HCOP : Modeling Distributed Constraint Optimization Problems with Holonic Agents

Fernando J. M. Marcellino, Jaime S. Sichman
Laboratório de Técnicas Inteligentes (LTI)
Escola Politécnica (EP) - Universidade de São Paulo (USP)
Av. Luciano Gualberto, 158 – trav. 3
05508-970 – São Paulo – SP – Brasil
fmarcellino@usp.br, jaime.sichman@poli.usp.br

Abstract

This paper defines a new distributed optimization problem, called Holonic Constraint Optimization Problem (HCOP). It is based on the concepts of Distributed Constraint Optimization Problem (DCOP) and Holonic Agents. We present the background theory and the formalization of a HCOP, which rather than a mere generalization of a DCOP, represents a distinct paradigm. We also propose a meta-algorithm, called HCOMA, to solve this kind of problem, where several available DCOP algorithms, or even centralized algorithms, can be embedded and integrated in such a way to obtain the most fitting configuration for each case. In addition, a motivating application in the oil supply chain domain is presented in order to illustrate the new approach.

1 Introduction

Constraint satisfaction and optimization are powerful paradigms that model a large range of tasks like scheduling, planning, optimal process control, etc. Traditionally, such problems were gathered into a single place, and a centralized algorithm was applied in order to find a solution. However, problems are sometimes naturally distributed, so Distributed Constraint Satisfaction (DisCSP) was formalized by Yokoo et al. in [Yokoo *et al.*, 1992]. Here, the problem is divided among a set of agents, which have to communicate with each other to solve it. More recently this paradigm was extended to constraint optimization by replacing the logical constraints with valued ones, and it was formalized as a Distributed Constraint Optimization Problem (DCOP) [Modi *et al.*, 2006]. In general, an optimization problem is much harder to solve than a DisCSP, as the goal is not just to find any solution, but the best one.

If we analyze some real constraint optimization problems, we can notice that they own a *recursive* nature, which is not currently exploited by the available optimization frameworks and their associated algorithms. An example of this kind of problem is the supply chain management. Usually each entity in the chain is likely to act in its best interests to optimize its own profit. However, in general, that doesn't meet the goal of the optimization of the entire supply chain. On the other hand, the complexity of the whole chain integration makes the development of a single centralized system an unfeasible task. In addition, even if it were possible, the frequent and unforeseeable changes in the business environment would make the results of such a system obsolete and useless very fast.

This paper defines a Holonic Constraint Optimization Problem (HCOP) as a new paradigm to model distributed optimization problems which meet those features. Its main motivation is modeling such problems through the integration of solvable subproblems into which they may be naturally partitioned. Sections 2 and 3 synthesizes the basic concepts involved in the work, whereas section 4 introduces a problem of the oil supply chain industry, which was the original motivation for the proposed model. Section 5 describes and formalizes the HCOP, and suggests a meta-algorithm, called HCOMA, for its solution. Section 6 characterizes the problem presented in section 4 as a HCOP, and shows the advantage of this approach. Finally, the paper concludes with a summary of the results, and an outlook on future research activities.

2 Holonic Agents

MutiAgent System (MAS) has become a natural tool for modeling and simulating complex systems. However, in those systems there usually exist a great number of entities interacting among themselves, and acting at different *abstraction levels*. In this context, it seems unlikely that MAS will be able to faithfully represent complex systems without multiple granularities. That's why holonic systems have attracted the attention of researchers [Hilaire *et al.*, 2008]. The term holon was coined by Arthur Koestler [Koestler, 1967], based on the Greek words *holos* for whole and *on* for part. Thus, a holon is a self-similar or fractal structure that consists of several holons as components, and is itself a part of a greater whole. A holon (superholon) is composed of other holons (members or subholons) and should meet three conditions: (i) to be stable, (ii) to be autonomous and (iii) to be able to cooperate. Thus, according to Koestler [Koestler, 1967] a *holarchy* is a hierarchy of self-regulating holons that function first as autonomous wholes in supra-ordination to their parts, secondly as dependent parts in sub-ordination to controls on higher levels, called *echelon*s, and thirdly in coordination with their local environment.

Gerber et al. [Gerber *et al.*, 1999] propose three types of structures for holons, which vary with respect to the autonomy of the members. The moderated group is the intermediary structure, which was chosen for this work due to its greater flexibility. According to [Hilaire *et al.*, 2008] it specifies a holonic organization with three main roles: *head* role players are moderators of the holon, whereas represented members have two possible roles: *part*, whose players belong to only one superholon, and *multipart*, where subholons belong to more than one superholon. The *head* represents the shared intentions of the holon and negotiates them with agents outside the holon. The remainder of the holon, i.e. the set of parts and multiparts, is called *body.*

3 Distributed Constraint Optimization Problem (DCOP)

DCOP is a formalism that can model optimization problems distributed due to their nature. These are problems where agents try to find assignments to a set of variables that are subject to constraints. It is assumed that agents optimize their cumulated satisfaction by the chosen solution. This is different from other related formalisms involving self-interested agents, which try to maximize their own utility individually. Thus, the agents can optimize a global function in a distributed fashion communicating only with neighboring agents, and even in a asynchronous way.

3.1 Formalization

According to [Petcu *et al.*, 2007], a DCOP can be defined as a tuple $(A,\ V,\ D,\ F)$ where :

- $A = \{a_1, \ldots, a_n\}$ is a set of n agents,
- $V = \{v_1, \ldots, v_n\}$ is a set of n variables, one per agent,
- $D = \{D_1, \ldots, D_n\}$ is a set of finite and discrete domains each one associated with the corresponding variable,
- $F = \{f_1, \ldots, f_m\}$ is a set of valued constraints f_i, where $f_i : D_{\alpha_1} \times \ldots \times D_{\alpha_k} \rightarrow \mathbb{R},\ \alpha_k \in \{1 \ldots n\}$

The goal is to find a complete instantiation V^* for all the variables v_i that maximizes the objective function defined as

$$F = \sum_i f_i$$

3.2 Available Algorithms

The main complete algorithms developed for DCOP are:

ADOPT It is a backtracking based bound propagation mechanism [Modi *et al.*, 2006], which operates completely decentralized, and asynchronously. Its drawback is that it may require a very large number of small messages, thus producing considerable communication overhead.

OptAPO It is a hybrid between decentralized and centralized methods [Mailler and Lesser, 2004]. It operates as a cooperative mediation process, where agents designated as mediators centralize parts of the problem in dynamic and asynchronous mediation sessions. Message complexity is significantly smaller than ADOPT's. However, it may be inefficient with some mediators solving overlapping problems. Furthermore, the dynamic nature of the mediation sessions make it impossible to predict which part of the problem will be centralized.

DPOP It is an algorithm based on dynamic programming [Petcu and Faltings, 2005] as an evolution of the DTREE algorithm [Petcu and Faltings, 2004] for arbitrary topologies even with cyclic graphs. It generates only a linear number of messages, which, however, may be large and require large amounts of memory, up to space exponential. Therefore it was extended later, and a new hybrid algorithm called PC-DPOP was developed [Petcu *et al.*, 2007], that uses a customizable message size and amount of memory.

4 Motivating Scenario

As discussed in the introduction, an example of actual distributed optimization problem with a recursive organization is the supply chain management. Let us consider an oil company, which may be a single verticalized petroleum enterprise or a set of cooperating companies of the oil business. That enterprise system can purchase from the spot market (SM), which satisfies any extra demands of crude oil and its derivatives at higher prices. In the same way, SM can buy any exceeding inventories of those items at lower prices. A holonic model can be built according to geographical criteria, taking into account the transport integration. Thus there is a global holon, which comprises several continent holons, which are in their turn made up of region holons. These last holons may contain subholons like refineries, which are responsible for the production of oil derivative products, distribution terminals, which store those products, and oil extraction platforms, that yield crude oil (raw material). All the areas are connected by transportation modals, like ships and pipelines, and each area owns a specific logistic entity, which is responsible for planning the transportation of products.

In general, the refineries own their specific centralized optimization system for production planning, whereas the logistics of each echelon also has its respective optimization system for the corresponding transport planning. However, the different systems are not conveniently integrated to allow a global optimization. Figure 1 depicts the holonic echelons of this problem [Marcellino and Sichman, 2010].

5 Holonic Constraint Optimization Problem (HCOP)

Some distributed constraint optimization problems have a hierarchical and recursive structure, which is called *holarchy*. That organization is characterized by entities with great cohesion with respect to their fellows, but only a coupling relationship with their parents and childs along the hierarchy. Therefore, that kind of modeling allows that the optmization problem may be partitioned into a set of smaller optimization problems, which, although not independent from each other, present such a low coupling level that enable some parallelism. In addition it makes it easier to tackle the complexity of the whole system, which is modeled through a simpler model that repeats itself recursively throughout the complete model.

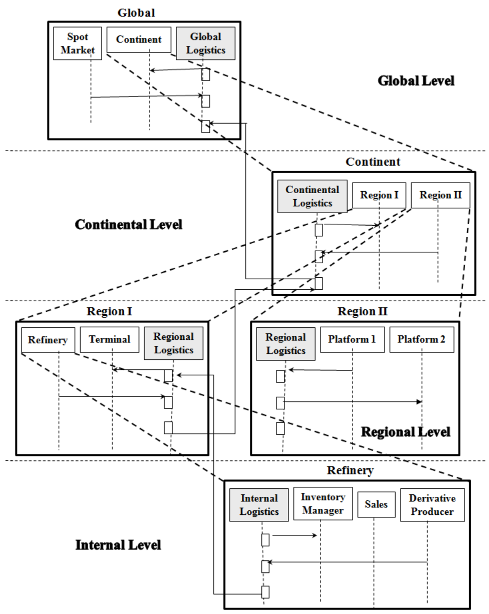

Figure 1: Diagram of an oil supply chain holonic model

5.1 Description

The HCOP consists of a set of agents that are called holons. Those holons are distributed into different abstract levels which are named *echelons*. By definition, a holon contains other holons (its subholons) and is part of another holon (its superholon). However, the most fundamental echelon ($\eta = 0$) comprises only atomic holons, i.e., a conventional agent that doesn't contain any other. Each holon is responsible for a variable. In the case of the atomic holon, it is a *decision variable*, which is an independent variable in the same sense of a DCOP variable. On the other hand, each holon belonging to a higher echelon ($\eta > 0$) is associated with an *emergent variable*, which is dependent on the internal variables of that holon, i.e., the variables associated with its subholons. Such dependency is specified by an *emergent function*.

The holonic organization adopted in this work classifies the subholons into 2 roles : *head*, which is unique for each holon, and *part*, which may be one or more. Here we don't consider the *multipart* role, as it will be seen later. It is also assumed that the *head* holons are atomic in all the echelons, for the sake of the model elegance, avoiding a recursive overload of that kind of holon as η increases. Due to the distinctive behavior of the *head* holon, which is responsible for the communication with the outside world, it is natural to set it apart from the remainder of the holon, which comprises all the *part* holons and is called *body*.

The internal strong cohesion of the holons, and the less intense coupling between a holon and its superholon or subholon, make it possible to view a HCOP as a partition of coupled optimization problems (OPs). Basically each holon may map to a corresponding OP.

5.2 Formalization

The HCOP is formalized as a tuple (H, R, V, D, E, F) where :

- $H = \{H_0, \ldots, H_{\eta_{max}}\}$, where η_{max} is the highest echelon, $H_\eta = \{h_{\eta 1}, \ldots, h_{\eta N_\eta}\}$ is the set of holons of the echelon η, H_0 is the set of atomic agents h_{0i} of the fundamental echelon, and $H_{\eta_{max}} = \{h_{\eta_{max}1}\}$ contains a single holon (*global* holon);
- $R = \{r_1, \ldots, r_R\}$ is the set of relations between the holons, where $r_i$is one of the two primal relations :
  - $headOf_\eta : H'_\eta \rightarrow H_{\eta+1}$
  - $partOf_\eta : H'_\eta \rightarrow H_{\eta+1}$
    where $\eta \in \mathbb{N}, \eta < \eta_{max},\ H'_\eta \subset H_\eta$

  Other important relations derived from these are :
  - $subholonOf_\eta : H_\eta \rightarrow H_{\eta+1}, \eta < \eta_{max}$, where $subholonOf_\eta \equiv headOf_\eta \cup partOf_\eta$
  - $superholonOf_\eta : H_\eta \rightarrow H_{\eta-1}, \eta > 0$, where $superholonOf_\eta \equiv subholonOf_\eta^{-1}$
- $V = \{V_0, \ldots, V_{\eta_{max}}\}$, where $V_\eta = \{v_{\eta 1}, \ldots, v_{\eta N_\eta}\}$ is the set of variables of echelon η (a variable per holon);
- $D = \{D_0, \ldots, D_{\eta_{max}}\}$, where $D_\eta = \{D_{\eta 1}, \ldots, D_{\eta N_\eta}\}$ is the set of discrete and finite domains associated with each variable of echelon η;
- $E = \{E_1, \ldots, E_{\eta_{max}}\}$, $E_\eta = \{E_{\eta 1}, \ldots, E_{\eta N_\eta}\}$ is the set of emergence functions of echelon η (one per holon, but the atomic), $E_{\eta i} : D_{\eta-1\alpha_1} \times \ldots \times D_{\eta-1\alpha_{B_{\eta i}}} \rightarrow D_{\eta i}$, where $B_{\eta i}$ is the *body* size of the holon $h_{\eta i}$, so that $v_{\eta i} = E_{\eta i}\left(v_{\eta-1\alpha_1}, \ldots, v_{\eta-1\alpha_{B_{\eta i}}}\right)$, where the domain is the cartesian product of the internal variables of holon $h_{\eta i}$, and $v_{\eta i}$ its emergent variable;
- $F = \{F_0, \ldots, F_{\eta_{max}}\}$, and $F_\eta = \{F_{\eta 1}, \ldots, F_{\eta N_\eta}\}$, $F_{\eta i} = \{f_{\eta i 1}, \ldots, f_{\eta i M_{\eta i}}\}$ is the set of $M_{\eta i}$ valued constraints between the members of holon $h_{\eta i}$, and $f_{\eta ij} : D_{\eta-1\alpha_1} \times \ldots \times D_{\eta-1\alpha_{M_{\eta ij}}} \rightarrow \mathbb{R}$, whose domain is the cartesian product of the $M_{\eta ij}$ variables$\{v_{\eta-1\alpha_1}, \ldots, v_{\eta-1\alpha_{M_{\eta ij}}}\}$, which is a subset of the set of internal variables of holon $h_{\eta i}$.

The goal is to find a complete instantiation V^* for all variables $v_{\eta i}$ that maximizes the objective function defined as

$$\mathcal{F} = \sum_{\eta=0}^{\eta_{max}} \sum_{i=1}^{N_\eta} \sum_{j=1}^{M_{\eta i}} f_{\eta ij} \tag{1}$$

That definition reflects the holonic feature that there is no direct constraint f between two *part* subholons belonging to different superholons.

Since each holon $h_{\eta i}$ is responsible for an emergent variable $v_{\eta i}$ via its *head* agent, if it is taken into account the emergence function $E_{\eta i}$, it is possible to say that the agent *head* is connected by an n-ary constraint $c_{\eta i}$ with all the members of its holon, so that the following equation must be true :

$$v_{\eta i} = E_{\eta i}\left(v_{\eta-1\alpha_1}, \ldots, v_{\eta-1\alpha_{B_{\eta i}}}\right) \tag{2}$$

5.3 Holonic Constraint Optimization Meta-Algorithm (HCOMA)

As already said, a HCOP can be seen as a holarchy whose holons may map to corresponding OPs. Each of these problems may be represented by a DCOP, or a centralized OP in the case of a holon with greater internal cohesion. Thus HCOP is a distributed OP, which may be modeled as a hybrid network of distributed and centralized optimization subproblems. Therefore, to take advantage of that feature, it is more appropriate a meta-algorithm for a HCOP, rather than a single algorithm. Thus, it is possible to embed into the more abstract framework different DCOP algorithms, or centralized optimization algorithms, in such a way to obtain the most fitting possibility for each case.

Since the holonic organization which was considered in this work does not include *multipart* holons, the macro graph made up of the several holons has a tree structure. In fact, its a connected graph without cycles. Therefore, it was developed a meta-algorithm, which was based on the DTREE algorithm [Petcu and Faltings, 2004]. Such a choice was due to the nature of that algorithm, which is free of backtracking, and hence evolves ininterruptedly upwards and then downwards, in a way compliant with the necessary independence between the optimization subproblems of the HCOP. On the other hand, algorithms like Adopt [Modi *et al.*, 2006] present a behavior which would interweave the holarchy echelons during the solving process and make that decoupling very hard.

At a first glance the exclusion of *multipart* holons seems an oversimplification, which aims at the reduction to a tree structure. However, in the same way DTREE evolved to DPOP (vide subsection 3.2) by arranging the relevant graph as a pseudotree, what is possible for any graph, it is straightforward to adapt HCOMA accordingly. Thus that enhancement would include *multipart* holons and support a general topology, keeping the backtracking free trait.

The proposed meta-algorithm has 3 phases, which are described in Algorithm 1. It is assumed that a generic and trustworthy optimization algorithm, distributed or centralized, is available in the scope of each pertinent holon. However, it must respect the protocol specified in Algorithm 2. For the sake of simplicity and readability, it was used another derived relation, as well as the predicate *head*, which are defined as :

$$headOfPart_\eta \equiv headOf_\eta(superholonOf_\eta) : H'_\eta \rightarrow H''_\eta$$

$$head(h_{\eta i}) := \exists h_{\eta+1\,k} \in H_{\eta+1} |\, h_{\eta i} = headOf(h_{\eta+1\,k}),$$
where $\eta \in \mathbb{N}, \eta < \eta_{max}, H'_\eta \subset H_\eta, H_\eta = H'_\eta \cup H''_\eta$

The phase 1 is a bottom-up process, which starts from the atomic holons and propagates upwards up to the global holon. In phase 2 the global holon owns the maximum utilities associated with each value of its emerging variable. That means it has the maximum values of the global objective function for each value of its variable. Then it will choose the highest value, which will represent the optimum value of the global objective function, whereas the associated value of its variable is the first assignment of the solution. Finally, in phase 3 the global holon will send the index of that solution, regard-

Algorithm 1 HCOMA - Holonic Constraint Optimization Meta-Algorithm

HCOMA(H, R, V, D, E, F)
Phase 1: Utility Computation
for all $h_{0i} \in H_0$ **and not** $head(h_{0i})$ **do**
  for all $ind \in \{1, \ldots, \mid D_{0i} \mid\}$ **do**
    $BestUtil_{0i}[ind] \leftarrow 0$
  end for
  send READY_msg($BestUtil_{0i}$, i) to $headOfPart_0(h_{0i})$
end for
return

READY_msg_handler($BestUtil$, j){by holon $h_{\eta-1\,k}$}
$h_{\eta\,i} \leftarrow superholonOf_{\eta-1}(h_{\eta-1k})$
for all $ind \in \{1, \ldots, \mid D_{\eta-1j} \mid\}$ **do**
  $BestUtilBody_{\eta\,i}[j][ind] \leftarrow BestUtil[ind]$
end for
if READY_msg received from $h_{\eta-1\,l}, \forall l, h_{\eta-1\,l} = partOf_\eta(h_{\eta i})$ **then** {received from all its parts}
  for all $I \in \{1, \ldots, \mid D_{\eta i} \mid\}$ **do**
    $c \leftarrow$**constraint**($d_{\eta i I} = E_{\eta i}(v_{\eta-1\alpha})$){$d_{\eta i I} \in D_{\eta i}$}
    call $OptAlgth_{\eta i}(c, BestUtilBody_{\eta i}, I)$
  end for
end if
return

UTIL_msg_handler($BestUtil$, ind){by holon $h_{\eta-1\,k}$}
$h_{\eta\,i} \leftarrow superholonOf_{\eta-1}(h_{\eta-1k})$
$BestUtil_{\eta i}[ind] \leftarrow BestUtil$
if UTIL_msg received for all $ind \in \{1, \ldots, \mid D_{\eta i} \mid\}$ **then**
  if $\eta < \eta_{max}$ **then**
    send READY_msg($BestUtil_{\eta i}\, i$) to $headOfPart(h_{\eta i})$ {to its head}
  else {Global Holon}
    Phase 2: Global Optimization
    $ind^* \leftarrow argmax_{ind}(BestUtil_{\eta i}[ind])$
    $OptimumUtil \leftarrow BestUtil_{\eta i}[ind^*]$
    $v^*_{\eta i} \leftarrow d_{\eta i\, ind^*}$
    Phase 3: Termination
    for all $h_{\eta-1\,k} = partOf_\eta(h_{\eta i})$ **do**
      send VALUE_msg(ind^*) to $h_{\eta-1\,k}$
    end for
  end if
end if
return

SOLUTION_msg_handler($SolInd$, $UpperInd$){by holon $h_{\eta i}$}
$SolInd_{\eta i}[UpperInd] \leftarrow SolInd$
return

VALUE_msg_handler($UpperInd^*$){by holon $h_{\eta i}$}
$ind^* \leftarrow SolInd_{\eta i}[UpperInd^*]$
$v^*_{\eta i} \leftarrow d_{\eta i\, ind^*}$ {$d_{\eta i\, ind^*} \in D_{\eta i}$}
if $\eta > 0$ **then**
  for all $h_{\eta-1k} = partOf_\eta(h_{\eta i})$ **do**
    send VALUE_msg(ind^*) to $h_{\eta-1\,k}$
  end for
end if
return

Algorithm 2 Optimization Algorithm of holon $h_{\eta\, i}$

$OptAlgth_{\eta\, i}(ctr_{\eta\, i\, ind},\ BestUtilBody_{\eta\, i},\ ind)$

Require: - n-ary constraint $ctr_{\eta\, i\, ind}$

- $BestUtilBody_{\eta\, i}[k][l]$ corresponding to each $d_{\eta-1 k\, l} \in D_{\eta-1\, k}$ associated with each subholon $h_{\eta-1\, k}$ of $h_{\eta\, i}$

Ensure: - the maximum $BestUtil$ corresponding to value $d_{\eta\, i\, ind} \in D_{\eta\, i}$ such that $BestUtil \leftarrow argmax_l(\sum_{j=1}^{M_{\eta i}} f_{\eta ij}(d_{\eta-1\, k\, l}) + +\sum_k BestUtilBody_{\eta i}[k][l])$

- the index $SolInd_{\eta-1\, k}$ of each variable $v_{\eta-1\, k}$ of the subholons of $h_{\eta\, i}$, corresponding to the solution.

executes the optimization algorithm of holon $h_{\eta\, i}$

send UTIL_msg($BestUtil,\ ind$) to $headOf_{\eta}(h_{\eta\, i})$

for all $h_{\eta-1\, k}$ = partOf($h_{\eta\, i}$) **do**

  send SOLUTION_msg($SolInd_{\eta-1\, k},\ ind$) to $h_{\eta-1\, k}$

end for

return

ing its variable domain, to all its parts via a VALUE message. By using that index, each part subholon will determine its own solution value, and recursively will send its respective index to its parts by VALUE messages. This phase is a top-down process, which is initiated by the global holon, propagates downwards down to all the atomic agents, when the meta-algorithm terminates. Figure 2 outlines HCOMA. The proof that it is sound and complete can be obtained in a straightforward way, as it will be shown next.

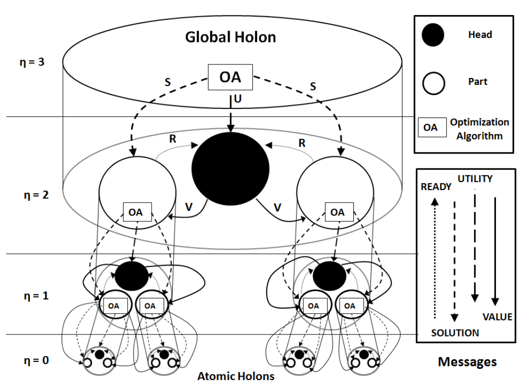

Figure 2: Outline of HCOMA

Proof of Correctness and Complexity

In any holon of any echelon, its *head* receives a UTIL message for each value of its variable domain. Each of these utility values is obtained from the execution of the optimization algorithm of the superholon, which uses the best utilities assigned to the domain values of the variables of each *part*. Since it is assumed that the distributed or centralized optimization algorithm is correct and terminates properly, if the best utilities associated with the *parts* domain values are correct, it can be concluded that the *head* will receive the best utilities for each value of its variable domain. But the atomic holons of echelon $\eta = 0$ own the best utilities for each value of their domains, for they depend only on themselves or internal optimization algorithms that are correct. Hence, by induction, it can be infered that any holon of any echelon will receive the best utilities for each value of its domain, since there is no *multipart* agent, i.e., there are no cycles in the graph made up of the holons and the constraints between them, and therefore the utility associated with each holon is considered only once. Thus, the global holon will choose the best utility for the entire holarchy, and then all the subholons will be informed and choose its respective domain values associated with that solution.

Due to its tree structure HCOMA has polynomial time complexity. As to the number of messages, it inherits the behavior of DTREE algorithm, which is linear in the number of agents (here holons) [Petcu and Faltings, 2004].

6 Modeling Example

As mentioned in section 4, the oil supply chain management is a real candidate problem to be modeled as a HCOP. In [Marcellino and Sichman, 2010] this problem was modeled as a DCOP, where the objective function is the total profit in the whole chain; it is also used a holonic approach: some holons, like the Transport Planners and the Derivative Producers, wrapped centralized optimizers.

In all the levels (echelons) of the supply chain the *logistics* is the *head* of each holon i.e., the *internal logistics* for the holon *refinery* or *terminal*, the *regional logistics* for the holon *region*, and so on, up to the *global logistics* for the holon *global*. Each *logistics* is responsible for the balance of products between the *suppliers* and the *clients*, and manages the transportation planning. The *supplier* role refers to any entity that provides products to other entity of the chain, such as a *refinery,* whereas the *client* role is played by any entity which needs these products, such as a *terminal*. The difference between availabilty and need of a product represents the emergent variable of the corresponding holon.

A refinery can produce multiple derivatives, and it does so according to different production plans, which are characterized by processing a definite quantity of a particular type of crude oil and producing a certain quantity of each resulting derivative. In addition, each refinery or terminal is responsible for the management of its inventories of each product. Thus, the decision variables of the model in the basic echelon ($\eta = 0$) are the *production plan* adopted by each refinery during each period of time, and the *inventory level* of each product in each refinery or terminal at the end of each period of time.

The higher the echelon, the larger the spatial scope of the corresponding holons. Similarly the higher the echelon, the longer the period of time considered by the head logistics in its planning. Thus, the holons result from a spatial and temporal discretization along growing abstract levels. On the other hand, the problem comprises different OPs: the production optimization of each refinery, and the transport optimization of the logistics in each holon. These latter are associated with growing echelons, and gradually embody larger geographic areas and longer planning periods of time. In fact, the inter-

	Ref. 1	Ref. 2	Income	Cost	Profit
Local Optim.	Plan B	Plan B	5920	629	5291
Holonic Model	Plan C	Plan C	10080	4162	5918

Table 1: Integrated holonic X Conventional approach

nal logistics is responsible only for a refinery or terminal on a day-by-day basis, the regional logistics takes care of an entire region with the week as the time unit, and so on up to the global logistics which focuses on the whole enterprise with a planning horizon of semesters or even years.

Let us consider a case study, which is simple but representative. It includes all the relevant entities of the chain and a significant set of products. It contains one continent with three regions, and one overseas SM. The first two regions comprise one refinery and one terminal, whereas the third region contains only one oil extraction area. Inside the regions, entities are connected by pipelines, but regions and SM are connected to each other by ships. The refineries produce three derivatives (gasoline, diesel and naphtha) by processing three types of crude oil (pet1, pet2 and pet3). The refineries can operate according to three production plans, which are specific to each refinery: plan A, plan B, and plan C. Although it is not a real situation, it is representative and fits for a proof-of-concept, which is accomplished by comparing the holonic model with a usual approach to manage the oil supply chain, which is based only on local optimization. The results of applying both approaches are presented in Table 1.

The local optimization approach recommends plan B in refineries 1 and 2, since it leads to the highest supposed profit (5291). On the other hand, the holonic model presents as optimum choice to adopt production plan C in both refineries, with a total profit of 5918. Such a discrepancy comes from the myopia of the local approach, which is unable to consider aspects of higher echelons, such as the additional profit resulting from sales of surplus products to SM, or the penalties incurred by not having product enough to supply all customer demands. Therefore, the proposed model generates a global gain of about 12 % on account of the whole chain integration.

7 Conclusions and Future Work

In this paper, we have defined a Holonic Constraint Optimization Problem (HCOP), which combines the distributed optimization constraint approach with the holonic multi-agent approach to take advantage of the best of both worlds. On one hand, since the constraint model provides a tight integration of the involved entities, it allows optimization. On the other hand, the holonic approach makes it possible to represent the intrinsic recursive nature of a category of optimization problems, such as the supply chain management. In addition, it was developed the meta-algorithm HCOMA for the solution of the HCOP. Since it is a meta-algorithm, it makes it possible to integrate different optimization algorithms, which may be chosen according to each specific problem.

In a future work the model will be extended to include more complex holarchies with multipart agents. Furthermore, it will be treated the environmental parameters and their influence on the stability of the holonic solution. In other words, it will be studied how an environmental perturbation propagates between echelons, and the possible advantages of the proposed model to tackle such kind of changes. Another point to be investigated is how the communication problems between neighboring echelons may harm the quality of a HCOP solution. Finally, it will be developed a prototype based on a case study of the oil supply chain, where the HCOMA will be implemented using as components centralized optimization algorithms already available in the oil industry.

Acknowledgment

Jaime Simão Sichman is partially supported by CNPq/Brazil.

References

[Gerber *et al.*, 1999] C. Gerber, J. Siekmann, and G. Vierke. Holonic multi-agent systems. *Research Report*, 99(3), 1999.

[Hilaire *et al.*, 2008] V. Hilaire, A. Koukam, and S. Rodriguez. An Adaptative Agent Architecture for Holonic Multi-Agent Systems. *ACM Transactions on Autonomous and Adaptive Systems*, 3(1), 2008.

[Koestler, 1967] A. Koestler. *The Ghost in the Machine*. Hutchinson 'I&' Co, London, 1st edition, 1967.

[Mailler and Lesser, 2004] R. Mailler and V. Lesser. Solving distributed constraint optimization problems using cooperative mediation. In *Proceedings of Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2004)*, pages 438–445. IEEE Computer Society, 2004.

[Marcellino and Sichman, 2010] F. J. M. Marcellino and J. S. Sichman. Oil industry supply chain management as a holonic agent based distributed constraint optimization problem. In *AILog - ECAI 2010*, Lisbon, Portugal, 2010.

[Modi *et al.*, 2006] P. J. Modi, W. Shen, M. Tambe, and M. Yokoo. Adopt: Asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligence*, 161:149–180, 2006.

[Petcu and Faltings, 2004] A. Petcu and B. Faltings. A distributed, complete method for multi-agent constraint optimization. In *Proceedings of the Fifth International Workshop on Distributed Constraint Reasoning (DCR2004) in CP 2004*, 2004.

[Petcu and Faltings, 2005] A. Petcu and B. Faltings. A scalable method for multiagent constraint optimization. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 266–271, 2005.

[Petcu *et al.*, 2007] A. Petcu, B. Faltings, and R. Mailler. Pc-dpop: a new partial centralization algorithm for distributed optimization. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 167–172, Hyderabad, India, 2007.

[Yokoo *et al.*, 1992] M. C. Yokoo, E. H. Durfee, T. Ishida, and K. Kuwabara. Distributed constraint satisfaction for formalizing distributed problem solving. In *International Conference on Distributed Computing Systems*, pages 614–621, 1992.