i)
1	50	0	50	60	50	0	180
2	50	0	50	80	100	0	240
3	33	0	50	60	50	0	163
4	33	0	50	80	50	0	173
5	33	0	100	100	50	0	233

Figure 1:

cover wide variations of possible (real) worlds. The first set of simulations was chosen to show that the algorithm works in cases that are similar to our expectations regarding the nature of predicates in the real world. The second set of simulations was chosen to show that the algorithm works even if the representations that the agents (respectively) use for the world are so different that the only common information they can use is whether two values are equal or not.

A. Different number of values, binomial and dependency distribution, ordered values:

In this set of simulations, the world had the following details (in addition to the general structure described above):

1. The possible values in the world were distributed using a binomial distribution.
2. Some of the predicates were statistically dependent on other predicates. For example, most trees are green, and only very few are blue.
3. The predicates' values were ordered.

The dependency among predicates was defined as follows: we chose randomly[9] $\frac{1}{3}$ of the predicates to be dependent on the predicates before them, in the following manner. Assume $P_j(x, v_j)$ and $P_{j+1}(x, v_{j+1})$; then $v_j = \lfloor \frac{v_{j-1}}{3} \rfloor + r$ where r was randomly chosen between 0 and 1.

B. Different number of values, even distribution, non-ordered:

In this set of simulations, the world had the following details (in addition to the general structure described above):

1. The possible values in the world were evenly distributed.
2. No predicate was statistically dependent on other predicates.
3. The predicates did not have an internal order: given two different values for a predicate, the agents could not assign an order to them, they could only distinguish between them. For example, consider the predicate *type*. The values "table" and "chair" could not be ordered, but the agents could tell them apart. Therefore, when using the algorithm described above, the agents could not use the extremeness term, only the rarity term.

In all other respects, this set of simulations was similar to the previous one.

[9] In each run, different predicates were chosen.

Results

The results of cases A and B are presented in Figures 2 and 3 respectively. In these figures, the rows correspond to the number of objects in the tested configuration and the columns correspond to the number of predicates in the tested configuration.

In general, the results of the simulations show that the success rate of the algorithm was very high. For example, in case A (Figure 2) if there are at least 2 predicates and at least 3 objects in the configuration, then in more than 97% of the worlds, the algorithm will succeed. The only cases where there is a low success rate is when there is only one predicate or two objects in the configuration.[10]

In case B (Figure 3), where there is no order on the values of predicates in the configuration, if there are at least 4 predicates and at least 7 objects in the configuration, then in more than 94% of the worlds the algorithm will succeed. It is clear that when there are only two objects there is no way to choose one of them based only on rarity; therefore, we have the zeros in row 1 in Figure 3.[11] Also, when there is only one predicate the probability that the algorithm will succeed is very low.[12]

In general, the success rate of the algorithm rises as the number of predicates or objects increases in the world. However, when the number of predicates in the world is very small the success rate of the algorithm doesn't necessarily rise as the number of objects increases. It may be that with a small number of predicates this specific algorithm tends to repeat the same FP values for different objects, and therefore the chance of finding an object with a unique focal point value is relatively low.

[10] As we explained above, in these cases the agents may use a different algorithm to increase their probability of choosing the same object (see the section "Extension of the Algorithm to Probabilistic Choice" below for details).

[11] Recall that in case B the values of the predicates are not ordered, so that the extremeness term in the focal point formula in Algorithm 1 cannot be calculated.

[12] The chance of finding a unique object in a world that has only one predicate is obviously low.

No. of	No. of Predicates								
Objects	1	2	3	4	5	6	7	8	9
2	81	72	82	79	86	85	90	89	92
3	95	99	99	99	99	99	100	99	99
4	90	97	98	98	99	100	99	99	100
5	93	99	100	100	100	100	100	100	100
6	91	99	99	100	100	100	100	100	100
7	89	100	100	100	100	100	100	100	100
8	87	99	100	100	100	100	100	100	100
9	87	99	100	100	100	100	100	100	100
10	89	99	100	100	100	100	100	100	100
11	87	100	100	100	100	100	100	100	100
12	84	100	100	100	100	100	100	100	100
13	85	100	100	100	100	100	100	100	100
14	82	100	100	100	100	100	100	100	100
15	83	99	100	100	100	100	100	100	100
16	80	99	100	100	100	100	100	100	100
17	84	100	100	100	100	100	100	100	100
18	78	99	100	100	100	100	100	100	100
19	78	99	100	100	100	100	100	100	100

Figure 2: Probability of Definitely Choosing the Same Object in Case A

Short Description of Other Experiments

As mentioned above, in addition to experiments A and B, many additional experiments were conducted. Some of these experiments are concisely described here.

Noise: Worlds with partial knowledge were modeled by adding random noise to one of the agent's information (before the addition of noise, the agents had the same knowledge, but used different representations). As can be expected, the algorithm does not perform as well in these cases as in cases A and B; however, it still performs much better that random choice.

Expanded range of values: In other experiments we explored the relationships between the number of values a predicate can have and the success rate of the algorithm. We discovered that in general the success rate of the focal point algorithm increases with the number of values that a predicate can receive. In addition, we discovered that even a small number of possible values was generally enough for a high success rate.

Exact knowledge: Another experiment was performed to test the added utility of using exact knowledge: both agents see the exact values of the predicates in the world. The probability of success in these cases was very similar to the success rate shown in this paper, although the chances of success are significantly higher when the number of objects, predicates, or possible values in the world, is very small.

Extension of the Algorithm to Probabilistic Choice: In previous sections, the focal point algorithm provided the agents with a mechanism for guaranteeing their common choice of the same object. However, even when the focal point algorithm fails, the information that it provides can be used to increase the probability of choosing the same object. The agents can look for the smallest set of objects such that their focal point value is the same (if there's more than one such set, they can choose the one with the largest focal point value) and then choose one of them randomly. Simulations we performed showed an increase in success rate over cases A and B.

Conclusions

We have presented the concept of focal point solutions to cooperation problems. An algorithm was presented for discovering unique focal points, and a series of simulations were run over various randomly generated worlds that demonstrated the usefulness of the focal point algorithm as a heuristic for multi-agent coordination.

The algorithm is shown to be successful in a wide variety of cases, including cases where the only information that the agents can use for their coordination is the non-ordered values of a few properties (predicates).

Acknowledgments

This research has been partially supported by the Israeli Ministry of Science and Technology (Grant 032-8284 and Grant 4210), by the Israel Science Foundation (Grant 032-7517), and by the National Science Foundation under Grant Number IRI-9311988.

References

Bond, A. H., and Gasser, L. 1988. An analysis of problems and research in DAI. In Bond, A. H., and

No. of Objects	No. of Predicates								
	1	2	3	4	5	6	7	8	9
2	0	0	0	0	0	0	0	0	0
3	31	54	68	77	80	87	88	91	91
4	8	33	42	55	62	71	75	82	82
5	17	55	76	86	90	93	94	97	97
6	9	49	68	81	88	93	95	96	98
7	11	65	88	94	97	98	98	99	100
8	9	61	82	96	97	99	99	99	100
9	10	80	92	99	99	99	99	100	100
10	10	75	92	98	99	99	100	100	100
11	11	81	97	99	99	100	100	100	100
12	12	80	97	99	100	100	100	100	100
13	11	86	99	99	100	100	100	100	100
14	11	83	97	99	100	100	100	100	100
15	11	89	99	99	100	100	100	100	100
16	12	87	99	100	100	100	100	100	100
17	12	92	100	100	100	100	100	100	100
18	10	90	99	100	100	100	100	100	100
19	10	92	100	100	100	100	100	100	100

Figure 3: Probability of Definitely Choosing the Same Object in Case of Different Number of Values, Even Distribution (B), Non-Ordered Values

Gasser, L., eds., *Readings in Distributed Artificial Intelligence*. San Mateo, California: Morgan Kaufmann. chapter 1, 3–56.

Devore, J. L. 1991. *Probability and Statistics for Engineering and Sciences*. Pacific Grove, California: Brooks/Cole Publishing Company.

Genesereth, M. R.; Ginsberg, M. L.; and Rosenschein, J. S. 1986. Cooperation without communication. In *Proceedings of The National Conference on Artificial Intelligence*, 51–57.

Kraus, S., and Rosenschein, J. S. 1992. The role of representation in interaction: Discovering focal points among alternative solutions. In *Decentralized AI, Volume 3*. Amsterdam: Elsevier Science Publishers B.V./North-Holland.

Levy, R., and Rosenschein, J. S. 1992. A game theoretic approach to distributed artificial intelligence and the pursuit problem. In *Decentralized Artificial Intelligence III*. Amsterdam: Elsevier Science Publishers B.V./North-Holland. 129 146.

Rasmusen, E. 1989. *Games and Information*. Cambridge, Ma: Basil Blackwell Ltd.

Schelling, T. C. 1963. *The Strategy of Conflict*. New York: Oxford University Press.