Dynamic Fact Communication Mechanism: A Connectionist Interface

Proceedings of the Annual Meeting of the Cognitive Science Society, 13(0)

Aaronson, Jeffrey S.

1991

Dynamic Fact Communication Mechanism: A Connectionist Interface

Jeffrey S. Aaronson
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104
aaronson@linc.cis.upenn.edu

Abstract

Shastri and Ajjanagadde have proposed a biologically plausible connectionist rule-based reasoning system (hereafter referred to as a knowledge base, or KB), that represents a dynamic binding as the simultaneous, or in-phase, activity of the appropriate units [Shastri & Ajjanagadde 1990]. The work presented in this paper continues this effort at providing a computational account of rapid, common-sense reasoning. The Dynamic Fact Communication Mechanism (DFCM) is a biologically plausible connectionist interface mechanism that extracts a temporally-encoded fact (i.e. a collection of dynamically-encoded bindings) from a source KB and incorporates the fact into a destination KB in a manner consistent with the knowledge already represented in the latter. By continually interpreting source KB activity in terms of target KB activity, DFCM is able to transfer facts between distinct KBs on the same time scale needed to perform a single rule application within a single KB. Thus, DFCM allows the benefits of decomposing a phase-based reasoning system into multiple KBs, each with its own distinct phase structure, while rendering the inter-module communications costs negligible. A simple modification to DFCM allows the unit of transfer to be *groups* of facts. Finally, the number of units that compose DFCM is linear in the size of the KB.

Introduction

Shastri and Ajjanagadde have suggested a computational account of how a human agent, seemingly without conscious intervention, can perform a broad class of inference with remarkable efficiency (within a few hundred milliseconds) [Shastri & Ajjanagadde 1990]. This class of reasoning is described as *reflexive* reasoning, since the inferences seem to be drawn automatically and effortlessly [Shastri 1990]. Reflexive reasoning proficiency is independent of the size of the body of knowledge upon which the reflexive inferences are drawn.[1] As argued in [Shastri & Ajjanagadde 1990], this strongly suggests that reflexive reasoning is a manifestation of a massively parallel reasoning system in which response time is governed by the length of the chain of inference, and that multiple potentially relevant inference paths are explored simultaneously. In light of results from complexity theory, this implies that human agents are not general purpose reasoners, as all valid inferences cannot be derived proportional to the length of the proof. The connectionist knowledge base described by Shastri and Ajjanagadde not only offers a biologically plausible, computational account of reflexive reasoning, but suggests a limited class of inference by which humans do reason.

A brief description of their system follows. Each constant and predicate argument in the KB is represented by a unit. Long-term, or static, facts are physically encoded as an ensemble of units with structured interconnections that bind together the appropriate filler/role pairs. Rules are represented as a mapping between arguments of the antecedent predicate(s) and arguments of the consequent predicate. As such, there is a physical link reflecting argument mappings for each rule. Inferences are drawn as a result of activation spreading through the system in parallel, by way of the links between predicate arguments. During each episode of reasoning, many transient facts are created, propagated, and eventually discarded. Any attempt at modeling reflexive reasoning must have a method of creating and propagating these dynamic bindings (see [Shastri & Ajjanagadde 1990] for an in-depth discussion) that allows the inference process to perform in real-time, without succumbing to a complex form of cross-talk called the *variable binding* problem. A hard-wired solution to this problem is adopted in order to represent *static, long*-term facts. However, this method is not a feasible means of representing *dynamic* bindings in *short*-term memory. Consider that an episode of reasoning takes on the order of a few hundred milliseconds. This suggests that the creation of the new bindings introduced by each rule application can take only on the order of tens of milliseconds. It has yet to be shown that synaptic modification occurs anywhere near this quickly. Hence, reflexive reasoning does not permit us the luxury of allowing synaptic modification to recruit new units and build up the necessary connectivity "on the fly".

The solution chosen by Shastri and Ajjanagadde represents a dynamic binding as the simultaneous, or in-phase, activity of the appropriate units [Shastri & Ajjanagadde 1990]. Note that a system in which nodes characteristically fire in short slices at a given frequency *does not require a controlling global clock*. Rather, the only requirement is that we have units that fire with a fixed period of oscillation. Thus the inference mechanism consists of a rhythmic spread of activation through the network, each set of bindings "lighting up" during its distinct phase within a fixed-length period, dynamically creating a set of bindings with more member units as the inference process continues. An analogous spatial solution would be realized by creating ρ copies of each unit in the KB, one for each of the ρ phases. If a unit were active in the former system in phase i, then the i^{th} copy of the unit would be active in the latter system. While this method would increase the size of the KB by a factor of ρ, the speed of the system would presumably increase by a factor of ρ as well. Were we not concerned with a biologically plausible system, this might

---

[1] If this were not the case, then it would take you longer to understand this sentence today than it did 5 years ago!

well be an attractive tradeoff. However, the firing characteristics of neurons are such that they fire in short bursts followed by longer periods of latency. Thus, by adopting the temporal synchrony approach we reduce the size of the knowledge base without sacrificing speed. Furthermore, there is mounting biological evidence of synchronous neural activity in the brain, and that we encode the unity of objects via the proposed temporal method [Freeman 1981, Eckhorn *et al.* 1988, Gray & Singer 1989, Eckhorn *et al.* 1989]. Note that this paper proposes an idealized biological system. We do not claim that facts, rules, and dynamic bindings are encoded by the brain via the methods described herein. However, we do think it is significant that there are mechanisms in the brain that support synchronous activity, and that this temporal synchrony may be an encoding utilized by the animal brain to process information.

The bottleneck in such a phased system is ρ, the ratio between period duration and phase duration. This ratio determines the number of distinct individuals (or entities) that can be attuned to during an episode of reasoning. Furthermore, this ratio is KB dependent; different KB, with different characteristic ratios, will be able to support a different number of distinct individuals at one time during an episode of reasoning. Hereafter, a binding will refer to a single entity filling a single role. A Binding Set (BSet) will refer to a set of bindings involving the same entity. Thus the characteristic ratio determines the number of temporal BSets that may be supported during an episode of reasoning. It is important to note that the number of roles to which an entity can be dynamically bound is unlimited. Hence, each BSet may include an arbitrary number of bindings.

Dynamic Fact Communication Mechanism

The animal brain is a highly structured, modular architecture in which specific areas are delegated toward processing one type, or one form, of information. Despite this clear separation, intelligent agents have the ability to draw reflexive inferences that can only be reached as a result of an integration of data from distinct sensory processing "stations". An interface mechanism that is able to communicate between 2 distinct repositories of information, for e.g., the "visual" KB and the "auditory" KB must be a component in not only the reasoning process performed by an intelligent agent, but in the *reflexive* reasoning process performed by an intelligent agent. A typist glancing down at his keyboard is reflexively aware that the repetitive clacks he hears are the result of the depression of keys on his keyboard. This conclusion could not be reached without a mechanism that allows the visual and auditory stations to confer with the requisite efficiency that enables such an inference to be drawn reflexively.

The Dynamic Binding Communication Mechanism (DBCM) [Aaronson 1991] makes the first attempt at designing such an interface mechanism. DBCM is a biologically plausible, connectionist communication system that operates between 2 distinct phase-based KBs. The unit of transfer is a BSet (i.e. a single entity and all of the roles to which it is bound). The interface mechanism presented in this paper continues the effort begun by DBCM. The Dynamic Fact Communication Mechanism (DFCM) is a biologically plausible connectionist interface mechanism that extracts a temporally-encoded fact (i.e. a collection of dynamically-encoded bindings) from a source KB and incorporates the fact into a destination KB in a manner consistent with the knowledge already represented in the latter. DFCM continually monitors the entity-to-phase assignments within both KBs in order to interpret source KB activity in terms of target KB activity. Effectively, DFCM processes each potential request *prior to the arrival of the request itself*. The dynamic bindings that compose each interpreted fact flow along the DFCM links, poised at the threshold of the target KB. The arrival of a predicate transfer request acts much like the opening of a door to that target KB predicate, enabling the transfer to take place instantly. DFCM is thus able to transfer facts between distinct KBs *on the same time scale needed to perform a single rule application within a single KB*. This allows us to enjoy the benefit of decomposing a phase-based reasoning system into multiple KBs, each with its own distinct phase structure, while rendering the inter-module communications costs negligible. Moreover, the number of units that compose DFCM is linear in the size of the KB.

Later, we discuss a simple modification to DFCM that allows the transfer unit to be *all* active facts associated with a predetermined group of predicates. Despite this increased bandwidth, the time needed to effect a transfer remains unchanged. Such a system could naturally operate between KBs that allow multiple instantiations of predicates [Shastri & Ajjanagadde 1990]. A more drastic modification allowing dynamic grouping of predicates is also discussed.

Setting the Scene

All links in DFCM are unidirectional. Arrows and filled circles designate excitatory and inhibitory links, respectively. Links between sets of units indicate pairwise connections. DFCM makes use of 2 simple processing elements:

- **ρ-btu** ρ-btu nodes take the weighted sum of their inputs. If this total meets or exceeds the threshold of the node, the node begins to fire synchronously with the driving input. Node continues to fire in this fashion as long as driving input persists. ρ-btu nodes are represented diagrammatically by a circle.
- **τ-or** τ-or nodes take the weighted sum of their inputs. If this total meets or exceeds the threshold of the node, the node fires with a duration equal to the period of oscillation. τ-or nodes are represented diagrammatically by a triangle.

As stated in the introduction, the representation and propagation of dynamic bindings can be realized without the use of a global clock. Units firing synchronously belong to the *same* BSet; units firing asynchronously belong to *different* BSets. A global clock is not needed to support this encoding; rather, all we require are units which fire with a fixed period of oscillation, according to the ρ-btu activation function described above. Hereafter, we will make reference to a synchronous set of units occupying a distinct phase within a period of oscillation. This is done merely for explanatory convenience, and in no way implies the presence of a global clock regulating unit activity.

Let ρ_s and ρ_t be the number of active phases in the source and target KBs, respectively. For simplicity, we assume that both KBs have the same period of oscillation and thus can encode the same number of phases. This characteristic ratio will be denoted by ρ. If an entity is active in more than one KB, we say that the phases which encode the entity in each KB are *entity equivalent*. An active source KB phase for which there is no entity equivalent target KB phase will be called a *foreign phase*. Bindings represented in a foreign phase are called *foreign bindings*.

DFCM delivers target KB input into a special-purpose buffer predicate associated with each target KB predicate. A single rule application will instantiate a target KB predicate with the contents of its buffer. A rule application is performed every period of oscillation [Shastri & Ajjanagadde 1990]. The i^{th} role of the buffer predicate e.g., GIVE, will be denoted $\mathcal{B}_i^{GIVE}$.

The responsibility of generating transfer requests is application dependent; both source and target KBs may play this role. Signal duration should equal period of oscillation. Multiple requests should not be signaled simultaneously.

Architecture

There are 2 categories of bindings that must be processed. Bindings in entity equivalent phases require supplementing the number of bindings within a target KB BSet. Foreign bindings require allocating a new phase within the target KB, thereby creating a new BSet. Interpreting these latter bindings requires care. Note only must an *unused* phase be allocated for each binding, but we must satisfy a higher-level encoding as well. A given dynamic fact may require several foreign phases to represent its collection of bindings. Properly interpreting such a fact includes allocating a distinct, new phase *for each such foreign phase*. Failure to do so would result in cross-talk being introduced into the target KB. The interpretation mechanism of DFCM reflects these issues. DFCM is functionally divided into $p+1$ planes, where p is the arity of the predicate of maximum arity in the source KB. Let $\mathcal{P}_i$ denote the i^{th} plane. $\mathcal{P}_0$ has 2 responsibilities: i) interpreting all bindings in entity equivalent phases, ii) distributing the foreign phases amongst the remaining planes such that the distinct foreign phases which compose each dynamic fact are assigned to distinct planes. $\mathcal{P}_1, \ldots, \mathcal{P}_p$ are responsible for interpreting foreign bindings. Each of these planes is assigned a currently available phase in which to re-encode its assigned foreign bindings. In order to ensure that cross-talk is not introduced into the target KB, a phase may not be assigned to more than one plane. Figure 1 illustrates how the responsibility of interpreting the filler/role pairs is distributed amongst the different planes for the dynamic fact *give(John,Mary,Book1)*.

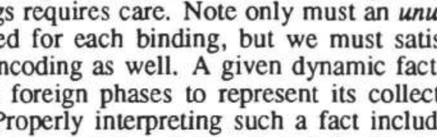

Figure 1: Here, $p = 3$. *give(John,Mary,Book1)* is a dynamic source KB fact. The entity *Mary* is active in the target KB; the entities *John* and *Book1* are inactive in the target KB. $\mathcal{P}_0$ interprets the *recipient* role in the entity-equivalent phase given by the activity of its filler *Mary* within the target KB. $\mathcal{P}_1$ interprets the filler/role pair *Book1/give-object*, $\mathcal{P}_2$ interprets the pair *John/giver*. Each of the latter planes are assigned a unique, available target KB phase in which to encode their assigned bindings.

DFCM relies on several subnetworks that monitor the content of both the source and target KBs in order to provide each plane with the phase information it needs to act as desired. The Free Phase Map (FPM) communicates unique free phases to $\mathcal{P}_1, \ldots, \mathcal{P}_p$. The Entity Map (EM) provides $\mathcal{P}_0$ with a mapping between entity equivalent phases across the source and target KBs.

Free Phase Map

The function of the Free Phase Map is to monitor the target KB and assign unique, unused phases to distinct planes. There are 2 sets of units that compose the FPM. See figure 2.

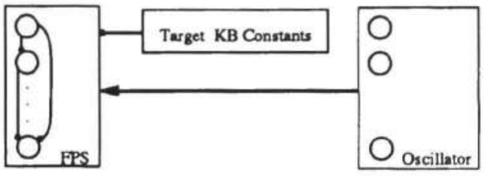

Figure 2: The Free Phase Map

First, there is a set of ρ ρ-btu units that oscillate within each period, according to the phase structure of the target KB. The i^{th} unit of this Oscillator, denoted Osc_i, fires during the i^{th} phase of each period. The behavior of these units never varies. They are present in order to communicate the phase structure of the target KB to other parts of DFCM. One of these parts is the second set of units composing the FPM. This group consists of p "free phase selector" (FPS) units, one for each plane $\mathcal{P}_1, \ldots, \mathcal{P}_p$. FPS_i firing in synchrony with its (lone) driving input Osc_j informs $\mathcal{P}_i$ to interpret bindings using phase j. Each of the ρ oscillating units connect to each FPS unit. In the absence of any inhibition, activation along each link is sufficient to incite the destination unit to fire. There are 2 constraints that must be placed on the firing of the FPS units to prevent cross-talk from being introduced into the target KB. First, we cannot allow an FPS unit to signal its associated plane with a phase that is already in use within the target KB. Second, we cannot allow multiple FPS units to fire in concert, thereby signaling multiple planes with the same phase. The former constraint is achieved by establishing a strong inhibitory connection from each target KB constant to each FPS unit. The latter constraint is enforced via a downstream inhibition technique among the FPS units, whereby there is a strong inhibitory connection from FPS_i to FPS_j, for all $j > i$.

Entity Map

DFCM monitors the entity-to-phase mapping within both source and target KBs, maintaining a record of entity equivalent phases. $\mathcal{P}_0$ relies on this mapping to provide the correct phase in which interpret its assigned roles. The Entity Map has ρ units, $EM_0, \ldots, EM_{\rho-1}$. EM_i firing in phase j indicates that phase i in the source KB and phase j in the target KB are entity equivalent. That is, the entity belonging to the BSet represented in phase i in the source KB is the same entity belonging to the BSet represented in phase j in the target KB. Since each BSet includes a constant, it is sufficient for the Entity Map to monitor the constants in the source and target KBs. See figure 3 to see how the EM monitors the source and target KB representations of a sample constant, labeled S and T, respectively. There is a link from T to each EM unit. There are ρ τ-or units for each constant. The i^{th} τ-or unit fires in response

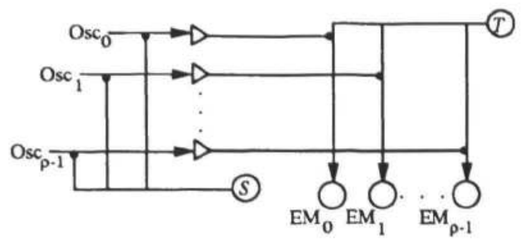

Figure 3: Connections for only a single constant are shown.

to Osc_i, provided that the Osc_i signal is not blocked by an inhibitory connection from S. Similarly, EM_i fires in response to T, provided that activation along the incoming link is not blocked via activation traveling along an inhibitory connection from the i^{th} τ-or unit. The net result of this architecture is that EM_i fires in phase j if and only if some constant is active in phase i of the source KB and phase j of the target KB.

Initial Plane

$\mathcal{P}_0$ is responsible for interpreting roles that are active in entity equivalent phases. The activity of the EM units precisely identifies these roles. If EM_i is active then $\mathcal{P}_0$ must interpret all source KB roles active in phase i. Moreover, the entity equivalent phase in which we want to introduce these roles is given by the phase in which EM_i fires. Figure 4 illustrates how $\mathcal{P}_0$ processes the k^{th} role of the source KB predicate GIVE. There are ρ subnetworks for each role, each acting as a temporal pattern matcher. Thresholds and connection weights are such that each unit must receive input from 2 sources in order to become active. Thus, if a role is active in phase i, then only the τ-or unit in the i^{th} subnetwork will fire. If EM_i is firing in phase j, then the associated ρ-btu unit will fire in phase j, thereby communicating its activation to $\mathcal{B}_k^{GIVE}$. The onset of the GIVE transfer signal enables $\mathcal{B}_k^{GIVE}$ to respond in this entity equivalent phase. Later, we see how $\mathcal{B}_k^{GIVE}$ may be incited by activity from other planes.

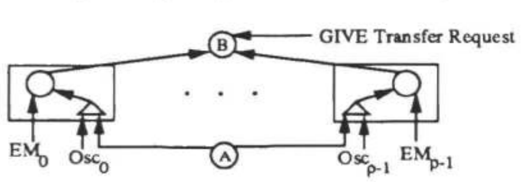

Figure 4: Unit A is $GIVE_k$ in the source KB. Unit B is $\mathcal{B}_k^{GIVE}$.

The second task facing $\mathcal{P}_0$ is to distribute the foreign phases to the remaining planes. Since the DFCM unit of transfer is a fact, it is sufficient if this distribution is internally consistent at the fact-level. Accordingly, there is a dedicated director network, composed of n ρ-btu units, for each n-ary predicate.

Figure 5 details the director $\mathcal{D}^{GIVE}$. The idea is analogous to that used in the Free Phase Map. Whereas there we wanted the FPS units to latch onto unique available phases, here we want the director units to latch onto unique foreign phases. Each predicate's role units are connected pairwise to the corresponding director units. Each director is endowed with downstream inhibition so that no 2 planes will be assigned the same foreign phase. We use the Entity Map to inhibit director units during the the entity-equivalent phases processed by $\mathcal{P}_0$. If $\mathcal{D}_i^{GIVE}$ fires in phase j, then $\mathcal{P}_i$ is responsible for interpreting *all* the phase j foreign bindings that contribute to the representation of the dynamic fact GIVE in the source KB. These will be encoded in the phase specified by FPS_i.

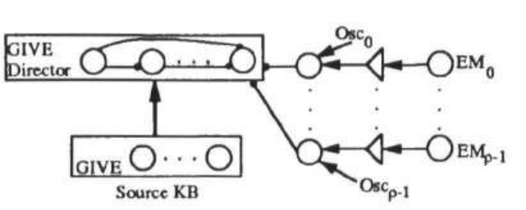

Figure 5: The director network for predicate GIVE

Remaining Planes

$\mathcal{P}_0$ processes bindings in entity equivalent phases, thereby only needing to incorporate *roles* into the target KB. However, the remaining planes process foreign bindings, thereby needing to incorporate *fillers* and *roles* into the target KB.

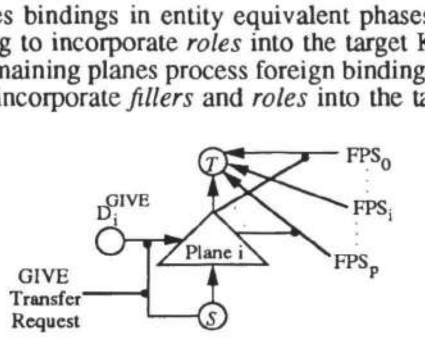

Figure 6: Predicates other than GIVE are suppressed.

Figure 6 shows how $\mathcal{P}_i, i > 0$ processes a sample constant. There is a τ-or unit in each plane for each constant in the source KB. Each such $\mathcal{P}_i$ unit receives activation from its corresponding source KB constant as well as from the $\mathcal{D}_i$ unit corresponding to each source KB predicate. Activation along each of these latter links is effectively blocked via inhibition from the source KB constant *should the phases coincide*. This inhibition may itself be blocked for a sample predicate should that predicate transfer signal be active. Weights and thresholds are such that the τ-or unit in $\mathcal{P}_i$ corresponding to a constant S fires if and only if i) S is active in the source KB in phase j, ii) some $\mathcal{D}_i$ is firing in phase j, and iii) the predicate transfer signal corresponding to that $\mathcal{D}_i$ is active. A simple inhibitory technique ensures that the target KB constant is incited in the free phase specified by the appropriate FPS unit. Each FPS unit is connected to each target KB constant. Each $\mathcal{P}_i$ τ-or unit is connected upwards to the corresponding target KB constant, and inhibits all but the FPS_i unit from sending activation to that constant. Weights and thresholds are such that a target KB constant cannot fire in the absence of input from a τ-or plane unit.

Figure 7 illustrates how $\mathcal{P}_i$ processes the k^{th} role of the source KB predicate GIVE. There is a τ-or unit in each plane for each role in the source KB. This unit receives activation from the corresponding source KB role as well as from the director unit indexed by that predicate and plane. Weights and thresholds are such that the τ-or unit fires in

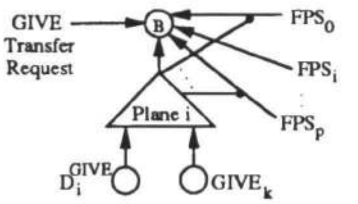

Figure 7: Unit B is $\mathcal{B}_k^{GIVE}$. $\mathcal{P}_0$ connections are suppressed.

response to the simultaneous activation of both sources. Activity of this unit indicates $\mathcal{P}_i$ is responsible for interpreting the source KB role $GIVE_k$. Each τ-or units connects to the appropriate $\mathcal{B}$ role. We use the analogous inhibitory technique as above to ensure that each $\mathcal{B}$ role is incited in the free phase specified by the appropriate FPS unit.

Performance Evaluation

A buffer predicate is fully instantiated one period following the arrival of the predicate request signal. New target KB entities are activated (if required) 2 periods following the arrival of a predicate request signal.

The director corresponding to a source KB fact whose dynamic bindings require representation in $f > 0$ foreign phases needs $2f - 1$ time periods *following onset of the fact* to reach desired state. New phase allocation within the target KB requires the resetting of each director that has a unit firing in the entity equivalent phase within the source KB. In the worst case, resetting requires $2f - 1$ time periods. Since dynamic facts are typically active for tens of time periods during an episode of reasoning [Shastri & Ajjanagadde 1990], and f will characteristically be quite small, this delay should not detract from the effectiveness of the system. The FPS units require $2 * (\rho - \rho_t) - 1$ time periods to be *fully* functional following new phase allocation within the target KB (although multiple phases may be allocated seamlessly within one request). Since this allocation rate will typically be quite slow, this should not detract from the effectiveness of the system. We hope to develop quicker methods of resetting the FPS and director networks, although these limitations are likely to become inconsequential in future versions of DFCM, in which the unit of transfer is increased from a *single* fact to a *group* of facts.

Extension

The foreign phase to plane assignments made by each predicate director ensures that DFCM interprets each fact in an internally consistent fashion. However, arbitrary sets of facts, in general, will not be interpreted consistently, as there is no cooperation between directors. This limits the bandwidth of transfer to a single fact. The unit of transfer, however, can be increased to be *all* active facts associated with a predetermined group of predicates by allowing each group to *share the same director*. Moreover, increasing the transfer bandwidth does not increase the transfer time, and the number of units needed is still linear in the size of the KB. Such a system could naturally operate between KBs that allow multiple instantiations of predicates [Shastri & Ajjanagadde 1990]. The increased bandwidth limits the urgency of being able to process successive requests quickly. A more drastic modification of DFCM would permit dynamic grouping of predicates, allowing for arbitrarily defined groups at run-time. Such a change would alter the philosophy of DFCM, rendering it more of a reactive mechanism, as the entity-to-phase assignments would have to wait until the predicate composition of the requested group is known. Despite this, DFCM would still be able to satisfy a transfer request in small constant time (proportional, but bounded by ρ, to the number of distinct foreign phases per group).

Conclusion

The Dynamic Fact Communication Mechanism (DFCM) is a biologically plausible connectionist interface mechanism that operates between 2 KBs that dynamically encode variable bindings by means of temporal synchrony. DFCM continually monitors the entity-to-phase assignments within both KBs in order to interpret source KB activity in terms of target KB activity. Effectively, DFCM processes each potential request *prior to the arrival of the request itself.* This active approach enables a fact to be transferred between distinct KBs *on the same time scale needed to perform a single rule application within a single KB*. The interface thus allows the benefits of decomposing a phase-based reasoning system into multiple KBs, each with its own distinct phase structure, while rendering the inter-module communications costs negligible. Moreover, the number of units composing DFCM is only linear in the size of the KB. We believe this system to be an important component in a general model of reflexive, i.e., rapid common sense reasoning.

Acknowledgements

I'd like to thank Lokendra Shastri for the many discussions that provided the motivation for this work, and for his guidance along the way.

References

Aaronson, J.S. 1991. Dynamic Binding Communication Mechanism, Technical Report MS-CIS-91-16, CIS Dept., University of Pennsylvania.

Eckhorn, R., Bauer, R., Jordan, W., Brosch, M, Kruse, W. Munk, M., and Reitboeck, H.J. 1988. Coherent Oscillations: A mechanism of feature linking in the visual cortex. *Biological Cybernetics* 60:121-130.

Eckhorn, R., Reitboeck, H.J., Arndt, M., and Dicke, P. 1989. A Neural Network for feature linking via synchronous activity: Results from cat visual cortex and from simulations. In Model of Brain Function, Cotterill RMJ (ed), Cambridge Univ. Press.

Freeman, W.J. 1981. A physiological basis of perception. In Perspectives in Biology and Medicine. Univ. of Chicago Press.

Gray, C.M. and Singer, W. 1989. Stimulus-specific neuronal oscillations in orientation specific columns of cat visual cortex. *PNAS*, 86:1698-1702.

Shastri, L. 1990. Connectionism and the Computational Effectiveness of Reasoning. *Theoretical Linguistics* 16(1):65-87.

Shastri L, and Ajjanagadde, V. 1990. From Simple Associations to Systematic Reasoning: A Connectionist Representation of Rules, Variables and Dynamic Bindings, Technical Report MS-CIS-90-05, CIS Dept., Univ. of Pennsylvania.