Supplementary material for Modelling and parameter inference of predator-prey dynamics in heterogeneous environments using the direct integral approach

Itai Dattner*[1], Ezer Miller †[2], Margarita Petrenko[3], Daniel E. Kadouri‡[4], Edouard Jurkevitch§[3] and Amit Huppert¶[2,5]

[1]Department of Statistics, University of Haifa, 199 Abba Khoushy Ave, Mount Carmel, Haifa 3498838, Israel
[2]Bio-statistical Unit, The Gertner Institute for Epidemiology and Health Policy Research, Chaim Sheba Medical Center, Tel Hashomer, 52621 Israel
[3]Department of Agroecology and Plant Health, The Robert H. Smith Faculty of Agriculture, Food and Environment, The Hebrew University of Jerusalem,Jerusalem, Israel
[4]Department of Oral Biology, Rutgers School of Dental Medicine, Newark, NJ, USA
[5]Department of Epidemiology and Preventive Medicine at the School of Public Health, the Sackler Faculty of Medicine, Tel-Aviv University Israel

November 3, 2016

1 Direct integral estimation

In the following we apply the direct integral approach for estimating model parameters using predator prey data. The dynamic model is given by

$$\begin{aligned} P'(t) &= ksC(t) - dP(t), \\ C'(t) &= a(N(t) - r)P(t) - sC(t), \\ N'(t) &= -a(N(t) - r)P(t), \end{aligned} \tag{1}$$

*idattner@stat.haifa.ac.i
†ezermiller@gmail.com
‡kadourde@sdm.rutgers.edu
§edouard.jurkevitch@mail.huji.ac.il
¶amit.huppert@gmail.com

where the notation $P'(t)$ stands for dP/dt, and the same for C and N. Recall the abstract form given in equation (10) of the main text,

$$F(x(t);\theta) = g(x(t);\theta_{NL})\theta_L,$$

where $\theta = (\theta_{NL}^\top, \theta_L^\top)^\top$. Since the nonlinearity of the parameters in model (1) has a multiplicative form (i.e., ks and ar), there exist more than a single possible formulation for the matrix g and the parameters $\theta = (\theta_{NL}^\top, \theta_L^\top)^\top$. For example, let $x(t) = (P(t), C(t), N(t))^\top$, $\theta_{NL} = (k, r)^\top$, and $\theta_L = (s, d, a)^\top$. Then the matrix $g(x(t);\theta_{NL})$ corresponding to model (1) takes the form

$$\begin{pmatrix} kC(t) & -P(t) & 0 \\ -C(t) & 0 & (N(t)-r)P(t) \\ 0 & 0 & -(N(t)-r)P(t) \end{pmatrix}.$$

However, this is not the formulation used in practise, since that applying the method in steps and using other matrix forms can lead to better numerical stability of the algorithm. While studying the optimal formulation to use is of much interest, this is outside the scope of the current work. In the next section we describe in detail the estimation procedure we applied.

1.1 Nonparametric smoothing

It is assumed that the unknown system of equations can be approximated for any $t \in [0, T]$ by a linear combination of cubic B-spline functions denoted by $\phi_k(t)$, $k = 1, \ldots, K_\ell$, $\ell \in \{1, 2, 3\}$, namely,

$$\begin{array}{l} P(t) \approx \sum_{k=1}^{K_1} \beta_{1,k}\phi_k(t), \\ C(t) \approx \sum_{k=1}^{K_2} \beta_{2,k}\phi_k(t), \\ N(t) \approx \sum_{k=1}^{K_3} \beta_{3,k}\phi_k(t). \end{array} \tag{2}$$

In our experiment, both $P(\cdot)$ and $N(\cdot)$ are observed, enabling estimation of the coefficients vectors $\beta_1 = (\beta_{1,1}, \ldots, \beta_{1,K_1})^\top$ and $\beta_3 = (\beta_{3,1}, \ldots, \beta_{3,K_3})^\top$ from (2) in a straight forward manner (the estimation of $C(\cdot)$ is described later). Let $\widetilde{P}_n = (\widetilde{P}(t_1), \ldots, \widetilde{P}(t_n))^\top$, and $\widetilde{N}_n = (\widetilde{N}(t_1), \ldots, \widetilde{N}(t_n))^\top$ be the observed noisy versions of $P(\cdot)$, and $N(\cdot)$ on the grid $0 \le t_1 < \cdots < t_n = T < \infty$. Denote by Φ_ℓ the $n \times K_\ell$ matrix composed of the column vectors $(\phi_j(t_1), \cdots, \phi_j(t_n))^\top$, $j = 1, \ldots, K_\ell$, $\ell \in \{1, 3\}$. Then we define

$$\begin{array}{l} \widehat{\beta}_1 = (\Phi_1^\top \Phi_1)^{-1} \Phi_1^\top \widetilde{P}_n, \\ \widehat{\beta}_3 = (\Phi_3^\top \Phi_3)^{-1} \Phi_3^\top \widetilde{N}_n. \end{array}$$

The estimators for $P(\cdot)$ and $N(\cdot)$ are given by

$$\begin{array}{l} \widehat{P}(t) = \sum_{k=1}^{K_1} \widehat{\beta}_{1,k}\phi_k(t), \ t \in [0, T], \\ \widehat{N}(t) = \sum_{k=1}^{K_3} \widehat{\beta}_{3,k}\phi_k(t), \ t \in [0, T]. \end{array} \tag{3}$$

Practically the evaluation of the estimators above is done on a finite grid $0 \le t_1 < \cdots < t_m = T < \infty$, not necessarily identical to (actually denser than) the original sampling points.

The following procedure was used for estimating the parameters of model (1). In the first step we estimate a and r; in the second we estimate k, and d over a grid of values for s. Finally, in the third step we choose the best s value and as a result, the final estimators for k and d.

1.2 Estimating a and r

Looking at the third equation in (1) one can see that $N(t) = r$ whenever $dN/dt = 0$, which in this case occurs in the tail of the experiment; see Figure 1 of the main text. However, data at the tails are sparse with only two observations. This fact made the estimation task quite challenging since effectively, there is not much information in the observations for estimating r. In practise, in each experiment the mean of the last two observations was used as an estimate for r. Given $\widehat{r}$, the estimator for r, it is possible to estimate a as follows. According to the third equation in (1) we have

$$N(t) - N(0) = -a \int_0^t N(v)P(v)\mathrm{d}v + ar \int_0^t P(v)\mathrm{d}v.$$

Recall that in our experiment, the initial values $P(0)$ and $N(0)$ are known. The form above implies that one can use a least squares type of estimator for a. Indeed, let $\widehat{G}_{3i} = (-\int_0^{t_i} \widehat{N}(v)\widehat{P}(v)\mathrm{d}v + \widehat{r}\int_0^{t_i} \widehat{P}(v)\mathrm{d}v)$, where $\widehat{N}(\cdot)$ and $\widehat{P}(\cdot)$ are given in (3). Let $\widehat{G}_3$ be the $m \times 1$ vector $(\widehat{G}_{31}, \ldots, \widehat{G}_{3m})^\top$. Let $\widehat{N} = (\widehat{N}(t_1) - N(0), \ldots, \widehat{N}(t_m) - N(0))^\top$, then we have

$$\widehat{a} = (\widehat{G}_3^\top \widehat{G}_3)^{-1} \widehat{G}_3^\top \widehat{N}.$$

1.3 Estimating k and d for a given s

Consider the first equation in (1) according to which we have

$$P(t) - P(0) = ks \int_0^t C(v)\mathrm{d}v - d \int_0^t P(v)\mathrm{d}v. \tag{4}$$

If we fix s, then plugging estimators for the system states leads naturally to an ordinary least squares type of solution for k and d. However, $C(\cdot)$ is an unobserved function. The estimation of $C(\cdot)$ is conducted by noting that according to (1), (2) and the fact that $C(0) = 0$ we have

$$\begin{aligned} N(t) - N(0) &= -C(t) - s \int_0^t C(v)\mathrm{d}v \qquad (5) \\ &\approx \sum_{k=1}^{K_2} [-\phi_k(t) - s \int_0^t \phi_k(v)\mathrm{d}v]\beta_{2,k}. \end{aligned}$$

Let $\Phi_i = (-\phi_1(t_i) - s\int_0^{t_i} \phi_1(v)\mathrm{d}v, \ldots, -\phi_{K_2}(t_i) - s\int_0^{t_i} \phi_{K_2}(v)\mathrm{d}v)$. Let Φ_s be the $m \times K_2$ matrix with rows $\Phi_1, \ldots, \Phi_m$. Then we define $\widehat{\beta}_2$ of (2) to be

$$\widehat{\beta}_2(s) = (\Phi_s^\top \Phi_s)^{-1} \Phi_s^\top \widehat{N}.$$

Using s in the notation above stresses that the estimator $\widehat{\beta}_2$ depends on s. Plugging in, one gets

$$\widehat{C}_s(t) = \sum_{k=1}^{K_2} \widehat{\beta}_{2,k}(s)\phi_k(t),\ t \in [0,T].$$

Now define $\widehat{P} = (\widehat{P}(t_1) - P(0), \ldots, \widehat{P}(t_m) - P(0))^\top$, $\widehat{G}_{1i} = (s\int_0^{t_i} \widehat{C}_s(v)\mathrm{d}v, -\int_0^{t_i} \widehat{P}(v)\mathrm{d}v)$, and let $\widehat{G}_1$ be the matrix with rows $\widehat{G}_{11}, \ldots, \widehat{G}_{1m}$. Then, with (4) in mind one obtains

$$(\widehat{k}(s), \widehat{d}(s))^\top = (\widehat{G}_1^\top \widehat{G}_1)^{-1} \widehat{G}_1^\top \widehat{P}. \tag{6}$$

1.4 Estimating s

Looking at the sum of (4) and (5) (i.e., based on the first and second equations of (1)) consider

$$\widehat{s} \ = \ \arg\min_s M(s), \tag{7}$$

where we have defined

$$\begin{aligned} M(s) \ = \ & \sum_{i=1}^{m} (\widehat{P}(t_i) - P(0) + \widehat{N}(t_i) - N(0) - s(\widehat{k}(s) - 1) \int_0^{t_i} \widehat{C}_s(v)\mathrm{d}v \\ & + \widehat{C}_s(t_i) + \widehat{d}(s) \int_0^{t_i} \widehat{P}(v)\mathrm{d}v)^2. \end{aligned} \tag{8}$$

Note that the above loss function is different from the Euclidean norm presented in equation (11) of the main text. In practise, equation (8) yields good estimations. Further note that this loss is the only nonlinear optimisation used in the fitting procedure. However, also here one can avoid using nonlinear optimisation algorithms by evaluating the above loss over a grid and looking for a minimum, both visually and numerically, see Figure 2. This is demonstrated in the main text for both real and synthetic data. The final estimators for k and d according to equations (6)-(7) are given by

$$\begin{aligned} \widehat{k} &= \widehat{k}(\widehat{s}), \\ \widehat{d} &= \widehat{d}(\widehat{s}). \end{aligned}$$

1.5 Performance of the estimation method for various noise levels

To study the robustness of the direct integral approach to various noise levels a simulation study was conducted using Monte Carlo experiments. Model (1) was solved for $\xi = (P(0), C(0), N(0))^\top = (10^6, 0, 10^8)^\top$, similar to the experimental design, and the 'true' $\theta = (k, s, d, a, r)^\top = (5, 0.05, 0.02, 4 \times 10^{-9}, 3 \times 10^5)^\top$ (values that are close to those estimated from the real data, see Table 2 of the main text). The statistical model considered is additive, meaning that Gaussian measurement errors were added to the deterministic solutions of equation (1). For instance, the measurements of the predator will be given by

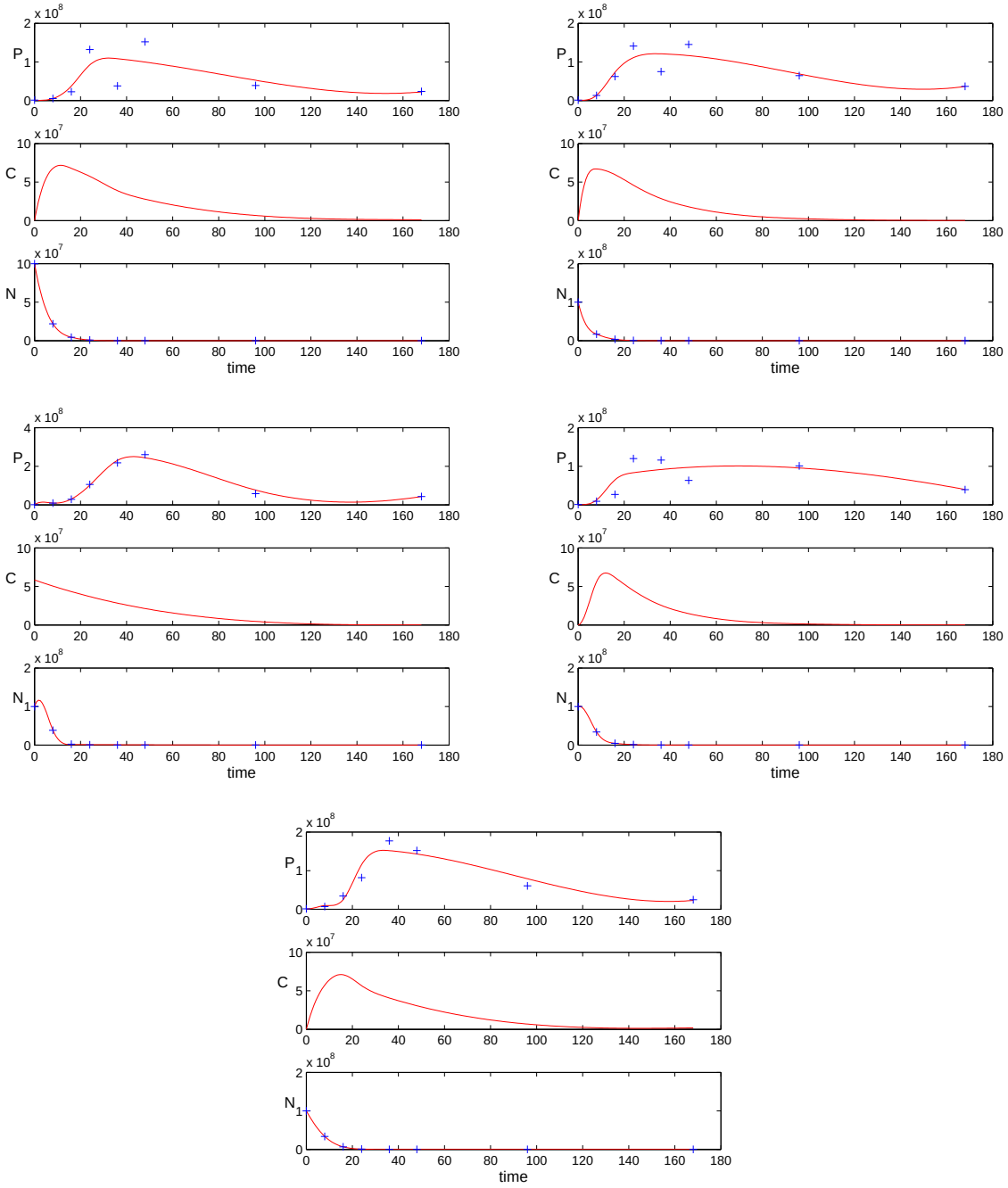

Figure 1: The first step of the method requires smoothing. The spline fits are displayed with the solid line. Observations are displayed with plus sign. Water content: upper left - 20%, upper right - 50%, middle left - 70%, middle right - 80%, bottom - 100%.

$\widetilde{P}(t_i) = P(t_i;\theta,\xi) + \epsilon_i$, $i = 1,\ldots,n$, where $P(t_i;\theta,\xi)$ is the deterministic solution of (1) at point t_i with respect to initial values ξ and parameter θ. The measurement error ϵ_i follows a Gaussian distribution with zero expectation and standard deviation that is proportional to both P and N: $\sigma_P(t_i) = \sigma \times P(t_i)$, and $\sigma_N(t_i) = \sigma \times N(t_i)$ where $\sigma \in \{0.01(0.01)0.1\}$. Note

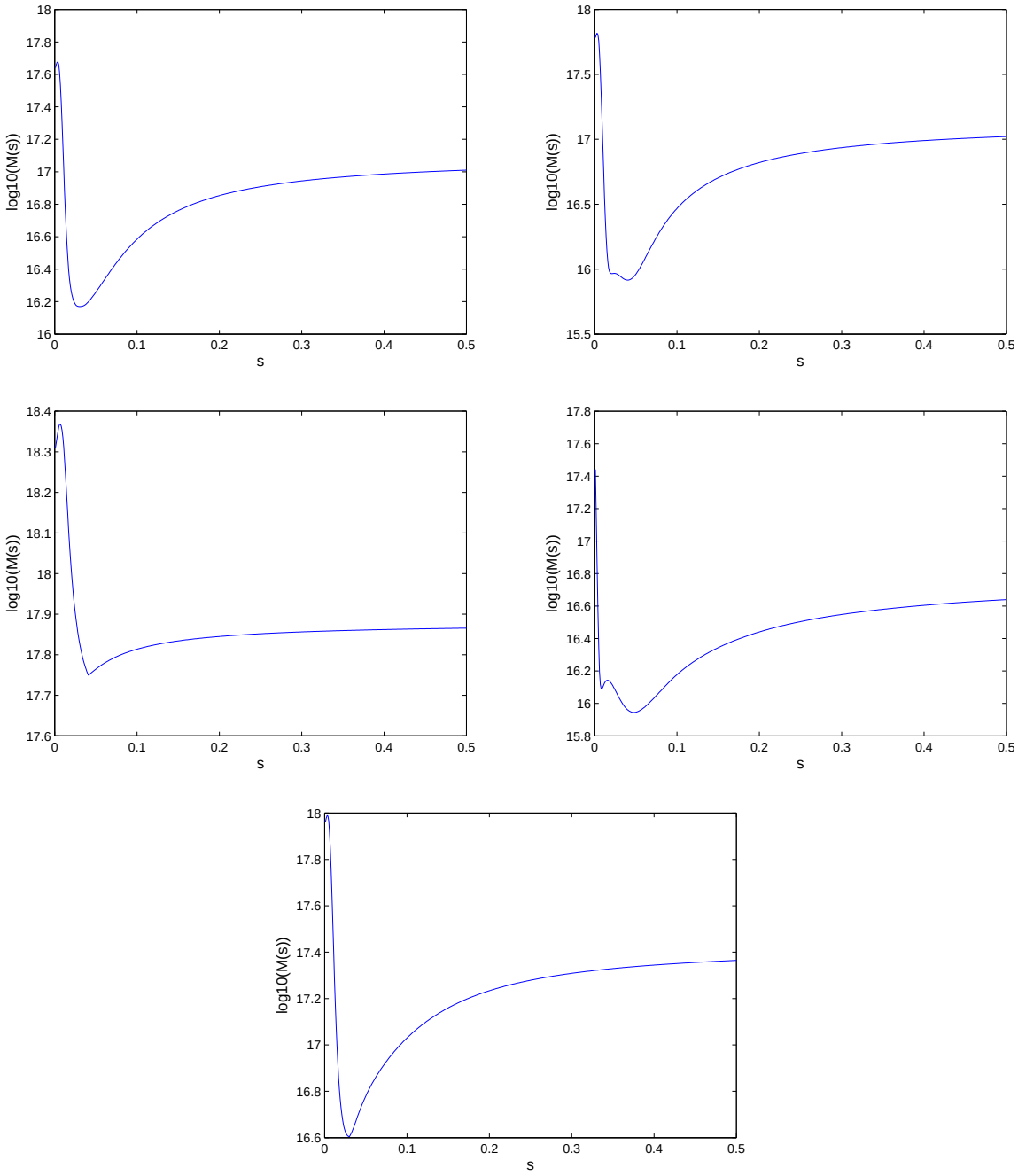

Figure 2: The loss function $M(s)$ given in (8). A clear minimum is apparent here. Water content: upper left - 20%, upper right - 50%, middle left - 70%, middle right - 80%, bottom - 100%.

that unlike the Monte Carlo experiment reported in the main text, here the variance was not constant over time, and we tested the performance for various noise levels. In the Monte Carlo experiment 100 different random samples were generated as described above. The time points were set to be exactly as in the real experiment $[0, 8, 16, 24, 36, 48, 96, 168]$, hence,

$n = 8$. Further, the number of spline bases was fixed to $K = \{K_1, K_2, K_3\} = \{11, 5, 9\}$. This bases setup was found to work well when the noise was as in the Monte Carlo experiment reported in the main text. Implying that the direct integral method as applied here was challenged with a non-optimal setup.

The simulation result is presented in Figure 3. On the vertical axis we see the square

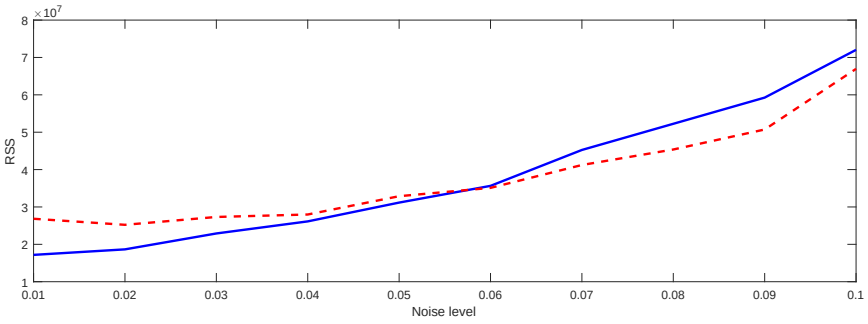

Figure 3: Square root of sum of squares for various noise levels. The direct integral method in solid line; the nonlinear least squares in dashed line.

root of sum of squares, and on the horizontal axis the noise level (σ above). The sum of squares was calculated as follows: The model equations (1) were solved using the estimated parameters values, and next we summed the squared differences between the solutions of the ODEs to the noisy observations, over the first and third equations. The performance of the direct integral method is presented in solid line and as expected, the sum of squares increases together with the noise level. In order to assess the performance of the direct integral method in this specific scenario, a weighted nonlinear least squares (NLS) was executed. Taking into account the fact that the variance of the measurement error is not fixed in time, we used the true values of the variance for the weights. Figure 3 suggests that for small noise levels the performance of the direct integral method is better than that of the NLS, while for larger noise levels, the NLS performs better. Note that in the setup considered here, the performance of the two methods was comparable without substantial differences. In Figure 4 one can see the 'true' simulated values versus the error-laden observations, enabling a visual sense of the imposed error.

1.6 Confidence intervals

Here we present a way to calculate confidence intervals for the parameters of interest. Specifically, due to the limited amount of data, we explore the reliability of the parametric bootstrap in our context, via a simulation study.

The following Monte Carlo experiment was preformed. The predator-prey ODE system given in equation (1) was solved for initial conditions given by $\xi = (P(0), C(0), N(0))^\top = (10^6, 0, 10^8)^\top$, and 'true' parameters given by $\theta = (k, s, d, a, r)^\top = (5, 0.05, 0.02, 4 \times 10^{-9}, 3 \times 10^5)^\top$. In this simulation the measurement error ϵ_i follows a Gaussian distribution with zero expectation and standard deviation that is proportional to both P and N: $\sigma_P = 0.01 \times \bar{P}$, and $\sigma_N = 0.01 \times \bar{N}$ where $\bar{P}$, and $\bar{N}$ are the means of $P(\cdot)$, and $N(\cdot)$ over the time interval of the experiment, respectively.

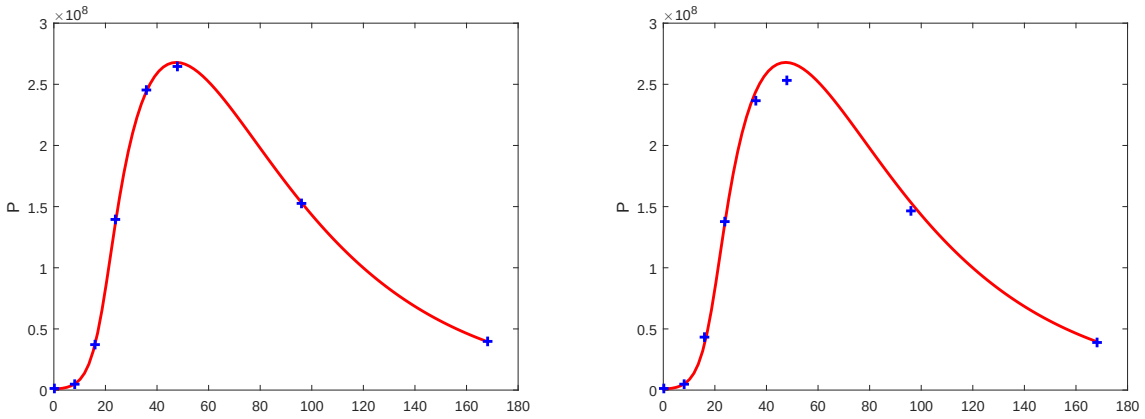

Figure 4: Visualising noise level. solution P of model (1) in solid, and observations in plus signs. Left: noise level $\sigma = 0.01$. Right: noise level $\sigma = 0.1$.

Recall that the parametric bootstrap procedure assumes that the distribution of the error is known, Gaussian in this case. Confidence intervals for k, s, d, a, r using the direct integral estimator, resulted in a coverage of $0.72, 0.37, 0.38, 0.005, 0.94$ for $n = 8$ and $0.78, 0.845, 0.81, 0.77, 0.725$ for $n = 16$, respectively (200 Monte Carlo simulations, 300 bootstrap samples in each). As expected, increasing (doubling) sample size improved the confidence intervals coverage for all parameters except for r in this setting. However, the bootstrap procedure had to assume that σ is known, otherwise, using the bootstrap by estimating σ with such a small sample size was numerically unstable. The above results are not surprising in the context of the complex nonlinear problem tackled here. To further investigate the coverage of the confidence intervals, we repeated the above exercise using nonlinear least squares (NLS). When executing NLS for $n = 8$ by **starting the nonlinear optimisation from the true parameters** the coverage of the confidence intervals for k, s, d, a, r was $0.91, 0.91, 0.88, 0.93, 0.9$, respectively (200 Monte Carlo simulations, 300 bootstrap samples in each). However, using the NLS by starting the nonlinear optimisation from a random point (normal random variable with mean values the true parameters k, s, d, a, r and standard deviation of $0.01 \times (k, s, d, a, r)$) resulted in a coverage of $0.76, 0.66, 0.76, 0.38, 0$. Thus, the coverage of the confidence intervals shrunk considerably with a relative small change in the parameters used for initiating the optimisation. For instance, starting the optimisation from the parameters estimated by the direct integral method, resulted in a dramatic reduction in the coverage, $0.26, 0.17, 0.27, 0.14, 0$, respectively. Here also we considered σ to be known, otherwise applying the bootstrap by estimating σ was numerically unstable. Although the NLS (or maximum likelihood) has theoretically all the required optimal statistical properties, it seems that unless a much more expensive optimisation procedure is carried out (e.g., global optimisation), the NLS may practically have no reliable uncertainty quantification in a complex setup as the one considered here. In a future work we plan to study further the problem of uncertainty quantification in this context.