

## Supplementary material to “POMAGO: Multiple Genome-wide Alignment Tool for Bacteria”

**Table S1.** Comparison of genome-wide alignments of the Aquificales dataset with different reference organisms. aae – *Aquifex aeolicus*, hsp – *Hydrogenobaculum sp*, pma – *Persephonella marina*, saz – *Sulfurihydrogenibium azorense*, ssp – *Sulfurihydrogenibium YO3AOP1*, tal – *Thermocrinis albus*,

	Reference Organism					
	aae	hsp	pma	saz	ssp	tal
No. of blocks	1202	1101	1258	1330	1305	1207
Coverage (%)	83.4	77.3	70.3	88.9	84.2	84.1
Block length (nt)	1104	1094	1108	1096	1186	1045
Mean no of seq	4.97	5.16	4.91	4.84	4.83	5.00
Gaps/column	0.09	0.08	0.08	0.08	0.12	0.07
Weighted SoP	2.81	2.95	2.80	2.82	2.77	2.84

---

**Algorithm 1: Decomposition**

---

**Input:** orthology graph  $G = (V, E)$ , set of connected components  $\Gamma = \{C_i\}$  of  $G$ , neighborhood  $x$   
**Output:** cleaned orthology graph  $G'$

```
for all the  $e = (A_i, B_k) \in E$  do
  set  $e$ .similarity to the E-value from the pairwise blast of  $A_i$  and  $B_k$ ;
  set  $e$ .synteny to 0;
  for  $j = i - x \rightarrow i + x$  do
    if  $\exists B_l : (A_j, B_l) \in E \wedge ntDistance(A_i, A_j) \approx ntDistance(B_k, B_l)$  then
       $e$ .synteny ++;
  for  $l = k - x \rightarrow k + x$  do
    if  $\exists A_j : (A_j, B_l) \in E \wedge ntDistance(A_i, A_j) \approx ntDistance(B_k, B_l)$  then
       $e$ .synteny ++;

set  $G' = (V' = V, E' = \emptyset)$ ;
for all the  $C_i = (V_{C_i}, E_{C_i}) \in \Gamma$  do
  set  $\mathcal{D}_i = (V_{\mathcal{D}_i} = V_{C_i}, E_{\mathcal{D}_i} = \emptyset)$ ;
  sort  $E_{C_i}$  with respect to  $e$ .synteny and  $e$ .similarity;
  for all the  $e \in E_{C_i}$  in sort order do
    if  $(V_{\mathcal{D}_i}, E_{\mathcal{D}_i} \cup e)$  consists of connected components with one color per node then
      set  $E_{\mathcal{D}_i}$  to  $E_{\mathcal{D}_i} \cup e$ ;
  set  $E'$  to  $E' \cup E_{\mathcal{D}_i}$ ;

return  $G'$ ;
```

---

---

**Algorithm 2: Synteny-based Insertion**

---

**Input:** orthology graph  $G = (V, E)$ , set of connected components  $\Gamma = \{C_i\}$  of  $G$ , neighborhood  $x$ , threshold  $t$

```
for all the  $e = (A_i, B_k) \in E$  do
  for all the  $e' = (A_j, B_l) \in E$  with  $|i - j| = 1 = |k - l|$  do
    let  $C_{A_j}, C_{B_l} \in \Gamma$  be the connected components containing  $A_j$ , resp.  $B_l$ ;
    if  $ntDistance(A_i, A_j) \approx ntDistance(B_k, B_l)$  and  $C_{A_j}, C_{B_l}$  are color distinct
    then
      compute pairwise alignment of  $A_j$  and  $B_l$ ;
      if alignment score is better than  $t$  then
        exchange  $C_{A_j}$  and  $C_{B_l}$  in  $G$  by  $C_{A_j} + C_{B_l}$  the graph join;

for all the  $e_1 = (A_i, B_k), e_2 = (A_j, B_l) \in E$  with
 $|i - j| < x \wedge |k - l| < x \wedge \forall n \in (i, j), m \in (k, l) e' = (A_n, B_m) \notin E$  do
  for all the  $e' = (A_n, B_m)$  with  $n \in (i, j), m \in (k, l)$  do
    let  $C_{A_n}, C_{B_m} \in \Gamma$  be the connected components containing  $A_n$ , resp.  $B_m$ ;
    if  $ntDistance(A_i, A_n) \approx ntDistance(B_k, B_m)$  and
 $ntDistance(A_n, A_j) \approx ntDistance(B_m, B_l)$  and  $C_{A_n}, C_{B_m}$  are color distinct
    then
      compute pairwise alignment of  $A_n$  and  $B_m$ ;
      if alignment score is better than  $t$  then
        exchange  $C_{A_n}$  and  $C_{B_m}$  in  $G$  by  $C_{A_n} + C_{B_m}$  the graph join;
```

---

---

**Algorithm 3:** Synteny-based Annotation

---

**Input:** orthology graph  $G = (V, E)$ , set of connected components  $\Gamma = \{C_i\}$  of  $G$ , neighborhood  $x$ , threshold  $t$

**forall the**  $e_1 = (A_i, B_k), e_2 = (A_{i+1}, B_l) \in E$  *with*  $1 < |k - l| < x$  **do**

**if**  $ntDistance(A_i, A_{i+1}) \approx ntDistance(B_k, B_l)$  **then**

**forall the**  $B_m, m \in (k, l)$  *with*  $\forall n : e = (A_n, B_m) \notin E$  **do**

            compute pairwise alignment of  $B_m$  and the sequence between  $A_i$  and  $A_{i+1}$ ;

**if** *alignment score is better than*  $t$  **then**

                create a new node  $A_{i'}$  with the start and stop positions taken from the alignment;

                merge  $C_{B_m} \in \Gamma$  containing  $B_m$  with the singleton graph  $(A_{i'}, \emptyset)$  by a graph join;

---