BIOINFORMATICS

Vol. 00 no. 00 2011
Pages 1–4

Supplementary Material to: A comprehensive computational model of facilitated diffusion in prokaryotes

Nicolae Radu Zabet [1,2,*] and Boris Adryan [1,2]

[1]Cambridge Systems Biology Centre, University of Cambridge, Tennis Court Road, Cambridge CB2 1QR, UK; [2]Department of Genetics, University of Cambridge, Downing Street, Cambridge CB2 3EH, UK

1 IMPLEMENTATION OF ONE DIMENSIONAL RANDOM WALK

A TF molecule of type x which is bound on the DNA at position j will wait an exponentially distributed time with average τ_x^j before it performs a new action. Having multiple agents in the system (TF molecules bound to the DNA), we store all the waiting times in a PriorityQueue. The head of this queue is the soonest event. We call this in our application the First Reaction Method.

Alternatively, by selecting an option the user can simulate the system with the Direct Method, which is an adaptation of Gillespie's Direct Method (Gillespie, 1977). In this method, each bound TF molecule will have a rate to move from its current position, which is inversely proportional to the waiting time, $\kappa_x^j = 1/\tau_x^j$, and the simulator will select the time a molecule will move from its current position and the ID of that molecule. In addition, to keep the move rate for each position and total sum, we also keep intermediate sums for sectors of DNA which reduces the search from $\sum_x TF_x$ to $\sqrt{\sum_x TF_x}$, similar to our approach on locating a free site on the DNA; see section 2.2. For some values, it seems that this approach is slightly better than the First Reaction Method, but, in general, we found that the First Reaction Method outperforms the Direct Method.

2 LOCATING A FREE BINDING POSITION

The simplest implementation of TF binding to the DNA consist to draw a random number identifying a position on the DNA and checking if that position is free (Chu *et al.*, 2009). In addition, all TF_x^{size} right base pairs need to be free as well, in order for the TF to bind at that position. The time to find a free location is dependent on the the DNA occupancy and, even if a free position is found, there is no guarantee that there are enough uncovered base pairs on the right side. When implementing genome-wide simulations, crowding of various molecules on the DNA can create problems when searching for a free spot to bind a molecule.

An alternative implementation was presented by Barnes and Chu (2010). They proposed that each long enough run of free base pairs to be stored in an array, while small fragments to be stored in a different array. On each TF binding, unbinding or random walk event, the software will keep these arrays updated so that, when a free spot is required, the simulator will look only in the array of free spots and, thus, only one random number is generated per binding event. This *memory model* was able to achieve a speed-up and was reported that it could perform up to 10^5 events per second on a DNA of $4.6\ Mbp$ with 10^6 TF molecules. Their simulator was implemented in C++ and the simulations were run on a Mac Pro 3GHz quad-core Intel Xeon with 8G memory running Mac OSX 10.4.

Here, we propose a slightly different model to store free positions. We create an array list of boolean values for each TF species (x) and this array stores whether a TF molecule of species x is allowed to bind at position j, i.e., if $A[x][j]$ is true then a TF molecule of species x can bind at position j, and otherwise a molecule will not be allowed to bind. This array is updated after each event is performed. The memory model is looking through the entire list of fragments or continuous runs to locate where the molecule made more room for another TF molecule to bind, or where it reduced/removed the binding space. In our model, we look only in neighbouring positions where the TF moved or from where it moved. Thus, searching in an entire list of fragments (which can be significantly high) is reduced to searching only through neighbouring positions (which depends on the size of the molecules).

The purpose of this mechanism is to eliminate the need to check if enough base pairs (TF_x^{size}) on the right side of the selected position are not covered by other molecules. This, however, comes at the cost of maintaining a series of arrays updated after each event is processed; see below.

2.1 Update Available Position Array

Initially any TF can bind anywhere on the DNA. After each event is performed, we need to update the availability arrays as follows:

- After a TF of type x **binds** at position j do

$$A[y][i] = false,\ \forall i \in (j - TF_y^{\text{size}}, j + TF_x^{\text{size}});\ \forall y$$

  This update is applied for all TF species, once a molecule of type x bound to the DNA.

*to whom correspondence should be addressed

- After a TF of type x **slides left** from position j to position j', with $j' \in (j - TF_x^{\text{size}}, j)$, do

$$A[y][i] = false, \quad \forall i \in (j' - TF_y^{\text{size}}, j - TF_y^{\text{size}}); \quad \forall y$$
$$A[y][i] = true, \quad \forall i \in (j' + TF_x^{\text{size}}, posEnd); \quad \forall y$$

  where $posEnd = findFirstCoveredBP(j + TF_x^{\text{size}}, j + TF_x^{\text{size}} + TF_y^{\text{size}}) - TF_y^{\text{size}}$ if there is a covered base pair between positions $(j + TF_x^{\text{size}}, j + TF_x^{\text{size}} + TF_y^{\text{size}})$ or $posEnd = j + TF_x^{\text{size}}$ otherwise. The function $findFirstCoveredBP(s, e)$ returns the first covered base pair between positions s and e.

- After a TF of type x **slides right** from position j to position j', with $j' \in (j, j + TF_x^{\text{size}})$, do

$$A[y][i] = false, \quad \forall i \in (j + TF_x^{\text{size}}, j' + TF_x^{\text{size}}); \quad \forall y$$
$$A[y][i] = true, \quad \forall i \in (posStart, j' - TF_y^{\text{size}}); \quad \forall y$$

  where $posStart = findLastCoveredBP(j - TF_y^{\text{size}}, j)$ if there is a covered base pair between positions $(j - TF_y^{\text{size}}, j)$ or $posStart = j - TF_y^{\text{size}}$ otherwise. The function $findLastCoveredBP(s, e)$ returns the last covered base pair between positions s and e.

- After a TF of type x **unbinds** from position j do

$$A[y][i] = true, \quad \forall i \in (posStart, posEnd); \quad \forall y$$

  where $posStart = findLastCoveredBP(j - TF_y^{\text{size}}, j)$ if there is a covered base pair between positions $(j - TF_y^{\text{size}}, j)$ or $posStart = j - TF_y^{\text{size}}$ otherwise and $posEnd = findFirstCoveredBP(j + TF_x^{\text{size}}, j + TF_x^{\text{size}} + TF_y^{\text{size}}) - TF_y^{\text{size}}$ if there is a covered base pair between positions $(j + TF_x^{\text{size}}, j + TF_x^{\text{size}} + TF_y^{\text{size}})$ or $posEnd = j + TF_x^{\text{size}}$ otherwise.

2.2 Randomly Select a Free Position

The simplest way to select a free position is selecting a random number in the interval $[0, M)$, where M is the length of the analysed DNA. In a crowded environment, when a significant part of the DNA is covered by TF the simulator might experience a lot of failed attempts to find a free position.

Alternatively, one can store the current number of free position for each species A_x^{current}.This is computed before the simulation starts and whenever any item in the array of available positions is updated, the current number of available positions is also updated. This does not put any significant load on the simulation, but reduces the search time for a free position. When we need to locate a position for a TF molecule of type x on the DNA, we have an array with all available positions $A[x]$ and the current number of available positions A_x^{current}. In order to select a free position, we draw a random number z in the interval $[0, A_x^{\text{current}})$ and then we count through the array until we find the zth available position. This method guarantees that a free position is found using only one random number, which represents a significant improvement of the simulation speed.

Since random numbers are uniformly distributed, on average we need to search through $M/2$ elements, where M is the length in base pairs of the DNA sequence. Thus, the search time for a free position increases linearly with increasing the length of the DNA sequence. The search of a the zth free position can be optimised by dividing the DNA into sectors of size R and by keeping updated the current number of free positions for each sector. This is similarly implemented as keeping updated the current number of free positions for the entire DNA and the load on simulation time is negligible.

When looking for a free position, we first need to search through the current number of available positions for each sector until we locate the sector which contains the position of interest and then look inside that sector. Assuming the same uniform distribution, the number of steps is

$$\langle steps \rangle = \frac{1}{2}\frac{M}{R} + \frac{1}{2}R$$

The minimum number of steps is obtained by checking where the first derivative with respect to the sector size is zero, $\partial\langle steps\rangle/R$

$$\frac{1}{2}\frac{M}{R^2} + \frac{1}{2} = 0 \;\Rightarrow\; R = 0,\; R = \sqrt{M} \tag{1}$$

The only positive solution is $R = \sqrt{M}$, which leads to an average number of steps equal to $\langle steps \rangle = \sqrt{M}$.

3 AFFINITY LANDSCAPE

The affinity landscape can be computed using one of the three following methods: (i) mismatch energy (Gerland *et al.*, 2002), (ii) PFM and information theory (Stormo, 2000) and (iii) PFM and energy affinity (Berg and von Hippel, 1987). For exemplification purposes, we consider that the DNA binding motif of lacI tetramer is the following

`AATTGTNNNNNNNNNACAATT`

, a spaced inverted repeat. We consider that the dimeric lacI recognizes `5'-AATTGT-3'` and that the binding site has 21 bp (Turner, 2001).

We used this sequence in conjunction with mismatch energy (Gerland *et al.*, 2002) and for the PFM we constructed an equivalent position frequency matrix $\mathbf{PFM}_{\text{lacI}}$. To construct the $\mathbf{PFM}_{\text{lacI}}$ we consider 40 identical sequences and in the gap we assumed that all nucleotides have equal probabilities; see $\mathbf{PFM}_{\text{lacI}}$ Figure 1.

Figure 2 confirms that the computed binding energy is Gaussian distributed and only a few number of sites have high affinity.

We investigated the correlation between these three versions for an affinity landscape of lacI tetramer on the *E.coli* K-12 genome and found that there is a high correlation between all three methods (Gerland *et al.*, 2002; Berg and von Hippel, 1987; Stormo, 2000); with a Pearson coefficient of correlation equal to 0.99. Although the binding energies of the three methods are highly correlated, their mean and variance differ significantly. However, these differences can be corrected, by selecting the appropriate τ_0 term. This will scale the actual waiting times, so the waiting times will have similar values for the three methods.

$$\mathbf{PFM}_{\text{lacI}} = \begin{array}{c|cccccccccccccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 0 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 \\ A & 40 & 40 & 0 & 0 & 0 & 0 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 40 & 0 & 40 & 40 & 0 & 0 \\ C & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 0 & 40 & 0 & 0 & 0 & 0 \\ G & 0 & 0 & 0 & 0 & 40 & 0 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ T & 0 & 0 & 40 & 40 & 0 & 40 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 10 & 0 & 0 & 0 & 0 & 40 & 40 \end{array}$$

Fig. 1. *PFM*. Equivalent PFMs for the AATTGTNNNNNNNNNACAATT binding motif and 40 binding sequences. In the gap region (between position 7 to 15) we consider equal probability to see any of the nucleotides.

We also investigated the affinity landscape computed with the aforementioned methods in a biased genome (70% AT and only 30% CG content). The results showed that there is a high correlation between Gerland *et al.* (2002) and Berg and von Hippel (1987) methods (a Pearson correlation coefficient of 1) and a very low between these two and the Stormo (2000) approach (a Pearson correlation coefficient of 0.69). This result can be explained by the fact that the Stormo (2000) method takes into account the frequency of a nucleotide in the entire genome, which does not appear in the two other methods, Gerland *et al.* (2002) and Berg and von Hippel (1987).

4 ESTIMATING MODEL PARAMETERS

4.1 Estimating Specific Waiting Time from the Affinity Landscape

For $s_l^{\text{obs}} = 90\ bp$, a residence time of $t_R = 5\ ms$ (Elf *et al.*, 2007) and an average binding energy of the lacI tetramer $\langle \exp(-E_x) \rangle \in \{3.73e-06, 3.44, 2.10e-12\}$ (see Figure 2) we obtain the following specific waiting times

$$\begin{aligned} \tau^0_{\text{lacI}} &= 0.33\ s \text{ for Gerland } \textit{et al.} \text{ (2002)} \\ \tau^0_{\text{lacI}} &= 3.58e-07\ s \text{ for Stormo (2000)} \\ \tau^0_{\text{lacI}} &= 5.87e+05\ s \text{ for Berg and von Hippel (1987)} \end{aligned} \quad (2)$$

Note that, for Stormo (2000) and Berg and von Hippel (1987), we weighted the energy contribution by $\varepsilon^*_{\text{lacI}} = 1\ K_B T$.

In addition, we computed the specific waiting time for non-cognate species (see Figure 2) as $\tau^0_{\text{nc}} = 0.33\ s$, where the average exponential binding energy was $\langle \exp(-E_{\text{nc}}) \rangle = 3.72e-06$.

All the above mentioned parameters, are listed in Table 1.

5 VALIDATION OF THE PARAMETERS ESTIMATION APPROACH

We systematically investigated the accuracy of the proposed method to estimate model parameters. Figure 3 confirms that our proposed approach leads to the results of simulations deviating only negligible from the ones predicted mathematically. In particular, we considered four parameters: (i) sliding length, (ii) observable sliding length, (iii) residence time and (iv) proportion of time bound to the DNA. Figure 3 shows that only the residence time has higher variability, but the average residence time of multiple (or longer) simulations matches well the value computed analytically. This variability can be reduced by running the simulations for longer times, which would lead to more accurate averages for the measures.

We also compared the one dimensional diffusion coefficient from our simulations to the one proposed in (Elf *et al.*, 2007). Figure 4 confirms that our simulations reproduces the value proposed by Elf *et al.* (2007) with negligible error.

6 SIMULATION APPROXIMATIONS

The binding rate (k_x^{bind}) has to be evaluated whenever the number of available positions on the DNA change, which includes binding, unbinding, hopping and sliding events.

This has a negative effect for the performance of the simulation, because we need to draw another random number for next binding event each time a TF molecule slides on the DNA. From the parameters that we estimated above, it means that we may draw up to 4000 random numbers before a new binding event actually takes place, which is highly inefficient.

Alternatively, one might consider an approximate system, in which the binding of TF molecules is affected by occupancy, but the update is performed only when a molecule binds/unbinds and not when any other event (sliding or hopping) would lead to change in the number of available binding sites on the DNA. This approximation reduces the load of updates caused by the sliding events, but at the same time seems to follow with good accuracy the behaviour of the exact system. In Figure 5, red and blue lines represent the approximate system, while green and orange lines is the exact one.

Henceforth, when computing and comparing the speed, we will use only the approximate system, since the difference between the approximation and the exact system is negligible. Nevertheless, the user has the possibility to use the exact system in any simulation.

7 SIMULATION SPEED

7.1 DNA Sectors and Event List Subgroups

The simulation speed is measured as the number of events simulated per second and the simulator was run for $T\ s$. We used a Mac Pro 2x2.26GHz quad-core Intel Xeon with 32GB memory running Mac OSX 10.6.8.

In our model section, we proposed that searching in an array of available positions can be optimised by maintaining them for smaller sectors. The left part of Figure 6 confirms that there is an optimal sector size ($R = \sqrt{M}$), which can lead to significant increase in simulation speed (in our case the number of operations performed per second doubles).

Furthermore, we tested the effect of dividing the event list in smaller sub-groups and found that, for the *Direct Method*, the speed increased significantly when the size of the sub-group equalled the square root of the non-cognate copy number, while in the case of the *First Reaction* method breaking the event list into smaller sub-lists did not increase the speed; see right part of Figure 6.

7.2 TF Abundance

Finally, we observed that the simulation speed can depend strongly on the number of TF molecules in the system. This mainly comes from the fact that the event list storing the events associated with each molecule increases with the number of TF molecules. For the *First Reaction* method, keeping the queue sorted is the most time consuming step, while in the *Direct Method* the bottleneck is represented by the search of the next TF molecule to move on the DNA. Since *First Reaction* uses a PriorityQueue, which has a search time equal to $\log_2\left(\sum_x TF_x\right)$, while our implementation of the *Direct Method* has a search time of $\sqrt{\sum_x TF_x}$, one can see that for long lists the *First Reaction* is much more efficient compared to the *Direct Method*; see right end of Figure 7. However, Figure 7 shows that for intermediate number of molecules in the system ($\approx 10^4$) the *Direct Method* algorithm seems to slightly outperform the *First Reaction* method.

Barnes and Chu (2010) indicated that their implementation could simulate $100000\ events/s$ for 1000000 non-cognate molecules. Figure 7 indicates that our program can simulate approximately 4 times more events per second ($400000\ events/s$) for same number of non-cognate molecules. One might argue that, by using molecules that cover 46 base pairs on the DNA, only a small fraction of our TFs will bind to the DNA. In fact we observed that only 8% of the molecules bind to the DNA in the case of $TF_{\text{nc}} = 10^6$. To investigate whether this is the reason behind the speed-up, we reduced the size of the non-cognate molecules to $4\ bp$ and changed the association rate so that we get $\approx 85\%$ of the molecules are bound to the DNA. The results showed that even when we accommodate a higher number of molecules on the DNA, our simulator will still perform $\approx 400000\ events/s$.

REFERENCES

Barnes, D. J. and Chu, D. F. (2010). An efficient model for investigating specific site binding of transcription factors. In *Bioinformatics and Biomedical Engineering (iCBBE), 2010 4th International Conference on*, pages 1–4, Chengdu, China. IEEE Xplore.

Bell, C. E. and Lewis, M. (2000). A closer view of the conformation of the lac repressor bound to operator. *Nature Structural Biology*, **7**, 209–214.

Berg, O. G. and von Hippel, P. H. (1987). Selection of DNA binding sites by regulatory proteins statistical-mechanical theory and application to operators and promoters. *Journal of Molecular Biology*, **193**(4), 723–750.

Chu, D., Zabet, N. R., and Mitavskiy, B. (2009). Models of transcription factor binding: Sensitivity of activation functions to model assumptions. *Journal of Theoretical Biology*, **257**(3), 419–429.

DeSantis, M. C., Li, J.-L., and Wang, Y. M. (2011). Protein sliding and hopping kinetics on DNA. *Physical Review E*, **83**, 021907.

Elf, J., Li, G.-W., and Xie, X. S. (2007). Probing transcription factor dynamics at the single-molecule level in a living cell. *Science*, **316**, 1191–1194.

Gerland, U., Moroz, J. D., and Hwa, T. (2002). Physical constraints and functional characteristics of transcription factor-DNA interaction. *PNAS*, **99**(19), 12015–12020.

Gillespie, D. T. (1977). Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, **81**, 2340–2361.

Riley, M., Abe, T., Arnaud, M. B., Berlyn, M. K., Blattner, F. R., Chaudhuri, R. R., Glasner, J. D., Horiuchi, T., Keseler, I. M., Kosuge, T., Mori, H., Perna, N. T., Plunkett, G., Rudd, K. E., Serres, M. H., Thomas, G. H., Thomson, N. R., Wishart, D., and Wanner, B. L. (2006). Escherichia coli k-12: a cooperatively developed annotation snapshot - 2005. *Nucleic Acids Research*, **34**(1), 1–9.

Stormo, G. D. (2000). DNA binding sites: representation and discovery. *Bioinformatics*, **16**(1), 16–23.

Stormo, G. D. and Fields, D. S. (1998). Specificity, free energy and information content in protein-DNA interactions. *Trends in Biochemical Sciences*, **23**(3), 109–113.

Turner, B. (2001). *Chromatin and gene regulation: mechanisms in epigenetics*. John Wiley & Sons.

Wunderlich, Z. and Mirny, L. A. (2008). Spatial effects on the speed and reliability of protein-DNA search. *Nucleic Acids Research*, **36**(11), 3570–3578.

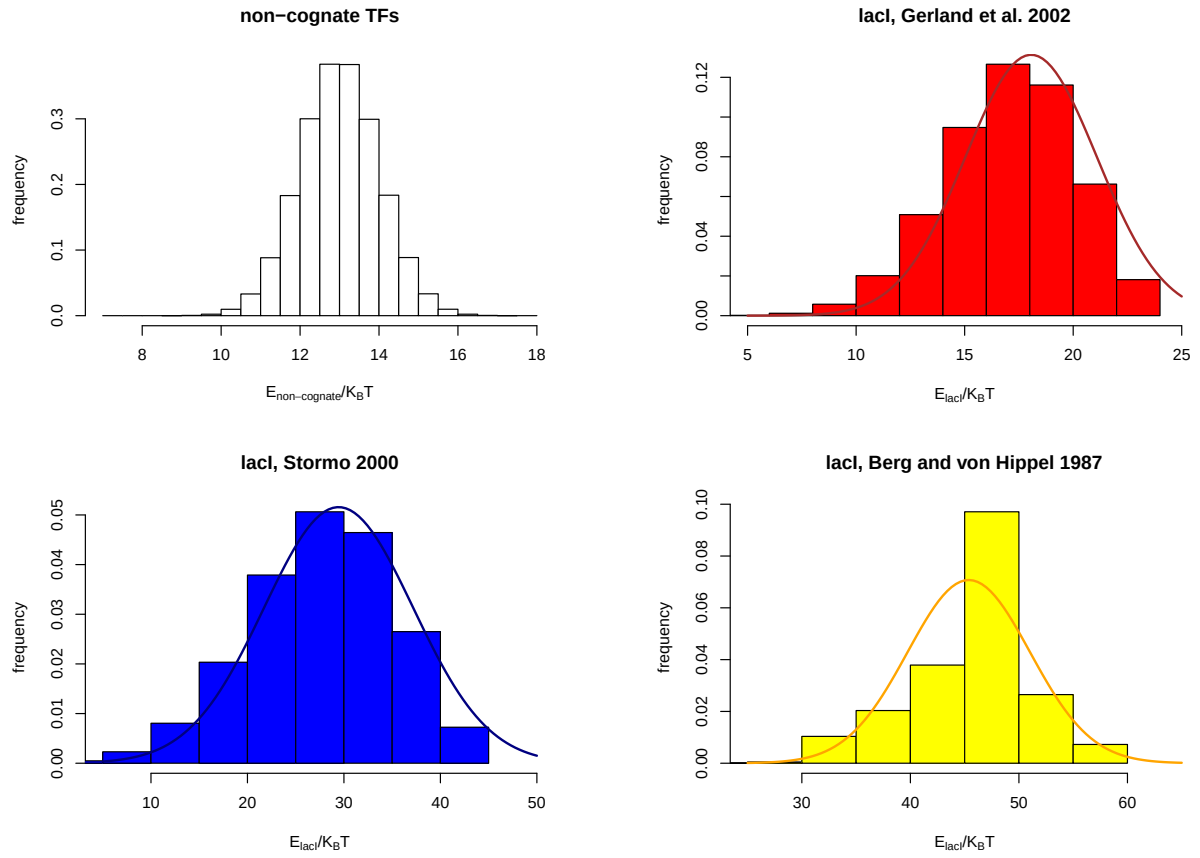

Fig. 2. *Density plot of binding energies*. We computed the binding energy for the genome of *E.coli* K-12 genome of: (*white*) a non-cognate TF specie, (*red*) lacI reressor using the mismatch energy method and the O_1 operator site (Gerland *et al.*, 2002), (*blue*) lacI repressor and (Stormo, 2000) and (*yellow*) lacI repressor and (Berg and von Hippel, 1987). We used $\mathbf{PFM}_{\text{lacI}}$ for the third and forth plots and the motif consensus AATTGTNNNNNNNNNACAATT for the second plot. The density plot for non-cognates follows a Gaussian distribution and all values were allowed. For lac repressor, since we selected an energetic penalty of $\varepsilon^*_{\text{lacI}} = 2\ K_BT$ we get the values in bins separated by $2\ K_BT$, but which follows a Gaussian distribution as well. The mean values for the binding energy are: (*white*) $\langle E \rangle \approx 13$, (*red*) $\langle E \rangle \approx 18$, (*blue*) $\langle E \rangle \approx 5$, (*yellow*) $\langle E \rangle \approx 22.8$. The means of the exponential energy are: (*white*) $\langle \exp(-E) \rangle \approx 3.72e-06$, (*red*) $\langle \exp(-E) \rangle \approx 3.73e-06$, (*blue*) $\langle \exp(-E) \rangle \approx 13.46$, (*yellow*) $\langle \exp(-E) \rangle \approx 4.46e-09$.

parameter	description
M	the length of the DNA measured in base pairs ($M \approx 4.6\ Mbp$) (Riley *et al.*, 2006)
TF_x^{free}, TF_x^{bound} and TF_x	the number of free (TF_x^{free}), bound (TF_x^{bound}) or total ($TF_x = TF_x^{\text{free}} + TF_x^{\text{bound}}$) molecules of species x
f	the relative time a TF molecule stays bound to the DNA ($f \approx 0.9$) (Elf *et al.*, 2007)
k_x^{bind}	the rate at which a TF molecule of species x will bind to the DNA
k_x^{assoc}	the association rate constant for species x, see equation (19) in main text
k^{dissoc}	the dissociation rate constant between a TF molecule and the DNA, see equation (18) in main text, ($k_x^{\text{dissoc}} = 200$)
s_l^{obs}	observed sliding length, the average number of base pairs scanned during a slide including small dissociations from the DNA (hops), ($s_l^{\text{obs}} = 90\ bp$) (Elf *et al.*, 2007)
s_l	sliding length, the average number of base pairs scanned during a slide, see equation (14) in main text, ($s_l \approx 40\ bp$)
$N_{\text{se}}^{\text{obs}}$	number of one dimensional random walk events performed until the TF molecule performs a full dissociation from the DNA, see equation (15) in main text, ($N_{\text{se}}^{\text{obs}} \approx 4000$)
N_{se}	number of one dimensional random walk events performed until the TF molecule performs a dissociation from the DNA (including a hop), see equation (12) in main text, ($N_{\text{se}} \approx 700$)
t_R	residence time, the time a TF spends performing one dimensional random walk before it unbinds ($t_R = 5\ ms$) (Elf *et al.*, 2007)
P_{unbind}	the probability to unbind from the DNA, see equation (13) in main text, $P_{\text{unbind}} = 0.0012$
P_{left}, P_{right}	the probability to slide left (P_{left}) or right (P_{right}) on the DNA, see equation (16) in main text, ($P_{\text{left}} = P_{\text{right}} = 0.4994$)
P_{jump}	the probability that the molecule completely dissociates from the DNA during an unbinding event, while $(1-P_{\text{jump}})$ represents the probability that a molecule will rebind fast after a dissociation (hop), ($P_{\text{jump}} = 0.1675$) (Wunderlich and Mirny, 2008) or ($P_{\text{jump}} = 0.001$) (DeSantis *et al.*, 2011)
σ_{hop}^2	the variance of the hop distance, which is Gaussian distributed around previous position ($\sigma_{\text{hop}}^2 = 1\ bp$) (Wunderlich and Mirny, 2008)
d_{jump}	the distance over which a hop becomes a jump ($d_{\text{jump}} = 100\ bp$) (Wunderlich and Mirny, 2008)
τ_x^j, κ_x^j	the average waiting time (τ_x^j) and the average move rate ($\kappa_x^j = 1/\tau_x^j$) at position j for species x
τ_x^0	the average waiting time for species x when bound specifically, see equation (17) in main text.
E_x^j	the binding energy of specie x at position j
S_x	the specific binding sequence of TF species x
$\varepsilon_x^j(k)$	the energetic penalty at position k within the motif
ε_x^*	the constant energetic penalty for a mismatch. ($\varepsilon_x^* \approx 2 \cdot K_B T$) (Gerland *et al.*, 2002)
$n_{0,k}^x$	the highest number of occurrences of any nucleotide in position k of all known high affinity binding sequences of species x (Berg and von Hippel, 1987)
$n_{j,k}^x$	the number of occurrences of the nucleotide at position $j+k$ on the DNA in position k of all known high affinity binding sequences of species x (Berg and von Hippel, 1987)
ζ	the pseudo-count term ensures that the energy penalty is not $\varepsilon_x^j(k) = -\infty$. ($\zeta = 1$) (Berg and von Hippel, 1987)
$\nu_{j,k}^x$	the frequency of occurrences of the nucleotide at position $j+k$ on the DNA in position k of all known high affinity binding sites (Stormo, 2000)
ν_j	the frequency of the nucleotide at position j in the entire genome (Stormo, 2000)
TF_x^{size}	the number of base pairs covered by a bound TF molecule. ($TF_{\text{nc}} \geq 20\ bp$ (Stormo and Fields, 1998))
TF_x^{motif}, TF_x^{left} and TF_x^{right}	the number of base pairs covered by the DNA binding domain (TF_x^{motif}), to the left (TF_x^{left}) of the DNA binding domain and to the right (TF_x^{right}) of the DNA binding domain. We analysed the crystallographic structure of the lacI-DNA complex (PDB ID: 1EFA, (Bell and Lewis, 2000))and found that $TF_x^{\text{left}} = TF_x^{\text{right}} = 0$.
$A[x][j]$	array that stores whether a TF molecule of type x can bind at position j
A_x^{current} and A_x^{max}	the current (A_x^{current}) and maximum (A_x^{max}) number of free sites on the DNA where the molecule of type x can bind
R	the length of the DNA sector used to speed-up the search process of a free spot on the DNA ($R = \sqrt{M}$)
$c_{x,j}^{y,j'}$	cooperativity term for DNA mediated cooperative behaviour term between a molecule of type x bound at position j and a molecule of type y bound at position j'
c_x^y	cooperativity term for direct TF-TF cooperativive behaviour between a molecule of type x and a molecule of type y
K_B and T	Boltzmann constant (K_B) and temperature (T).

Table 1. *Nomenclature*. We usually denote species indexes by x or y and positions on the DNA or motif by j, j', i or k. To avoid increase in notations, we did not add an explicit index to the one dimensional random walk parameters (such as P_{unbind} or s_l), but it is implicitly assumed that these parameters are specific to each TF species. Where there is a default value of the parameter, we specified it in the parentheses at the end of the description.

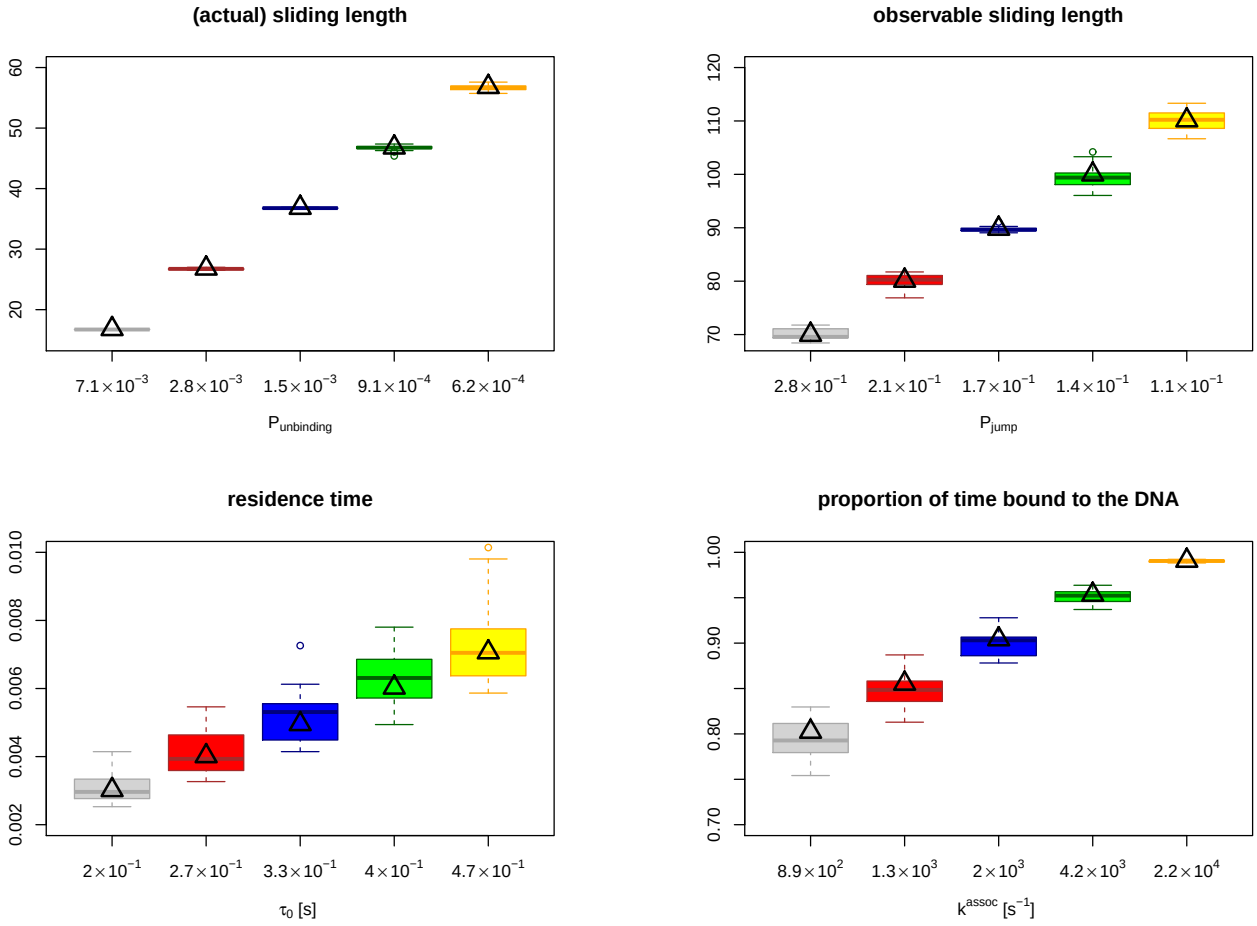

Fig. 3. *Validation of the parameters estimation approach (1).* We compare the values obtained from simulations with the ones estimated from our approach. The triangles represent the computed values, while the box plots the simulation ones. To compute the affinity landscape we used the (Gerland *et al.*, 2002) mismatch approach. In addition, we considered 10^5 non-cognate TFs (each covering 46 bp of DNA) with the default parameters and 5 lacI molecules with the default parameters. We considered four measurable parameters, namely: (i) sliding length, (ii) observable sliding length, (iii) residence time and (iv) proportion of time bound to the DNA. We kept everything fixed and for each of these measurable parameters we varied one microscopic parameter which: (i) for sliding length we varied unbinding probability ($P^{\text{lacI}}_{\text{unbind}} \in \{7.09E-3, 2.78E-3, 1.47E-3, 9.13E-4, 6.20E-4\}$), (ii) for observable sliding length we varied the jumping probability ($P^{\text{lacI}}_{\text{jump}} \in \{0.28, 0.21, 0.17, 0.13, 0.12\}$), (iii) for residence time we varied the specific waiting times ($\tau^{0}_{\text{lacI}} \in \{0.20, 0.27, 0.33, 0.40, 0.47\}$ s) and (iv) for proportion of time bound to the DNA we varied the association rate ($k^{\text{assoc}\,0}_{\text{lacI}} \in \{892, 1263, 2000, 4238, 22082\}$ s^{-1}). For each set of parameters we performed 20 simulations each running for 1 s (for sliding length and observable sliding length) and 10 s (for residence time and proportion of time bound to the DNA). The blue error bars represent the original system (the one using our default parameters estimates). The errors between the computed values and the ones measured from the simulations are negligible.

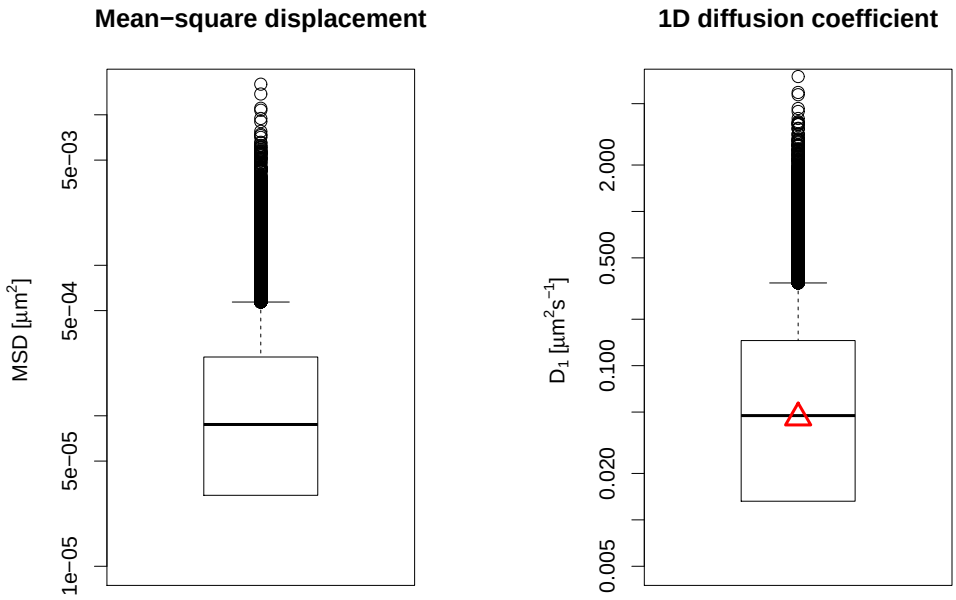

Fig. 4. *Validation of the parameters estimation approach (2).* We compare one dimensional diffusion measurements of our simulations (box plots) that we obtained with the ones proposed by Elf *et al.* (2007) (triangle). To compute the affinity landscape we used the (Gerland *et al.*, 2002) mismatch approach. In addition, we considered 10^5 non-cognate TFs (each covering 46 *bp* of DNA) with the default parameters and 5 lacI molecules with the default parameters. The mean value for our one dimensional diffusion coefficient is $0.046\ \mu m^2 s^{-1}$ which is the same value Elf *et al.* (2007) proposed. Since the resolution of our method is much higher than the one of the experimental measures, we discretized the sample data to fit resolution in (Elf *et al.*, 2007) (we disregarded sliding events that lasted less than $1\ ms$). Without removing those data points the mean one dimensional diffusion coefficient is $0.056\ \mu m^2 s^{-1}$ which is still close to the value proposed in (Elf *et al.*, 2007).

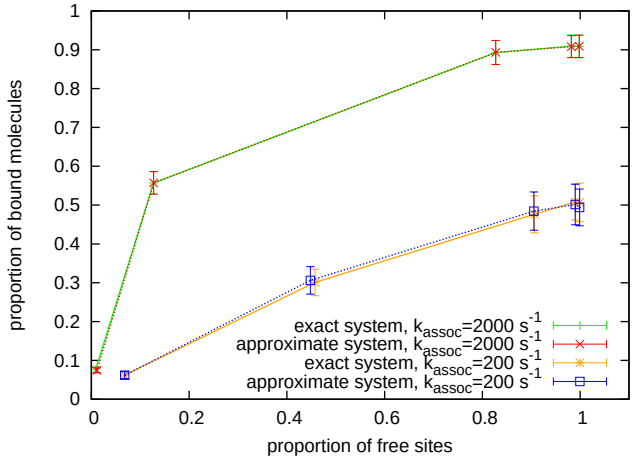

Fig. 5. *DNA occupancy affects binding rate.* This graph presents the steady state bound proportion for two systems: (*green* and *orange*) a system which updates the the binding events after each binding, unbinding, sliding or hopping event and (*red* and *blue*) a system which updates the the binding events after each binding or unbinding event only. We also considered two cases for the constant association rate: (*green* and *red*) $k_{\text{assoc}} = 2000\ s^{-1}$ and (*orange* and *blue*) $k_{\text{assoc}} = 200\ s^{-1}$. The differences between the exact and approximate systems are negligible.

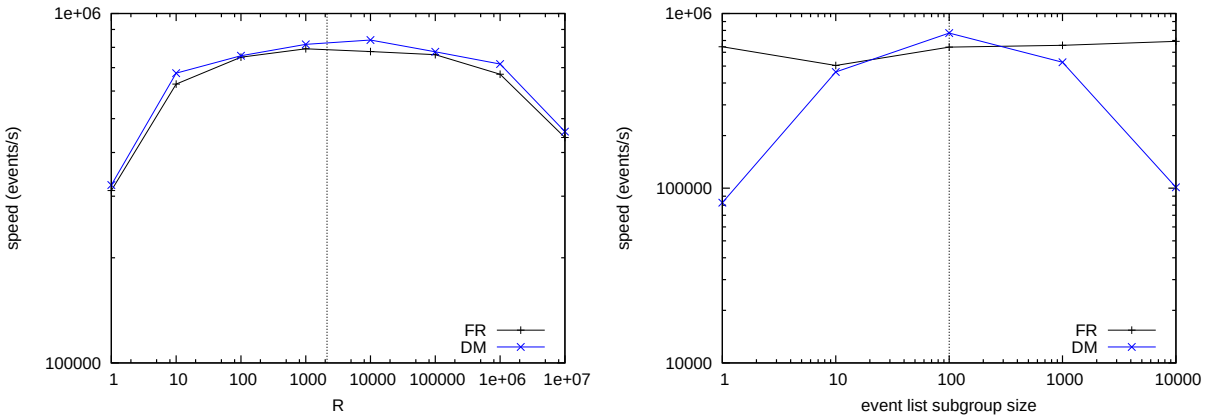

Fig. 6. *Influence of sector size on simulation speed*. The system consists of 10000 non-cognate TFs and 5 cognate ones (lacI) and the *E.coli* K-12 genome. **Left** The DNA sector size plays an important role for the speed, in the sense that there is an optimal sector size ($R = \sqrt{M}$) which maximizes number of events performed per second. **Right** The sub-group size of the random walk event list plays an important role only for Direct Method, where the optimal event list subgroup size is $\sqrt{\sum_x TF_x}$.

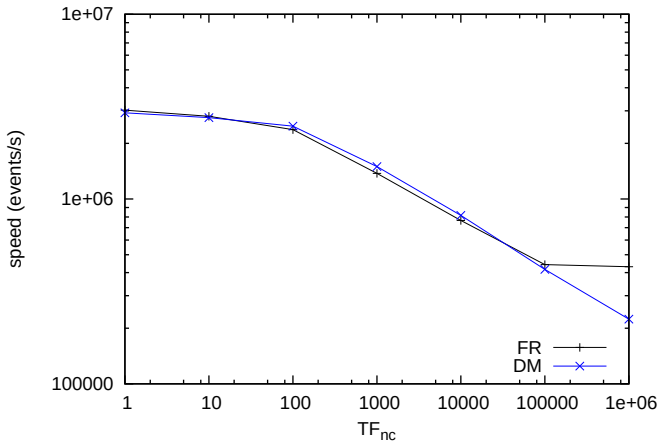

Fig. 7. *Simulation speed is inversely proportional with the number of TF molecules*. The system consists of 5 cognate TF and the number of non-cognate TF molecules is varied between 1 and 10^6.