Review Form

Number:	
Paper Title:	

Should this be considered for a "Best Paper" award?

( ) Yes

(X) No

I. OVERALL EVALUATION (Comments to Author below)

Contribution	Exceptional	Significant	Minor	Questionable	None
New Results	( )	(X)	( )	( )	( )
Significant Applications	( )	( )	( )	( )	(X)
Survey or Tutorial	( )	(X)	( )	( )	( )

II. OVERALL RECOMMENDATION:

( ) ACCEPT: Article is acceptable for publication in its present form

(X) REVISE AND ACCEPT: Required changes are minor and may be made by author without further review.

( ) REVISE AND RESUBMIT: Article is acceptable for publication provided that significant changes are made as indicated in the reviewer's comments. Revision will be carefully checked by the editor or associate editor to ascertain whether the changes are acceptable.

REJECT:

( ) Article is so poorly written that contributions, if any, are obscure.

( ) Article is not appropriate for the IEEE Transactions on Evolutionary Computation.

( ) The results are not correct.

( ) Article is not suitable as a tutorial for reasons given below.

( ) Other

III. CONFIDENTIAL COMMENTS TO EDITORS

This is a high quality paper, but the authors seem a little unfamiliar with the paradigm. I have suggested some things they need to look at again and clarify. In general though I believe it is the kind of paper you would want to publish.

IV. COMMENTS TO AUTHOR(S)

First of all: the title. The word "Engineered" is not so unique, or even so appropriate, that it needs to be italicized. The reader, given this title, expects to find out that the authors have a particular view or definition of engineering, which they are going to apply to PSO in order to produce something unique. I would recommend the word "eclectic" as a substitute, or "patchwork," or maybe something sardonic like "grave-robbing." At least, please do not italicize the word in the title.

The first sentence of the introduction misrepresents the particle swarm algorithm a little bit. It is not necessary to compute velocity and then update the position, in fact, it can be done without velocity at all:

x_id(t+1) = x_id(t) + w(x_id(t)-x_id(t-1)) + differences etc.

Of course it is usually done in two steps, and there is nothing wrong with saying it this way, but the two rules do not really define the algorithm.

The descriptions of the algorithms are for the most part all right, and the selection of variants featured is good enough, though that is a matter of chance and taste, as will be noted below.

I do think it would be better not to call the constricted version "canonical." It is widely used, but the inertia-weight version may be more so. Call it "constricted." The two versions are identical if you shuffle the algebra around a little; the difference between Clerc's version and the inertia weight version you use is simply the decreasing coefficient.

There is an important fact that affects your experimental design and interpretation of results, and that has to do with the way the topology works in FIPS. A best-neighbor particle is influenced by itself and one neighbor per iteration. Even if the topology is defined as gbest, with full connectivity, any target particle i is only affected by i and g, the population best particle. In FIPS, however, the particle is affected by all its neighbors. So, for instance, a gbest topology means that i is influenced by all the particles in the population, all at once. No wonder then that the result is essentially random behavior. Many statements in this paper attempt to generalize about smaller and larger neighborhoods, but these generalizations do not apply in the same way to FIPS and best-neighbor swarms.

In Section III you mention that "it is known that some PSO variants tend to search near" the origin, and cite Seppi and Monson's paper. A careful reading of that paper finds no such tendency in any particle swarm. In fact, the paper doesn't really test any ordinary particle swarm, and further the conclusions are based on subjective inspection of some graphs, which in most cases don't show any visible origin-seeking bias. The two versions those researchers tested that do resemble particle swarms,e.g., the TRIBES version and a kind of Gaussian model, both showed no origin-seeking bias. The rumor of this bias, encoded in the title of their paper, should not be perpetuated in these Transactions.

Is it important that "large population sizes provide higher initial diversity"? In a case of, say, a 30-dimensional function, there are 2^30 or 1,073,741,824 corners to the search-space hypercube. The statements about the diversity of the initial population seem to imply that somehow a population of 60 particles significantly blankets the search space better than one of 20; it doesn't. You might try comparing function results at iteration-zero for different population sizes to see if this is correct -- I don't think you will find significant differences. The different results for different population sizes probably come from something else, having to do with particle interactions during the search; for instance, in a distributed topology like a ring lattice, problem solutions will take longer to propagate through the entire population and so there is more parallel searching going on.

In fact, I don't think you say anywhere how you initialize the population. Is it random? It is common in particle swarm research to initialize the particles in a region of the search space that does not contain the global optimum -- did you do that?

The description of RTDs is insufficient. It's good marketing to encourage the reader to buy your book, but I think the editors of this journal are better served if their readers get all the information they need from the article itself. You can still cite yourself, for more information.

The reference to convergence is similar. Not everyone has Engelbrecht's book handy. In PSO there are two phenomena, which are sometimes referred to as clustering and convergence. The first of these refers to the distance between the particles in the search space, and the second describes the size of their steps through the space. The fact that these two measures are correlated is very important to understanding the functioning of the swarm. As they achieve consensus on where to search for optima, they take smaller steps; thus the functionality of the strategy parameter vector from ES is inherent to the structure of the particle swarm.

The concept of "restarts" is woven through this paper, but never really explained. What do you restart? Do you reinitialize the positions of the whole swarm, keeping their previous bests? Do you throw out the previous bests? Initialize one at a time with some probability? There is also an unjustified assumption, stated in section III, that "it is possible to improve an algorithm's performance by using occasional restarts..." First, I doubt that "an algorithm" is specific enough -- I'm sure there are some algorithms that do not benefit from restarts (a random search algorithm, for instance). Second, there is no strong tradition in the literature supporting such a statement. Of course people have used restarts of various kinds, but I suspect there is a strong interaction of the effectiveness of this technique with features of the problem, and probably features of the algorithm itself.

A formula is given for "effort." Is this used anywhere in the paper?

Equation 9, first hitting time, raises a question: what if the trial doesn't hit? How do you manage failed trials in your analysis? Hopefully you use medians rather than means, for instance, and nonparametric statistics.

The statement in Section III that "all PSO variants have a strong stagnation tendency" will come back to haunt you. The point of Clerc's coefficients is to induce convergence in the particle's trajectory; by adjusting those coefficients you could easily produce a PSO that doesn't stagnate. It may also not solve problems, but it wouldn't stagnate. A PSO with no coefficients at all and no VMAX will not stagnate, for instance. Really, the holy grail with the particle swarm is to adjust the dynamics of the system so that it converges at the time you want it to, not too fast, not too slowly. But you do want it to converge, and correspondingly you want the population to cluster in one region, which you are calling (with, I think, unnecessary negativity) "stagnation."

Statements as in Result 1 need to explain clearly that they refer to one test function. Also in Result 1, a generalization about "low connected topologies:" you might look more closely at Mendes' thesis results (which you cite), where he finds mean degree k=4 to 4.25 to be optimal for FIPS populations, as I recall. Again, you won't be able to make statements about particle degree that apply equally to FIPS and best-neighbor populations. Be careful.

Tables were screwed up in this manuscript. Some appear at the end, out of order.

There is a sentence that says: "This is in contradiction with the assumptions of our experimental setting." What does that mean?

I think there is something wrong, or at least missing or hidden, in your interpretation of the descending-inertia-weight effect. Let's say that when the inertia weight is very high, the search is chaotic. When it is in a reasonable range, say around 0.7, search is dynamically interesting and reasonable, and the particle oscillates with enough variance to escape local optima. And when it is lower, the ability to escape local optima is impaired but the particle is able to descend a local gradient. At least this is how I have always understood this time-dependent variant.

Now, what this means is that your search happens in three stages, roughly, as the weight value descends. First phase is chaotic search, large jumps around the search space; second is good robust search with some leaps and some hill-climbing, and last is strictly local search. When you change the pace of decreasing the inertia coefficient, you are changing the amount of time that the particle spends in each stage or phase. It is no surprise that some global searching goes on in the chaotic phase, as the particle is leaping all over the place, so that when it enters a more reasonable condition (2nd phase) it is in a better starting-place. This seems to explain why your graphs look the way they do: performance seems stagnant during the chaotic phase, then improves suddenly in the reasonable phase (last phase the same in all). Discussion of the steepness of the change in inertia weight seems like a distraction from understanding the real cause of these differences in behavior and performance. The word "greedy" is a distraction, too, and does not appear to be justified. The first hitting time reflects the phase when the algorithm is behaving reasonably, e.g., when the coefficient is around 0.7, and of course it gets to that phase sooner when the coefficient is being reduced faster. Is that really greediness? At least you might want to explain your usage of that word a little.

In fact, since, as seen in Figure 6, all of your versions "stagnate" when the coefficient gets small -- why don't you try only dropping the inertia weight to 0.6 or 0.7?

Funny, Frankenstein's version introduces adaptive topology for the first time: so this is not an organ taken from a graveyard, but part of the monster's own corpse. Sorry about that gruesome metaphor.

Now -- it appears that you are using a FIPS interaction method (according to Algorithm 1), and so the fully-connected topology induces random particle behavior. Thus the statement that "a highly connected topology during the first iterations give an algorithm a fast convergent behavior ..." is incorrect. In FIPS, a highly connected topology drives the particle crazy, trying to "sit between two chairs," as my old professor used to say. In a best-neighbor version, yes, gbest leads to fast convergence; in FIPS, no.

There is also a strange statement in Section V.A: "For short runs, FIPS's best performance is obtained with the fully connected topology..." I don't think FIPS ever does well with a fully connected topology. Do you have data showing that? Please indicate where it is.

V.B's second sentence is not a sentence: "In particular, on the effects of the topology adaptation and the inertia weight schedules."

Also in that section, what does "extreme configurations" mean?

Later, same section, I don't think "fasten" is an English word as used here. Maybe "accelerate" or "speed up" is what you mean -- fasten is attaching something to something.

Last paragraph of VI, I recommend you be careful with your conclusions. You say "... if properly parameterized, is faster and more reliable than ..." It will take a lot more than these few functions to support that statement. You have shown good potential here, but Frankenstein's version will not, I predict, take the world by storm overnight.

Similarly, in the second paragraph of VII, you say you have compared it with "the most influential and some of the most promising PSO variants." I won't complain about your choice of versions, but I'll bet Thiemo Krink or Andries Engelbrecht or P. N. Suganthan or even Maurice Clerc would, and some other researchers who feel they have made important breakthroughs in the paradigm which are ignored by your selection of models. There is a lot of ground to cover out there, and this is just a beginning, so be careful about overstating your findings.

You say you find "high quality solutions using fewer function evaluations and ... higher frequency ..." Remember, the cynical reader will be aware that every grad student's paper reports that its innovation is better than the competition as the author himself has coded it. Sometimes the claim is supported by further research, and sometimes it's not. So, again, be careful.

The last paragraph's mention of restarts seems out of place. It's easy enough to do -- why didn't you do it? My guess is that it won't help, your guess is that it would, maybe it would be better not to mention it.

As a methodological approach, the manipulation of this many independent variables makes the causes of performance differences impossible to identify. Your topology adaption is counterintuitive -- does it help, or would you have better results with something else? There is no way to know, because so many things vary.

And as a final comment, it does not appear that the Frankenstein algorithm is ever really described in the paper. Algorithm 1 gives it, but the narrative never says until the end, for instance, that you are using a FIPS type of neighborhood averaging.

All in all, I am glad to see this kind of paper in the literature. There are now many variations on the original core algorithm, and it makes sense to start putting the pieces together in different ways. Some of these experiments will result in improvement of the paradigm and perhaps will lead to new insights about how the different levels of functioning work together. With a little more attention, this paper will be publishable.

Files for Associate Editor Confidential files for Associate Editor ONLY	Files for Author Files viewable by Associate Editor and sent to Author with final decision
files	files

Review Form

Number:	
Paper Title:	

Should this be considered for a "Best Paper" award?

( ) Yes

(X) No

I. OVERALL EVALUATION (Comments to Author below)

Contribution	Exceptional	Significant	Minor	Questionable	None
New Results	( )	( )	(X)	( )	( )
Significant Applications	( )	( )	( )	( )	( )
Survey or Tutorial	( )	( )	( )	( )	( )

II. OVERALL RECOMMENDATION:

( ) **ACCEPT**: Article is acceptable for publication in its present form

(X) **REVISE AND ACCEPT**: Required changes are minor and may be made by author without further review.

( ) **REVISE AND RESUBMIT**: Article is acceptable for publication provided that significant changes are made as indicated in the reviewer's comments. Revision will be carefully checked by the editor or associate editor to ascertain whether the changes are acceptable.

REJECT:

( ) Article is so poorly written that contributions, if any, are obscure.

( ) Article is not appropriate for the IEEE Transactions on Evolutionary Computation.

( ) The results are not correct.

( ) Article is not suitable as a tutorial for reasons given below.

( ) Other

III. CONFIDENTIAL COMMENTS TO EDITORS

IV. COMMENTS TO AUTHOR(S)

The first part (experimental comparison) is technically very good. All the "results/rules" about the behaviour of different PSO versions are already known, but here they are more rigorously supported than in most of previous works.

In particular, the influence of the neighbourhood topology is well examined.
However, strangely, the authors do not consider an important class of topologies: the stochastic ones.

The second part is less convincing.
The main weakness is that it works well only if the user have a good idea of the number of iterations to perform to find an acceptable solution.
For real problems, it is usually not the case.
Also the way the "best configuration" is chosen for each PSO is quite questionable.

On the whole, it means that the paper is more or less acceptable, but also that it could easily be more complete and better.

----
p. 6
"We considered three population topologies: fully connected, square, and ring"

What about random ones? See for example

Miranda, V. & Oo, N. W. New Experiments with EPSO - Evolutionary Particle Swarm Optimization IEEE Symposium on Swarm Optimization, 2006

Standard PSO 2006 (Particle Swarm Central http://www.particleswarm.info, Programs)

Mohais, A. S., Mendes, R., Ward, C. & Posthoff, C. Neighborhood re-structuring in particle swarm optimization, Lecture Notes in Computer Science, 2005, 3809 LNAI, 776-785

p. 15
"PSO is composed of three main algorithmic components,"

Note that this is not the first attempt. See for example the adaptive parameter-less PSO called TRIBES, which adapts the swarm size, the information network topology and the strategy to use for the next move.

Onwubolu, G. C. TRIBES application to the flow shop scheduling problem New Optimization Techniques in Engineering, Springer, 2004, 517-536

Clerc, M. Particle Swarm Optimization ISTE (International Scientific and Technical Encyclopedia), 2006

More important, by combining these three components, you may also combine their weaknesses. The main one is that weight evolution is always the same, only depending on the number of iterations (i.e. not even depending on the number of fitness evaluations nor on the function to minimise).
And the same for the topology evolution.

p. 15
"We schedule the adaptation of the topology so that in k iterations ..."

How do you know k in advance for a real problem? This is an important parameter, for you use it both to modify the topology and the inertia weight.

And this can certainly not be called "adaptation", for it is not depending on any information found during the iterative process.

Note that adapting the topology is an idea that can also be found in "niching" PSOs. In these PSOs the topology of the information network is automatically modified during the process, for it is depending on the distances between particles.

p. 15
"If we refer to Table VI, we see that the only velocity- and position-update mechanism that is ranked among the best and that uses different topologies is FIPS"

"The only modification to this component is that Clerc and Kennedy's constriction factor is not used. We employ a decreasing inertia weight..."

To be consistent, you should show FIPS+decreasing_weight is still "ranked among the best".

p. 19 Performance validation

According to Table X, the choice of decreasing weight for FPSO is quite strange, for Increasing-IW is better than Decreasing-IW (on the whole, over the four cases described in the table)

Similarly, the choice of FIPS as a component of FPSO is also not justified: Canonical

PSO is better than FIPS (mean ranking 17/4 versus 19/4).

One could argue that, anyway, FPSO seems globally better than the others. However, it is written
"For each PSO algorithm we consider all its configurations resulting from our experimental setup"
but "all its configurations" means "just three swarm sizes (20, 40, 60)", and, unfortunately, for a given problem and a given search effort, the optimum swarm size is rarely one of these values.
As a result, for each PSO, the chosen configuration is almost certainly _not_ the best one.

Note that, anyway, it is quite strange to choose a configuration that is depending on the termination criterion.

Files for Associate Editor **Confidential files for Associate Editor ONLY**	Files for Author **Files viewable by Associate Editor and sent to Author with final decision**
files	**files**

Review Form

Number:	
Paper Title:	

Should this be considered for a "Best Paper" award?

( ) Yes

(X) No

I. OVERALL EVALUATION (Comments to Author below)

Contribution	Exceptional	Significant	Minor	Questionable	None
New Results	( )	( )	(X)	( )	( )
Significant Applications	( )	( )	( )	( )	(X)
Survey or Tutorial	( )	( )	(X)	( )	( )

II. OVERALL RECOMMENDATION:

( ) **ACCEPT**: Article is acceptable for publication in its present form

( ) **REVISE AND ACCEPT**: Required changes are minor and may be made by author without further review.

(X) **REVISE AND RESUBMIT**: Article is acceptable for publication provided that significant changes are made as indicated in the reviewer's comments. Revision will be carefully checked by the editor or associate editor to ascertain whether the changes are acceptable.

REJECT:

( ) Article is so poorly written that contributions, if any, are obscure.

( ) Article is not appropriate for the IEEE Transactions on Evolutionary Computation.

( ) The results are not correct.

( ) Article is not suitable as a tutorial for reasons given below.

( ) Other

III. CONFIDENTIAL COMMENTS TO EDITORS

The authors combined three components from three PSO variants together through experiments, and therefore introduce a new PSO variants. Even though the performace of the resulted PSO is better from their experiments but still very similar to that of the original PSOs, because the new PSO does not introduce any new mechanism to improve its performace essentially.

IV. COMMENTS TO AUTHOR(S)

1) This paper introduced a new PSO variant through combining three components from three different PSO variants through experimental comparison. The new PSO variant provides better performance from the provided simulation results on several benchmark functions. Essentially, the three componnets, especially the adaptive topology and decreasing inertia weight, play the role to have more exploration capability at the beginning while to have more exploitation capability at the end of each run, therefore, it is intuitive to have the performance for the proposed new PSO variant.

2) In the paper, the authors claimed that they provide an answer to the question "is it possible to combine different algorithmic components that seem to provide good performance into a single PSO variant ...". From the reviewer's point of view, it will be much more convincing, before claiming an answer to the question, if PSO variants with other possible combinations are tested and compared.

3) Usually, the search process is nonlinear, and therefore a search process with monotonous transition from exploration to exploitation does not reflect the nature of the search process. Therefore a dynamic adjustment of the transition will be prefered, if possible. Fixed way of adjustment of both topology and inertia weight does not guarantee a better PSO. It should be problem-dependent.

4) For adopted FIPS algorithm component, should the current position of a particle provide some contribution to its velocity's change?

Files for Associate Editor **Confidential files for Associate Editor ONLY**	Files for Author **Files viewable by Associate Editor and sent to Author with final decision**
files	**files**

Review Form

Number:	
Paper Title:	

Should this be considered for a "Best Paper" award?

(X) Yes

( ) No

I. OVERALL EVALUATION (Comments to Author below)

Contribution	Exceptional	Significant	Minor	Questionable	None
New Results	( )	(X)	( )	( )	( )
Significant Applications	( )	(X)	( )	( )	( )
Survey or Tutorial	( )	(X)	( )	( )	( )

II. OVERALL RECOMMENDATION:

( ) **ACCEPT**: Article is acceptable for publication in its present form

(X) **REVISE AND ACCEPT**: Required changes are minor and may be made by author without further review.

( ) **REVISE AND RESUBMIT**: Article is acceptable for publication provided that significant changes are made as indicated in the reviewer's comments. Revision will be carefully checked by the editor or associate editor to ascertain whether the changes are acceptable.

REJECT:

( ) Article is so poorly written that contributions, if any, are obscure.

( ) Article is not appropriate for the IEEE Transactions on Evolutionary Computation.

( ) The results are not correct.

( ) Article is not suitable as a tutorial for reasons given below.

( ) Other

III. CONFIDENTIAL COMMENTS TO EDITORS

A good study and a comparison of a number of PSO algorithms. Some interesting conclusions and a good combination of certain characteristics of those algorithms to build a good optimizer.

IV. COMMENTS TO AUTHOR(S)

There are some very interesting aspects that I hadn't though about in this research. I had the idea that FIPS was simply bad when using dense topologies. I had no idea that it had a steep descent before stagnating. That was very interesting.

I like the idea of RLDs instead of using success probability.

I think the part about the restarts should be explained in more detail. It is not easy to replicate with the explanations given.

I don't like your naming the topology adaptive. In my book, adaptive means that the topology evolves in a way connected to its performance. What happens is that the topology gradually transforms itself from a densely connected topology into a sparsely connected one. But the proceeding is simply random.

Files for Associate Editor **Confidential files for Associate Editor ONLY**	Files for Author **Files viewable by Associate Editor and sent to Author with final decision**
files	**files**