```
For each generation do {
    + For each wasp of the population do {
        • Build a new chromosome of 3 genes: G1, G2, G3 (real numbers);    // each wasp holds a chromosome
        • Compute the plasticity cost (using G2) and modify subsequently the initial coordinates of the
          wasp in the fecundity-lifespan trade-off;
        • Repeat 20 times{                                                 // each wasp is tested 20 times
          • Decide whether the wasp is born inside a patch or not;
          • while (alive = TRUE){
            If (inpatch = TRUE){                                           // the wasp is on a patch
              o compute the number N0 host ->𝒩(NB_EGGS,SD) of the patch;
              o compute the optimal time to leave the patch: Tleave = Topt*N0/NB_EGGS; //Marginal value
                Theorem
              o nbpontes_old = 0.0;
              o patch_timer  = 0;
                             // The Linear estimator is updated every time step
              o while (Tleave - patch_timer > 0.0 )AND(eggs_laid < fecundity_limit)AND
                        (lifespan - Cur_LifeTime > 0) {
                ▪ patch_timer++  ;                                          // time spent on the patch
                ▪ Cur_LifeTime++ ;                                          // wasps' age
                ▪ eggs_laid_new = Nt = N0*(1-exp(-taux*patch_timer/N0));   //Number of eggs laid
                ▪ delta_clutch      = eggs_laid_new - eggs_laid_old;
                ▪ Update λti= delta_clutch and compute μt = λt*G3 + (1-G3)*μt-1;
                ▪ Modify the position in the fecundity-lifespan trade-off;
                ▪ eggs_laid_old  = eggs_laid_new;
                ▪ clockStop      = patch_timer;
                ▪ eggs_laid     += delta_clutch;
              }EndWhile

              o if (nbpontes >= fecundity_limit)OR(Cur_LifeTime >= lifespan) alive = FALSE; //end of life
                else
                    // the out patch travel time is greater than the rest of its life
                    // the wasp stays on the patch and attempts to find further hosts
                  if (lifespan - Cur_LifeTime) < out_pa_ttime) AND (nbpontes < lim_fec){
                   do{
                     ▪ patch_timer++;
                     ▪ Cur_LifeTime++;
                     ▪ eggs_laid_new = Nt = N0*(1-exp(- taux*patch_timer/N0));
                     ▪ delta_clutch  = eggs_laid_new - eggs_laid_old;
                     ▪ if ((patch_timer - clockStop) = 0){
                         • Update λti= delta_pontes and compute μt = λt*G3 + (1-G3)*μt-1;
                         • Modify the position in the fecundity-lifespan trade-off;
                       }EndIf
                     ▪ eggs_laid_old = eggs_laid_new;
                     ▪ eggs_laid    += delta_clutch;
                    }while(eggs_laid < fecundity_limit) AND ((lifespan - Cur_LifeTime) > 0);
                   alive = FALSE;
                  }EndElseIfDo
              else inpatch = FALSE;                                        // the wasp leaves the patch
              }
            else {                                                         // the wasp is out of a patch
              out_patch_timer = 0;
              while(lifespan - cur_LifeTime > 0)AND(out_patch_time - out_patch_timer > 0) {
                  cur_LifeTime++ ;
                  out_patch_timer++ ;
                  Update λti= 1/(clock_time_out_patch) and compute μt = λt*G3 + (1-G3)*μt-1;
                  Modify the position in the fecundity-lifespan trade-off;
              }EndWhile
              residual_time  = out_patch_time - out_patch_timer;
              cur_LifeTime  += residual_time;
              clock_time_out_patch += residual_time;
              inpatch = TRUE;     // the wasp will meet a new patch
              if(cur_LifeTime >= lifespan)OR(eggs_laid >= fecundity_limit) alive = FALSE;//end of life
              }
          }EndWhile (alive = TRUE)
        }EndRepeat
    }EndFor (wasp)
    + Compare the scores (i.e. the number of eggs laid) and select the n best of them as genitors;
    + With genitors' chromosomes produce n offsprings, applying mutation and crossing-over procedures
    + Replace the worst score owners by the progeny;
}EndFor (generation)
```

Fig. SD1. Simplified algorithm (in the C style) of the numerical model. NB_EGGS = average number of eggs per patch in the environment; SD = standard deviation; Tleave = time to leave the current patch; Topt = optimal time to leave a patch within an environment of NB_EGGS (Marginal Value Theorem); λ_t, μ_t = parameters of the linear estimator of the richness of the environment (prior and posterior respectively). The number of generation and the population size are 300 and 100 respectively.

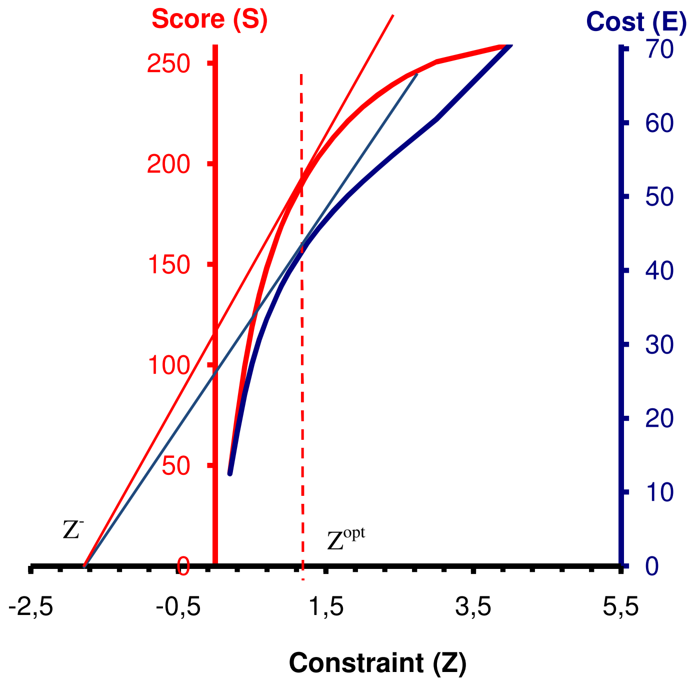

Fig. SD2. Scores and costs (coherent scales) in the space of the constraint *Z*. From figure 6e of the main paper we have at the minimum of the (*E*/*S*) ratio: $\frac{d(E/S)}{d(Z)}(Z^{opt}) = 0$.

According to the derivative rule, it follows that $E'(Z^{opt}).S(Z^{opt}) = E(Z^{opt}).S'(Z^{opt})$ and then

$$\frac{E'(Z^{opt})}{E(Z^{opt})} = \frac{S'(Z^{opt})}{S(Z^{opt})}, \qquad (1)$$

where Z^{opt} is the corresponding abscissa. Next, from the graph above, we ascertain that the two tangent lines at $Z = Z^{opt}$ ($\approx$1.25) cross the horizontal axis at the same abscissa Z^-.

Proof:

$S'(Z^{opt}) = \frac{S(Z^{opt})}{(Z^{opt} - Z_1)}$ and $E'(Z^{opt}) = \frac{E(Z^{opt})}{(Z^{opt} - Z_2)}$.

Thus, $\frac{S'(Z^{opt})}{S(Z^{opt})} = \frac{E'(Z^{opt})}{E(Z^{opt})}$ (Eqn (1)) is true iff $Z_1 = Z_2 = Z^-$.

Tangent lines (*S* =58.022(Z-1.25) + 197.965 and *E* = 13.486(Z-1.25) + 44.367) were obtained from the derivatives of polynomial regressions (3rd order) on score and cost respectively, in the vicinity of Z^{opt}. Equation (1) indicates that in the vicinity of Z^{opt}, a variation of *Z* corresponds to a variation of the score *S* which induces an almost proportional variation of the cost *E* in such a way that the gains (or losses) are nearly negligible. Consequently, Z^{opt} defines a pseudo-equilibrium. We interpret $|Z^-|$ as the energy that should be invested into the system to shift from a non plastic state to the ideal plasticity to face environment fluctuations.

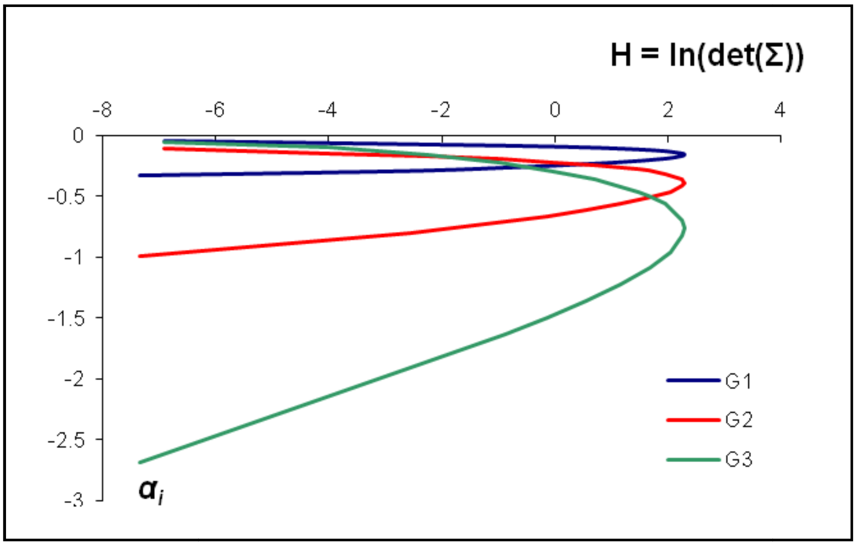

Fig. SD3. Parameters α_i of the genes' response functions versus H value. Maximum of H indicates the best values of the three parameters in the space of the constraint Z ({0.151, 0.374, 0.724} respectively for G1, G2 and G3. Small variations were observed, depending on the type of fluctuation we tested (see the paper for more details).

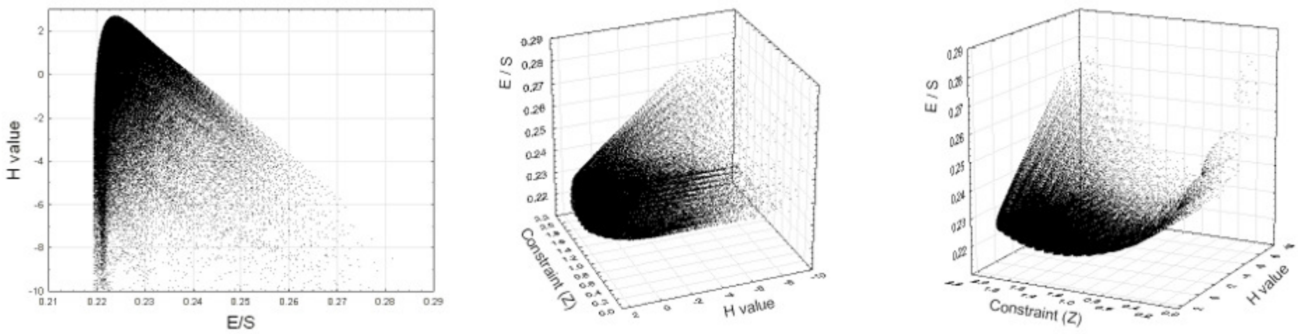

Fig. SD4. Numerical experiments. In addition to the Lagrange's constrained optimisation method, we performed some numerical experiments. Few results of these computations are shown here. 124,491 combinations of the 3 parameters α_i (steps = 0.01) in the space of Z ($Z \in [0.2; 2.0]$; step = 0.01) were generated. Next, H = ln[det(Σ)] and the *E*/*S* ratio generated by each combination were numerically computed (periodic environment variation (*g*)). **Left**. H value versus Cost/Score (*E*/*S*). One can see from this graph that H admits a single maximum (H ≈ 2.28). This maximum confirms the value we found (*E*/*S* ≈ 0.224) by means of the Lagrange's constrained optimisation method. Identical results were obtained using either the uniform or the Gaussian variations of the environment. **Right**. Two points of view of the Cost/Score (*E*/*S*) ratio versus H value and Constraint (*Z*).

We briefly recall the idea of the MaxEnt method. Let us assume that $p(x)$ is an unknown distribution function of a multidimensional random variable X and suppose we want to determine $p(x)$. Let us assume that we have only single information about it, say the average value $\overline{A_k}$:

$$\overline{A_k} = \int A_k(x)p(x)dx\,, \tag{1}$$

where, $k = 1,..,N$, where N is the number of constraints, and $\int p(x)dx = 1$. (2)

Equations (1) and (2) are insufficient to determine the distribution of $p(x)$. Jaynes showed that the most objective (i.e., unprejudiced) method consists of maximizing the informational entropy:

$$H = -\int p(x)\log(p(x))dx \tag{3}$$

H can be maximized by means of a standard method (Lagrange's constrained optimization method) using both condition (1) and constraint (2). The maximization leads to the results:

$$p(x) = \frac{1}{T}\exp\left\{-\sum_k \lambda_k A_k(x)\right\}, \tag{4}$$

with $T = \int \exp\left\{-\sum_k \lambda_k A_k(x)\right\}dx$. (5)

λ_k are the so-called Lagrange multipliers that can be determined from Eq. 1.

Fig. SD5. The Maximum Informational Entropy (MaxEnt) method.