1 A 7/5-approximation for discrete BGT

By using $R = \frac{7}{5}$ instead of $R = \frac{10}{7}$ in Algorithm 1 we can improve the approximation ratio to $\frac{7}{5}$. The proof remains mostly the same but we will use some computer assistance to verify the individual cases. The following lemma will help determine how long to run our program.

Lemma 1. *Given a porous schedule of length λ, whenever there are 5λ consecutive periods with $\frac{h}{p} \geq D_A$, then $h_p \geq D_A$ also holds for all subsequent periods.*

Proof. Let $p^* = 5\lambda a + b$. Then $\frac{h}{p} = \frac{7h(\lambda)a+h(\lfloor\frac{10}{7}b\rfloor)}{7\lambda a+\lfloor\frac{10}{7}b\rfloor}$.

Assume $\frac{h(\lambda)}{\lambda} \geq D_A$. In this case if $\frac{h}{p} = \frac{7h(\lambda)a+h(\lfloor\frac{10}{7}b\rfloor)}{7\lambda a+\lfloor\frac{10}{7}b\rfloor} \geq D_A$ then we also have $\frac{h}{p} = \frac{7h(\lambda)(a+1)+h(\lfloor\frac{10}{7}b\rfloor)}{7\lambda(a+1)+\lfloor\frac{10}{7}b\rfloor} \geq D_A$ and the result follows.

On the other hand, if $\frac{h(\lambda)}{\lambda} < D_A$, then by setting $b = 0$ we also have $\frac{7h(\lambda)a}{7\lambda a} < D_A$ for any $a \in \mathbb{N}$, which means there can never be 5λ consecutive periods with $\frac{h}{p} \geq D_A$. □

Theorem 1. *Algorithm 1 with $R = \frac{7}{5}$ is a $\frac{7}{5}$-approximation.*

Proof. Let $p_i^* = \lfloor\frac{K}{v_i}\rfloor$ and $p_i = \lfloor\frac{7p_i^*}{5}\rfloor$. Assume there is a schedule of height K. Then there is a solution to the pinwheel problem $p^* = (p_1^*, ..., p_n^*)$ and thus $\sum_{i=1}^{n} \frac{1}{p_i^*} \leq 1$.

We need to show that there is always a schedule for the pinwheel problem $p = (p_1, \ldots, p_n)$. In this case $\frac{1}{p_i} \leq \frac{3}{4p_i^*}$ holds for all periods except 2, 4 and 7.

We consider all cases that involve these periods by presenting porous schedules and determining for which periods p^* we have $\frac{h}{p} \geq D_A$.

Verifying these cases involves looking at a large number of periods with the help of the program in appendix B.

To keep the cases short we simply state the schedule and a value p^* such that $\frac{h}{p} \geq D_A$ holds for this p^* and all subsequent periods. We may also omit the value of p^* if a schedule completely solves a case and no subcases need to be considered.

2 We have $p/p^* \geq 6/5$ for all p. This means the density in the pinwheel-problem of Alg is at most 5/6. Since $p_1 = 2$ this pinwheel-problem can be solved.

3,3,4 and 3,4,4 In these cases there are only three plants.

$\mathbf{3, 4, p_3^* \geq 5}$ we schedule plants 1 and 2 using the schedule 1020. $\frac{h}{p} \geq D_A$ holds for $p^* \geq 5$. This solves all cases beginning with 3 and 4 that do not include 7 but we still need to consider cases that do include 7.

3,4,5,7 In this case there are only 4 plants. We give a short argument: Consider 7 consecutive positions and assume plant 5 is scheduled in position 4. Clearly plants 1 and 2 must each occur twice, once to the left and right of plant 5. Then plants 3 and 4 can each only occur once in these 7 positions. However, if plant 3 is to occur only once it must be in position 3 or 5 (w.l.o.g position 3). Then plant 1 must occur in positions 2 and 5 which leaves not enough room to schedule plant 2.

3,4,6 and 3,4,7 using the schedule 13201020 we get $p^* \geq 9$.

3,4,6,6 to 3,4,7,8 using the schedule 13201420 we get $p^* \geq 6$. Any additional periods of value 7 would lead to a negative density.

3,7 using the schedule 1020 1000 we get $p^* \geq 5$. (This also solves the cases 3,5,7 and 3,6,7.)

3,7,7 using the schedule 1020 1030 we get $p^* \geq 2$.

3,7,7,7 using the schedule 1024 1030 we get $p^* \geq 5$.

3,7,7,7,7 using the schedule 1024 1035 we get $p^* \geq 5$. Any additional periods of value 7 would lead to a negative density.

4 using the schedule 10000 we get $p^* \geq 9$.

4,4 follows from 2.

4,5 using the schedule 12000 10020 10002 10000 we get $p^* \geq 9$.

4,5,5 using the schedule 12003 10020 13002 10300 we get $p^* \geq 4$.

4,5,5,7 using the schedule 12043 10020 13402 10304 12003 10420 13002 14300 we get $p^* \geq 5$. This also solves any case where there are additional periods 5 or 6 before the 7.

4,5,5,7,7 using the schedule 12043 15020 13452 10304 12503 10425 13002 14350 we get $p^* \geq 5$. Any additional periods of value 7 would lead to a negative density.

4,5,6 using the schedule 31002 01300 21003 01200 we get $p^* \geq 7$.

4,5,6,6 follows from $p_1^* = 3, p_2^* = 4, p_3^* = 5$.

4,5,6,7 using the schedule 12043 10020 13402 10034 12000 13420 10032 14030 we get $p^* \geq 8$.

4,5,6,7,7 using the schedule 12043 10520 13402 15034 12005 13420 10532 14035 we get $p^* \geq 5$. Any additional periods of value 7 would lead to a negative density.

4,5,7 using the schedule 12030 10020 10302 10003 12000 10320 10002 13000 we get $p^* \geq 7$.

4,5,7,7 using the schedule 12430 10020 14302 10043 12000 14320 10042 13000 we get $p^* \geq 10$.

4,5,7,7,7 to 4,5,7,7,10 using the schedule 12430 15020 14302 15043 12005 14320 10542 13005 we get $p^* \geq 7$. Any additional periods of value 7 would lead to a negative density.

4,5,8 using the schedule 21030 01200 01032 01000 we get $p^* \geq 7$.

4,6 using the schedule 21000 01020 01000 we get $p^* \geq 9$.

4,6,6 follows from $3, 4$.

4,6,7 using the schedule 1020 1030 we get $p^* \geq 6$.

4,6,7,7 using the schedule 10304 10200 13042 10003 12400 10032 14002 we get $p^* \geq 7$. Any other cases involving 4, 6 and multiple periods 7 follow from the cases with 4 and multiple periods 7 below.

4,6,8 using the schedule 21030 01020 01003 21000 01023 01000 we get $p^* \geq 9$.

4,6,8,8 using the schedule 21003 01024 01003 21004 01023 01004 we get $p^* \geq 4$.

4,7 using the schedule 10200 10000 12000 10002 10000 10020 10000 we get $p^* \geq 10$.

4,7,7 and 4,7,8 using the schedule 10203 10000 12030 10002 10300 10020 13000 we get $p^* \geq 9$.

4,7,7,7 to 4,7,8,8 using the schedule 10204 10300 12040 13002 10403 10020 14030 we get $p^* \geq 6$.

4,7,7,7,7 using the schedule 15204 10305 12040 13052 10403 10520 14030 we get $p^* \geq 3$.

4,7,7,7,7,7 using the schedule 15204 10365 12040 13652 10403 16520 14036 we get $p^* \geq 5$. Any additional periods of value 7 would lead to a negative density.

4,7,9 using the schedule 10203 10000 12000 13002 10000 10320 10000 we get $p^* \geq 9$.

4,8 using the schedule 10200 10000 we get $p^* \geq 6$.

5,7 using the schedule 100200 we get $p^* \geq 9$.

5,5,7 and 5,6,7 using the schedule 1020300 we get $p^* \geq 5$. This also solves any case where there are additional periods 5 or 6 before the 7.

5,7,7 follows from case $7, 7$ below. The same is true for any other case involving 5 and multiple periods 7 except the following case.

5,7,7,7,7 using the schedule 503012040 513002041 we get $p^* \geq 5$.

6,7 using the schedule 10002000 we get $p^* \geq 8$.

6,6,7 follows from $3, 7$.

6,7,7 follows from case $7, 7$ below. The same is true for all other cases involving 6 and multiple periods of 7.

7 using the schedule 100000000 we get $p^* \geq 15$.

7,7 using the schedule 100002000 we get $p^* \geq 5$. Any other case involving multiple periods of 7 can be solved with analogous simple schedules. This also solves any case with additional periods 5 or 6 before the 7s.

7,8 using the schedule 120000000 100200000 100002000 100000020 100000002 100000000 we get $p^* \geq 17$.

7,8,8 follows from $4, 7$.

7,8,9 using the schedule 100020030 100000200 130020000 100030200 we get $p^* \geq 12$.

7,8,9,9 to 7,8,9,11 using the schedule 140020030 100040200 130020040 100030200 we get $p^* \geq 7$.

7,8,10 using the schedule 100020030 100000200 130020000 100030200 we get $p^* \geq 15$.

7,8,10,10 using the schedule 140020030 100040200 130020040 100030200 we get $p^* \geq 7$.

7,8,10,11 using the schedule 140020030 100040200 130020040 100030200 we get $p^* \geq 12$.

7,8,10,11,11 using the schedule 140025030 100040205 130020040 105030200 we get $p^* \geq 13$.

7,8,10,11,11,11 and 7,8,10,11,11,12 using the schedule 140025030 106040205 130026040 105030206 we get $p^* \geq 9$.

7,8,10,12 using the schedule 103040200 100000230 140000200 100300240 100003200 we get $p^* \geq 17$.

7,8,10,12,12 and 7,8,10,12,13 solved using the schedule 120000003 150200400 100032005 104000023 100000502 140030000 120500043 100200000 150432000 100000523 140000002 100530004.

7,8,10,12,14 using the schedule 120000053 100200400 100032005 104000023 100000052 140030000 120000543 100200000 100432050 100000023 140000052 100030004 we get $p^* \geq 16$.

7,8,10,12,14,14 and 7,8,10,12,14,15 using the schedule 120000653 100200400 100032605 104000023 100006052 140030000 120006543 100200000 100432650 100000023 140000652 100030004 we get $p^* \geq 12$.

7,8,10,12,15 and 7,8,10,12,16 solved using the same schedules as $7, 8, 10, 12, 14$.

7,8,10,13 and 7,8,10,14 using the schedule 102004003 102000000 102034000 102000003 102004000 102030000 we get $p^* \geq 12$.

7,8,11 using the schedule 100000003 120000000 100203000 100002000 103000020 100000023 100000000 120003000 100200000 103002000 100000023 100000002 100003000 120000000 103200000 100002003 100000020 100003002 100000000 123000000 100200003 100002000 100003020 100000002 103000000 120000003 100200000 100023000 100000200 103000002 we get $p^* \geq 17$.

7,8,11,11 using the schedule 124000003 100200004 100023000 100004200 103000002 104000003 120000004 100203000 100024000 103000200 104000023 100000024 100003002 100004002 103000000 we get $p^* \geq 10$.

7,8,11,12 and 7,8,11,13 solved using the schedule 124000003 100200004 100023000 100004200 103000002 104000003 120000004 100203000 100024000 103000200 104000023 100000024 100003002 100004002 103000000.

7,8,11,14 to 7,8,11,16 using the schedule 100040003 120000000 102043000 100200000 103024000 100020003 100042000 100003200 100040020 103000002 we get $p^* \geq 13$.

7,8,12 using the same schedule as $7, 8, 11$ we get $p^* \geq 17$.

7,8,12,12 using the same schedule as $7, 8, 11, 11$ we get $p^* \geq 18$.

7,8,12,12,12 using the schedule 124050003 100200004 150023000 100004250 103000002 104050003 120000004 150203000 100024050 103000200 104050023 100000024 150003002 100004052 103000000 we get $p^* \geq 18$.

7,8,12,12,12,12 using the schedule 124050003 100260004 150023000 160004250 103000062 104050003 120060004 150203000 160024050 103000260 104050023 100060024 150003002 160004052 103000060 we get $p^* \geq 9$.

7,8,12,12,12,13 to 7,8,12,12,12,17 solved using the schedules for $7, 8, 13$ to $7, 8, 17$.

7,8,12,12,13 to 7,8,12,12,17 solved using the schedules for $7, 8, 13$ to $7, 8, 17$.

7,8,12,13 to 7,8,12,16 solved using the schedules for $7, 8, 13$ to $7, 8, 17$.

7,8,13 to 7,8,15 using the schedule 120000000 100203000 100002000 100003020 100000002 100003000 we get $p^* \geq 11$

7,8,16 and 7,8,17 using the same schedule as above we get $p^* \geq 12$.

7,9 using the schedule 100002000 100000002 100000000 102000000 we get $p^* \geq 15$.

7,9,9 using the schedule 100002000 103000002 100003000 102000003 we get $p^* \geq 7$.

7,9,10 to 7,9,12 using the same schedule as above we get $p^* \geq 11$.

7,9,10,10 using the schedule 140002000 103040002 100003040 102000003 we get $p^* \geq 12$.

7,9,10,10,10 and 7,9,10,10,11 using the schedule 140052000 103040052 100003040 152000003 we get $p^* \geq 11$.

7,9,10,10,10,10 using the schedule 140052060 103040052 160003040 152060003 we get $p^* \geq 8$.

7,9,13 and 7,9,14 using the schedule 120003000 100020000 100003020 100000000 we get $p^* \geq 12$.

7,10 using the schedule 100000002 100000000 100020000 we get $p^* \geq 15$.

7,10,10 and 7,10,11 using the schedule 100030002 100000003 100020000 we get $p^* \geq 9$

7,10,12 using the schedule 100000023 100000000 100200300 100000020 100030000 100200000 103000020 100000003 100200000 100000320 100000000 100230000 100000020 103000000 100200003 100000020 100000300 100200000 100030020 100000000 103200000 we get $p^* \geq 16$.

7,10,12,12 follows from $p_1 = 6$, $p_2 = 7$ and $p_3 = 10$.

7,10,12,13 and 7,10,12,14 using the schedule 100004023 100000000 100204300 100000020 100034000 100200000 103004020 100000003 100204000 100000320 100004000 100230000 100004020 103000000 100204003 100000020 100004300 100200000 100034020 100000000 103204000 100000023 100004000 100200300 100004020 100030000 100204000 103000020 100004003 100200000 100004320 100000000 100234000 100000020 103004000 100200003 100004020 100000300 100204000 100030020 100004000 103200000 we get $p^* \geq 15$.

7,10,12,13,13 to 7,10,12,14,14 using the schedule 100004023 100005000 100204300 100005020 100034000 100205000 103004020 100005003 100204000 100005320 100004000 100235000 100004020 103005000 100204003 100005020 100004300 100205000 100034020 100005000 103204000 100005023 100004000 100205300 100004020 100035000 100204000 103005020 100004003 100205000 100004320 100005000 100234000 100005020 103004000 100205003 100004020 100005300 100204000 100035020 100004000 103205000 we get $p^* \geq 15$.

7,10,12,13,13,13 to 7,10,12,14,14,14 using the schedule 100004023 160005000 100204300 160005020 100034000 160205000 103004020 160005003 100204000 160005320 100004000 160235000 100004020 163005000 100204003 160005020 100004300 160205000 100034020 160005000 103204000 160005023 100004000 160205300 100004020 160035000 100204000 163005020 100004003 160205000 100004320 160005000 100234000 160005020 103004000 160205003 100004020 160005300 100204000 160035020 100004000 163205000 we get $p^* \geq 9$.

7,10,13 solved using the schedule for $7, 13$.

7,10,14 using the schedule 100203000 100000020 100003000 100200000 100003002 100000000 we get $p^* \geq 15$.

7,10,14,14 using the schedule 100203000 100004020 100003000 100204000 100003002 100004000 we get $p^* \geq 6$.

7,11 using the schedule 100002000 100000000 102000000 100000002 100000000 we get $p^* \geq 16$.

7,11,11 using the schedule 103002000 100000003 102000000 100003002 100000000 we get $p^* \geq 16$.

7,11,11,11 using the schedule 103002040 100000003 102040000 100003002 140000000 we get $p^* \geq 16$.

7,11,11,11,11 using the schedule 103002045 100000003 102045000 100003002 145000000 we get $p^* \geq 9$.

7,11,11,11,12 using the same schedule as $7, 11, 11, 11, 11$ we get $p^* \geq 17$.

7,11,11,11,12,12 using the schedule 103002045 160000003 102045060 100003002 145060000 we get $p^* \geq 10$.

7,11,11,11,12,13 to 7,11,12,12,12,15 using the schedule 103002045 100600003 102045000 100603002 145000000 103602045 100000003 102645000 100003002 145600000 we get $p^* \geq 10$.

7,11,11,11,12,16 using the schedule 103002045 100600003 102045000 100603002 145000000 103602045 100000003 102645000 100003002 145600000 we get $p^* \geq 18$.

7,11,11,11,12,16,16 and 7,11,11,11,12,16,17 using the schedule 103702045 100600003 102745000 100603002 145700000 103602045 100700003 102645000 100703002 145600000 we get $p^* \geq 9$.

7,11,11,11,13 using the schedule 103002040 100500003 102040000 100503002 140000000 103502040 100000003 102540000 100003002 140500000 we get $p^* \geq 10$.

7,11,11,11,14 and 7,11,11,11,15 using the same schedule as $7, 11, 11, 11, 13$ we get $p^* \geq 10$.

7,11,11,11,16 using the same schedule as $7, 11, 11, 11, 13$ we get $p^* \geq 18$

7,11,11,11,16,16 using the schedule 103602040 100500003 102640000 100503002 140600000 103502040 100600003 102540000 100603002 140500000 we get $p^* \geq 9$.

7,11,11,12 using the same schedule as $7, 11, 11, 11$ we get $p^* \geq 17$.

7,11,11,12,12 follows from $6, 7, 11$.

7,11,11,12,13 and 7,11,11,12,14 using the schedule for $7, 11, 11, 11, 13$ we get $p^* \geq 10$.

7,11,11,12,15 and 7,11,11,12,16 The first case is solved using the same schedule as $7, 11, 11, 11, 13$ in the second case we get $p^* \geq 18$.

7,11,11,12,16,16 and 7,11,11,12,16,17 using the same schedule as $7, 11, 11, 11, 16, 16$ we get $p^* \geq 9$.

7,11,11,13 using the schedule for case $7, 13$ we get $p^* \geq 10$ which solves this case.

7,11,11,14 and 7,11,11,15 using the schedule 103042000 100000003 102040000 100003002 100040000 103002000 100040003 102000000 100043002 100000000 we get $p^* \geq 10$.

7,11,12 using the same schedule as $7, 11, 11$ we get $p^* \geq 17$.

7,11,12,12 follows from $6, 7, 11$.

7,11,12,13 using the schedule for case $7, 13$ we get $p^* \geq 10$ which solves this case.

7,11,12,14 to 7,11,12,16 using the schedule 103042000 100000003 102040000 100003002 100040000 103002000 100040003 102000000 100043002 100000000 we get $p^* \geq 18$.

7,11,12,14,14 to 7,11,12,16,17 using the schedule 103042000 100500003 102040000 100503002 100040000 103502000 100040003 102500000 100043002 100500000 we solve cases $7, 11, 12, 14, 14$ and $7, 11, 12, 14, 15$ and in the other cases we get $p^* \geq 17$.

7,11,12,14,16,16 to 7,11,12,16,16,16 These cases can be solved using the schedule 103042000 100500603 102040000 100503602 100040000 103502600 100040003 102500600 100043002 100500600.

7,11,13 using the schedule for case $7, 13$ we get $p^* \geq 10$ which solves this case.

7,11,14 and 7,11,15 using the schedule 100302000 100000000 102300000 100000002 100300000 100002000 100300000 102000000 100300002 100000000 we get $p^* \geq 17$.

7,11,14,14 to 7,11,15,16 using the schedule 100302000 100400000 102300000 100400002 100300000 100402000 100300000 102400000 100300002 100400000 solves all cases except $7, 11, 15, 16$ where we get $p^* \geq 17$.

7,11,15,16,16 using the schedule 100302050 100400000 102300050 100400002 100300050 100402000 100300050 102400000 100300052 100400000 we get $p^* \geq 16$.

7,12 using the schedule 100000002 100000000 100000200 100000000 100020000 100000000 102000000 we get $p^* \geq 17$.

7,12,12 follows from $6, 7$.

7,12,13 using the schedule for case $7, 13$ we get $p^* \geq 10$ which solves this case.

7,12,14 using the schedule 100003002 100000000 100003200 100000000 100023000 100000000 102003000 100000002 100003000 100000200 100003000 100020000 100003000 102000000 we get $p^* \geq 18$.

7,12,14,14 to 7,12,14,17 solved using the schedule 100003002 100004000 100003200 100004000 100023000 100004000 102003000 100004002 100003000 100004200 100003000 100024000 100003000 102004000.

7,12,15 and 7,12,16 using the schedule 100030002 100000000 100000230 100000000 100020000 130000000 102000000 we get $p^* \geq 18$.

7,12,15,15 to 7,12,16,17 using the schedule 100030002 100040000 100000230 100000040 100020000 130000000 142000000 we solve all cases except $7, 12, 16, 17$ where we get $p^* \geq 18$

7,12,16,17,17 using the schedule 100030002 105040000 100000230 100005040 100020000 130000005 142000000 we get $p^* \geq 17$.

7,13 using the schedule 100002000 100000000 we get $p^* \geq 10$.

7,14 using the same schedule as above we get $p^* \geq 18$.

7,14,14 to 7,14,17 solved using the schedule 100002000 100003000.

□

2 A case verification program for discrete BGT

In this section we give an overview over the program used in the proof of the $\frac{7}{5}$-approximation for discrete BGT, beginning with the input.

```
#include <iostream>
#include <math.h>
#include <list>
#include <vector>
#include <string>
using namespace std;


//INPUT:
//

float factor = 7.0 / 5.0;
int periodarray[] = { 3,7,7 };
string schedulestring = "1020␣1030";
```

```
//
/////////////////////////
```

After the input section there are two assisting functions. The first one tests the validity of a schedule and the second converts the schedule from a string into a vector.

```
void testschedule(vector<int> schedulevector, vector<int>
   periodvector){
        vector<int> dists;
        for (int j = 0; j < periodvector.size(); j++) {
                dists.push_back(0);
                float f = factor*periodvector[j];
                periodvector[j] = floorf(f);
        }
        int counter = 0;
        for (int i = 0; i < 2 * schedulevector.size(); i++) {   //We
            go through the schedule twice and measure the distance
           inbetween occurences of each period
                for (int j = 0; j < periodvector.size(); j++) {
                        dists[j]++;
                        if (schedulevector[counter] == j+1) {
                                dists[j] = 0;
                        }
                        if (dists[j] >= periodvector[j]) {
                                cout << "Invalid␣schedule!" << j <<
                                   "␣";
                                cin.get();
                        }
                }
                counter = counter+1;
                counter = counter % schedulevector.size();
        }
}


void convert(vector<int> * emptyvector) {
        for (int i = 0; i < schedulestring.length(); i++) {
                if (schedulestring[i] != '␣') {
                        emptyvector->push_back(stoi(schedulestring.
                           substr(i, 1)));
                }
        }
}
```

The next function determines the minimum number of holes in p consecutive positions in the schedule, i.e determines h for a given p.

```
int countholes(vector<int> schedulevector, int p) {
        //Determine the overall number of holes in the schedule
        int overallholes = 0;
        for (int i = 0; i < schedulevector.size(); i++) {
                if (schedulevector[i] == 0) {
                        overallholes++;
                }
        }
        int c = p / schedulevector.size();
        int d = p % schedulevector.size();
        //p=c*schedulesize+d
        //Determine the minimum number of holes in d consecutive positions
        int start = 0;
        int min = schedulevector.size();
        while (start < schedulevector.size()) {
                int holesinsection = 0;
                int counter = start;
                for (int i = 0; i < d; i++) {
                        counter = counter+1;
                        counter = counter % schedulevector.size();
                        if (schedulevector[counter] == 0) {
                                holesinsection ++;
                        }
                }
                if (holesinsection < min) {
                        min = holesinsection;
                }
                start++;
        }
        return c*overallholes+min;
}
```

In the main part we iterate over p^*, trying to find a sufficiently large set of consecutive periods where $\frac{h}{p} \geq D_a$ holds.

```
int main() {
        //Converting the input into vectors
        vector<int> schedulevector;
        convert(&schedulevector);
        vector<int> periodvector(std::begin(periodarray), std::end(periodarray));

        //Is this schedule valid?
        testschedule(schedulevector, periodvector);
        //Calculate D_A
        float DA = 1.0;
        for (int i = 0; i < periodvector.size(); i++) {
                DA = DA - 1.0 / periodvector[i];
        }
        cout << "DA␣" << DA << "␣";
        cin.get();

        //Determine a suitable amount of iterations and check h/p \
           geq D_A for these periods
        int iterations = schedulevector.size() * 5 + 100;
        int pstar = 0;
        for (int i = 1; i < iterations; i++) {
                float f = factor*i;
                int p = floorf(f);
                int h = countholes(schedulevector, p);
                float temp = h;
                temp = temp / p;
                temp = temp - DA;
                if (temp < 0) {
                        pstar = i;
                }
                cout << "Iteration:␣" << i << endl;
        }
        cout << "Result:␣p*␣>=␣" << pstar+1;
        cin.get();
}
```