I have several important concerns over this paper:

1) The conceptual framework is inappropriate since FLOCK is not an MCMC algorithm
2) There is only one dataset which is way too little to come to conclusions
3) There are basic flaws in the interpretation of the outputs of FLOCK , leading to invalid comparisons relative to K values other than the estimate
4) Comparisons of estimates of K cannot be performed with sufficient rigor
5) The speed comparisons are based on an ill-chosen processing unit

I now describe each of those concerns in more detail

1) Conceptual framework

It must be emphasized that FLOCK is not an MCMC based algorithm. It has no target function, no transition probabilities and does not move in small measured probabilistic steps. No amount of mathematical analysis will ever change this. In fact, similar iterative methods have been around for a very long time (e.g. Newton's method for finding roots), much before the first MCMC algorithms were designed (1950's). This family of methods should be analysed within a truly pertinent conceptual framework (attractors, basins of attraction, fixed points, orbits, etc). The book *A First Course in Discrete Dynamical Systems (RA Holmgren, 1994)* provides an excellent introduction to this topic. The current popularity of MCMC methods in the field of genetic analysis does not justify trying to cast every current algorithm into an MCMC framework.
For example:
Line 156: it is suggested that Flock can be trapped in a local mode, this is incorrect because Flock does NOT search for maximum values, either local or else
Line 254: it is suggested that Flock is searching over all possible partitions. Flock does not sample nor search the surface or space of all partitions.
Line 234: 'individuals would alternate between reference groups': this, precisely, is an orbit

2) Data

Performance comparisons between programs based on a single dataset are of little value. The fact that it is large and real does not compensate. For this purpose, one would expect that a large set of simulations spanning several parameter values with or without several real datasets would be used instead of a single empirical dataset. The advantage of simulations is, of course, that the true genetic structure of the data (essentially K) is known and so the relative performances may be compared on a solid basis. Here, unless I missed something, there is no independent (of the two programs) source of knowing with a reasonable degree of certainty the actual number of clusters represented within this collection of genotypes. In fact, there appears to be several candidate values for K with this particular dataset (see below).
Several (over)generalizations are based on this unique dataset:
(p.9) *FLOCK seems to find the same solution less consistently than does STRUCTURE*
(p.14) The problem of getting caught in local modes is not unique to FLOCK...; however our results suggest it is a bigger problem for FLOCK
(p.16) Given FLOCK'S tendency to converge to different solutions with large K and n...

(p.16) It is probably better to use STRUCTURE in such cases...

3) Interpretation of FLOCK outputs (purpose of FLOCK, best run, stochasticity of plateau lengths)

Purpose of FLOCK

I agree with the following statement (p. 11):

While we take the stance that estimates of K made with any unsupervised clustering algorithm from data on real (non-idealized) populations should always be interpreted and used cautiously...

I am fully aware that we are dealing with probabilistic statements here, not certainties. However, I am also convinced that the search for the number of genetic units (K) is a central issue. This is precisely what Flock intends to do, and it is also the main task that STRUCTURE was built for.
Thus, the basic purpose of FLOCK is to estimate K and if the estimate is not *undecided*, to obtain a reasonably accurate partition of the collection of genotypes. Consistency across different values of K is not intended. By considering outputs for any value of K, the authors make invalid comparisons. Comparisons among clustering solutions for different values of K provided by STRUCTURE are commonly reported in the literature, and the successive clustering solutions may at times make biological sense. That practice, however, is not recommended in FLOCK and therefore cannot serve as basis for evaluating its results outside the estimated value of K.

Misinterpretation of `best run`

However, the authors are in fact proposing to analyse the output of FLOCK for each K as if it were potentially informative. This, in turn, lead to a misunderstanding of what 'best run' means. For instance on p.12, they state:

For all groups of runs for K> 2 no plateaus were observed...it is unclear in these cases what constitutes the `best run` as all plateau lengths are equal to one.

The answer is simple: there is no best run. The term 'best run' applies exclusively to the run with the longest plateau corresponding to the estimate of K. First the stopping rules are applied then the estimation rules. If the estimate is not *undecided*, the only best run to be considered is the one associated with the estimated K. If the estimate is *undecided* there is no best run. The documentation and also the flow chart in the latest article (Duchesne and Turgeon 2012) make that clear. In fact, outside the estimated K, no partitions from FLOCK should be considered. Therefore comparisons between clusterings from FLOCK and STRUCTURE for all K except the estimated one are just irrelevant and not valid.

This same approach (= all K outputs are valuable) also brings the authors to lament `FLOCK's tendency to converge to different solutions with large K` and conclude that STRUCTURE would be a better choice for complex problems. This is a surprising conclusion since FLOCK is expected to signal the absence of K components precisely by not repeating solutions for that K and thus producing very short plateaus if any. Clearly, by increasing the value of K, the true value will eventually be surpassed and FLOCK will tend to generate numerous solutions and therefore short plateaus (< 6). As for the real dataset, there is no ground to believe that K > 4 (line 246-248), so that the reported large variability of the FLOCK results may just be the right signal. Here there is a huge misunderstanding that leads the authors to systematically misinterpret potentially correct answers as a sign of inconsistency or lack of power.

Misinterpretation of the stochasticity of plateau lengths

Another misinterpretation of FLOCK output shows when the plateau records for K = 2 are reported (p.12) for each of the six groups of (50) runs and described as varying greatly. However, the important statistic here is the length of the longest plateau for each K as this will determine the decision on K (the main purpose of FLOCK). Here the top lengths for the six groups are: (7, 5, 5, 4, 5, 3). The corresponding decisions are *undecided* for 5 groups and K = 2+ for one group. And so there is not so much variation among top lengths and, more importantly, practically none at the decision level (5/6 *undecided*).

4) Comparisons of estimates of K

(p. 12) *The (ln P(D)) values from STRUCTURE was largest at K = 6 which was further supported by the qi plots, however, the ΔK method supported a K of 3.*

With STRUCTURE, the authors do not provide a definite estimate for K, and they do not provide support for several alleged estimates.
As is, the authors used Flock to reach the decision that STRUCTURE supports an estimate of K=6. Flock needed to go up to K=6 to reach a stopping rule and it is on that basis that they decided to run STRUCTURE up to K=6. This is not consistent with what the method actually prescribes nor with what is often (or should be) reported in the literature: K should be estimated from the peak value for (ln *P*(D)), not just its highest value but one preceded and succeeded by lower values. It is well known that sometimes the (ln P(D)) values from STRUCTURE increase monotonically with K. If one applied the simple largest value rule then obviously the estimate for K would be as large as the largest K that is tested. Here, that led, incorrectly, to the decision that it is the best estimate because Ln(P(D)) is maximum.
As for K = 3 obtained from the ΔK method, the results presented do not allow to evaluate whether this estimate is truly supported or not. It is well-known that this method, being based on the rate of change of Ln(P(D)) as a function of K, cannot provide support for the lowest value of K tested (here, K=2). For that reason it is customary to run STRUCTURE from K=1 such that K=2 is a possible estimate.

Finally, I note that, based on the same dataset and STRUCTURE, K=5 was preferred in Garza et al. 2014 (Trans. Amer. Fish. Soc.) , but that required considering the regional distribution of clusters, and plots of Ln(P(D)) or ΔK were not provided.

At the very least, the results supporting the estimate of K=6 on the basis of the maximum value of Ln(P(D)) (and K= 3 on the basis of ΔK for K=1 to a large value) should be provided.

The bottom line is that this dataset does not allow valid comparisons of estimates of K from the STRUCTURE and FLOCK programs, such estimates being the primary task of both programs. There may certainly be cases where FLOCK will provide a different and sometimes less (or more) accurate estimate for K than STRUCTURE, but the comparison made here is not done properly and would in any case concern a single dataset.

3) Speed comparisons

(p. 11) *All runs of FLOCK completed in 1133 minutes. Each of the six groups of runs averaged 188.8 minutes. All runs of STRUCTURE were completed in 486 minutes and averaged 80.8 minutes to complete each of the six groups of runs.*

(line 281) *Though it is difficult to reliably assess differences in program run times when there is no clear consensus on what constitutes a reasonable comparison with respect to run length and number of runs, our results indicate that Flock did not carry a distinct run-time advantage over STRUCTURE*

I agree with the authors that it is difficult to decide on the unit of comparison.

At first sight, their comparison seems to indicate that STRUCTURE is 188.8/80.8 = 2.33 faster than FLOCK. My own opinion is that the right unit of comparison is the *run* since each run produces the output from a clustering program i.e. a partition of a collection of genotypes into K clusters. Consequently, the speed of execution of cluster programs should be compared on the basis of average run time and not on batches of runs of unequal numbers. If one takes the run as basic unit, given that FLOCK was required to perform 50 runs each time STRUCTURE was required to run once, then the correct speed ratio should be 80.8 X 50/188.8, which means that FLOCK was over 21 times faster than STRUCTURE in clustering this particular dataset. Theoretically, one might argue that one run of STRUCTURE is really the equivalent of 50 runs of FLOCK since 50 runs is the recommended (for validation purposes) number of runs when using FLOCK while STRUCTURE may be run any number of times, including just once. The reason that large numbers of runs are not a standard recommendation for STRUCTURE is precisely that it is too slow. FLOCK is much faster mainly because convergence is very fast (usually in less than 10 iterations = reallocations). It is doubtful that clusterings from STRUCTURE would be of even acceptable quality after 20 iterations. Otherwise why would the authors run a total of 200000 sweeps (one sweep = one iteration), a number consistent with customary usage and recommendations, rather than a mere 20?

Another option that has not been formally evaluated but probably deserves considering would be to compare the total computing time required for a user to reach a decision on the best K estimate, ideally with various datasets for which both programs provide the same K estimates. This would require deciding how many runs of STRUCTURE should be used as there are currently no strict rules on the matter. Perhaps the program should be run following the recommendation of Gilbert et al. (2012, Mol. Ecol. Res.), namely 20 replicates for each K, and a range of K-values truly allowing to estimate the peak or plateau value of Ln(P(D)). Using the dataset presented here, this recommendation implies testing for values larger than K=6, and the range used by Garza et al. (2014) in their original analysis, i.e. K=2 to K=10, seems reasonable. That would be a start for a comparison that truly matters to real users. Among the other people who have run the two programs, I feel that there is a general consensus that FLOCK is much faster than STRUCTURE , and I believe that they are referring to this 'real life' experience. It would be interesting to have a quantitative comparison based on processing time to reach a decision on K.

Additional comment:

The authors mention the existence of a set of stopping rules in FLOCK but not their absence in STRUCTURE and the possible consequences thereof. For example, if no peak has yet been observed for (ln P(D)) values from STRUCTURE, how long should one process ever larger values of K? Interestingly, this is exactly the situation with the empirical dataset discussed in the paper. The authors chose 6 as the upper bound for K, a value clearly derived from the upper bound from the FLOCK stopping rules. Now this in turn lead them to suggest that K = 6 was to be considered as a plausible number of populations since (ln P(D)) was highest at K = 6, which is obviously a misapplication of the estimation rules for K with STRUCTURE. Then they noticed the large number of solutions from FLOCK, while implicitly assuming the validity of the K = 6 estimate. From this they concluded that FLOCK `seems to find the same solution less consistently than does STRUCTURE`. As explained above, if K = 6 were wrong, and there is no evidence to the contrary, then FLOCK would be expected to produce numerous solutions and the more the better, thereby indicating that there is no support for K=6. And so the absence of stopping rules in STRUCTURE combined with a misapplication of the estimation rule for K for STRUCTURE and a misunderstanding of FLOCK plateau analysis lead the authors to a specific conclusion not supported by their data and, worse still, one that they generalized over the entire domain of application of FLOCK.

Recommendations:

Based on the numerous problems I have described, I think this paper unfit to be published in Molecular Ecology Resources. If the authors intend to establish that `FLOCK yields results similar to STRUCTURE with the no-admixture model and uncorrelated allele frequency prior`, they should do so by running FLOCK, STRUCTURE with the no-admixture model and STRUCTURE with the with-admixture model over a large collection of datasets spanning numerous combinations of pertinent parameter values. The object of the comparisons should be the K-value estimated by each algorithm, not the solutions for each processed K-value. The datasets should be mostly

from simulations so as to compare the performance of each algorithm on a solid basis and also, possibly, on some real datasets whose genetic structure is already well known. This, of course, takes time but there is no analytical shortcut to this work. This data-driven research would hit a double target: compare FLOCK to the two models of STRUCTURE and compare the performances of the two STRUCTURE models on a broad empirical basis. The results would be very useful to all potential users of clustering programs and would practically ensure that they are not misled into false conceptions. They should be published in a full-fledged article, not just a comment. I also suggest that the authors make an additional effort to better understand plateau analysis especially in connection with variability of solutions i.e. lengths of plateaus. Short plateaus (numerous solutions) are the right answer whenever K is the wrong answer to the 'number of populations' problem. If run times are to be compared, the comparisons should follow the above recommendation such that a real user gets a simple message out of it.

Pierre Duchesne