# Supplementary Information

Paul François[1*], Vincent Hakim[2*], Eric D. Siggia[1*]

October 15, 2007

[1] Center for studies in Physics and Biology, The Rockefeller University ,
1230 York Avenue, New-York NY 10065, USA.
[2] Laboratoire de Physique Statistique, Ecole Normale Supérieure,
24 rue Lhomond, 75231 Paris Cedex 05, France.

[*] Order is alphabetical.

## 1 Evolution in silico of segmentation networks

We provide here further details on the evolutionary algorithm described in the main text. It is similar to the algorithm of ref. [1].

### 1.1 Evolution

We model an embryo as a one-dimensional array of cells (typically 100 or 200 cells depending on the simulations) sharing the same "genome". Cells are indexed by their position $x$ in the array. The dynamics of the morphogen G is imposed on these cells. For simulations with static gradients, the concentration of G depends only on the position $x$ and is defined as:

$$G(x,t) = G(x) = \exp(0.05x). \tag{1}$$

For dynamic segmentation, the concentration of the morphogen G is imposed and depends both on the cell position $x$ and on time $t$ in the following way :

$$G(x,t) = \min[20\exp(0.15x - 0.6t), 500]. \tag{2}$$

This expression mimics the formation of a signalling gradient upon exit from a growth zone where the morphogen concentration is held constant and equal to 500. The cell at position $x$ exits from the growth zone when $20\exp(0.15x - 0.6t) = 500$, i.e. when $t = x/4 - 5\ln(25)/3$, so that the time of exit is proportional to the position of the cell. Then, the morphogen decays with a fixed degradation rate (chosen to be 0.6 in arbitrary units). Note that the definition of G imposes both a time and a length scale.

Each network includes a gene E which acts as a reporter gene. The expression profile of E serves to score the networks, as precisely described below. The initial networks at the start of an evolution run simply consist in the morphogen G and the reporter E with no interaction between the two.

Network behaviors are simulated with delayed-differential equations (DDEs) which are integrated in each cell of the array. The dynamics of individual cells differ due to differences in the time course of the concentration of morphogen G they experience.

The initial concentrations of all proteins other than the morphogen are zero. Therefore, the cells in an embryo are initially all in the same state (or phase when it is possible to define an oscillator). After a long time (chosen here to be 60 time units), the fitness function is computed from the states that the cells have reached, by counting the number of boundaries in the profile between low and high states of E, as detailed below.

The evolutionary process works on a collection of such networks (typically 100). At each step of the algorithm, for each network, the scoring or fitness function is computed. Networks are then ordered with respect to their fitness with a random order for networks of equal scores (the fitness is discrete). The least fit half of networks are discarded, the top half is duplicated, and the duplicated copies are mutated.

There are two kinds of mutation:

- mutations that do not change network topology. The possible mutations in this category are random modifications of parameters : transcriptional maximum activity, thresholds of activation and repression, Hill coefficients, delays, degradation constants.
- mutations that change network topology. The possible mutations are creation of new genes, creation of new activating or repressing regulation on existing genes, as well as removal of existing genes, or regulations. Either we add a gene along with a randomly chosen transcriptional regulator and target gene for it to regulate, or we add just the gene and allow independent mutations to link it into the transcriptional network. We assume that the rates of creation of new genes or new regulations is slightly higher than the rates of removal, so that the network is slowly growing.

Once the mutated networks have been produced, the process is iterated.

The mutation rates for all these processes were fixed for each run of the evolutionary simulations. Typically, the relative probability to modify a numerical parameter was roughly ten times higher than the probability to change network topology. Varying mutation rates does not appear to significantly change the evolution outcome but modifies the evolution speed ( see for instance Figs. S10 for a network obtained with comparable probabilities of changing network topologies and kinetic constants ).

Finally, as for the previous algorithm [1], evolutionary computations produce irrelevant interactions which do not affect the network fitness (“code bloat” phenomenon). In all figures, we present the best final network of the population, in which the irrelevant interactions have been automatically pruned. Note that pruning is just a convenient way to display the core working networks and is never done during the simulations.

## 1.2 Fitness

The fitness function used for selection is a key ingredient of the evolutionary computation. Several fitness functions were tried to obtain segmented patterns. After some preliminary tests, it appeared that too strict a fitness function, imposing for instance a pre-defined profile $E_0(x)$, and trying to minimize $\int(|E(x) - E_0(x)|)^{\alpha}\mathrm{d}x$, would not work efficiently. The best network obtained with such a fitness function (over 18 different evolutionary simulations) is displayed in

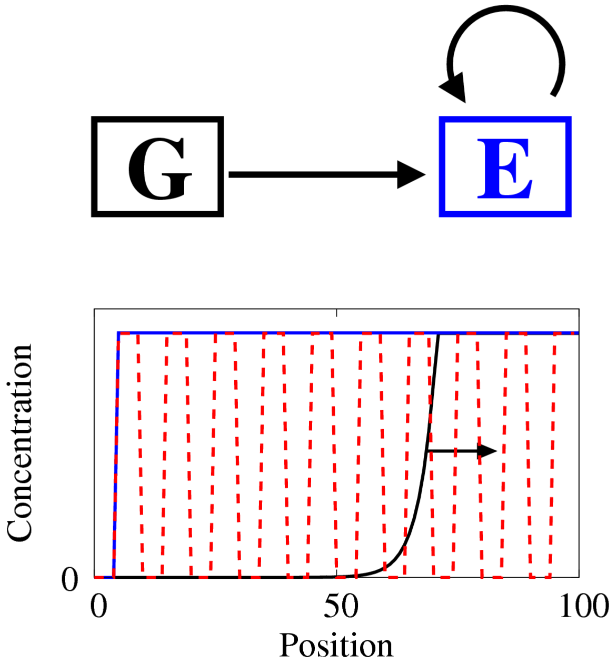

Figure S 1: Best network obtained with a fitness function imposing a fixed profile (the imposed profile is displayed in red).

Fig. S1: the algorithm managed to evolve a bistable network defining one boundary in less than 100 generations, but was then stuck and failed to produce a network creating more boundaries within the next 300 hundred generations.

Two examples of difficulties illustrating why such a fitness function failed to quickly evolve segmentation networks are displayed in Fig. S2. On both panels, the desired profile is in red and a possible evolved profile is in blue. In Panel (A), the blue profile is slightly shifted and its concentration is slightly higher than in the ideal profile. Depending on the relative weight between importance of position and importance of level, this profile may be selected or not. In Panel (B), the profile shape is fine, but it is translated relative to the ideal profile, so the fitness of the corresponding network would be very low. We believe that the main reason why these fitness functions failed was that imposing a specific profile puts too many evolutionary constraints since it requires fine tuning of both interactions and parameters to evolve. It therefore prevents incremental evolution.

Defining a fitness function that is 'blind' to numerical variation in the height, spacing, and positions of steps defines a much simpler target for the evolutionary search process. This is why we chose as a fitness function to simply count the number of boundaries $n_b$ between stripes.

Several fitness functions were used to detect the boundaries of stripes, and they gave similar results. In the following we describe the fitness function that was used to select for well-defined stripes of expression.

The fitness function is computed by looking at the values $E(x, t_f)$ of the protein E in the cell $x$ at the end of simulation after a long time $t_f = 60$. To select for expression of $E$ somewhere in the system, we increment $n_b$ by one if there is an $x$ such that $E(x, t_f) > E_{ref}$ where $E_{ref} = 20$

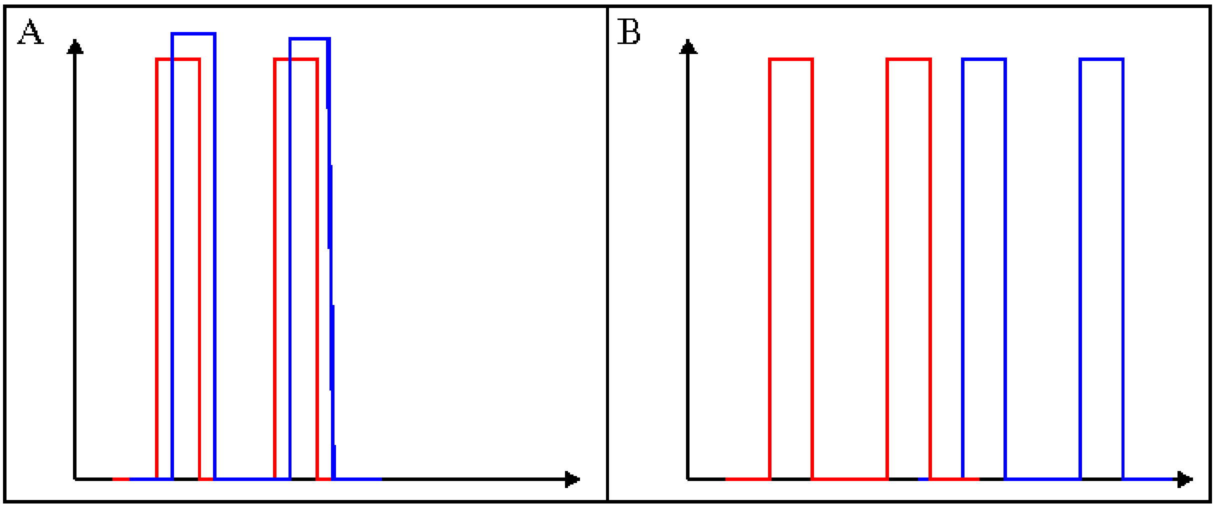

Figure S 2: Some situations where it might be difficult to select for an interesting network with too strict a fitness function.

is the minimum concentration above which $E$ is considered fully active (this part of the fitness did not appear to play a significant role in the selection and could be removed).

It is somewhat complicated to define computationally what a proper "stripe" is, as illustrated in Fig. S3. On Fig. S3 A and B, low amplitude oscillation defines up and down transition and potential stripes. There must be some minimum difference in concentrations to distinguish between an "on" and an "off" state. So we define two thresholds, $E_+ = 15$ and $E_- = 5$; if $E(x) > E_+$ the cell is considered "on", if $E(x) < E_-$, the cell is considered "off". The algorithm then detects regions of low and high E activity.

Once these thresholds are defined, one can track the extrema above and below thresholds. The alternation of local maxima and minima define succession of stripes. In that case, $n_b$ is incremented after succession of a maximum of the profile of E above $E_+$ and a minimum below $E_-$ (or if a maximum follows a minimum). This fitness function worked and gave results for the dynamic case. However, such a fitness function cannot distinguish between profiles of Fig. S3C and Fig. S3D (and we cannot really consider the profile of Fig. S3C as the boundary of a stripe). This was not a problem for the dynamic case, because bistable systems that spontaneously evolved naturally create very sharp boundaries. However, for the static case, it appeared more difficult to create sharp boundaries; so we explicitly selected for them.

The boundary of a stripe is therefore defined as a sharp transition between low and high regions. A sharp "up" boundary is a region where $E(x)$ quickly increases to reach a "plateau" in the "on" region. So we track positions where $E(x) - E(x-1) > \Delta_a$ and $E(x+1) - E(x) < \Delta_b$ (i.e. in continuous terms regions where both $E'(x)$ and $-E''(x)$ are high enough; $\Delta_a = 1$ and $\Delta_b = 0.1$ for the simulations presented here for the somites case, $\Delta_a = 2$ and $\Delta_b = 1.5$ for the static gradient case) . The number of boundaries $n_b$ is incremented by one if an "up" boundary is detected and if the previous boundary is a "down" boundary (and the symmetric computation is done to detect a "down" boundary).

The fitness is given by this number of boundaries $n_b$ so defined.

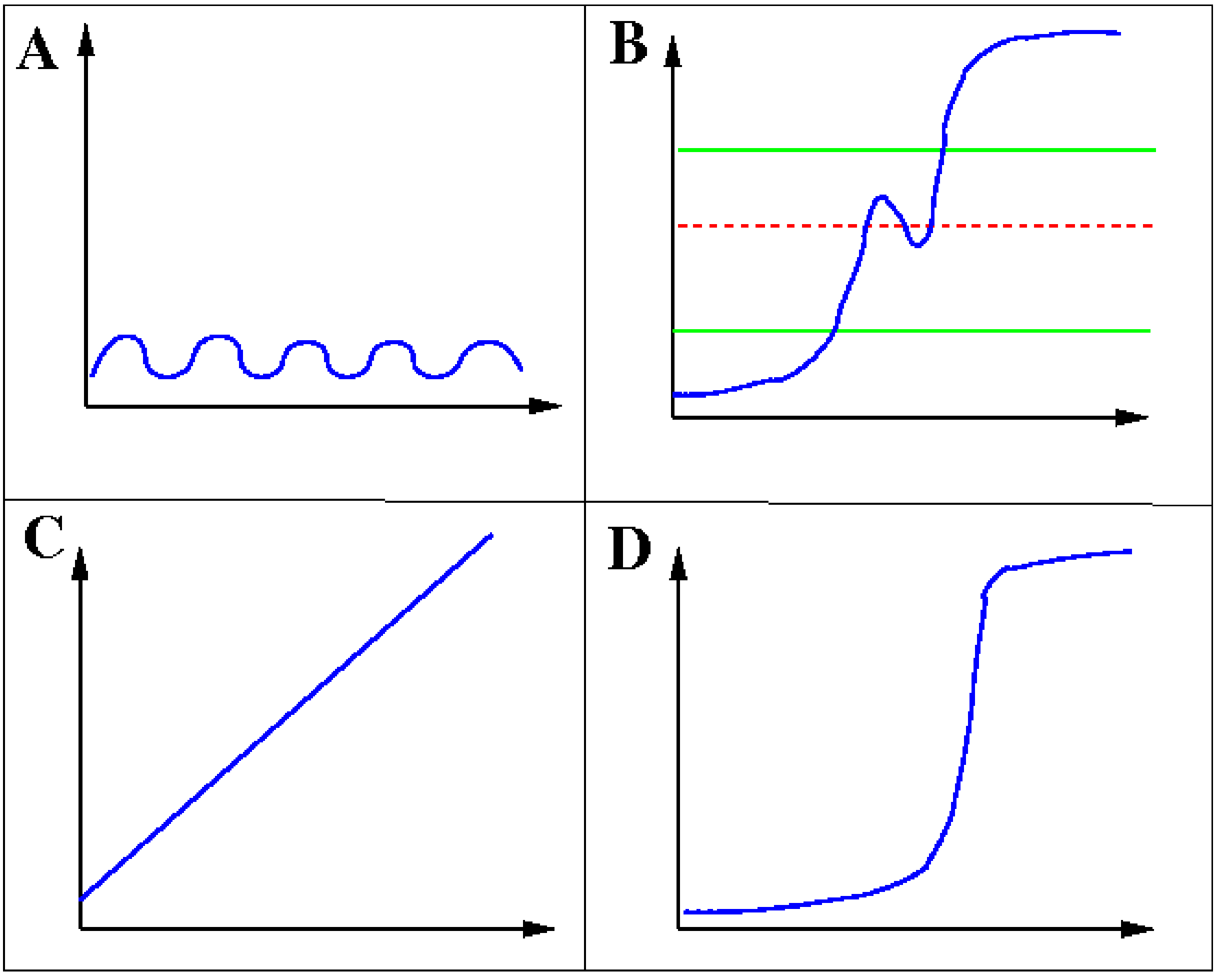

Figure S 3: Several examples (A-C) of profiles were definition of boundaries may be ambiguous, and a desirable profile D

$$F = n_b. \tag{3}$$

Practically, the term $n_b$ faithfully computes the number of "up" and "down" transitions defining a stripe.

The last difficulty is that when the number of stripes is computed in this way, the algorithm may as well select for "travelling waves" of genetic expression (which indeed happened in some early simulations, in parallel with the canonical example). So a penalty was added to the fitness function to favor a static stationary pattern, of the form

$$P = -\epsilon \int_{L,[t_f - t_0, t_f]} (E(x, t_f) - E(x, t))^2 \mathrm{d}x\mathrm{d}t, \tag{4}$$

where $L$ is the length of the embryo, $t_f$ the total time of integration and $t_0$ a very small time ($t_0 << t_f$). When E profile quickly reaches its steady state, this term is close to 0 and does not significantly change the fitness.

Computing the number of boundaries is enough to select for the topologies displayed here. A small term was however added to the fitness to ensure the production of clear cut stripes: we wanted E to reach concentration 0 in the "off" state and to be above concentration $E_{ref}$ in the "on" state. So, we defined a term $\Delta$ and incremented $\Delta$ by $|E(x)|^\alpha$ for cells in the "off" state and by $|\min(E(x), E_{ref}) - E_{ref}|^\alpha$ for cells in the "on" state (we took $\alpha = 0.5$). Then we incremented $F$ by $1 - \tanh(\Delta)$, which is always between 0 and 1 and therefore did not significantly modify the fitness while imposing a second order selective pressure to define these clear cut stripes. However, it was found in several simulations that this $\Delta$ term did not play an important role.

Note that :

- concentrations of $E(x)$ in the "on" state are free, as long as they are higher than $E_{ref}$.
- positions of the boundaries are completely free as well, so that the sizes of the stripes are free.

Other terms can be added to $\Delta$ to produce more regular stripes in the static gradient case. For instance, Fig. S13 displays a network similar in principle to the network of Fig. 3F that was selected with a term constraining the stripes and the interstripes to be 5 cells long : we first counted boundaries, then for each couple ($b_1$,$b_2$) of successive boundaries, we added to $\Delta$ a term proportional to $|b_2 - b_1 - 5|^\alpha$, with $\alpha = 0.5$.

### 1.3 Choice of parameters and methods

As our primary goal was to find working topologies for the formation of patterns, some arbitrary simplifying assumptions were made. We imposed the dynamics of the morphogen gradient, this fixed the typical time scale for the dynamics of the networks. All the other parameters were randomly chosen and varied in a specified range: for most of the presented simulations, the maximal transcriptional activities were lower than 300 in arbitrary units, degradation rates were imposed to be higher than 0.1, Hill coefficients were lower than 5. Delays were kept lower than 4; for computational reasons, delays take discrete values that vary incrementally in steps of 0.1 between 0 and 4. These assumptions were checked not to be crucial for the results and these range of parameters could be changed without consequence.

### 1.4 Numerical methods

All evolutionary implementations were written and performed in C++. Specific classes were developed to model all components of the evolutionary computation : genes, interactions between genes, and finally collection of networks. Because of the large number of equations to simulate in a complete embryo, integration was performed with a simple Euler method for speed.

The final behavior of the presented networks were double-checked independently with MATLAB, using the method dde23. All codes are available on request.

## 2 Movies

A few movies have been made to illustrate the dynamics of some networks, including networks with topologies similar to Fig. 4 in the main text. A movie where the period of the clock depends on another gradient is also displayed to show how phase waves can be generated. These movies can be visualized at the following address : http://www.physics.rockefeller.edu/˜pfrancois/movies.html .

## 3 Notations

In the following equations, the following notation conventions are used ( taking $R$ and $E$ as two arbitrary proteins):

- $\delta_E$ is the degradation constant of protein $E$,
- $S_E$ is the maximum production rate of protein $E$,
- $R_E$ is the threshold of activation/repression in the Hill function describing the regulation of protein $E$ by protein $R$. In case of multiple thresholds, we index them with small letters $a$, $b$,...
- $\tau_E$ is the delay in transcription of protein E.
- $n_i$ are Hill coefficients

## 4 Examples of evolved networks

In the equations for networks under the control of a static gradient, the dynamics of G is given by Eq. 1. Delays are not given for the static case, since they do not play any role in the dynamics of the system. In the equations for networks under control of dynamic gradients, the dynamics of G is given by Eq. 2. All equations are given for single cells, e.g. $E$ actually stands for $E(x, t)$.

### 4.1 Equations for the final network displayed in Fig. 3C

$$
\begin{aligned}
\frac{\mathrm{d}E}{\mathrm{d}t} &= \frac{G^{n_1}}{G^{n_1}+(G_E)^{n_1}}\frac{S_E}{1+(R1/R1_E)^{n_2}} - \delta_E E && (5)\\
\frac{\mathrm{d}R1}{\mathrm{d}t} &= \frac{G^{n_3}}{G^{n_3}+(G_{R1})^{n_3}}\frac{S_{R1}}{1+(R2/R2_{R1})^{n_4}} - \delta_{R1} R1 && (6)\\
\frac{\mathrm{d}R2}{\mathrm{d}t} &= S_{R2}\frac{G^{n_5}}{G^{n_5}+(G_{R2})^{n_5}} - \delta_{R2} R2 && (7)
\end{aligned}
$$

A selected set of parameters was $S_E = 104, S_{R1} = 119, S_{R2} = 38, G_E = 1, R1_E = 15, G_{R1} = 37, R2_{R1} = 4, G_{R2} = 65, n_1 = 5, n_2 = 2.5, n_3 = 5, n_4 = 2.8, n_5 = 5, \delta_E = 2, \delta_{R1} = 1$, $\delta_{R2} = 4.48$.

### 4.2 Equations for the final network displayed in Fig. 3G

$$
\begin{aligned}
\frac{\mathrm{d}E}{\mathrm{d}t} &= \frac{G^{n_1}}{G^{n_1}+(G_E)^{n_1}}\frac{S_E}{(1+(R1/R1_{Ea})^{n_2})(1+(R1/R1_{Eb})^{n_3})(1+(R2/R2_E)^{n_4})} - \delta_E E && (8)\\
\frac{\mathrm{d}R1}{\mathrm{d}t} &= \frac{G^{n_5}}{G^{n_5}+(G_{R1})^{n_5}}\frac{S_{R1}}{(1+(R2/R2_{R1})^{n_6})} - \delta_{R1} R1 && (9)\\
\frac{\mathrm{d}R2}{\mathrm{d}t} &= \frac{G^{n_7}}{G^{n_7}+G_{R2}^{n_7}}\frac{S_{R2}}{(1+(R1/R1_{R2})^{n_8})} - \delta_{R2} R2 && (10)
\end{aligned}
$$

A selected set of parameters was $S_E = 230, S_{R1} = 14, S_{R2} = 13, G_E = 0.6, R1_{Ea} = 82, R1_{Eb} = 2.9, R2_E = 0.07, G_{R1} = 82, R2_{R1} = 1.4, G_{R2} = 5.5, R1_{R2} = 0.13, n_1 = 4.7, n_2 = 2.2, n_3 = 3.4, n_4 = 2, n_5 = 3.7, n_6 = 2.4, n_7 = 4.3, n_8 = 2.7, \delta_E = 1, \delta_{R1} = 1$, $\delta_{R2} = 4.1$.

For this network, mutual repression between repressors R1 and R2 appear early in the evolution. However, this interaction does not play any role at first, as can be seen in Fig. S4, by comparing the complete system with the pruned system. Later in the evolution, this interaction helps sharpening the boundary and therefore improves the fitness (via the small term $\Delta$); this is the reason why it is included in the final representation of the network.

### 4.3 Equations for the final network displayed in Fig. 4

$$
\begin{aligned}
\frac{\mathrm{d}E}{\mathrm{d}t} &= \max\left[\frac{E^{n_1}}{E^{n_1}+E_E^{n_1}}, \frac{G^{n_2}}{G^{n_2}+G_E^{n_2}}\right]\frac{S_E}{1+(R/R_E)^{n_3}}(t-\tau_E) - \delta_E E && (11)\\
\frac{\mathrm{d}R}{\mathrm{d}t} &= \frac{G^{n_4}}{G^{n_4}+G_R^{n_4}}\frac{S_R}{1+(R/R_R)^{n_5}}(t-\tau_R) - \delta_R R && (12)
\end{aligned}
$$

A selected set of parameters was $S_E = 140, S_R = 54, E_E = 33.7, G_E = 83, R_E = 9.24, G_R = 52, R_R = 21, n_1 = 5, n_2 = 1.86, n_3 = 3.9, n_4 = 1.22, n_5 = 3.31, \tau_E = 0.5, \tau_R = 4, \delta_E = 0.45, \delta_R = 1.72$.

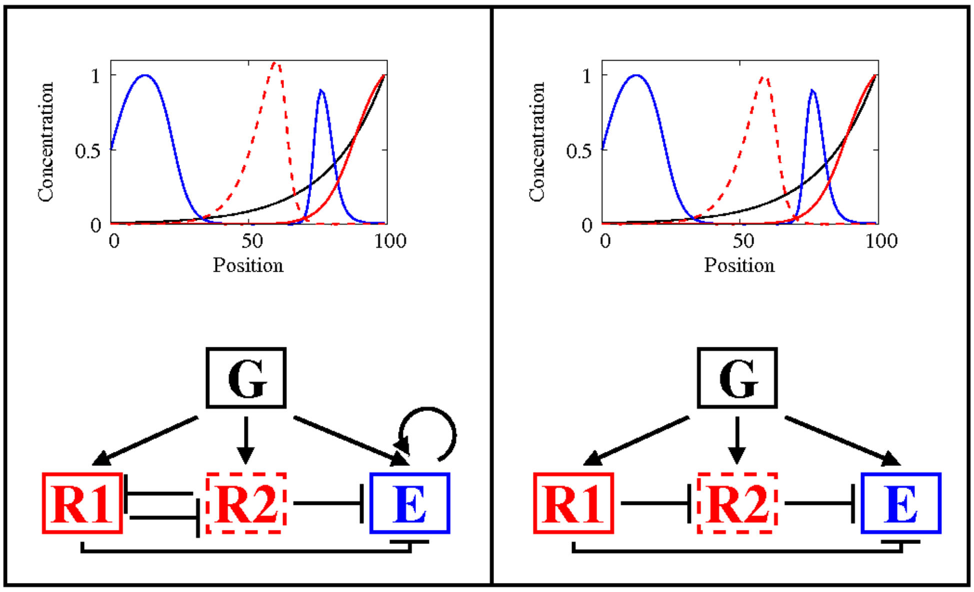

Figure S 4: Comparison between the complete network (before pruning) corresponding to Fig. 3 F, and the minimal pruned network with the same fitness displayed on Fig. 3 F. The profile of E (in blue) is not significantly affected by pruning: repression of $R2$ by $R1$ does not change the profile, while autoactivation of E is to weak to play any role. The pruning procedure used to display results only keeps the core interaction necessary to have a network with a given fitness.

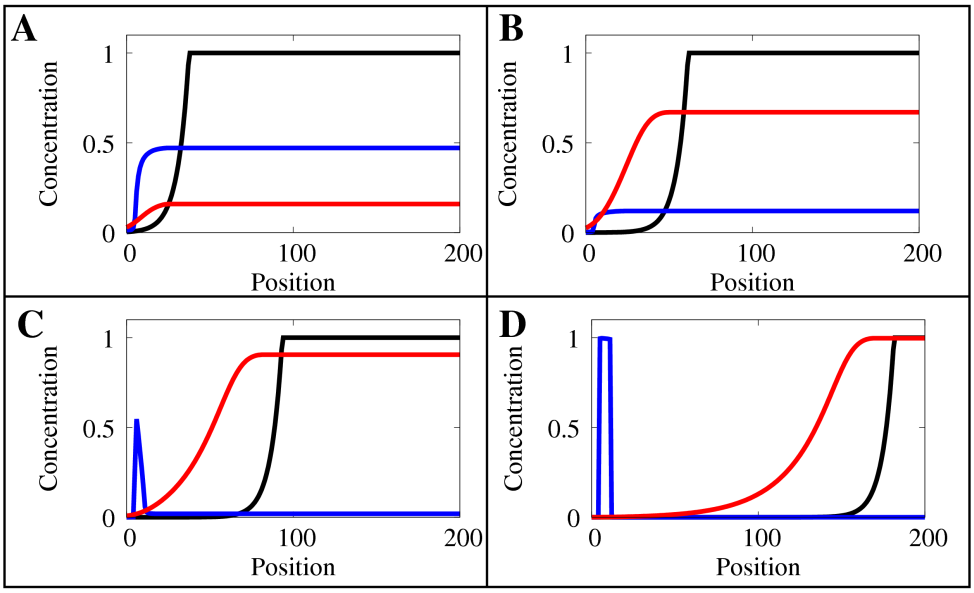

Figure S5: Illustration of the conditions required to make one stripe with the topology displayed on Fig. 4 C under control of a moving gradient. (A) E is first activated, R is activated later (B) R starts repressing E (C) While the front of G is moving, for low values of R,the concentration of E remains high enough for E for its autoactivation (close to small $x$); at high G concentration, R is fully induced and completely represses E which can no longer activates itself (higher values of $x$). (D) This creates a stripe close to small $x$.

### 4.4 Equations and evolutionary pathway for a network similar to the network of Fig. 4

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \max\left[\frac{E^{n_1}}{E^{n_1}+E_E^{n_1}}, \frac{G^{n_2}}{G^{n_2}+G_E^{n_2}}\right]\frac{S_E}{1+(R/R_E)^{n_3}} - \delta_E E \tag{13}$$

$$\frac{\mathrm{d}R}{\mathrm{d}t} = \left(\max\left[\frac{G^{n_4}}{G^{n_4}+G_{Ra}^{n_4}}, \frac{G^{n_5}}{G^{n_5}+G_{Rb}^{n_5}}\right]\frac{S_R}{1+(R/R_R)^{n_6}}\right)(t-\tau_R) - \delta_R R \tag{14}$$

A selected set of parameters was $S_E = 146, S_R = 29, E_E = 31.3, G_E = 47, R_E = 13.6, G_{Ra} = 40, G_{Rb} = 63, R_R = 4, n_1 = 2, n_2 = 5, n_3 = 2.2, n_4 = 0.57, n_5 = 4.2, n_6 = 4.24, \tau_R = 1, \delta_E = 2.24, \delta_R = 1.83$. Note that the algorithm sets to 0 the delay in the transcription of E ; however, a delay is required for R oscillations.

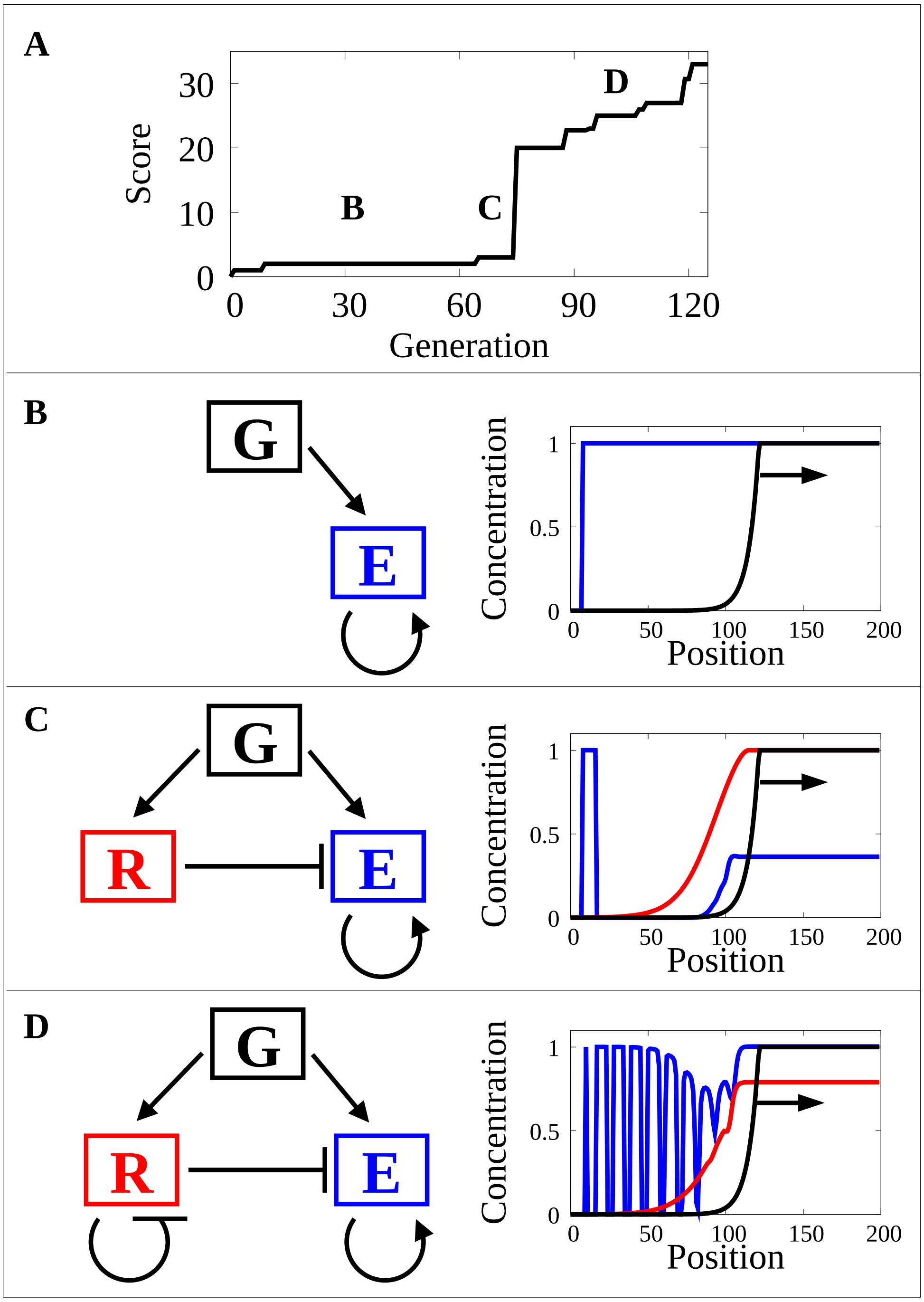

Figure S 6: Another example of an evolutionary pathway leading to a topology similar to the network of Fig. 4 .

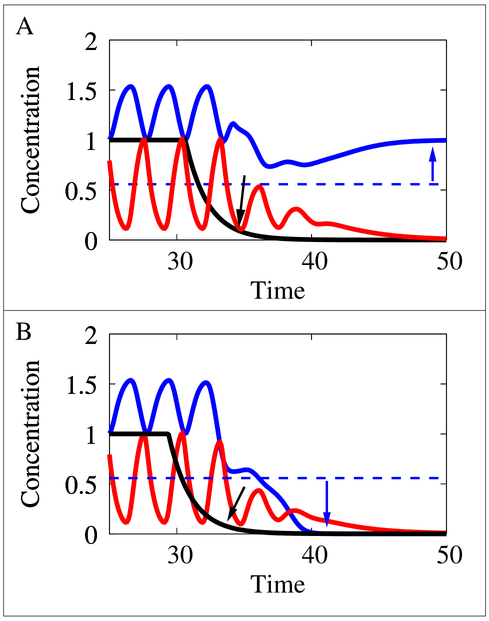

Figure S 7: Creation of high (A) or low (B) values of the segmentation marker $E$ in two cells by the coupled effects of oscillatory and bistable dynamics for the network displayed in Fig. S6. The cell fate is determined by the concentration of E relative to a threshold $E_*$ (shown by a dashed line) at the time (black arrow) of transition from oscillatory to bistable dynamics. The unstable fixed point $E_*$ (for $G = R = 0$) separates protein concentrations $E > E_*$ converging to the high state of E expression, from smaller values that end in the low state of E expression (as indicating by the blue arrows). In (A) E is high enough at the transition time so that G and R can disappear while leaving $E > E_*$. In (B) the concentration of E is lower at the transition time and subsequent oscillations of R push E under the threshold $E_*$.

### 4.5 Repressilator-like network

In all our evolutionary computations, the effect of selective pressure was to first select for a bistable system, then to create a repressor to delimit one stripe. Eventually, an oscillator appeared that “replicated” the stripe. In most of our simulations, oscillation was due to an auto regulatory feedback loop with delay. In much rarer cases, other kind of oscillators appeared. We obtained for instance several networks where the oscillation was due to a “repressilator”-like mechanism [2]. “Repressilators” were selected by the evolutionary process because cascades of repressors created several successive “up” and “downs” (as can be seen in Fig. S8). The final equations for one of the evolved repressilator-like networks (corresponding to Fig. 6 of the main paper) are:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \max\left[\frac{E^{n_1}}{E^{n_1}+E_E^{n_1}}, \frac{G^{n_2}}{G^{n_2}+G_E^{n_2}}\right] \frac{S_E}{(1+(R1/R1_{Ea})^{n_3})(1+(R1/R1_{Eb})^{n_4})} - \delta_E E \tag{15}$$

$$\frac{\mathrm{d}R1}{\mathrm{d}t} = \frac{S_{R1}G^{n_5}}{(G^{n_5}+G_{R1}^{n_5})(1+(R2/R2_{R1a})^{n_6})(1+(R2/R2_{R1b})^{n_7})} - \delta_{R1}R1 \tag{16}$$

$$\frac{\mathrm{d}R2}{\mathrm{d}t} = \frac{S_{R2}G^{n_8}}{(G^{n_8}+G_{R2}^{n_8})(1+(R3/R3_{R2})^{n_9})} - \delta_{R2}R2 \tag{17}$$

$$\frac{\mathrm{d}R3}{\mathrm{d}t} = \frac{S_{R3}E^{n_{10}}}{E^{n_{10}}+E_{R3}^{n_{10}}} - \delta_{R3}R3 \tag{18}$$

A selected set of parameters was $S_E = 75, S_{R1} = 68, S_{R2} = 112, S_{R3} = 300, E_E = 0.7, G_E = 91, R1_{Ea} = 3.8, R1_{Eb} = 7.5, G_{R1} = 86, R2_{R1a} = 3.11, R2_{R1b} = 88, G_{R2} = 154, R3_{R2} = 14, E_{R3} = 4.13, n_1 = n_2 = 5, n_3 = 2.6, n_4 = 1.8, n_5 = 2.9, n_6 = 3.4, n_7 = 0.47, n_8 = n_9 = 5, n_{10} = 1.21, \delta_E = 2.43, \delta_{R1} = 3, \delta_{R2} = 12, \delta_{R3} = 11.75$. Note that delays were eliminated in this specific evolutionary process, however they were required along the evolutionary pathway.

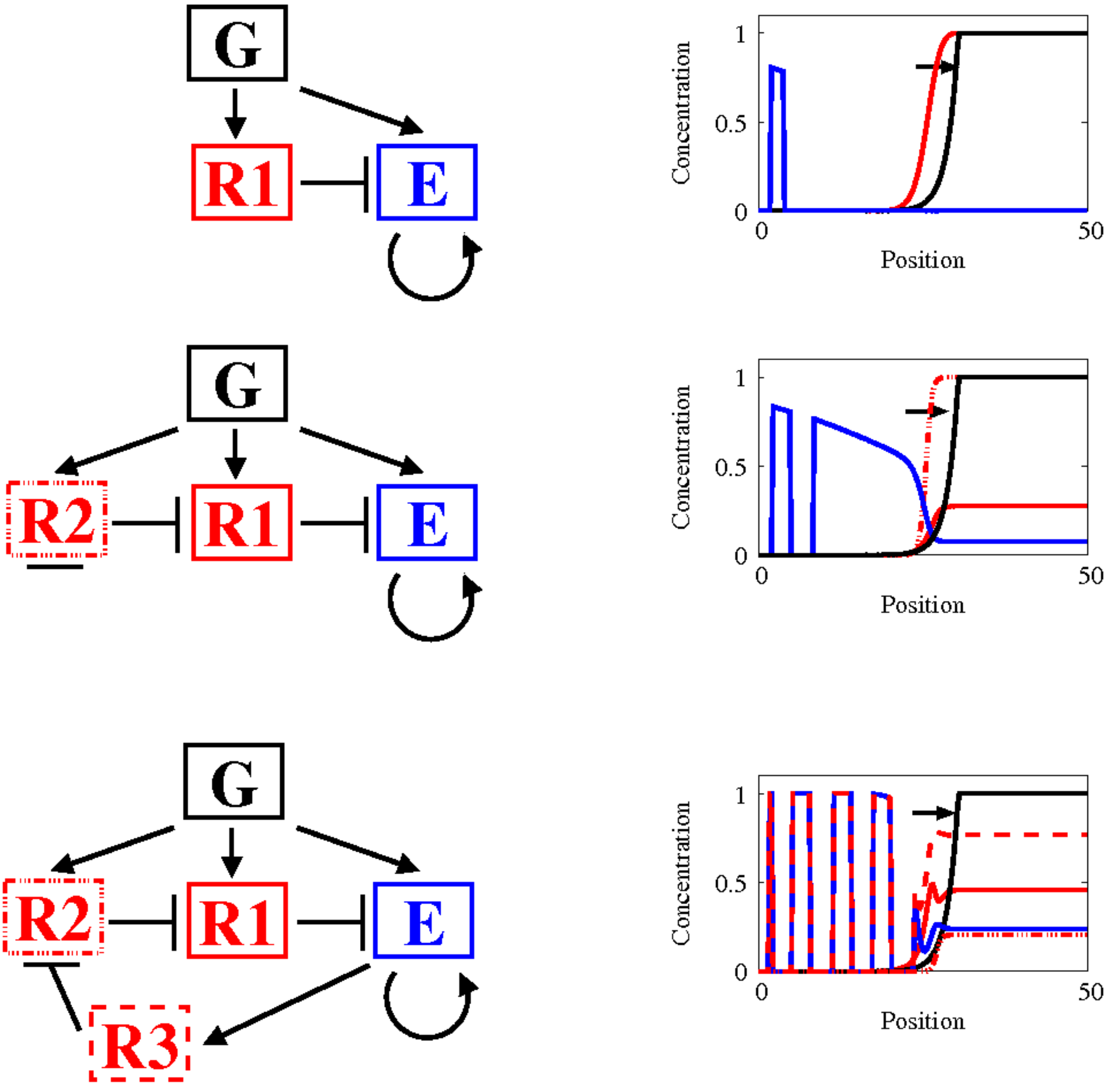

Figure S 8: Evolution of a repressilator-like network with corresponding network behaviors and topologies right after the main evolutionary transitions.

## 4.6 Network working on a principle similar to that the network of Fig. 4

Another example of a network working on a similar principle as the network of Fig. 4 is displayed with its evolutionary pathway on Fig. S9. Interestingly, for this network, the evolutionary pathway was slightly more complicated: the beginning of the pathway was the same, with first a bistable system coupled to a repressor to form one stripe (Fig. S9A). Then, the pathway bifurcated and a second repressor appeared to repress the first repressor, creating a new "up" transition (Fig. S9B). However, some networks similar to those of Fig. S9B remained in the population of networks, and in one of these, autorepression of the first repressor appeared, creating first damped and then sustained oscillations as parameters changed (Fig. S9C). Networks based on the topology displayed in Fig. S9B disappeared from the population.

Equations for the final network are:

$$\begin{aligned}
\frac{\mathrm{d}E}{\mathrm{d}t} &= T_E(t-\tau_E) - \delta_E E \qquad (19)\\
\frac{\mathrm{d}R}{\mathrm{d}t} &= \left(\frac{S_R G^{n_5}}{(G^{n_5}+G_R^{n_5})(1+(R/R_R)^{n_6})}\right)(t-\tau_R) - \delta_R R \qquad (20)
\end{aligned}$$

with

$$\begin{aligned}
T_E = \max &\left[\frac{E^{n_1}}{E^{n_1}+E_E^{n_1}}, \frac{G^{n_2}}{G^{n_2}+G_E^{n_2}}\right]\\
&\times \frac{S_E}{(1+(R/R_{Ea})^{n_3})(1+(R/R_{Eb})^{n_4})}
\end{aligned} \qquad (21)$$

A selected set of parameters was $S_E = 38, S_R = 29, E_E = 10.6, G_E = 26.7, R_{Ea} = 69, R_{Eb} = 99, G_R = 54, R_R = 4.36, n_1 = 3.28, n_2 = 0.5, n_3 = n_4 = 0.7, n_5 = 4.8, n_6 = 3.8, \tau_E = 0.1, \tau_R = 1.5, \delta_E = 1.56, \delta_R = 1.6$.

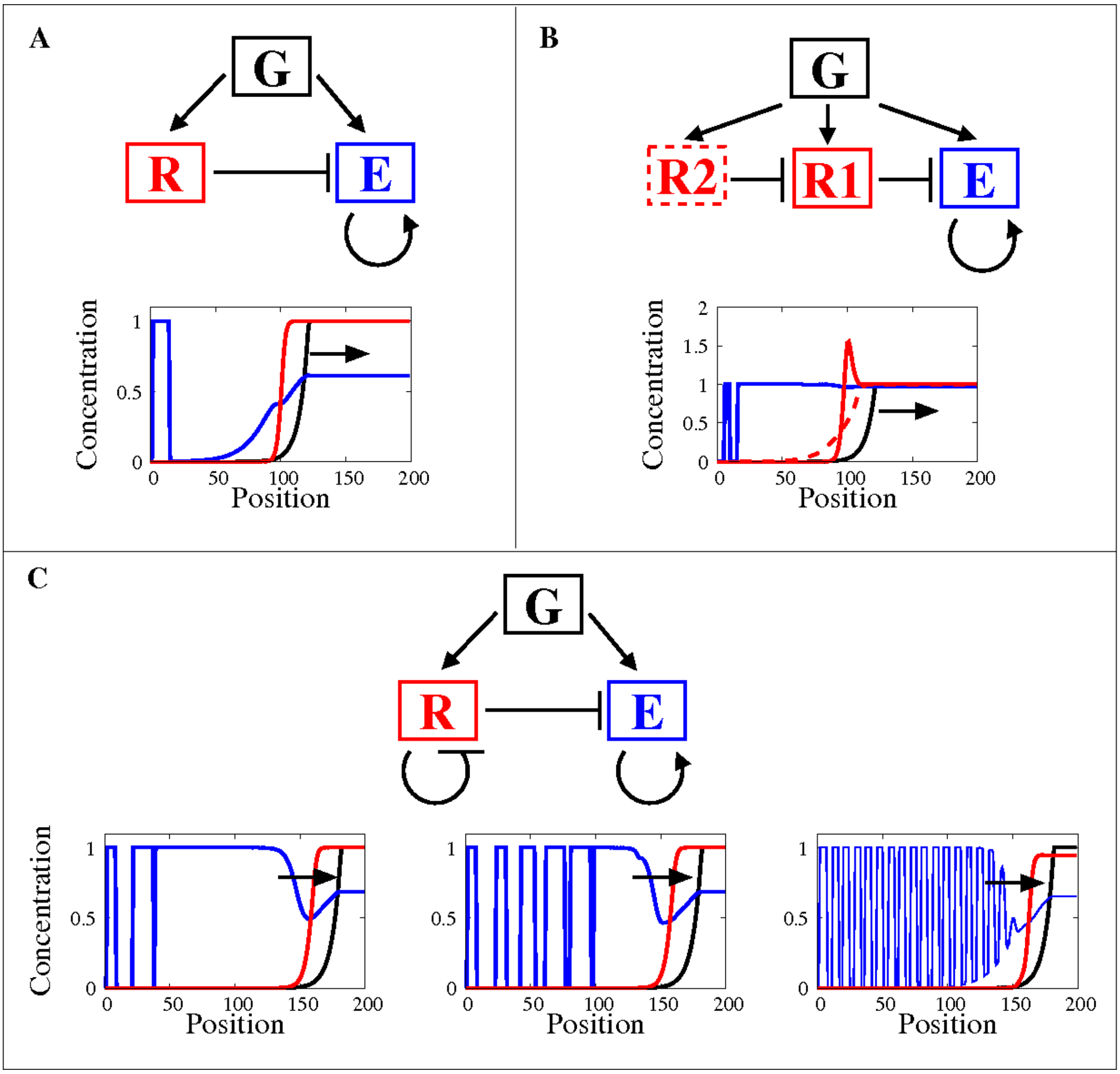

Figure S9: A more complex evolutionary pathway leading to a genetic network similar to Fig 4. (A) The evolutionary pathway first reached the two-gene network producing one stripe, similar to Fig. 4B.(B) Later in the evolution, a new repressor was created, in a similar way to Fig. 3C. This created a new boundary. (C) For later generations, oscillation on the first repressor appeared, so that the evolutionary innovation of panel (B) was forgotten - network of panel (C) is a “cousin” of the network of panel (B). First, oscillations were damped, creating only a few stripes (left). Then, parameters evolved, increasing the fitness, leading to less damped (middle), and finally short sustained oscillations (right).

### 4.7 Network creating two stripes for one oscillation of the repressor

Some improvements to the "clock and wavefront" model were found by the algorithm. In most of the evolutionary computations, once the final topology of Fig. 4D was found, the parameters evolved to create as many stripes as possible. An example of a different improvement is displayed in Fig. S10. In this example, the evolutionary pathways led to a clock coupled to a bistable system first as in the example of Fig. 4. Then, neutral evolution created a protein A repressed by the oscillating repressor. A therefore oscillated as well, but not in phase with E. Evolution then selected activation of E by A: because of the phase shift between E and A oscillation, this created a new zone of high E activity after sweeping of the morphogen, so that two stripes were created for one oscillation of the repressor.

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \left(\max\left[\frac{E^{n_1}}{E^{n_1}+E_E^{n_1}}, \frac{G^{n_2}}{G^{n_2}+G_E^{n_2}}, \frac{A^{n_3}}{A^{n_3}+A_E^{n_3}}\right]\frac{S_E}{1+(R/R_E)^{n_4}}\right)(t-\tau_E) - \delta_E E \quad (22)$$

$$\frac{\mathrm{d}R}{\mathrm{d}t} = \left(\frac{G^{n_5}}{G^{n_5}+G_R^{n_5}}\frac{S_R}{1+(R/R_R)^{n_6}}\right)(t-\tau_R) - \delta_R R \quad (23)$$

$$\frac{\mathrm{d}A}{\mathrm{d}t} = \left(\frac{G^{n_7}}{G^{n_7}+G_A^{n_7}}\frac{S_A}{1+(R/R_A)^{n_8}}\right)(t-\tau_A) - \delta_A A \quad (24)$$

A selected set of parameters was $S_E = 53, S_R = 70, S_A = 24, E_E = 2.2, G_E = 58.7, A_E = 23.7, R_E = 23, G_R = 13, R_R = 36, G_A = 71, R_A = 12.5, n_1 = 3.21, n_2 = 5, n_3 = 2.29, n_4 = 1.92, n_5 = 5, n_6 = 4.7, n_7 = 0.23, n_8 = 3, \tau_E = 0.5, \tau_R = 2.9, \tau_A = 0.1, \delta_E = 2.52, \delta_R = 0.87, \delta_A = 0.81$. Setting $S_A = 0$ gives single stripes.

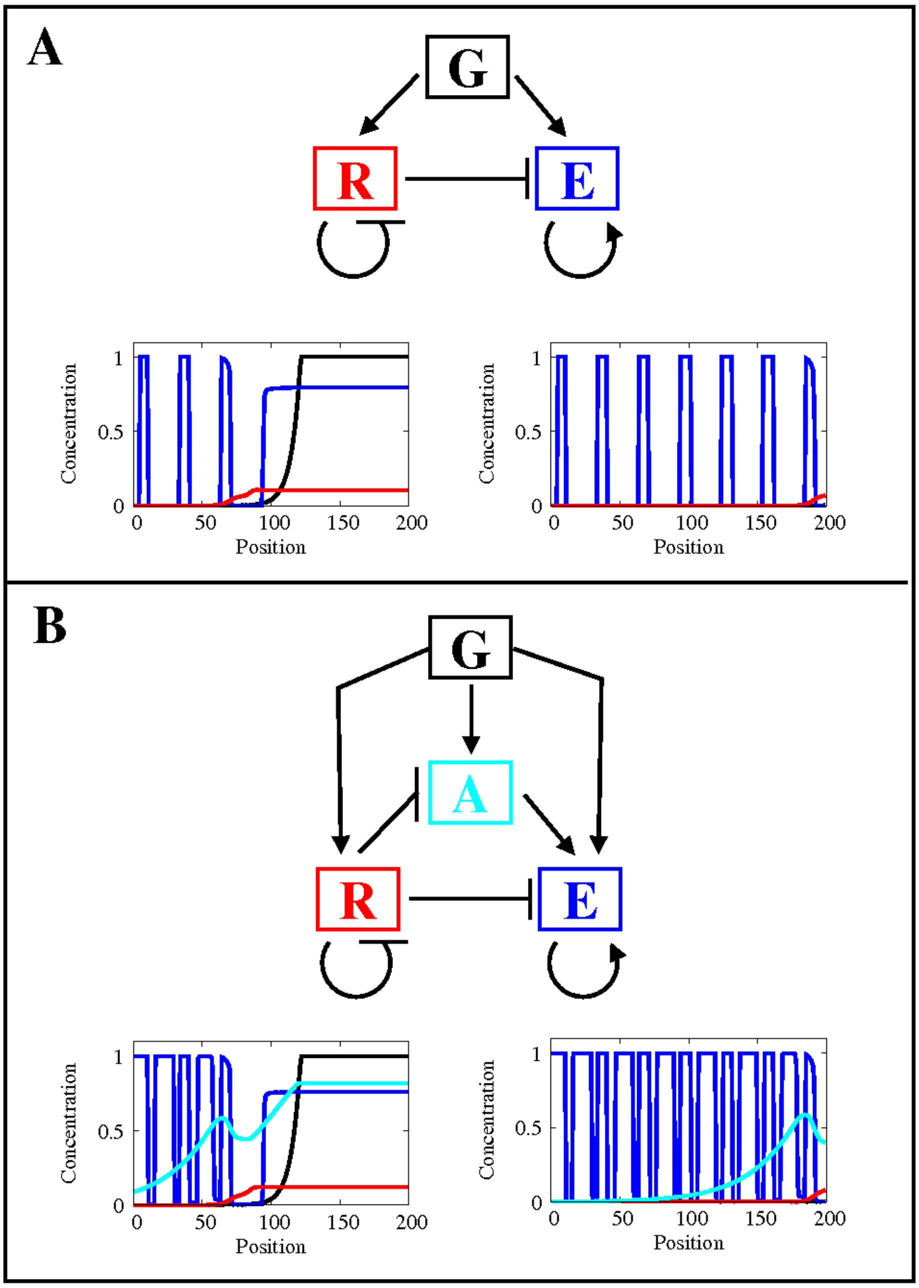

Figure S 10: Network creating two stripes for one oscillation of R. (A) Standard “clock and wavefront” mechanism, evolved before the apparition of the innovation. (B) A new protein A activates E between the stripes, creating twice as many stripes as in panel (A).

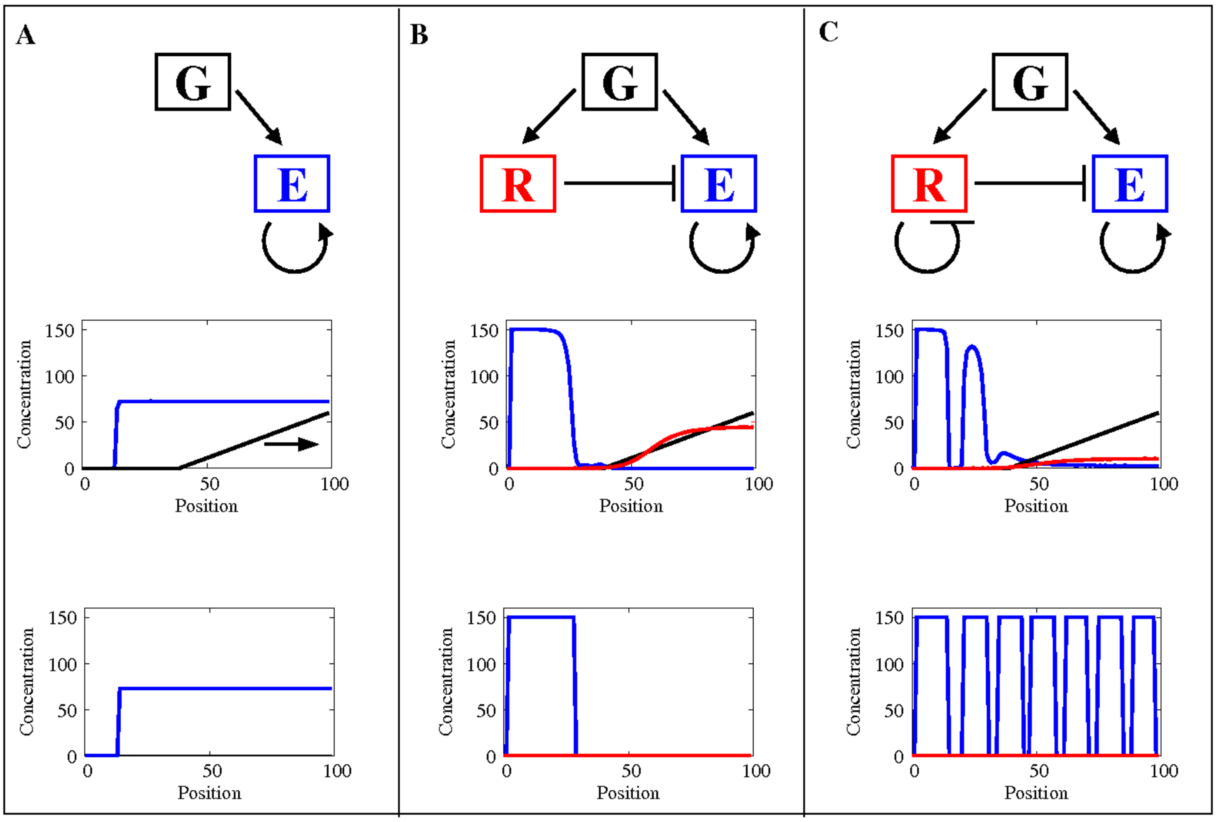

Figure S 11: Evolutionary pathway for a network under control of a dynamic linear morphogen gradient. (A) Evolution of bistability. (B) Evolution of a repressor creating one stripe. (C) Evolution of a segmentation clock.

### 4.8 Linear gradients

The presented results do not depend on the specific shape of the morphogen gradient. This procedure was applied as well with a linear gradient as an input. Evolution of G is given by $G(x,t) = \max(x - 4t, 0)$. Both the evolutionary pathways and topologies were similar. One working network obtained with such a gradient is :

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \left(\max\left[\frac{E^{n_1}}{E^{n_1}+E_E^{n_1}}, \frac{G^{n_2}}{G^{n_2}+G_E^{n_2}}\right]\frac{S_E}{1+(R/R_E)^{n_3}}\right)(t-\tau_E) - E \tag{25}$$

$$\frac{\mathrm{d}R}{\mathrm{d}t} = \left(\frac{S_R G^{n_4}}{(G^{n_4}+G_R^{n_4})(1+(R/R_R)^{n_5})}\right)(t-\tau_R) - R \tag{26}$$

A selected set of parameters was $S_E = 150, S_R = 150, E_E = 17, G_E = 5.7, R_E = 1.3, G_R = 28.8, R_R = 1.9, n_1 = 5, n_2 = 1.14, n_3 = 5, n_4 = 2.8, n_5 = 3, \tau_E = \tau_R = 1$. Evolutionary pathway is displayed in Fig. S11.

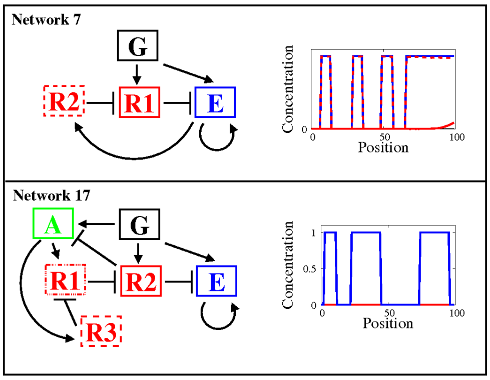

Figure S 12: Topologies of two networks of Table I, with the corresponding final profiles. In Network 7, stripes are due to delays in the two positive feedback loops that produce damped oscillations before stabilisation. In Network 17, the period of the oscillator is quite long, so that few stripes are created (but the network did have sustained oscillations).

## References

[1] P François and V Hakim. Design of genetic networks with specified functions by evolution in silico. *PNAS*, 101:580–585, 2004.

[2] M B Elowitz and S Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature.*, 403:335–338, 2000.

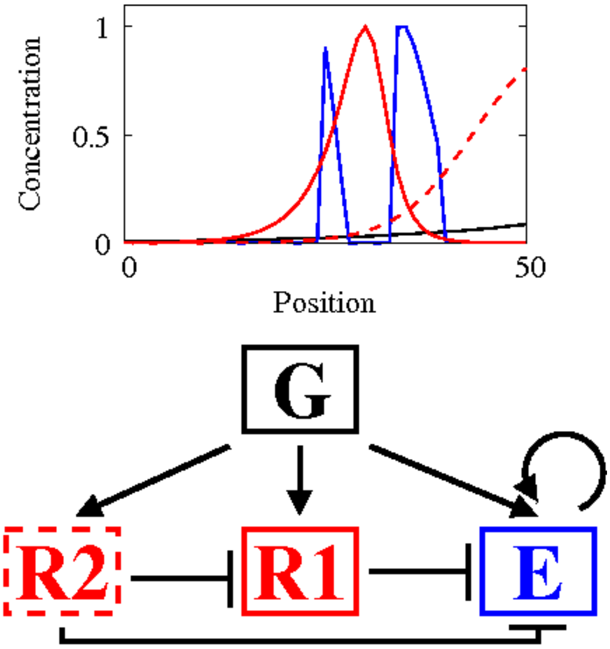

Figure S 13: Network obtained by adding a term in $\Delta$ that constrains stripes and interstripes to be more regular.