

1 Supplementary material

2 Latex and python codes of the Figures in our manuscript "Selection pressure by specialist and
3 generalist insect herbivores leads to optimal constitutive plant defense. A mathematical model".

4 1 Latex code of Figure 1

```
5
6 % Import the "tikz" package in the Latex file
7 \usetikzlibrary{arrows.meta}
8 \tikzset{%
9   >={Latex[width=2mm,length=2mm]},
10  % Specifications for style of nodes:
11     base/.style = {rectangle, rounded corners, draw=black,
12                   minimum width=5cm, minimum height=1cm,
13                   text centered, font=\sffamily},
14  % Color and width of different rectangular boxes
15  activityStarts1/.style = {base, minimum width=5cm, fill=magenta!30},
16  activityStarts2/.style = {base, minimum width=5cm, fill=cyan!40},
17  startstop/.style = {base, minimum width=5cm, fill=red!30},
18  activityRuns/.style = {base, minimum width=5cm, fill=green!30},
19  process1/.style = {base, minimum width=6cm, fill=orange!50},
20  process2/.style = {base, minimum width=6cm, fill=blue!30},
21  process3/.style = {base, minimum width=5cm, fill= orange!50},
22  suman/.style = {base, minimum width=5cm, fill=white!30},
23  papai/.style = {base, minimum width=5cm, fill=Gray!30}
24 }
25 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
26
27 % Begin code with \begin{tikzpicture}
28 [node distance=1.5cm,
29 every node/.style={fill=white, font=\sffamily}, align=center]
30 \node[inner sep=0pt] (L) at (12,9) []{\includegraphics[width=0.4\columnwidth]{Leaves.png}};
31 \draw[-, thick] (8.6,8.2) -- (8.6,11.5) -- (15.3,11.5)--(15.3,8.2) -- (8.6,8.2);
32
33 \node[inner sep=0pt] (G) at (26.5,9) []{\includegraphics[width=0.4\columnwidth]{generalists.png}};
34 % generalists.png is a image file, created seperately.
35
36 % Draw the arrows
37 \draw[->, thick] (15.3, 9.85) -- (23.1, 9.85)
38
39 % Insert text in the middle of the arrows
40 node[midway,fill= white] {\large Low constitutive defense};
41
42 % Draw the border by line segments
43 \draw[-, thick] (23.1,7.7) -- (23.1,11.5) -- (29.8,11.5) -- (29.8,7.7) -- (26.8,7.7)-- (26.5,7)
```

```

44 -- (26.2,7.7) --(23.1,7.7);
45
46 % Colorful rectangular box with text
47 \node[inner sep=0pt] (T1) at (26.5,6.5) [process2]{\large High total herbivore pressure};
48
49 \node[inner sep=0pt] (S) at (12,3) []{\includegraphics[width=0.4\columnwidth]{specialists.png}};
50 % specialists.png is a image file, created seperately.
51
52 \draw[->, thick] (11.9, 8.2) -- (11.9, 5.5)
53 node[midway,fill= white] {\large High constitutive defense};
54 \draw[-, thick] (8.6,1.7) -- (8.6,5.5) -- (15.3,5.5) -- (15.3,1.7) -- (12.3,1.7) -- (12,1)
55 -- (11.7,1.7) -- (8.6,1.7);
56 \node[inner sep=0pt] (T2) at (12,0.5) [process2]{\large High total herbivore pressure};
57
58 \node[inner sep=0pt] (O) at (26.5,3) []{\includegraphics[width=0.4\columnwidth]{opt-gen-spec.png}};
59 % opt-gen-spec.png is a image file, created seperately.
60
61 \draw[->, thick] (15.3,8.2) -- (23.1,3.6)
62 node[midway,fill= white] {\large Optimal constitutive defense};
63 \draw[-, thick] (23.1,1.7) -- (23.1,5.5) -- (29.8,5.5) -- (29.8,1.7) -- (26.8,1.7) -- (26.5,1)
64 --(26.2,1.7) -- (23.1,1.7);
65 \node[inner sep=0pt] (T3) at (26.5,0.5) [process1]{\large Low total herbivore pressure};
66
67 % End the code with \end{tikzpicture}.
68 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
69
70 # Instruction: Compile the given latex code in Overleaf.

```

2 Python code of Figure 2

```

71
72
73 #Import packages
74 import numpy as np
75 from scipy.integrate import odeint
76 import matplotlib.pyplot as plt
77 #####
78
79 # Define the ODEs and write "t" instead of "D" as independent variable, because odeint by default
80 take "t" #
81 def deriv(y, t, alpha, beta, gamma, eta, mu):
82     S, G, N = y
83     dSdt = alpha * t - mu * S * N # Attracted specialist population S
84     dGdt = - beta * G * t - eta * G * N # Deterred generalist population G
85     dNdt = gamma # Attracted natural enemies N
86
87     return dSdt, dGdt, dNdt
88
89 # Insert the parameter values
90 alpha = 0.04
91 beta = 0.05
92 gamma = 0.1

```

```

93 mu = 0.02
94 eta = 0.03
95
96 # Insert the initial conditions
97 S0, G0, E0 = 0.0, 20.0, 0.0
98 t = np.linspace(0, 100, 1500) # Grid of D
99 y0 = S0, G0, E0 # Initial conditions vector
100
101 # Integrate the equations over the time grid, t.
102 ret = odeint(deriv, y0, t, args=(alpha, beta, gamma, eta, mu))
103 S, G, E = ret.T
104
105 # Define the plots of S, G and N
106 def plotdc(t, S, G, E):
107     f, ax = plt.subplots(1,1,figsize=(8,2), dpi = 300)
108     ax.plot (t, S, 'darkcyan', alpha=0.9, linewidth=3.5, label='Specialist ($S$)')
109     ax.plot (t, G, 'tomato', alpha=0.9, linewidth=3.5, label='Generalist ($G$)')
110     ax.plot (t, E, 'orange', alpha=0.9, linewidth=3.5, label='Natural enemies ($N$)')
111
112 # Assign values on x and y axes
113 plt.tick_params(axis='y', labelsize=13)
114 plt.tick_params(axis='x', labelsize=13)
115
116 # Label on x and y axes
117 plt.xlabel('Plant defense ($D$)', {"fontsize": 13})
118 plt.ylabel('Populations', {"fontsize": 13})
119
120 # Label the variables (S, G and N) on the top of the Figure
121 legend = ax.legend()
122 legend.get_frame().set_alpha(0.5)
123 plt.legend( bbox_to_anchor=(0., 1.02, 1., .102), loc='lower left',
124            ncol=2, mode="expand", borderaxespad=0., fontsize=13 )
125
126 #Boundary
127 for spine in ('top', 'right', 'bottom', 'left'):
128     ax.spines[spine].set_visible(True)
129
130 plt.show();
131
132 plotdc(t, S, G, E) # Final figure visualization
133 #####
134
135 Instruction: Save the code as "Filename.py" and run the python code by the command
136 "python3 Filename.py" in your Linux (Ubuntu) terminal.
137

```

138 3 Python code of Figure 3

```

139 # Import packages
140 import numpy as np
141 import matplotlib.pyplot as plt

```

```

142 import math
143 #####
144
145 # Evenly sampled D at intervals
146 t = np.arange(0.0, 100.0, 0.02)
147
148 # Insert the parameter values
149 theta = 0.002
150 delta =0.003
151 beta = 0.05
152 alpha = 0.04
153 G_0 = 20
154
155 # Define the plot of Total herbivores (T_H)
156 f, ax = plt.subplots(1,1,figsize=(8,2), dpi = 300)
157 ax.plot(t, alpha/theta * (1 - np.exp(- theta * t * t)) + G_0 * np.exp(- (beta + delta) * t * t),
158 'steelblue', alpha=0.9, linewidth=3.5, label='$T_H$')
159
160 # Assign values on x and y axes
161 plt.tick_params(axis='y', labels=13)
162 plt.tick_params(axis='x', labels=13)
163
164 # Label on x and y axes
165 plt.xlabel('Plant defense ($D$)', {"fontsize": 13})
166 plt.ylabel('$T_H$', {"fontsize": 13})
167
168 # Label T_H inside the Figure
169 legend = ax.legend()
170 legend.get_frame().set_alpha(0.5)
171 plt.legend( bbox_to_anchor=(0.78, 0.05, 0.18, 0.1), loc='lower left',
172           ncol=1, mode="expand", borderaxespad=0, fontsize=13 )
173
174 #Boundary
175 for spine in ('top', 'right', 'bottom', 'left'):
176     ax.spines[spine].set_visible(True)
177
178 plt.show(); # Final figure visualization
179 #####
180
181 Instruction: Save the code as "Filename.py" and run the python code by the command
182 "python3 Filename.py" in your Linux (Ubuntu) terminal.
183

```

184 4 Python code of Figure 4

```

185 # Import packages
186 import numpy as np
187 import matplotlib.pyplot as plt
188 import math
189 #####
190

```

```

191 # Evenly sampled D at intervals
192 t = np.arange(0.0, 100.0, 0.02)
193
194 # Insert the parameter values
195 alpha = 0.05
196 theta = 0.0025
197 delta =0.001
198 beta = 0.001
199 G_0 = 20
200
201 # Define the plot of Total herbivores (T_H)
202 f, ax = plt.subplots(1,1,figsize=(8,2), dpi = 300)
203 ax.plot(t, alpha/theta * (1 - np.exp(- theta * t * t)) + G_0 * np.exp(- (beta + delta) * t * t),
204 'indianred', alpha=0.9, linewidth=3.5, label='$T_H$')
205
206 # Assign values on x and y axes
207 plt.tick_params(axis='y', labelsz=13)
208 plt.tick_params(axis='x', labelsz=13)
209
210 # Label on x and y axes
211 plt.xlabel('Plant defence ($D$)', {"fontsize": 13})
212 plt.ylabel('$T_H$', {"fontsize": 13})
213
214 # Label T_H inside the Figure
215 legend = ax.legend()
216 legend.get_frame().set_alpha(0.5)
217 plt.legend(fontsize=13)
218
219 #Boundary
220 for spine in ('top', 'right', 'bottom', 'left'):
221     ax.spines[spine].set_visible(True)
222
223 plt.show(); # Final figure visualization
224 #####
225
226 Instruction: Save the code as "Filename.py" and run the python code by the command
227 "python3 Filename.py" in your Linux (Ubuntu) terminal.
228

```

229 5 Python code of Figure 5

```

230 # Import packages
231 import numpy as np
232 import matplotlib.pyplot as plt
233 import math
234 #####
235
236 # Evenly sampled D at intervals
237 t = np.arange(0.0, 100.0, 0.02)
238
239 # Insert the parameter values

```

```

240 theta = 0.002
241 delta =0.003
242 beta = 0.05
243
244 # Define the plot of Total herbivores (T_H)
245 f, ax = plt.subplots(1,1,figsize=(8,2), dpi = 300)
246 ax.plot(t, 20 * (1 - np.exp(- theta * t * t)) + 30 * np.exp(- (beta + delta) * t * t), 'olive',
247 alpha=0.9, linewidth=3.5, label='$\u03B1/\u03B8 < G_0$')          # alpha/theta = 20, G_0 = 30
248 ax.plot(t, 30 * (1 - np.exp(- theta * t * t)) + 20 * np.exp(- (beta + delta) * t * t), 'peru',
249 alpha=0.9, linewidth=3.5, label='$\u03B1/\u03B8 > G_0$')          # alpha/theta = 30, G_0 = 20
250
251 # Assign values on x and y axes
252 plt.tick_params(axis='y', labelsz=13)
253 plt.tick_params(axis='x', labelsz=13)
254
255 # Label on x and y axes
256 plt.xlabel('Plant defense ($D$)', {"fontsize": 13})
257 plt.ylabel('$T_H$', {"fontsize": 13})
258
259 # Label T_H inside the Figure
260 legend = ax.legend()
261 legend.get_frame().set_alpha(0.5)
262 plt.legend( bbox_to_anchor=(0.66, 0.05, 0.3, 0.1), loc='lower left',
263            ncol=1, mode="expand", borderaxespad=0, fontsize=13 )
264
265 #Boundary
266 for spine in ('top', 'right', 'bottom', 'left'):
267     ax.spines[spine].set_visible(True)
268
269 plt.show();                                     # Final figure visualization
270 #####
271
272 Instruction: Save the code as "Filename.py" and run the python code by the command
273 "python3 Filename.py" in your Linux (Ubuntu) terminal.
274

```

275 6 Python code of Figure 6

```

276 # Import packages
277 import numpy as np
278 import matplotlib.pyplot as plt
279 import math
280 #####
281
282 # Evenly sampled D at intervals
283 t = np.arange(0.0, 100.0, 0.02)
284
285 # Insert the parameter values
286 alpha = 0.04
287 beta = 0.05
288 # Different values of theta

```

```

289 theta1 = 0.002
290 theta2 = 0.003
291 theta3 = 0.01
292 theta4 = 0.008
293 # Different values of delta
294 delta1 = 0.003
295 delta2 = 0.002
296 delta3 = 0.005
297 delta4 = 0.001
298
299 # Define the plot of Total herbivores (T_H) for different values of theta and delta
300 f, ax = plt.subplots(1,1,figsize=(8,2), dpi = 300)
301 # G_0 = 20
302 ax.plot(t, alpha/theta1 * (1 - np.exp(- theta1 * t * t)) + 20 * np.exp(- (beta + delta1) * t * t),
303 'y', alpha=0.9, linewidth=3.5, label='$G_0 = 20, \u03B8 = 0.002, \u03B4 = 0.003$')
304 # G_0 = 30
305 ax.plot(t, alpha/theta2 * (1 - np.exp(- theta2 * t * t)) + 30 * np.exp(- (beta + delta2) * t * t),
306 'b', alpha=0.6, linewidth=3.5, label='$G_0 = 30, \u03B8 = 0.003, \u03B4 = 0.002$')
307 # G_0 = 10
308 ax.plot(t, alpha/theta3 * (1 - np.exp(- theta3 * t * t)) + 10 * np.exp(- (beta + delta3) * t * t),
309 'm', alpha=0.7, linewidth=3.5, label='$G_0 = 10, \u03B8 = 0.01, \u03B4 = 0.005$')
310 # G_0 = 25
311 ax.plot(t, alpha/theta4 * (1 - np.exp(- theta4 * t * t)) + 25 * np.exp(- (beta + delta4) * t * t),
312 'g', alpha=0.7, linewidth=3.5, label='$G_0 = 25, \u03B8 = 0.008, \u03B4 = 0.001$')
313
314 # Assign values on x and y axes
315 plt.tick_params(axis='y', labelsize=13)
316 plt.tick_params(axis='x', labelsize=13)
317
318 # Label on x and y axes
319 plt.xlabel('Plant defence ($D$)', {"fontsize": 13})
320 plt.ylabel('$T_H$', {"fontsize": 13})
321
322 # Set legend on the top of the Figure
323 legend = ax.legend()
324 legend.get_frame().set_alpha(0.5)
325 plt.legend( bbox_to_anchor=(0.0, 1.02, 0.63, 0.1), loc='lower left',
326           ncol=1, mode="expand", borderaxespad=0., fontsize=12 )
327
328 #Boundary
329 for spine in ('top', 'right', 'bottom', 'left'):
330     ax.spines[spine].set_visible(True)
331
332 plt.show(); # Final figure visualization
333 #####
334
335 Instruction: Save the code as "Filename.py" and run the python code by the command
336 "python3 Filename.py" in your Linux (Ubuntu) terminal.
337

```