

EBTtool
Exercice d'introduction

Vincent Le Burlot David Claessen

5 décembre 2011

Exercice Introduction à EBTtool

Cet exercice a pour but de montrer par un exemple suivi comment utiliser le logiciel de modélisation EBTtool. Il est accompagné du guide rapide d'utilisation et de l'archive `TDebt-forstudents.zip`.

I — Ouverture de l'archive

Ouvrez l'archive fournie dans le dossier de votre choix. L'archive contient les différents fichiers utilisés par le logiciel EBTtool.

1. a) Quel est le nom et l'extension du fichier contenant le code source du modèle ?
b) Quel est le nom et l'extension du header ? Que contient-il ?
2. A quoi servent les fichiers `runkm` ?
3. Combien y a-t-il de fichiers graphiques ?
4. Quel est le fichier projet ?

II — Lecture du code source

Dans code source du modèle, repérez les éléments suivants.

1. Le nombre et le nom des variables d'environnement.
2. Le nom du 5^{eme} paramètre.
3. L'équation différentielle de la ressource et la partie correspondant à la consommation.
4. La ligne correspondant à la réponse fonctionnelle. De quel type de réponse s'agit-il ?
5. A quoi correspond `lstarve` ? Que se passe-t-il si la longueur d'une cohorte dépasse `lstarve` ?
6. Identifiez les équations différentielles de croissance individuelle, et du nombre d'individus.
7. Identifiez les équations différentielles des nouveaux-nés.
8. Identifiez la ligne définissant la 5^{eme} colonne du fichier de sortie `.out`. Quel est l'indice de cet sortie dans le code source ? Pourquoi ce décalage entre l'indice et le numéro de colonne ?

III — Ouverture d'un modèle sous EBTtool et première simulation
Démarrez le logiciel EBTtool et ouvrez le projet.

1. Comment s'appelle le modèle choisi ?
2. Comment s'appelle le run choisi ?
3. Comment s'appelle le fichier où seront enregistrées les sorties ?
4. Lancez une simulation en cliquant sur la flèche verte. Que se passe-t-il ? Pourquoi ?
5. Cliquez sur "compile", la compilation s'effectue et le modèle s'exécute.
 - a) Notez les paramètres utilisés.
 - b) Que peut-on dire de la dynamique de la population ?
 - c) Décrivez la structure de la population à $t=80$, $t=150$ et $t=250$ et exportez une image de chacune de ces distributions.
 - d) Proposez un mécanisme pour expliquer ces observations. Pour vous aider, tracez le nombre d'adultes en fonction du nombre de jeunes.
 - e) Qu'en est-il de la biomasse ?
6. Vérifiez numériquement la stabilité du cycle limite.
 - a) Exportez les conditions finales comme nouvel état initial (créez un nouveau fichier, sans écraser l'ancien).
 - b) Copiez le fichier `.cvf` et donnez lui le même nom qu'au nouveau fichier `.isf`. Lancez une simulation avec ces nouvelles conditions initiales. La dynamique est-elle modifiée ?
 - c) Perturbez le cycle limite en modifiant le nouvel état initial (par exemple en modifiant le nombre de cohortes, le nombre d'individus dans chaque cohortes. . .). Qu'est-ce que cela change ? Qu'en déduisez vous sur la stabilité du cycle ? Si la simulation n'est pas assez longue pour atteindre un équilibre, allongez le temps de simulation dans les paramètres.

IV — Effet de la mortalité et de la taille à maturité sur l'équilibre de la population

Nous allons ici tester l'effet des paramètres MUC et LJ sur la dynamique de la population et sur sa structure. Choisissez pour chaque groupes 10 jeux de paramètres différents (10 paires MUC, LJ). Répartissez ces paires entre les différents groupes afin de quadriller l'espace (MUC, LJ) entre les valeurs suivante :

– MUC $\in [0.01, 0.05]$

– LJ $\in [50, 250]$

1. Pour chaque paire de paramètres, effectuez une simulation jusqu'à atteindre l'équilibre. Caractérisez rapidement la dynamique de la population.
 - a) Cycles : Amplitude, période, pour les adultes, les juvéniles et la population totale ? Superposition des générations ou non ? Taille maximum réalisée ?
 - b) Point fixe : Taille de la population totale, d'adultes, de juvéniles ?
Forme de la distribution de la taille ? Taille maximum réalisée ?
2. Conservez dans un tableau pour chaque paire de paramètres le nombre maximum et minimum d'individus, d'adultes et de juvéniles, ainsi que l'amplitude et la période des oscillations en cas de cycles (0 en cas de point fixe). Une fois que toutes les simulations sont terminées, récupérez ces tableaux auprès des différents groupes.
 - a) Tracez sur un même graphique (dans R ou Excel) le nombre maximum et minimum d'individus en fonction de la mortalité pour une valeur intermédiaire de LJ. Commentez.
 - b) Tracez sur un même graphique (dans R ou Excel) le nombre maximum et minimum d'individus en fonction de la taille à maturité pour une valeur intermédiaire de MUC. Commentez.
 - c) Tracez l'amplitude des oscillations en fonction de MUC et LJ (utilisez par exemple la fonction `levelplot` de la librairie `lattice` sous R <http://stat.ethz.ch/R-manual/R-devel/library/lattice/html/levelplot.html> ou les graphiques en surface sous excel <http://office.microsoft.com/fr-fr/excel-help/creation-d-un-graphique-en-surface-HA001117939.aspx?CTT=1>). Tracez la courbe délimitant la zone de cycles et la zone de points fixes. Commentez.s

Solution de l'exercice

I — Ouverture de l'archive

1. a) nom : `km`, extension : `.c`
b) nom : `km`, extension : `.h`. Ce fichier contient des variables statiques du modèle, telles que le nombre de populations, le nombre de variable d'états, le nombre de paramètres, etc...
2. Les fichiers `runkm` contiennent les paramètres du modèle (`runkm.cvf`) et l'état initial de la population (`runkm.isf`).
3. Il y a trois fichiers graphiques. Deux sont des graphiques de sortie (`.ebtogr`) et un est un graphique d'état (`.ebtsgr`).
4. Le fichier projet est `kmodel.ebtp`.

II — Lecture du code source

1. Il y a deux variables d'environnement, `time` et `resource` (respectivement `env[0]`, et `env[1]`).
2. Le cinquième paramètre est `LM`. Attention, il est indicé `parameter[4]` car les indices commencent à 0 en C.
- 3.

$$\text{envgrad}[1] = \text{RHO} * (\text{K} - \text{resource}) - \underbrace{\text{IM} * \text{Fr} * \text{grazing} / \text{LAKEVOLUME}}_{\text{consommation}}$$

4. `Fr = resource / (RH + resource)`; il s'agit d'une réponse fonctionnelle de type II.
5. `lstarve` détermine la longueur au-delà de laquelle un individu (cohorte) est en famine. Il est calculé par `lstarve = (LM * resource / (RH + resource)) / KAPPA`. Lorsqu'une cohorte est plus grande que `lstarve`, les valeurs suivantes sont données aux équations différentielles :
`popgrad[0][i][number] = -pop[0][i][number];`
`popgrad[0][i][length] = 1.0;`
dans la fonction `Gradient`, puis est supprimée
(`if (pop[0][i][length] > lstarve) pop[0][i][number] = 0.0;`)
dès le passage suivant par la fonction `InstantDynamics`.
6. `popgrad[0][i][number] = -mort * pop[0][i][number];`

- ```

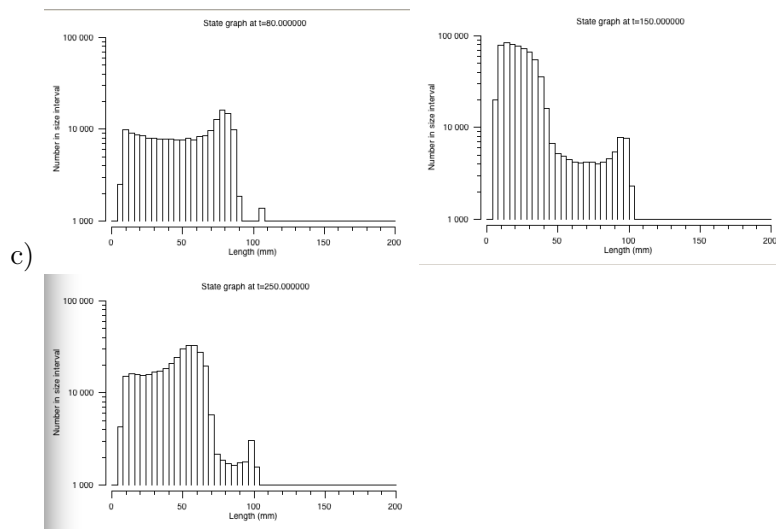
popgrad[0][i][length] = max(GAMMA*(lmax-len),0.0);
7. ofsgrad[0][0][number] = -mort*ofs[0][0][number] + RM*newborns;
ofsgrad[0][0][length] = -mort*ofs[0][0][length] +
ofs[0][0][number]*GAMMA*(lmax - LB) - GAMMA*ofs[0][0][length];
8. La cinquième colonne est définie à la ligne output[3] += mass; et
correspond à la biomasse totale de la population. L'indice est 3 dans
le code source car les indices commencent à 0 en C et la première
colonne correspond au temps, ce qui décale le compte de 2.

```

### III — Ouverture d'un modèle sous EBTtool et première simulation

1. Le modèle s'appelle `km` (pour Koojmann & Metz).
2. Le run s'appelle `runkm`.
3. Les sorties seront enregistrées dans le fichier `runkm.out`.
4. En cliquant sur la flèche verte, une fenêtre s'ouvre proposant de compiler le modèle car cela n'ayant pas été fait au par avant, il n'y a pas de fichier exécutable dans le dossier courant.
5. a)

|                                                                |         |          |
|----------------------------------------------------------------|---------|----------|
| Semi-chemostat renewal rate (1/d)                              | RHO     | " 0.1    |
| Resource carrying capacity (g/L)                               | K       | " 3.0E-4 |
| Length at birth (mm)                                           | LB      | " 7.0    |
| Length at maturation (mm)                                      | LJ      | " 110.0  |
| Length at infinite food (mm)                                   | LM      | " 300.0  |
| Maximum ingestion rate scaling constant (g/d.mm <sup>2</sup> ) | IM      | " 1.0E-4 |
| Functional response half-saturation constant (g/L)             | RH      | " 1.5E-5 |
| vonBertalanffy growth rate parameter (/d)                      | GAMMA   | " 0.006  |
| Adult fecundity scaling constant (1/d.mm <sup>2</sup> )        | RM      | " 0.003  |
| Consumer background mortality (1/d)                            | MUC     | " 0.015  |
| Fraction of assimilation to growth+metabolism (-)              | KAPPA   | " 0.7    |
| Scaling constant of starvation mortality (1/d)                 | MUS     | " 0.2    |
| Threshold fraction for onset starvation mortality (-)          | QS      | " 0.2    |
| Weight-cubed length scaling constant (g/mm <sup>3</sup> )      | LWCONST | " 9.0E-6 |
- b) On observe que la population oscille autour d'un cycle limite stable.

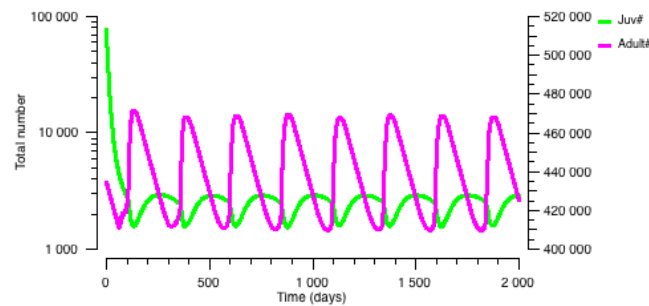


c)

On observe une vague d'individus de juvéniles qui se propage jusqu'à atteindre la maturité où il se reproduisent et sont remplacés par la vague suivante.

d) A cause des hypothèses sur la gestion du budget énergétique, et en particulier du fait que le métabolisme est proportionnel à  $l^3$  alors que l'ingestion est proportionnelle à  $l^2$ , les individus sont d'autant plus compétitifs qu'ils sont petits. Ainsi, lorsqu'un grand nombre de juvéniles sont présents dans la population, ils ont tendance à exclure les adultes, provoquant un ralentissement de la reproduction et créant la vague observée. Lorsque cette vague atteint la maturité, on a d'un seul coup beaucoup d'individus reproducteurs qui produisent un grand nombre de jeunes, se faisant du coup exclure par la compétition, produisant la vague suivante. On appelle cela des cycles de génération.

e) Le cycle limite est aussi visible pour la biomasse, mais l'amplitude des oscillations es beaucoup plus faible chez les juvéniles.



6. a) -  
b) La dynamique est identique.  
c) La période transitoire dépend de la condition initiale choisi, mais le système converge vers le même cycle limite. Avec des conditions initiales très différentes, par exemple  
0.000000 6.543571E-06 2.632635

881.400749 7.075620 4999.850059 0.504272 0.000000 1.313700E-05  
0.000000

881.400749 115.075620 4999.850059 0.504272 0.000000 1.313700E-05  
0.000000

On peut obtenir des cycles différents, sans superposition de générations.