Improving resource exploitation via Collective Intelligence by assessing agents' impact on the community outcome

Fabio Boschetti,
CSIRO CMAR

Abstract

A Collective Intelligence (COIN) can improve the exploitation of a limited renewable resource compared to fully cooperative or fully competitive approaches. The main strength of a COIN lies in approximating the impact of an agent on the short-term behaviour of a Complex Adaptive System. By penalising behaviours which lead to no measurable impact, COIN simplifies the implementation of an appropriate cost function which each agent needs to optimize in order to reach a global, community-wide goal. On a number of virtual experiments mimicking a fishing fleet operating in areas of different fishing capacity, a COIN provides optimal catches for the fleet while at the same time each individual vessel also maximizes its own profit: no individual sacrifice is required to achieve the common goal. In the view of possible application by real human agents, I propose a simplified implementation of a COIN, which involves only elementary numerical operations and minimum bookkeeping and can thereby be carried out simply by 'pen and paper', with no help of electronic devices.

Introduction

In a world of limited resources and ever expanding demands, the contribution that scientific research can offer to resource exploitation and management is not limited to the study of the dynamics of the resource itself, but, at least as important, includes understanding how humans interact with the resource and compete to access it. Today, for a large section of the scientific community, understanding and modelling are synonymous; it is via comparing modelling results to reality that we check whether our assumptions about a problem are correct. A 'good' model, which includes satisfactory approximation of main factors, dynamics and causal relations, encloses our understanding of a problem and our (currently best) hope for prediction. Within the framework of natural resource exploitation, this implies that we need to model the drivers which lead humans towards a resource, the way they compete, the way they obtain and process information about the resource and eventually the way they take decisions on how to act upon it. I address this problem by simplifying and porting to ecological modelling ideas taken from the Collective Intelligence literature (Wolpert and Tumer 1999) and linking them to more established agent-based and game-theoretical tools.

The Collective Intelligence (COIN) main strength lies in approximating the impact of an agent's action on the dynamics of the overall population. This may be more or less difficult depending on the management style. Under a strong top-down management scenario, simulating a community of agents is fairly simple, since the action of each agent is strongly rule-based and thus fairly predictable. If however the management style allows for competition and adaptation, then the agents' behaviour is far less constrained and the resulting dynamics much more complicated. In this scenario, local interactions among agents may result in large scale community-wide behaviour whose dynamics is difficult to predict (at least without modelling) from the knowledge of each agent's action. The arising of large scale dynamics from fundamentally different small scale dynamics is often defined as emergence[1] (Boschetti et al, 2005). Emergence is thus crucial to both the manager and the agent; the manager needs to 'engineer' policies in order to achieve resource-wide or community-wide outcomes with the understanding that the relation between the policies (which act at the small scale of the agent) and the aimed outcomes (at the scale of a community and resource) is not trivial. Each agent also needs to understand this emergent process in order to choose how, where and when is best to access a resource, depending on the behaviour of the other agents and the resource itself. Thus, the question of how an agent's action affects the community behaviour is crucial to both the manager and the agent. Here is where COIN plays a role.

COIN's crucial insight lies in discriminating between the agent's *contribution* to the community outcome and its *impact* on it. Here the contribution is the part the agent plays in the final outcome. The impact is how the agent directly affects the outcome or, said differently, what the outcome would be without the agent's intervention. An example clarifies the difference. Two agents, Paul and Mary, wish to collect apples from an orchard. Each can carry at most 2 bags of apples, one per hand. In the first scenario the orchard produces 2 bags of apples. Paul and Mary collect one bag each. Paul contribution is one bag. His impact however is zero, since had Paul not been there, Mary would have been able to collect both bags. In the second scenario the orchard produces 4 bags of apples. Now Paul and Mary collect two bags each. Paul's contribution is two bags. His impact now is also 2 bags, since had Paul not been there, Mary would have been able to carry only 2 bags, and 2 bags would have been left uncollected.

Previous work in COIN (Wolpert et al., 2000, Wolpert and Tumer, 2001) shows that the apparently minor difference between contribution and impact plays a major role in simplifying optimisation problems in which agents need to take local decisions in order to solve a global problem. This is particularly relevant to this work since the management of a limited resource can indeed be seen as an optimisation problem in which the manager aims to optimise (or at least improve) global exploitation and sustainability and the agents aim to optimise (or at least improve) their local return and long term gain. A vast literature (Hardin, 1968; Batten, 2005) and an even vaster set of real world examples suggest that these two aims are in direct conflict: the selfish (local) interest of each agent often goes against the public (global) good.

The main result in this work is to show that this is not necessarily the case if the difference between *contribution* and *impact* is accounted for; via modelling a fishery exploitation problem, I show that the use of COIN leads to improved resource exploitation not only for the overall community but *also for each individual* (on average); that is, no personal sacrifice is required for the good of the community. This has the

---

[1] Emergence is a hardly debated term in the physical and philosophical community. In this work I use it in the weak sense described in the text (see also Bedau, 1997).

potential to offer a radical shift in the way communal resources are managed and is worth an in-depth investigation, of which this work represents a first step.

Apart for porting COIN to ecological modelling, this work provides two further contributions. First, despite the simplicity of underlying idea, COIN literature is fairly cryptic and rich of terminology not easily accessible to ecological modellers; here I strive to describe the COIN algorithm in the simplest possible fashion. Second, COIN was not designed with human agents in mind; I present a simplified COIN which could potentially be employed by real people with *no need of computer aid*, by simply performing elementary calculation with pen and paper. The results could naturally be extended to the exploitation of resources other than fisheries.

The paper is organised in the following way. First, I cast the management of a limited renewable resource within a game-theoretical framework by describing the Minority Game and its self-referential and self-defeating nature. I then describe the agent-based model employed and how this can simulate four different virtual fishing fleets; a fully competitive one, a fully cooperative one, one which follows COIN ideas and one which takes fully random actions. After testing the four approaches on a number of fishing scenarios, I conclude by discussing the current limitations of the method and some directions for future study.

1. COMPETITION FOR LIMITED RESOURCES IN THE VIRTUAL FISHERY - THE MINORITY GAME

In the last few years a considerable body of work has been published on the study of the Minority Game (Zhang, 1998; see also the pioneering work by Arthur (1994) on the related El Farol Bar Problem and http://www.unifr.ch/econophysics/minority/ for an exhaustive collection of papers on the subject). Despite employing largely unrealistic simplifications in the behaviour of the agents, this tool has allowed scientists (mostly physicists) to highlight the unexpected and often counterintuitive dynamics displayed by the community of competitive agents as a whole, and how this changes dramatically as a function of the agents' behaviours (Savit et al, 1999).

The fundamental reason for the complexity of the Minority Game (and of the competition for limited resource in general) is its self-referential and self-defeating nature (Batten, 2005; Batten and Boschetti, 2006). Imagine a group of individuals who routinely choose where to access a limited resource. The amount of the resource an individual can obtain depends on how many other individuals choose to access the resource at the same location (since the resource is limited and needs to be shared). How many individuals choose a specific location depends in turn by the expectation that the location will be more or less exploited. Such expectation will guide the individuals' choice and consequently its final level of exploitation. Consequently, the expectation actually determines the outcome: this is the self-referential aspect. Also, the more people expect a location to be profitable, the more people will access it and the less profitable the location will result (since too many individuals have to share the limited resource). The expectation actually determines the *opposite* outcome: this is the self-defeating aspect.

A typical simulation of the Minority Game is shown in Figure 1. It shows a virtual community of 20 agents who access a resource distributed equally at two locations (A and B). Each agent strives to select the location with the lowest attendance, that is, the location chosen by the minority (from which the Minority Game name). If it guesses correctly, it will share the resource with the minority, rather than the majority, of agents and its return will be higher. Obviously, at each iteration, the majority of agents receives a lower return; the game by construction can satisfy only the minority of the players.

Each agent chooses its strategy (at what location it will access the resource) as a function of past returns, which is the only available information to the agents. Suppose location A was chosen by the minority of agents. These agents obtain a good return and will tend to stay in location A at the next iterations. The agents who chose location B (the majority) obtain bad returns and soon will tend to move to Location A. This will result in location A attracting the majority of agents and becoming unprofitable. Consequently, any winning strategy will soon turn into a losing strategy, which represents the self-referential and self-defeating nature of the game. The result, as shown in Figure 1, is that the number of agents at each location tends to fluctuate in a complex fashion, never settling to any fixed value, and in the long run the average of the attendance to each location will even out.

So far we discussed the return for each agent, let's now analyse the return for the entire community. At each iteration, the global return to the community is optimised by maximising the size of the minority[2]. This happens when the difference between the size of the majority and of the minority is lowest. Optimal resource exploitation is then obtained when the agents spread evenly among the available locations. Thus, the smaller the fluctuations in Figure 1, the larger the return to the overall community; the amplitude of the fluctuations is a measure of the resource waste in the community which results from agents behaving in a purely competitive fashion.

2. PROBLEM SETTING

The agent-based model employed in this work represents a generalisation of the Minority Game described in the previous section. A fishing fleet composed of competitive or cooperative vessels targets a limited renewable resource of a single fish species. The fleet includes *N* fishing vessels $n = 1..N$ (agents). At each iteration of the simulation, the vessels choose where to fish among *Z* available zones (a vessel has to choose one and only one zone at each iteration). In each zone a certain amount of fish $Fish_z, z = 1..Z$ is available. This amount can change from zone to zone but, unless otherwise stated, is fixed in time. This means that we are not modelling the population dynamics of the fish species and we assume a constant amount of fish is present in the zones at each iteration, independently of how much has been caught in the past. Clearly, this is an unrealistic simplification of

[2] As explained below, in this work I assume that each agent has a maximum capacity, so that not all the resource available at one location can be exploited when only a few agents attend to it. In standard Minority Game, agents obtain a return only if they attend the minority choice and no return otherwise.

the problem and was chosen in order to discriminate the effect of the vessels actions from the fish population dynamics.

The amount of fish available at a particular zone is shared equally among all the vessels which chose to operate in that specific zone. Consequently, the catch of each vessel depends on the action of all other vessels (the larger the number of vessels which access an area, the smaller their individual catch). Also, each vessel has a maximum fishing capacity, that is, it is not able to catch more than a predetermined amount (the same for each vessel). Thus the catch for each vessel is given by:

$$Catch_n = Min(Fish_{zone_n} / Crowd_{zone_n}, MaxCatch) \tag{1}$$

where *Min(a,b)* is the smaller number between *a* and *b*, $Catch_n$ is the amount of fish caught by vessel *n*, $zone_n$ is the fishing zone chosen by vessel *n*, $Fish_{zone_n}$ is the amount of fish available in $zone_n$, $Crowd_{zone_n}$ is the number of vessels which chose to fish in $zone_n$ and *MaxCatch* is the fishing capacity of each vessel.

Naturally, the total catch of the fleet is given by the sum of each individual catch,

$$TotalCatch = \sum_{n=1,N} Catch_n \tag{2}$$

Because each vessel has a maximum fishing capacity, we have:

$$\sum_{n=1,N} Catch_n \le Min(\sum_{z=1,Z} Fish_z, N * MaxCatch) \tag{3}$$

that is, the fleet can catch the maximum allowed amount of fish only provided the vessels spread their fishing effort strategically among the different available zones. As a simple example, if the entire fleet targets only *zone 1*, then $TotalCatch = Min(Fish_1, n * MaxCatch)$ which is $< Min(\sum_{k=1,K} Fish_k, n * MaxCatch)$, provided $\sum_{k=1,K} Fish_k \le n * MaxCatch$ (notice that $\sum_{k=2,K} Fish_k \ne 0$). In particular, since all vessels in our test have the same fishing capacity and the amount of fish at each zone may differ, an optimal catch can be achieved only provided the vessels spread their effort proportionally to the amount of fish available in each fishing zone. The fleet has thus to reach this optimal spread with no a priori information about fish distribution.

3. THE AGENT BASED MODEL

The agent based model can be summarised as follows:

1) Initialisation: a random strategy (W_z) is assigned to each vessel[3].
2) At each iteration *t* and for each vessel, update the value of W_z based on the catches in the last *T* steps. This simulates the effort in analysing catch records in order to plan the fishing operation at *t+1*. The value of *T* can be seen as a measure of the length of the history available in the record, or the 'memory' of the vessel crew, or the effort which is employed in the planning. In all our tests, we used *T=20*.
3) Each vessel randomly chooses which zone to fish next, with $P_z \propto W_z$, where P_z is the probability of choosing zone *z*.
4) Calculate the number of vessels which aims towards each zone (*Crowd*).
5) Calculate the catch for each vessel according to Equation 1 and the total catch according to Equation 2.
6) Go back to point 2 and iterate for a predetermined number of steps.

The crucial component of the algorithm is the criterion used to update the strategies W_z at point 2. In this work, a strategy is represented as a single set of weights, one for each fishing zone $W_z, z = 1..Z$. W_z gives a measure of the expectation of obtaining a good catch at zone *z*. At each iteration, a vessel performs a random pick among the *Z* zones, with the probability of selecting zone *z* proportional to W_z .

Within an optimisation framework, this parameterization allows for a balance between *exploration* and *exploitation*. Exploitation is represented by W_z : the more a vessel expects a specific zone to provide a good catch, the more likely the zone will be chosen. Exploration is represented by the random pick which allows non optimal zones to be chosen with lower probability. In the statistical physics parlance, this corresponds to a Boltzmann sampling with no temperature decay (Kirkpatrick et al, 1983).

At each iteration *t*, each vessel evaluates the catches it obtained in the past *T* iterations and it uses this information to set W_z (that is, to predict the most profitable fishing zone at time *t+1*). In order to account for non stationarity in the predictions, the past catches at times $t_p = t - T + 1...t$ are discounted proportionally to $t_d = t - t_p$, that is, the longer ago the catch was taken the less it influences the prediction. This implementation can be seen as a simplification of the ones proposed in Wolpert and Tumer (1999) and Wolpert and Tumer (2001) as it will discussed in more details in Section 5.

The difference between a fully competitive approach (as used in the standard Minority Game), a fully collaborative approach and the COIN lies in how past catches are analysed in order to determine W_z , which is the topic of next section.

---

[3] A random sequence of past catches is also selected. The purpose of this sequence is to provide a set over which the performance of the strategies can initially be evaluated. Sufficiently long runs of the algorithm are unaffected by this random initialization.

4. OPTIMISING THE WORK OF A COMMUNITY

In this paper we ask what strategy individual vessels should employ in order to achieve the best global exploitation of a resource (and possibly best individual return too). This can be cast as an optimization problem. Here the word optimization has a broader meaning that just maximizing an economic return. Rather we refer to the search, within the space of all possible vessels' strategies for set(s) which results in a specific global outcome (in our case best global catches).

There are several tools available to tackle optimization problems within the applied mathematics, engineering and computer science community. However, before choosing one among them, we need to appreciate the difficulty underlying this specific problem.

One of the main challenges in numerical optimization is how to design a suitable 'cost function', that is a measure of how good a certain outcome is, compared to the outcome we wish to achieve. This measure is used by optimization algorithms to guide their search in the parameter space. In our problem, one option for a 'cost function' could be to maximize the catch of each individual vessel. In this approach each individual vessel would try to maximize its return, without accounting for other vessels' action. This is the approach used by agents in the standard Minority Game; as seen in Section 1, when each agent tries to optimise its own return, the return of the entire community (and as a result of each agent in average) is far from optimal. In optimization jargon, this is described by saying that the global goal and the individual goals are not *aligned*, that is, by improving the performance on a single individual goal we do not necessarily improve the performance of the global goal. In everyday jargon, we would say that the individuals act at cross purposes or "tread on each other toes". In the rest of the paper I will call this approach *MG*, as short-hand for Minority Game.

Another natural choice for a cost function could be to assign to each *individual* agent a share of the *global* catch (that is, the better the global catch, the more each vessel is rewarded). In game theory this is called a 'team game' and represents a fully cooperative approach. This solution would work well with small communities. When the number of agents increases, 'team game' performances tend to worsen quickly (Wolpert and Tumer, 2001). The reason for this lies, once again, in the self-referential nature of the problem. When many agents are modelled, it is hard for an individual agent to determine how much its own action has affected the global catch. Agent *n1* may have taken a very unprofitable action, but the actions of all other vessels may have compensated for it by still producing a good global catch. Agent *n1* may thus 'think' its action was profitable and may adopt it again in the future, thereby preventing the community from improving its performance. In Wolpert et al. (2004) this problem is called lack of 'intelligence', and is referred to the inability of the vessels to obtain useful information ('intelligence') about the problem. In the rest of the paper I will call this approach *TG*, as short-hand for Team Game

A third option for a 'cost function' could be for vessels to choose strategies in a completely random fashion. This represents a situation in which no information is available to the vessels in order to make an informative choice. In the rest of the paper I will call this approach RAND.

Finally a fourth choice is represented by accounting for the difference between each agent's contribution and impact as in the COIN approach.

5. COIN IMPLEMENTATION

COIN addresses the two problems describe in the previous section ('greed' and lack of 'intelligence') and, remarkably, the solution it proposes is very simple. Here we illustrate briefly the method, while we refer the reader to Wolpert and Tumer (1999) for a detailed description of the mathematics, together with the theorems and their proof which give a solid base to the theory.

COIN overcomes the 'team game' problem by attempting to estimate the impact of each individual vessel to the global catch. This can be achieved by calculating what the catch of the fleet would be if a specific vessel *n1* (say) did not exist. In the rest of the paper we will use the superscript '*-n*' to refer to values calculated for the fleet as if vessel *n* did not exist. For example $TotalCatch^{-n1}$ is the (hypothetical) catch of the entire fleet in the absence of vessel *n1*. Notice that, as we explained in the Introduction via the example of Paul and Mary in the orchard, this is different from calculating a quantity for the entire fleet minus the same quantity for vessel *n1*: $TotalCatch^{-n1} \neq TotalCatch - Catch_{n1}$. Rather, the imapct of vessel *n1* to the total catch should be calculated as:

$$E_{n1} = TotalCatch - TotalCatch^{-n1} \tag{4}$$

In Wolpert and Tumer (1999) it is proved that any difference function of the form of Equation 4 is *aligned*, that is, by increasing E_{n1} we can not decrease *TotalCatch*. It results that if we could optimize E_n for each individual vessel then we would automatically optimize the catch of the entire fleet. Furthermore, by construction, this would also optimise the catch of each vessel.

In practice, this approach would be extremely complicated to implement, if not impossible. Because of the self-referential nature of the problem, it is very hard to calculate what the behaviour of the fleet would be if vessel *n1* did not exist. This is because the behaviour of each vessel in the fleet is affected by the presence/action of *n1*. Basically, we would have to account for the dynamics of the game and rerun the entire simulation from the starting point.

It turns out that a remarkable (computationally feasible) simplification is possible which allows for the general idea to be useful. In Wolpert and Tumer (1999) it is shown that the impact of an agent (in our case vessel) on the community can be approximated by simply replacing the result of its action (in our case its catch at the chosen fishing action) with a default value. Different options for such value are discussed in Wolpert and Tumer (2001). This means that we can circumvent the problem of accounting for the dynamics of the problem and still implement the idea with meaningful results. The simplest option for this approach (and the one which make more sense in our application) is to use 0 as default value. In our example (Equation 4), this corresponds to vessel *n1* not going fishing and calculating the resulting $TotalCatch^{-n1}$. Notice that this calculation is fictional, because in our model vessel *n1* can not choose not to go fishing. In the COIN literature this approximation is given the fanciful name of Wonderful Life Utility (WLO), in the name of a famous old movie.

Since the WFO is aligned and avoids the 'team game' problem, both self-referentiality and self-defeatingness are circumvented. The Wonderful Life Utilities does indeed make Life 'wonderfully' easy. Equation (4) thus becomes

$$E_{n1} = TotalCatch - TotalCatch^{-n1} = \sum_{n} Catch_n - \sum_{n \neq n1} Catch_n^{-n1} =$$

$$\sum_{n} Min(Fish_{zone_n} / Crowd_{zone_n}, MaxCatch) - \sum_{n \neq n1} Min(Fish_{zone_n} / Crowd_{zone_n}^{-n1}, MaxCatch) =$$

$$\sum_{k} Crowd_{zone_n} * Min(Fish_{zone_n} / Crowd_{zone_n}, MaxCatch) - \sum_{k} Crowd_{zone_n}^{-n1} * Min(Fish_{zone_n} / Crowd_{zone_n}^{-n1}, MaxCatch) =$$

$$\sum_{k \neq zone_{n1}} Crowd_{zone_n} * Min(Fish_{zone_n} / Crowd_{zone_n}, MaxCatch) + Crowd_{zone_{n1}} * Min(Fish_{zone_{n1}} / Crowd_{zone_{n1}}, MaxCatch) -$$

$$(\sum_{k \neq zone_{n1}} Crowd_{zone_n}^{-n1} * Min(Fish_{zone_n} / Crowd_{zone_n}^{-n1}, MaxCatch) + Crowd_{zone_{n1}}^{-n1} * Min(Fish_{zone_{n1}} / Crowd_{zone_{n1}}^{-n1}, MaxCatch)$$

since $Crowd_z = Crowd_z^{-n1}, \forall z \neq zone_{n1}$ we can write

$$E_{n1} = Crowd_{zone_{n1}} * Min(Fish_{zone_{n1}} / Crowd_{zone_{n1}}, MaxCatch) - Crowd_{zone_{n1}}^{-n1} * Min(Fish_{zone_{n1}} / Crowd_{zone_{n1}}^{-n1}, MaxCatch)$$

also, since $Crowd_{zone_{n1}}^{-n1} = Crowd_{zone_{n1}} - 1$, we have

$$E_{n1} = Crowd_{zone_{n1}} * Min(Fish_{zone_{n1}} / Crowd_{zone_{n1}}, MaxCatch) - (Crowd_{zone_{n1}} - 1) * Min(Fish_{zone_{n1}} / (Crowd_{zone_{n1}} - 1), MaxCatch) \quad (5)$$

Notice that in order to implement Equation 5, a vessel only needs to know how many vessels shared its fishing zone ($Crowd_{zone_{n1}}$) and their catch.

Referring back to the agent based algorithm in Section 3, we can now describe the 4 implementations we test, which differ according to how the strategies W_z are updated:

1) In the fully competitive approach (*MG*) W_z is represented by $Catch_{n1}^{z}$, that is, the catch of vessel *n1* in zone *Z* during the last *T* steps, discounted by how far in the past the catch was taken.
2) In fully cooperative approach (*TG*) W_z is $TotalCatch_z / N$, that is, the global catch of the entire fleet shared equally by all vessels, discounted by the how far in the past the catch was taken.

3) In the ‘blind’ approach (RAND) in which no information processing is available, W_z is a random set of numbers. This model is useful in our numerical test since it represent a sort of ‘null hypothesis’;
4) In the COIN approach, W_z is given by E_{n1} in equation 5.

In order to appreciate the working of COIN it is important to understand the difference between E_{n1} and $Catch_{n1}$. If vessel *n1* chooses a zone which is under exploited, that is a zone where vessels can fish at their maximum capacity (this is clearly a profitable action to take), E_{n1} in Equation 5 will be large, since vessel *n1* will be able to catch its share without affecting the other vessels’ catch. In this case $E_{n1} = Catch_{n1}$. If vessel *n1* choose a zone which is over exploited, that is a zone where vessels can not fish to their maximum capacity (this is clearly an action to avoid), $E_{n1} < Catch_{n1}$ since all vessels need to share the limited amount of fish available and no extra catch is possible. Had the vessel not gone fishing, its catch $Catch_{n1}$ would have been collected by other the vessels which did not succeed in fishing to maximum capacity. It is possible that several zones could give a high return to vessel *n1*, since more than one zone may be underexploited. In this case several vessels, all choosing different zones, may all receive high return E_n. This spreading of the fishing effort is exactly what we want to achieve. Conversely, a pure competitive scenario (like the one depicted in the *MG*) in which all vessels aim for the potentially most profitable zone, prevents such spreading and induces a ‘Tragedy of the Commons’-like outcome.

Finally, notice that by trying to optimise E_n, *each individual vessel still tries to maximize its own return*, since each vessel looks for underexploited zones. The vessels are still acting *competitively*, but this time their action does not impact negatively on the outcome of the community; the entire community obtains a larger catch, and consequently each vessel (on average) can obtain a larger individual catch.

6. IMPLEMENTING COIN WITH PEN AND PAPER

Conceptually, the core of the COIN approach lies in Equation 4, which implements the idea of evaluating the effect that vessel *n1* has on the fleet by comparing the current catch to that of a hypothetical fleet without *n1*. Computationally, the crucial step lies in mimicking the behaviour of the hypothetical fleet without vessel *n1* by using the Wonderful Life Utility with default factor 0. The fact that this (very rough) approximation works, immensely simplifies the approach and makes the conceptual idea behind COIN actually implementable. In this work, we simplified the COIN algorithm further by imposing some (albeit minor) modifications. In particular:

1) unlike in Wolpert et al. (2000), the weights W_z are not *adjusted*, but rather *reset*, at each iteration according to previous catches;
2) in previous applications of COIN to the Minority Game (Wolpert et al., 2000, Wolpert and Tumer, 2001) the return *R* of each agent for attending a certain zone is

given by $R = y * \exp(-y/c)$, where R is $Catch_n$ and y is $Fish_z$ in our problem. In this equation, c plays the part of a pre-determined ‘ideal’ number of vessels attending each zone. The presence of the parameter c helps to guarantee a balanced allocation of the vessels among the zones. In order to simplify the calculation, as well as to avoid using a parameter which needs to be set a priori, we did not use this option;

3) finally, in Wolpert et al. (2000) and Wolpert and Tumer (2001), in order to account for non stationarity in the predictions, the past catches are discounted by an exponential function while we used a simpler, linear one.

The net result of these modifications is that all that is needed in order to implement a COIN strategy are some basic bookkeeping and a handful of elementary operations (+,-,*,/). While this makes little difference when COIN is run on a computer, it may make a difference if real human agents want to test/use the procedure. Basically, the entire COIN approach could be performed with pen and paper by agents with only primary school training. This may not be relevant to modern fishing fleet in developed countries (which are geared with sophisticated equipment), but it could broaden the application to other resource management problems in the developing world. In particular, here is the ‘pen and paper’ pseudo-algorithm which vessel *n1*’s crew needs to perform:

1) keep a record of the day catch, and of how many other vessels fished in the same zone.
2) Calculate E_{n1} via equation 5 and store it.
3) Retrieve the values E_{n1} from the last T fishing days/periods and call them $E_{n1}^t, t = 1..T$ where t is day/period, with $t=1$ being the most recent day/period and $t=T$ the least recent.
4) Calculate the weights W_z as $W_z = \sum_{t=1..T} \delta_z^t E_{n1}^t \frac{T-t+1}{T}$, (6)

   where $\delta_z^t = 1$ if the vessel fished in zone z at time t in the past, and $\delta_z^t = 0$ otherwise.
5) Normalise the cumulative sum of the weights W_z , $CumW_z = \sum_{i=1}^{z} W_i$ and

   $Norm\_CumW_z = \frac{CumW_z}{CumW_Z}$;
6) Pick a random number r between [0,1].
7) Find the smallest z for which $Norm\_CumW_z \geq r$.
8) Next iteration, fish in zone z.

7. NUMERICAL RESULTS

In this section we compare the performance of the different algorithms we introduced above.

In the first test I aim to replicate the setting of a common Minority Game as described in Section 1. We model a fleet of 20 vessels (*N=20*) and 2 available fishing zones (*Z=2*) equally sharing resource of 50 units of fish ($Fish_{1,2} = 50$). Each vessel can catch at most 5 units (*MaxCatch=5*). Following Equation 3, we can estimate the maximum possible catch for the fleet:

$TotalCatch \leq Min(\sum_{k=1,K} Fish_k, N * MaxCatch) = 100$ units.

A summary of the results can be seen in Table 1. For each algorithm, we show the mean catch for the vessel which caught the largest amount of fish (column 1), the mean catch for the vessel which caught the smallest amount (column 2), the total mean catch of the entire fleet over the entire simulation (column 3) and the mean catch of the entire fleet at the last iteration step (column 4). In order to account for stochastic fluctuations inherent in the algorithms, all results are averaged over 10 different runs.

Table 1. Summary of the comparison of the different algorithms in the test with *N=20, Z=2, MaxCatch=5,* $Fish_{1,2} = 50$. In this and following table Max IC= Maximum Individual Catch, Min IC= Minimum Individual Catch, GMC=Global Mean Catch, FGC=Final Global Catch. All values are averaged over 10 runs.

	Max IC	Min IC	GMC	FGC
COIN	4.97	4.96	99.29	99.50
TG	4.75	4.75	95.07	97.00
MG	4.91	4.72	96.27	97.00
RAND	4.59	4.52	91.07	90.50

We can notice that the COIN outperforms all other algorithms, with an average Global Mean Catch very close to the maximum allowed. The worst performing vessel under COIN performs better that the best performing vessel under any other algorithm. We also notice that the performance on the *TG* and *MG* are quite similar; under this scenario fully competitive and fully collaborative approaches result in similar outcomes. Finally, the performance of the randomly behaving fleet *RAND* is considerably worse than all other approaches, confirming that the information processing inherent in the COIN, MG and TG pays back in terms of catches.

In the second test I aim at a more realistic fishery scenario. I model a fleet of 50 vessels (*N=50*) and 4 available fishing zones (*Z=4*). The resources available in each zone differ: zone 1, 3 and 4 can provide 50 units of fish ($Fish_{z \neq 2} = 50$), while zone 2 can offer 100 ($Fish_2 = 100$). Each vessel can catch at most 5 units (*MaxCatch=5*)., which results in

$TotalCatch \leq Min(\sum_{k=1,K} Fish_k, N * MaxCatch) = 250$ units.

A summary of the results can be seen in Table 2. In this case, in columns 5 and 6 I also show the average number of vessels who fished at zones 1,3 and 4 ($Crowd_{z \neq 2}$) and the average number of vessels which fished at zone 2 ($Crowd_2$).

We can notice the following:

1) COIN provides the best catch for the entire fleet as well as for the vast majority of individual vessels: the worst performing vessel obtains a catch only slightly worse than the best performing vessel for the MG. This suggests that no individual sacrifice is necessary in order to achieve good global catches when the COIN approach is used.
2) COIN performs best because it achieves the best average spread of vessels among the available fishing zones (column 5 and 6). The ratio between the vessels fishing at the different zones matches almost exactly the ratio of available resource (which is unknown to the vessels).
3) The second best catches are provided by the *MG*. However, with this approach the average difference between the catches of the best performing and the worst performing vessels is one order of magnitude larger than in the COIN approach. Fully competitive, selfish behaviour results in lower global catches, lower average catches for the worst performing vessels and larger disparity in performances.
4) The vessels' inability to obtain feedback about their contribution to the global performance in the *TG* results in global catches which are lower than two competitive approaches of the *MG* and COIN. Also the difference in performance between the *MG* and *TG* is now larger than in the first test. This seems to confirm the fact that the team game approach's performance decreases rapidly with increasing number of agents.

Table 2. Summary of the comparison of the different algorithms in the test with *N=50, Z=4*, *MaxCatch=5,* $Fish_{z \neq 2} = 50$ $Fish_2 = 100$ .In this and following table Max IC= maximum Individual catch, Min IC= minimum Individual catch, GMC=Global Mean Catch, FGC=Final Global Catch. All values are averaged over 10 runs.

	Max IC	Min IC	GMC	FGC	$Crowd_{z \neq 2}$	$Crowd_2$
COIN	4.96	4.91	246.84	247.50	9.99	20.04
TG	4.27	4.27	213.49	211.50	12.05	13.86
MG	4.92	4.53	237.45	237.50	10.21	19.38
RAND	4.23	4.07	207.39	199.50	12.52	12.45

It is also worthwhile to analyze the time series of the Global Mean Catch (GMC) produced by the different algorithms (Figure 2):

1) The *RAND* curve (thin dashed line) oscillates around a fixed exploitation baseline as expected from a random behaviour.
2) The *TG* (thick dashed line) shows oscillations of lower magnitude and consequently better performance than *RAND.* At the beginning of the simulation the performance of *RAND* and *TG* are quite similar. After some 100 steps the *TG* performance improves to reach a sort of rough plateau characterised by both short range and long range departures, suggesting that continuous learning is not possible fir the *TG* because of the agents' poor information processing.
3) The *MG* (thin line) also reaches maximum performance within some 100 steps with oscillations of generally decreasing amplitude. This shows a reasonably good level of adaptation to the modelled conditions.

4) The COIN's performance is the worst during the very first iterations, when little information is provided for training, but within a few time steps it overtakes all other algorithms. It then keeps on improving, converging towards the maximum allowed catch (250) with small amplitude oscillations.

In the third test I examine how the algorithms adapt to change in resource distribution. I use the same fishery scenario as in the first test (*N=50*, *Z=4*, *MaxCatch=5*). At the beginning of the run I set $Fish_{z\neq 2}=50$ $Fish_2=100$. However, after 220 time steps, I impose $Fish_2=50$ and $Fish_3=100$, while zones 1 and 4 are unaltered. Basically, the larger stock moves from zone 2 to zone 3. The time series of the Global Mean Catch can be seen in Figure 3. Obviously, the behaviour of the curves in the first 220 steps is similar to the one described in Figure 2. Also, the behaviour of *RAND* is unaffected by the stock migration because of its totally random nature. However, soon after time step 220 the catches of COIN, *MG* and *TG* decrease suddenly since their fleets had 'trained' themselves to exploit more heavily zone 2 and its drop in stock affects their catch negatively. All three approaches are able to adapt to the new resource distribution and adjust their fleet allocation to exploit it, but how they do so varies considerably. The *TG* drop at iteration 220 is smaller than for *MG* and COIN since it had adapted itself less to the previous stock distribution, and recovers in roughly 80 time steps. The *MG* drop is much sharper since it had adapted itself better to the previous stock distribution. However, its recovery is very slow and by the end of the run it has not recovered its pre-migration catch level. The COIN drop is also very sharp as expected, since COIN also had adapted to stock distribution. However, the COIN fleet is much faster in retraining to the new scenario and its GMC recovers in less than 20 iterations. This different adaptability to changing conditions explains the final performance displayed in Table 3.

Table 3. Summary of the comparison of the different algorithms in the test with *N=50*, *Z=4*, *MaxCatch=5.* At the beginning of the run we have $Fish_{z\neq 2}=50$ and $Fish_2=100$. After 220 steps the fish stock distribution changes to $Fish_{z\neq 3}=50$ $Fish_3=100$. All values are averaged over 10 runs.

	Max IC	Min IC	GMC	FGC
COIN	4.95	4.87	245.61	249.00
TG	4.19	4.19	209.75	209.50
MG	4.91	4.36	232.20	234.50
RAND	4.23	4.07	207.63	207.50

Finally, I test the algorithm under more challenging conditions. In the fourth test I model a larger fleet (*N=200)*, a larger fishery (*Z=16)* and a random stock distribution. Also, every 100 time steps, the stock at 4 random locations migrates to 4 other random locations. The results can be seen in Figure 4. The larger fleet results in the *TG* performance to further worsen and almost match the fully random fleet (*RAND*). The continuous change in stock distribution does not allow the MG to ever reach maximum fishing capacity, while the speed in recovery of COIN allows its fleet to recover optimal catches in between each migration event. The exact numerical outcome of this simulation is reported in Table 4.

Table 4. Summary of the comparison of the different algorithms in the test with *N=200*, *Z=16*, *MaxCatch=5*, random stock distribution and random migration every 100 time steps. All values are averaged over 10 runs.

	Max IC	Min IC	GMC	FGC
COIN	3.82	3.38	738.54	647.77
TG	2.99	2.99	598.93	598.12
MG	3.79	3.12	703.55	631.75
RAND	3.12	2.81	592.75	590.24

8. DISCUSSION, LIMITATIONS, AND DIRECTION FOR FURTHER WORK

The results in the previous section are very encouraging. Nevertheless, before more expectations can be placed on this technique more detailed modelling needs to be done. Here we present a "high-priority" list of directions for further work, some of which I endeavour to address in my future research:

1) the resource dynamics needs to be included into the simulations. How the fish population responds to the COIN exploitation, how the COIN can adapt to the population dynamics itself and how the management policies need to adapt to variable circumstances, are all factors which need to be carefully explored.
2) Human behaviour also needs to be modelled more accurately. For example, we should explore what would happen if one or more vessels in the COIN behaved greedily (by following standard MG strategies), or communication between nearby vessels was disrupted (either purposely or accidentally). Basically, we should model a fleet consisting of vessels of mixed strategies, some adopting a COIN approach, some a purely competitive one, some fully cooperating within a smaller group. Vessels may also decide to change strategy during the run, depending on their returns. It is important to evaluate COIN's robustness and within what limits it can still prove viable.
3) In the current implementation we are not modelling the availability of public information or 'a priori' expectations of fish population behaviour. It is worthwhile to study how such information could alter the effect of COIN. This could be done by casting the calculation of the W_z within a Bayesian framework, thereby catches modify prior information or the new information which may become available during the run from external sources is accounted for.
4) The potential receptivity of real communities to a COIN approach needs to be assessed. To our knowledge, no experience is reported in the literature on this subject. Planning to initially test this within a game theoretical setting in a lab with a number of volunteers is currently under way.
5) It is important to stress, once again, that the strength of COIN lies in disentangling impact from contribution and in recognising and *penalising contributions which do not lead to impact*. This need not be limited to the agents' behaviour, but could be

extended to policies in a straightforward manner. This would lead to discriminating the impact from the mere contribution of management decisions like fishery closure, whether complete of partial, its timing, the setting of quotas, the selection of allowed gear, the establishment of penalties for infringements.

6) Discriminating between contribution and impact is a first step towards the *very* difficult task on detecting causal relations within a Complex Adaptive System. Causality is a famously slippery concept in the philosophy of science. Pearl (2000) and Pattee (1997) propose an operative distinction between causality and mere correlation; causality, unlike correlation, involves control, that is, the possibility to affect a process by acting on the causal component. According to this definition, contribution which leads to impact has a direct causal effect on the global outcome of the community. A contribution which leads to no immediate impact might have a more subtle causal effect, which may or may not manifest itself in the longer run. COIN thus can provide a tool, albeit an approximate one, to detect the components with potentially highest causal power and act upon them.

We conclude by discussing by far the most important concern: in this work we assumed that the resource is fully renewable. In many resource management scenarios, maximizing the exploitation of a limited resource is *not* what decision makers want, especially in environments under considerable ecological pressure. In a certain sense, this issue is not directly related to whether COIN is a valuable approach, unless a less effective fishing strategy is seen as a management tool for stocks control. What is important is that COIN is an optimization technique. This, as mentioned before, does not necessarily imply maximizing exploitation, rather *finding a best strategy for a goal of our choice*. From this perspective, the challenge is to formulate an ecologically responsible and economically valuable goal and see whether COIN can help achieving it by acting on all components which may influence on it, agents and managers alike. Technically, this means finding a WLU 'cost function' which would result, *at the same time*, in a sustainable behaviour for the entire fleet and an individual 'selfish' goal which is worthwhile for each vessel to pursue. The results I presented suggest that such combined aims are not necessarily contradictory, rather they may be implemented in a way which results in their being *aligned* and COIN can offer a valuable approach to finding such alignment. This is surely the single most important direction for future work I aim to address.

9. CONCLUSIONS

We propose a simplified version of the Collective Intelligence which can be easily employed by a community of human agents in order to plan the exploitation of a limited resource. We compared COIN against other game theoretical approaches on a number of virtual fishery scenarios. In all tests the COIN not only guaranteed optimal global catch but also maximised the catch of each individual vessel. Achieving this in a competitive environment may be a key factor in this method's acceptance by real communities. In the view of actual implementations, we described a pseudo algorithm, which allows the COIN to be carried out by 'pen and paper', with minimum bookkeeping and only elementary calculations.

10. REFERENCES

Arthur, W.B., 1994. Inductive behaviour and bounded rationality. The American Economic Review 84, 406-411.

Batten, D., 2006, Are some human ecosystems self-defeating?, Environmental Modelling and Software, (in press)

Batten D., and Boschetti F., 2006, Self-defeating human ecosystems, in Understanding Human Behaviour In Complex Adaptive Systems: Building a foundation for promoting adaptive governance, (in press).

Bedau, M.A., 1997, Weak emergence. In Philosophical Perspectives, vol. 11, edited by J. Tomberlin, Blackwell, Oxford, pp. 375-399.

F. Boschetti, M. Prokopenko, I. Macreadie, and A.-M. Grisogono. Defining and detecting emergence in complex networks. In R. Khosla, R. J. Howlett, and L. C. Jain, editors, Knowledge-Based Intelligent Information and Engineering Systems, 9th International Conference, KES 2005, Melbourne, Australia, September 14-16, 2005, Proceedings, Part IV, volume 3684 of Lecture Notes in Computer Science, pages 573-580, 2005.

Hardin, G., 1968, The tragedy of the commons, Science 162, 1243-8.

Kirkpatrick, S., C. D. Gelatt Jr., M. P. Vecchi, Optimization by Simulated Annealing, Science, 220, 4598, 671-680, 1983.

Pattee, H., 1997, Causation, Control, and the Evolution of Complexity, Downward Causation, P.B. Anderson, P.V Christiansen, C. Emmeche, and N.O. Finnemann.

Pearl, J., 2000, Causality: models, reasoning and inference, Cambridge, Mass., MIT Press, 2000, 384 pages.

Savit, R., Manuca, R., and Riolo, R., 1999, Adaptive Competition, Market Efficiency, and Phase Transitions, Physical Review Letters, 82(10):2203-2206, 1999.

Wolpert, D. and Tumer, K., 1999, An Introduction To Collective Intelligence, Tech Report: NASA-ARC-IC-99-63, http://ic.arc.nasa.gov/ic/people/kagan/coin pubs.html

Wolpert, D., K. Wheeler, and K. Tumer, 2000, Collective Intelligence for Control of Distributed Dynamical Systems. Europhys. Lett., 49 (6), p. 708.

Wolpert, D. and Tumer, K., 2001, Optimal Payoff Functions for Members of Collectives. Advances in Complex Systems, 4(2/3):265–279, 2001.

Wolpert, D., Tumer, K, and Bandari, D., 2004, Improving Search Algorithms by Using Intelligent Coordinates. Physical Review E, 69, 2004.

Zhang, Y., 1998, Modeling Market Mechanism with Evolutionary Games, Europhys. News, 29, 51.

Figures

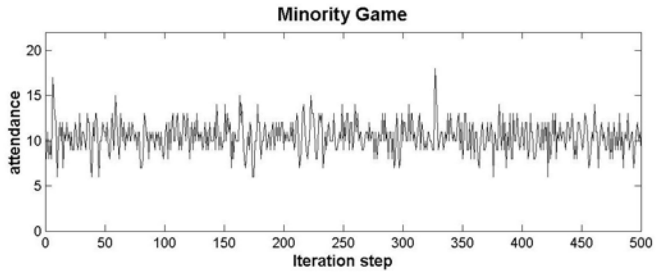

Figure 1. Typical run of a standard Minority Game, with N=20 agents and resources spread equally among two locations. The X axis shows the iteration number and the Y axis shows the attendance (number of agents) at location A. The attendance fluctuates in an irregular fashion around N/2, which represents the average attendance at each location in the long run, despite the curve never converges to this value.

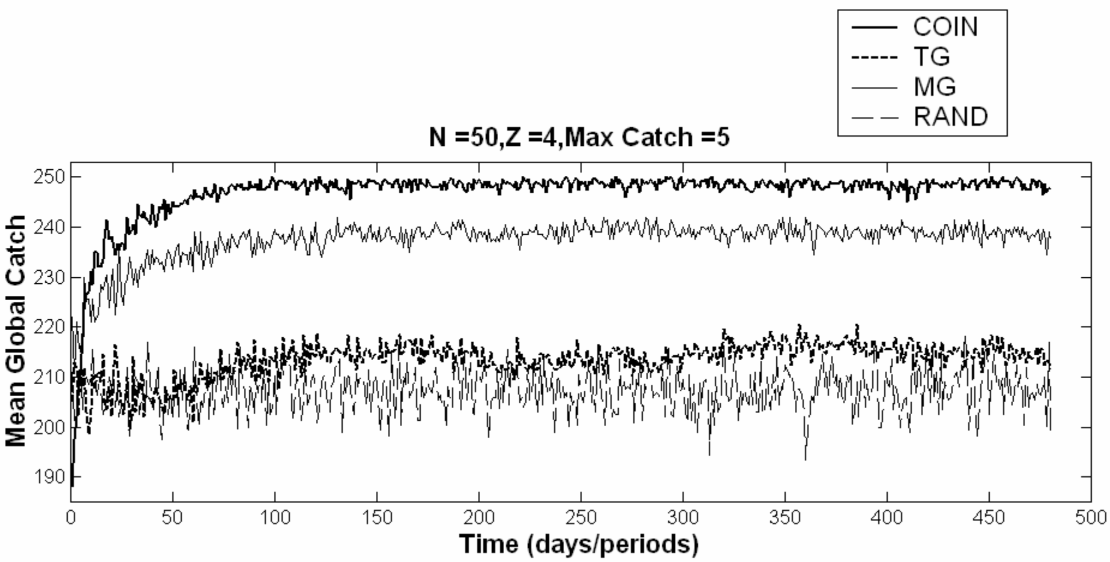

Figure 2. Mean Global Catch for the different algorithms in the first test case, N=50, Z=4, MaxCatch=5,   and . The curves are averaged over 100 runs.

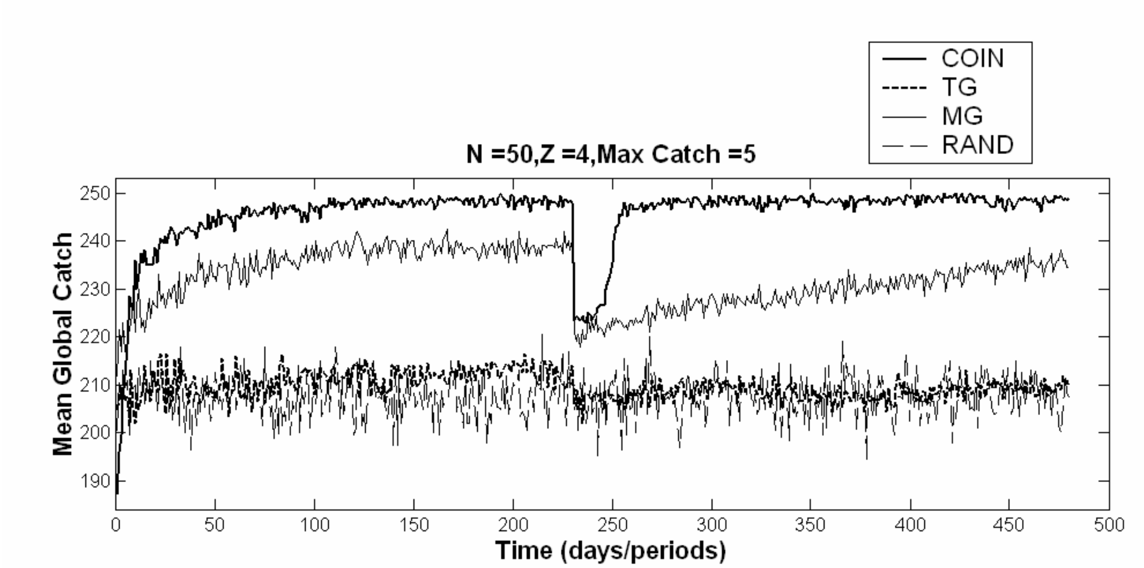

Figure 3. Mean Global Catch for the different algorithms in the second test case, N=50, Z=4, MaxCatch=5. At the beginning of the run we have $Fish_{z \neq 2} = 50$ and $Fish_2 = 100$. After 220 steps the fish stock distribution changes to $Fish_{z \neq 3} = 50$ $Fish_3 = 100$. All values are averaged over 10 runs.

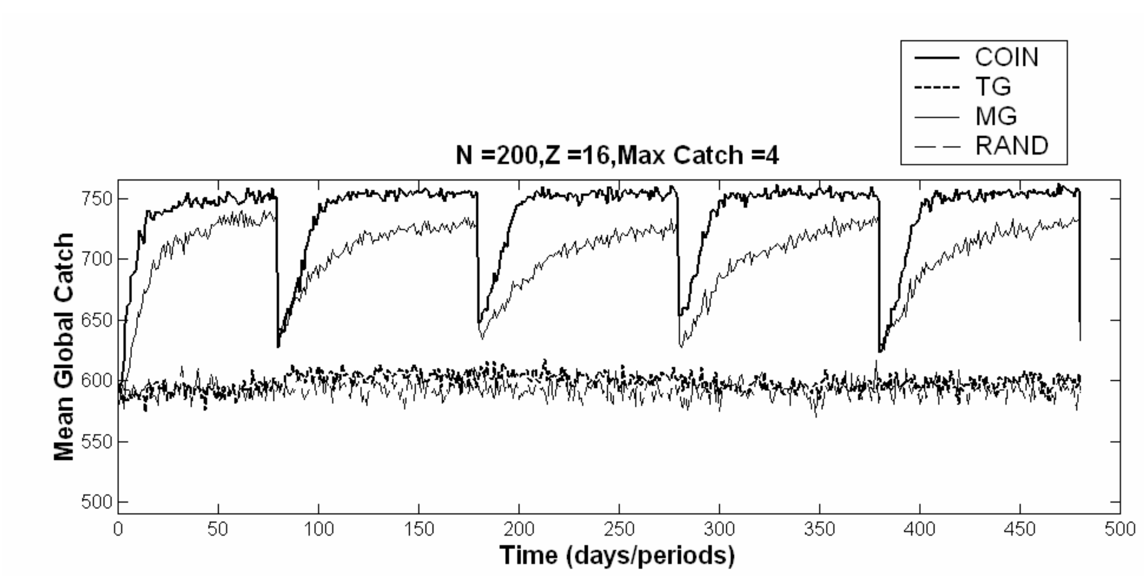

Figure 4. Mean Global Catch for the different algorithms in the final test case, *N=200, Z=15, MaxCatch=5,* random stock distribution and random migration every 100 time steps. The curves are averaged over 10 runs.