Swarm v2: highly-scalable and high-resolution amplicon clustering

Frédéric Mahé[1], Torbjørn Rognes[2,3], Christopher Quince[4], Colomban de Vargas[5,6], and Micah Dunthorn[1]

[1]Department of Ecology, University of Kaiserslautern, Kaiserslautern, Germany.
[2]Department of Informatics, University of Oslo, Oslo, Norway.
[3]Department of Microbiology, Oslo University Hospital, Rikshospitalet, Oslo, Norway.
[4]Warwick Medical School, University of Warwick, Coventry CV4 7AL, United Kingdom.
[5]CNRS, UMR 7144, EPEP – Évolution des Protistes et des Écosystèmes Pélagiques, Station Biologique de Roscoff, 29680 Roscoff, France.
[6]Sorbonne Universités, UPMC Univ Paris 06, UMR7144 Station Biologique de Roscoff, Roscoff, France.

ABSTRACT

Previously we presented Swarm v1, a novel and open source amplicon clustering program that produced fine-scale molecular operational taxonomic units (OTUs), free of arbitrary global clustering thresholds and input-order dependency. Swarm v1 worked with an initial phase that used iterative single-linkage with a local clustering threshold (d), followed by a phase that used the internal abundance structures of clusters to break chained OTUs. Here we present Swarm v2 that has two important novel features: 1) a new algorithm for $d = 1$ that allows the computation time of the program to scale linearly with increasing amounts of data; and 2) the new fastidious option that reduces under-grouping by grafting low abundant OTUs (e.g., singletons and doubletons) onto larger ones. Swarm v2 also directly integrates the clustering and breaking phases, dereplicates sequencing reads with $d = 0$, outputs OTU representatives in fasta format, and plots individual OTUs as two-dimensional networks.

Keywords: environmental diversity, barcoding, molecular operational taxonomic units

INTRODUCTION

Traditional *de novo* amplicon clustering methods that can handle large high-throughput sequencing datasets (e.g., Edgar, 2010; Ghodsi et al., 2011; Fu et al., 2012) suffer from two fundamental problems. First, they rely on an arbitrary fixed global clustering threshold to group amplicons into molecular operational taxonomic units (OTUs). Global clustering thresholds have rarely been justified and are not applicable to all taxa and marker lengths (e.g., Caron et al., 2009; Dunthorn et al., 2012; Nebel et al., 2011). Second, there is variability in the clustering results due to amplicon input order (Koeppel and Wu, 2013; Mahé et al., 2014).

To solve these problems, we previously introduced the open source Swarm v1 that implemented an initial clustering phase written in C++, then a breaking phase written in Python (Mahé et al., 2014). Swarm's clustering phase (Fig. 1a) was novel in its approach to single linkage clustering in that, instead of using a global clustering (e.g., Hartmann et al., 2012; Huse et al., 2010), amplicons were iteratively added together using a small local clustering threshold (d) until no more amplicons could be added. Using $d = 1$ produced the highest resolution OTUs. Swarm's breaking phase (Fig. 1b) was novel in that it used the abundance of amplicons to reveal the internal structure of potentially chained OTUs (i.e., a low abundant link between high abundant amplicons). These chained OTUs were then refined by splitting them.

Since its introduction, Swarm v1 has been used in a variety of datasets (de Vargas et al., 2015; Filker et al., 2015; Lima-Mendez et al., 2015; Mahé et al., 2015; Oikonomou et al., 2015). However, since the breaking phase was written in Python, it lacked scalability and was cumbersome to use. Kopylova et al. (prep) also found that in comparison to other clustering methods, Swarm v1 tended to produce relatively more low abundant OTUs; e.g., singletons and doubletons. And most importantly, Swarm v1 and other current *de novo* algorithms could not cluster today's largest high-throughout sequencing datasets within

PeerJ PrePrints | https://dx.doi.org/10.7287/peerj.preprints.1222v2 | CC-BY 4.0 Open Access | rec: 10 Jul 2015, publ: 10 Jul 2015

a reasonable amount of time (Rideout et al., 2014). Here we introduce Swarm v2 to help solve these problems, as well as introduce new and useful features.

MATERIAL AND METHODS

Linear complexity *de novo* clustering approach

Today's largest amplicon datasets contain hundreds of millions of amplicons and pose a computational challenge to *de novo* clustering methods. Because of this scalability problem, Rideout et al. (2014) proposed using a mixed clustering approach with an initial closed-reference clustering that compares the amplicons to what is known in taxonomic reference databases to capture most of the data, followed by a *de novo* clustering with the remaining amplicons. We feel that using only *de novo* clustering is the most powerful approach when working with amplicons from unexplored environments that lack sufficient taxonomic reference databases or with rare taxa that were previously missed in already-sampled environments. We therefore worked to improve Swarm's scalability.

Like other current *de novo* clustering approaches, Swarm v1 presented an apparent quadratic behavior in that it needs to perform a number of comparisons that grows as the square of the number of amplicons. In Swarm v2 we first reduced computational time by improving the multithreading and making a better usage of multi-core CPUs. We further reduced computational time by using a novel algorithmic approach. This linear complexity approach only applies for $d = 1$, which is Swarm's default and preferred parameter as it produces the highest resolution clusters.

As background to this linear approach, let us consider a nucleic sequence S made of As, Cs, Gs and Ts. A "microvariant" is a sequence with one difference ($d = 1$) to the original sequence S. How many distinct microvariants can derive from S? In a sequence S of length L, each position can be substituted with 3 different bases, so there are $3L$ possible microvariants due to substitutions. Each position in S can be deleted once, so there are L possible microvariants due to deletions. Insertions are more complicated. An insertion can happen before or after each position in the sequence S, and four different nucleotides can be inserted resulting in $4(L+1)$ microvariants. However, some insertions will result in the same microvariant: for example, inserting a "G" before or after a "G" will result in the same sequence "GG". As that situation arises for all positions in S but one, the maximum number of distinct insertions is not $4(L+1)$, but $3(L+1)+1 = 3L+4$. In total, the maximum number of microvariants that can be obtained from a given sequence S of length L is $3L+L+3L+4 = 7L+4$.

As stated above, different sequence modifications can produce the same microvariant. The final number of distinct microvariants depends on the number of homopolymer stretches in the sequence. In the extreme situation where the sequence is entirely made of one type of nucleotide, the number of microvariants due to deletions drops from L to 1. For example, if S is entirely made of "G", all possible deletions yield the same microvariant. The total number of distinct microvariants then drops to its minimum value: $3L+1+3(L+1)+1 = 6L+5$.

The number of distinct microvariants that can be obtained from a sequence S of length L then varies from $6L+5$ to $7L+4$. In practice, it means that a typical high-throughput sequencing 16S rRNA sequence of 130 nucleotides will yield at least 785 microvariants and at most 914, and that the number of microvariants will increase linearly with the sequence length. With current sequencing technologies read length increases at a slower rate than read number, and is safe to assume it will continue to do so in the foreseeable future.

Based on these characteristics of microvariants, we switched from an approximate-string comparison approach to an exact-string comparison approach. That is, for a given amplicon, instead of doing an exact pairwise alignment comparison against all available amplicons in the pool that have yet to be subsumed into an OTU, Swarm v2 generates all possible microvariants of the amplicon and checks whether or not they are present in the amplicon pool using a hash table. This exact-string search approach is extremely fast, and does not depend on the number of available amplicons in the pool. Therefore, the apparent computational complexity changes from n^2 to $n \times L$, where L is the average amplicon length.

Reducing under-grouping

As observed by Kopylova et al. (prep), Swarm v1 tended to produce relatively more low abundant OTUs; e.g., singletons and doubletons. This problem is due to Swarm's approach that depends on the existence of a continuous path of linked amplicons. Linking amplicons can be missing, especially when sequencing is shallow. When this occurs, there can be under-grouping of closely related amplicons leading to small OTUs surrounding a larger OTU.

To address this problem in Swarm v2, we introduced a new step—called the fastidious option—to graft low abundant OTUs onto more abundant ones by postulating a linking amplicon (Fig. 1c). In practice, microvariants (independent of the microvariants produced in the clustering phase) are produced for all the amplicons belonging to low abundant OTUs and stored in a Bloom filter (a probabilistic dictionary). An OTU abundance lower than 3 is the default threshold value (user-controllable parameter); i.e., the fastidious option will target OTUs with an abundance of one (singletons) or two (doubletons). Microvariants are then produced for high-abundant amplicons and cross-checked against the Bloom filter. The fastidious option can consume a large amount of memory, but is apparently linear in terms of computation time (see Results). The user does have control over memory usage and can exchange memory space for computation time. When using $d = 1$, the fastidious option is highly recommended.

The fastidious option can be viewed as a way to reduce data loss, as many researchers conservatively consider low abundant OTUs as spurious errors and remove them from downstream analyses (Behnke et al., 2011; Kunin et al., 2010). With the fastidious option, though, one can retain many of these amplicons by attaching them to more abundant OTUs. In contrast with an increase of d, the fastidious option does not degrade the clustering resolution; i.e., it reduces the under-grouping of amplicons without inducing over-grouping.

Other new and useful features

In Swarm v2 we introduce a number of options improving both speed and usability. First, there is a simpler user command line interface. For example, the breaking phase is now written in C++ and is performed directly during the growth phase, which further significantly reduces computation time. We chose to implement a strict, simple, non-parametric breaking model that prevents any increase in abundance along a continuous amplicon path (Fig. 1b). Breaking of linked chains can be deactivated.

Second, Swarm v2 extends the notion of clustering by allowing the option $d = 0$. Users can now dereplicate their sequencing reads into strictly identical amplicons (sensu Mahé et al. (2015); i.e., reads that have exactly the same sequences with no mutations, insertions, deletions). This fast dereplication approach uses the same algorithm as in VSEARCH (https://github.com/torognes/vsearch).

Third, Swarm v2 can output OTU representative amplicons in fasta format. A representative is the most abundant amplicon of an OTU, and its abundance is updated to reflect the total OTU abundance. OTU representatives are normally used for downstream community-comparative, novel-diversity, and ecosystem-functioning questions.

Fourth, Swarm v2 offers the possibility to visualize the internal structure of OTUs, which allows the user to gain further knowledge of its underlying genetic and ecological diversity (Fig. 2, 3). These plots are in the form of a network projected in two-dimensions. Edges in these networks only represent the parameter d used; the length of the edges carries no information. The nodes in the networks represent amplicons. The abundance information of these amplicons is represented in three ways: the size of the node, the color of the node, and text when its abundance value is 10 or more.

Analyses

To demonstrate the apparent linear complexity of Swarm v2, we analyzed 16S rRNA reads from the Earth Microbiome Project (Gilbert et al., 2014), which is the largest amplicon dataset currently available. The following swarm commands were used: `swarm -d 1 in.fasta`, and `swarm -d 1 -f in.fasta`. To illustrate over- and under-grouping of amplicons, the importance of the breaking phase, high-resolution clustering, and Swarm's ability to visualize OTUs' internal structures, we used 18S rRNA amplicon data from the BioMarKs consortium (Logares et al., 2014) that sampled European near-shore marine sites. The PR2 v203 reference database was used for taxonomic assignment (Guillou et al., 2013). The full methods can be found online in html format (Supplementary File 1).

RESULTS AND DISCUSSION

Time and space benchmarks

For $d = 1$, Swarm's default parameter, using the full- and sub-datasets of the Earth Microbiome Project we were able to evaluate Swarm v2's clustering time and memory usage. These timing experiments were obtained with Swarm v2.1.1 on a machine with 1024 GB of RAM running Red Hat CentOS v6.6 and Linux kernel v3.9.1 on four Intel Xeon E5-4620 processors (2.2 GHz) having a total of 32 physical cores. Swarm was run with 8 threads (option "-t 8") and memory limited to 240 GB ("-c 245760"). The times indicated below are the averages of four runs. With the sub-dataset of 154,896,650 strictly identical amplicons (representing 1,277,640,415 reads), Swarm without the fastidious option took 1 hour and 45 minutes ± 1 minute. With the full-dataset of 314,871,149 strictly identical amplicons (representing 2,254,207,945 reads), Swarm without the fastidious option took 3 hours and 41 minutes ± 1 minute. Doubling of dataset size approximately doubles the run time, confirming the apparent linear time complexity. Therefore, if the size of the Earth Microbiome Project were to increase ten times, it should take about ten times longer to cluster it (less than two days). These fast times of Swarm v2 contrast with the estimated computational time of UCLUST v6.1 as inferred by Rideout et al. (2014). Using a smaller partial-dataset of the Earth Microbiome Project consisting of only 660,000,000 reads (that dereplicate into a unspecified number of strictly identical amplicons), Rideout et al. (2014) estimated UCLUST's runtime to 150 days on an 8 core computer.

With the sub-dataset representing 24 GB of input data, the memory usage of Swarm v2 with $d = 1$ was 41 GB. With the full-dataset representing 49 GB of input data, the memory usage was 86 GB. Memory requirements can therefore be estimated to be approximately equal to the size of the input dataset plus 2/3.

When clustering at $d = 1$ and using the fastidious option, the total computational time of the sub-dataset was 4 hours and 59 minutes ± 1 minute, which resulted in 40.0% fewer OTUs in total. The total computational time of the full-dataset took 11 hours and 28 minutes ± 5 minutes, which resulted in 38.3% fewer OTUs in total. This considerable reduction in the number of singletons and doubletons in both datasets helps solve the problem found by Kopylova et al. (prep). The computation time is about three times longer when using the fastidious option than without it.

The total memory usage of $d = 1$ with the fastidious option for the sub-dataset was 114 GB, while it was 239 GB (capped) for the full-dataset. This amount of memory might not be available to all users. Therefore we have implemented two options to control and cap memory usage of the fastidious option: by defining the maximum memory footprint, and by adjusting the size of the Bloom filter entries. Both of these options allow users to trade computational time for memory space.

OTU visualizations

We provide examples of Swarm v2's graphical representation of the internal structure of its high-resolution OTUs by using V4 and V9 18S rRNA amplicons. In both cases the breaking phase and fastidious option were turned off. With the V9 data (about 129 bp in length), the graph shows two high abundant OTUs linked by one lower abundant amplicon (Fig. 2). The number of nucleotide differences between these two linked OTUs is only two, or about 98.4% similarity. If the breaking phase and fastidious option were applied to these V9 amplicons, nine separate OTUs would have been formed: two high abundant, and seven low abundant. These two high abundant OTUs are taxonomically assigned to different genera of Collodaria (Radiolaria). On the same V9 amplicons, UCLUST v6 (as well as v7 and v8) using a global clustering threshold of 97% similarity produced 37 OTUs (one high abundant, and 36 low abundant). The one high abundant OTU from UCLUST lumped the two Collodaria genera, thus masking meaningful biological data.

With the V4 amplicons (about 380 bp in length), the graph shows three high abundant OTUs linked by one to three low abundant amplicons (Fig. 3). The number of nucleotide differences between these three linked OTUs is only two and four, or about at least 98.9% similarity. If the breaking phase and fastidious option were applied to these V4 amplicons, seven separate OTUs would have been formed: three high abundant, and four low abundant. These three high abundant OTUs are assigned to different taxa of Cnidaria. On the same V4 amplicons, UCLUST v6 (as well as v7 and v8) produced only one OTU with the widely used global clustering threshold of 97% similarity, again masking meaningful biological data.

These amplicon data show that, compared to UCLUST, Swarm v2 can distinguish higher-resolution clusters and reduces both over-grouping and under-grouping on a range of marker lengths. In both of these amplicon examples, Swarm v2 is able to distinguish different taxa, while UCLUST conceals them.

Outlook

We are currently working on a number of fronts to continue making Swarm harder, better, faster, stronger. For example, preliminary experiments indicate that with a novel multithreading approach for $d \geqslant 2$ a ten-fold increase in speed could be obtained (although $d \geqslant 2$ will still be quadratic in behavior). Internally encoding nucleotides on two bits instead of eight bits may help reduce both memory consumption and computational time. Additional computation time can be saved by merging the fastidious option with the initial clustering phase. To facilitate its usage, Swarm v2 can be included in QIIME (Caporaso et al., 2010), which already offers Swarm v1.2, and in Galaxy (Goecks et al., 2010).

In summary, Swarm v2 is a highly-scalable approach that uses a local clustering threshold to produce high-resolution *de novo* OTUs and reduces low abundant OTUs. Swarm v2 is an optimized C++ program able to handle many hundreds of millions of amplicons. It is open source and freely available at https://github.com/torognes/swarm under the GNU Affero General Public License version 3.

ACKNOWLEDGMENTS

We would like to thank the Earth Microbiome Project for the use of their data and their constructive comments. Daft Punk provided the background music. We are grateful to the computational resources at the Regional Computing Center at the University of Kaiserslautern, and the Abel computing cluster at the University of Oslo.

REFERENCES

Behnke, A., Engel, M., Christen, R., Nebel, M., Klein, R. R., and Stoeck, T. (2011). Depicting more accurate pictures of protistan community complexity using pyrosequencing of hypervariable SSU rRNA gene regions. *Environmental Microbiology*, 13(2):340–349.

Caporaso, J. G., Kuczynski, J., Stombaugh, J., Bittinger, K., Bushman, F. D., Costello, E. K., Fierer, N., Pena, A. G., Goodrich, J. K., Gordon, J. I., Huttley, G. A., Kelley, S. T., Knights, D., Koenig, J. E., Ley, R. E., Lozupone, C. A., McDonald, D., Muegge, B. D., Pirrung, M., Reeder, J., Sevinsky, J. R., Turnbaugh, P. J., Walters, W. A., Widmann, J., Yatsunenko, T., Zaneveld, J., and Knight, R. (2010). QIIME allows analysis of high-throughput community sequencing data. *Nature Methods*, 7(5):335–336.

Caron, D. A., Countway, P. D., Savai, P., Gast, R. J., Schnetzer, A., Moorthi, S. D., Dennett, M. R., Moran, D. M., and Jones, A. C. (2009). Defining DNA-Based Operational Taxonomic Units for Microbial-Eukaryote Ecology. *Applied and Environmental Microbiology*, 75(18):5797–5808.

de Vargas, C., Audic, S., Henry, N., Decelle, J., Mahé, F., Logares, R., Lara, E., Berney, C., Le Bescot, N., Probert, I., Carmichael, M., Poulain, J., Romac, S., Colin, S., Aury, J.-M., Bittner, L., Chaffron, S., Dunthorn, M., Engelen, S., Flegontova, O., Guidi, L., Horák, A., Jaillon, O., Lukeš, J., Malviya, S., Morard, R., Mulot, M., Scalco, E., Siano, R., Vincent, F., Zingone, A., Dimier, C., Picheral, M., Searson, S., Kandels-Lewis, S., Tara Oceans coordinators, Acinas, S. G., Bork, P., Bowler, C., Gorsky, G., Grimsley, N., Hingamp, P., Iudicone, D., Not, F., Ogata, H., Pesant, S., Raes, J., Sieracki, M. E., Speich, S., Stemman, L., Sunagawa, S., Weissenbach, J., Wincker, P., and Karsenti, E. (2015). Eukaryotic plankton diversity in the sunlit global ocean. *Science*, 348(6237).

Dunthorn, M., Klier, J., Bunge, J., and Stoeck, T. (2012). Comparing the Hyper-Variable V4 and V9 Regions of the Small Subunit rDNA for Assessment of Ciliate Environmental Diversity. *Journal of Eukaryotic Microbiology*, 59(2):185–187.

Edgar, R. C. (2010). Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, 26(19):2460–2461.

Filker, S., Gimmler, A., Dunthorn, M., Mahé, F., and Stoeck, T. (2015). Deep sequencing uncovers protistan plankton diversity in the Portuguese Ria Formosa solar saltern ponds. *Extremophiles*, 19(2):283–295.

Fu, L., Niu, B., Zhu, Z., Wu, S., and Li, W. (2012). CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, 28(23):3150–3152.

Ghodsi, M., Liu, B., and Pop, M. (2011). DNACLUST: accurate and efficient clustering of phylogenetic marker genes. *BMC Bioinformatics*, 12(1):271.

Gilbert, J., Jansson, J., and Knight, R. (2014). The Earth Microbiome project: successes and aspirations. *BMC Biology*, 12(1):69.

Goecks, J., Nekrutenko, A., Taylor, J., and The Galaxy Team (2010). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology*, 11(8):R86.

Guillou, L., Bachar, D., Audic, S., Bass, D., Berney, C., Bittner, L., Boutte, C., Burgaud, G., de Vargas, C., Decelle, J., del Campo, J., Dolan, J., Dunthorn, M., Edvardsen, B., Holzmann, M., Kooistra, W., Lara, E., Le Bescot, N., Logares, R., Mahé, F., Massana, R., Montresor, M., Morard, R., Not, F., Pawlowski, J., Probert, I., Sauvadet, A.-L., Siano, R., Stoeck, T., Vaulot, D., and Zimmermann, P. Christen, R. (2013). The Protist Ribosomal Reference database (PR2): a catalog of unicellular eukaryote Small Sub-Unit rRNA sequences with curated taxonomy. *Nucleic Acids Research*, 41(D1):D597–D604.

Hartmann, M., Howes, C. G., VanInsberghe, D., Yu, H., Bachar, D., Christen, R., Henrik Nilsson, R., Hallam, S. J., and Mohn, W. W. (2012). Significant and persistent impact of timber harvesting on soil microbial communities in Northern coniferous forests. *ISME Journal*, 6(12):2199–2218.

Huse, S. M., Mark Welch, D., Morrison, H. G., and Sogin, M. L. (2010). Ironing out the wrinkles in the rare biosphere through improved OTU clustering. *Environmental Microbiology*, 12(7):1889–1898.

Koeppel, A. F. and Wu, M. (2013). Surprisingly extensive mixed phylogenetic and ecological signals among bacterial Operational Taxonomic Units. *Nucleic Acids Research*, 41(10):5175–5188.

Kopylova, E., Navas-Molina, J. A., Mercier, C., Xu, Z., Mahé, F., Rognes, T., Caporaso, J. G., and Knight, R. (in prep.). Open-source sequence clustering methods improve the state of the art.

Kunin, V., Engelbrektson, A., Ochman, H., and Hugenholtz, P. (2010). Wrinkles in the rare biosphere: pyrosequencing errors can lead to artificial inflation of diversity estimates. *Environmental Microbiology*, 12(1):118–123.

Lima-Mendez, G., Faust, K., Henry, N., Decelle, J., Colin, S., Carcillo, F., Chaffron, S., Ignacio-Espinosa, J. C., Roux, S., Vincent, F., Bittner, L., Darzi, Y., Wang, J., Audic, S., Berline, L., Bontempi, G., Cabello, A. M., Coppola, L., Cornejo-Castillo, F. M., d'Ovidio, F., De Meester, L., Ferrera, I., Garet-Delmas, M.-J., Guidi, L., Lara, E., Pesant, S., Royo-Llonch, M., Salazar, G., Sánchez, P., Sebastian, M., Souffreau, C., Dimier, C., Picheral, M., Searson, S., Kandels-Lewis, S., Tara Oceans coordinators, Gorsky, G., Not, F., Ogata, H., Speich, S., Stemmann, L., Weissenbach, J., Wincker, P., Acinas, S. G., Sunagawa, S., Bork, P., Sullivan, M. B., Karsenti, E., Bowler, C., de Vargas, C., and Raes, J. (2015). Determinants of community structure in the global plankton interactome. *Science*, 348(6237).

Logares, R., Audic, S., Bass, D., Bittner, L., Boutte, C., Christen, R., Claverie, J.-M., Decelle, J., Dolan, J. R., Dunthorn, M., Edvardsen, B., Gobet, A., Kooistra, W. H. C. F., Mahé, F., Not, F., Ogata, H., Pawlowski, J., Pernice, M. C., Romac, S., Shalchian-Tabrizi, K., Simon, N., Stoeck, T., Santini, S., Siano, R., Wincker, P., Zingone, A., Richards, T. A., de Vargas, C., and Massana, R. (2014). Patterns of rare and abundant marine microbial eukaryotes. *Current Biology*, 24(8):813–821.

Mahé, F., Mayor, J., Bunge, J., Chi, J., Siemensmeyer, T., Wahl, B., Paprotka, T., Filker, S., and Dunthorn, M. (2015). Comparing high-throughput platforms for sequencing the V4 region of SSU-rDNA in environmental microbial eukaryotic diversity survey. *Journal of Eukaryotic Microbiology*, 62(3):338–345.

Mahé, F., Rognes, T., Quince, C., de Vargas, C., and Dunthorn, M. (2014). Swarm: robust and fast clustering method for amplicon-based studies. *Peer Journal*, 2:e593.

Nebel, M., Pfabel, C., Stock, A., Dunthorn, M., and Stoeck, T. (2011). Delimiting operational taxonomic units for assessing ciliate environmental diversity using small-subunit rRNA gene sequences. *Environmental Microbiology Reports*, 3(2):154–158.

Oikonomou, A., Filker, S., Breiner, H.-W., and Stoeck, T. (2015). Protistan diversity in a permanently stratified meromictic lake (Lake Alatsee, SW Germany). *Environmental Microbiology*, 17(6):2144–2157.

Rideout, J. R., He, Y., Navas-Molina, J. A., Walters, W. A., Ursell, L. K., Gibbons, S. M., Chase, J., McDonald, D., Gonzalez, A., Robbins-Pianka, A., Clemente, J. C., Gilbert, J. A., Huse, S. M., Zhou, H.-W., Knight, R., and Caporaso, J. G. (2014). Subsampled open-reference clustering creates consistent, comprehensive OTU definitions and scales to billions of sequences. *PeerJ*, 2:e545.

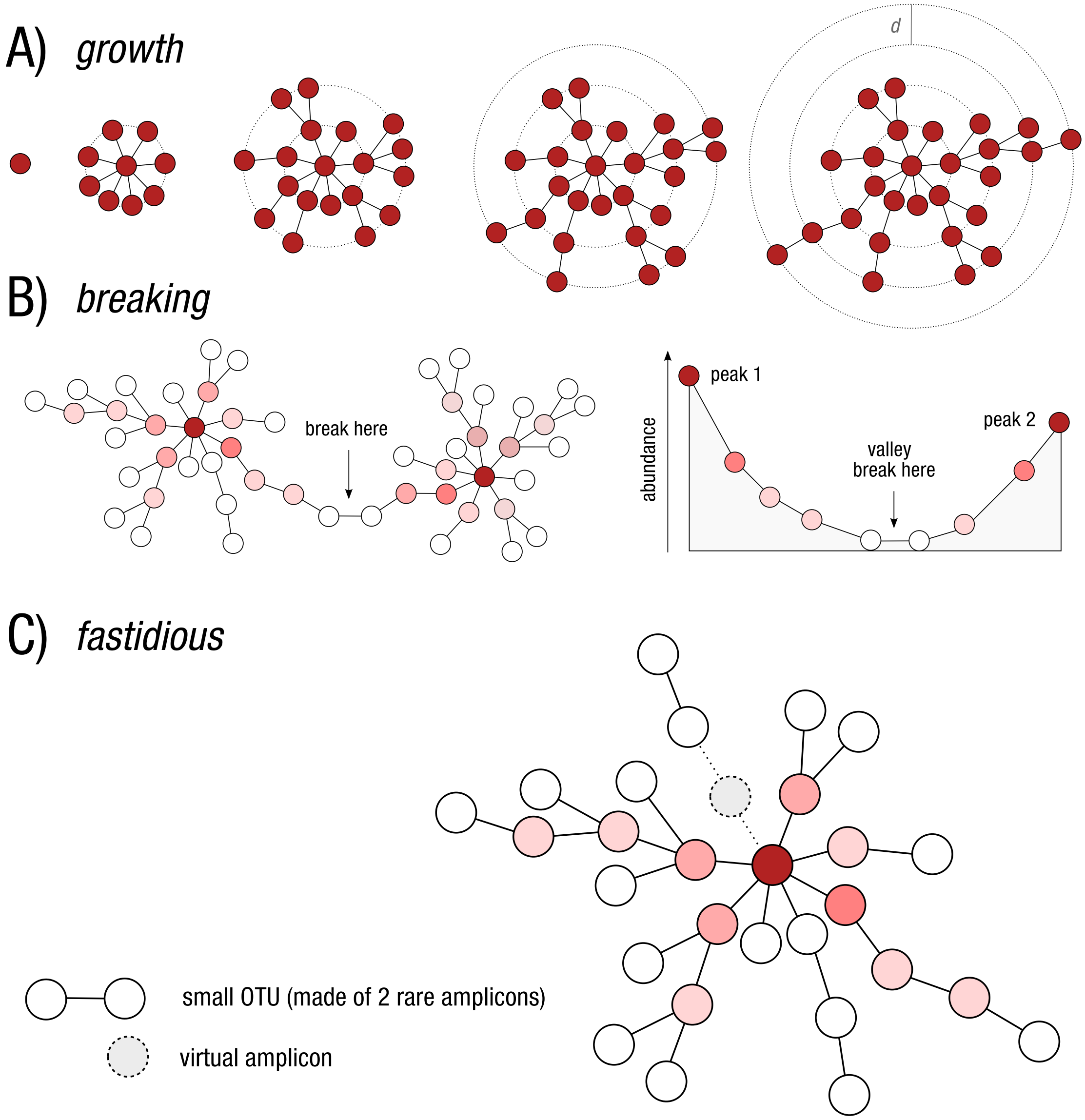

Figure 1. Schematic view of Swarm's clustering and refinement approach. (A) Swarm clusters amplicons iteratively by using a small user-chosen local threshold, *d*, allowing OTUs to grow to their natural limits, where no other amplicons can be added. (B) Swarm takes into account the abundance of each amplicon to produce higher resolution clusters, by not allowing the formation of amplicon chains. The darker the red, the higher the abundance. (C) The fastidious option avoids under-grouping (e.g., the production of small OTUs such as singletons and doubletons) by postulating the existence of virtual linking amplicons to graft smaller OTUs onto larger ones.

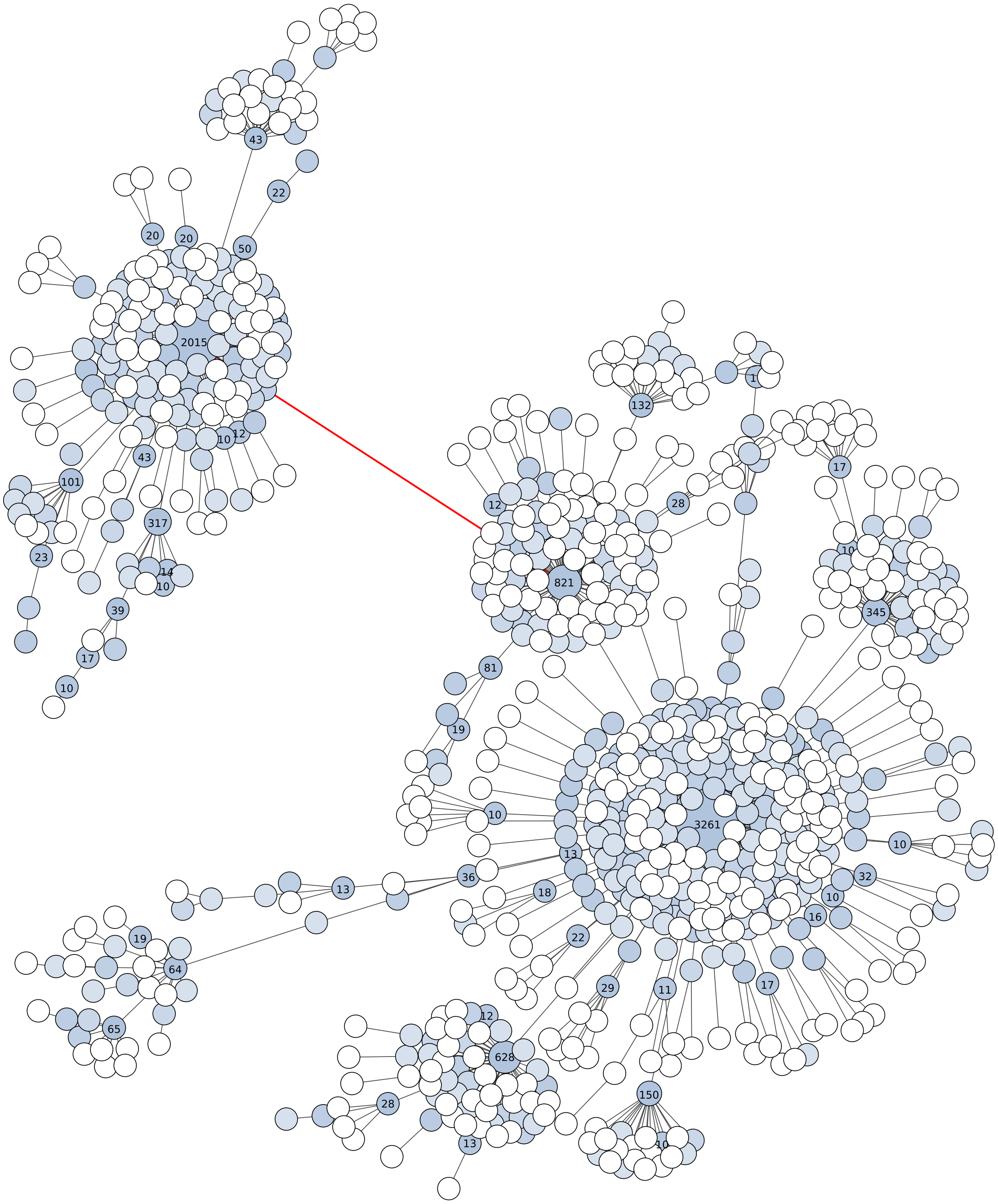

Figure 2. Graphical representation of an OTU produced by Swarm (breaking and grafting phases deactivated) when clustering the BioMarKs 18S rRNA V9 dataset (amplicons are appr. 129 bp in length). Nodes represent amplicons. Node size, color and text annotations represent the abundance of each amplicon. Edges represent one difference (substitution, deletion or insertion); the length of the edges carries no information. The edge colored in red indicates the breaking point between the two major sub-OTUs, each being assigned to a different genus of Collodaria (Radiolaria).

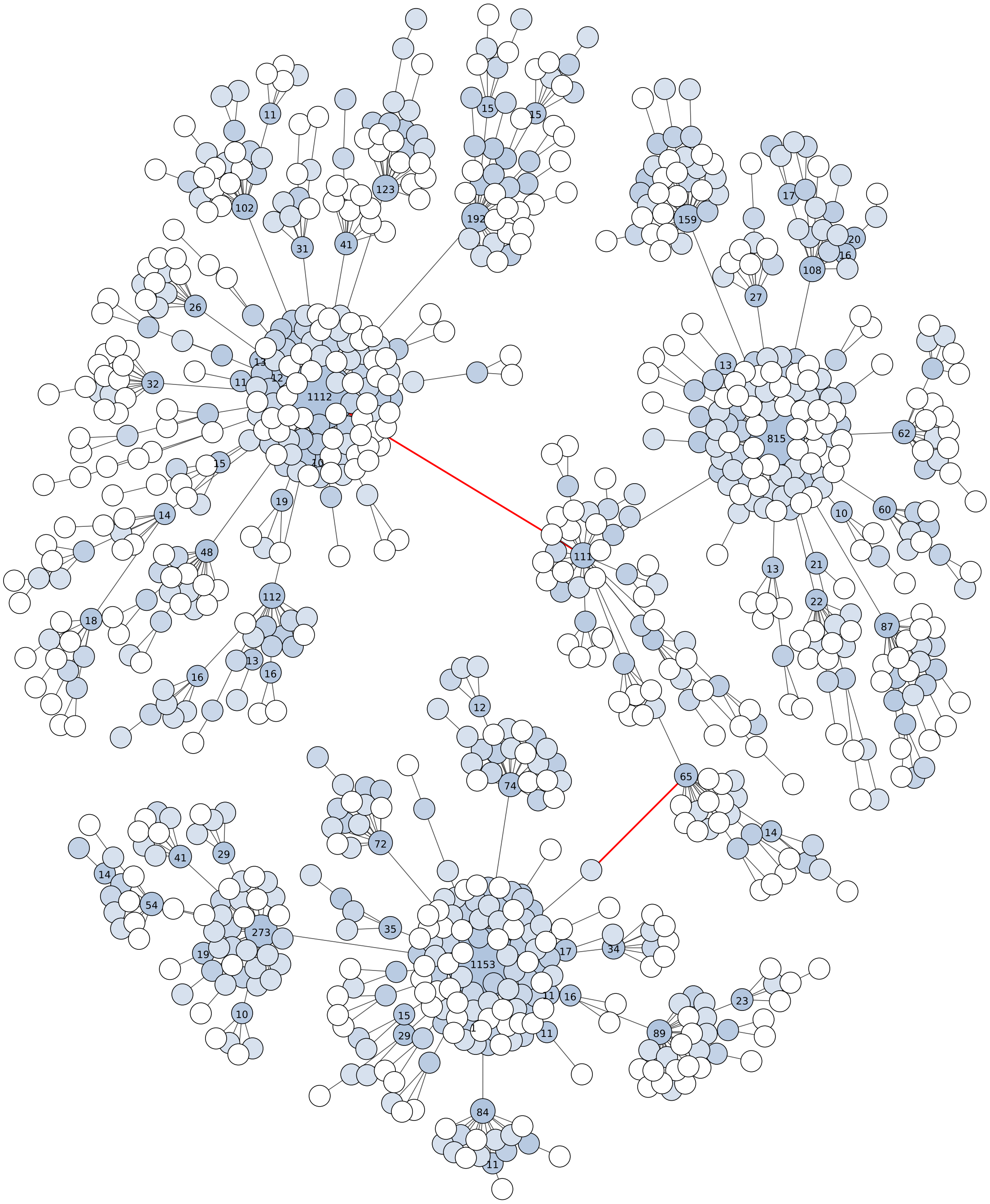

Figure 3. Graphical representation of an OTU produced by Swarm (breaking and grafting phases deactivated) when clustering the BioMarKs 18S rRNA V4 dataset (amplicons are appr. 380 bp in length). Nodes represent amplicons. Node size, color and text annotations represent the abundance of each amplicon. Edges represent one difference (substitution, deletion or insertion); the length of the edges carries no information. The two edges colored in red indicate the breaking point between the three major sub-OTUs, each being assigned to a different taxa of Cnidaria (Metazoa).