# DAFTAR PUSTAKA


Hutabarat, S. dan S.M, Evans, 1985. **Pengantar Oseanografi**. Universitas Indonesia Press Jakarta.

Hutagalung, H.P., D. Setiapermana dan S.H. Riyono., 1997. **Metode Analisis Air Laut, Sedimen dan Biota**. Buku 2. Puslitbang Oseanologi LIPI. Jakarta.

Nontji, Anugrah. 2005. **Laut Nusantara Djambatan**. Jakarta.

Nontji, Anugrah. 2008. **Plankton Laut**. Lembaga Ilmu Pengetahuan Indonesia Pusat Penelitian Osenografi. Jakarta.

Nybakken, J. W. 1992. **Biologi Laut Suatu Pendekatan Ekologi**. Gramedia Jakarta

Stewart, R. 2004. **Marine Genomics Europe**, http://www.marine-genomics-europe.org. diakses pada 26 Januari 2011

Verhulst, Ferdinand, 1990, **Nonlinear Differential Equation and Dynamical System**, Springer-Verlag., Jerman.

# Lampiran

## *Lampiran 1: Analitik*

```
> restart : with(linalg) :
> μ[1] := 1.9 : μ[2] := 1.5 : K[N1] := 0.15 : K[N2] := 0.6 : K[I1] := 30 : K[I2] := 5 : m[p] := 0.145 : g := 1.5 : K[Z] := 1.4 : β := 0.2 : ε := 0.06 : τ := 0.1 : C[0] := 1.2 :
```

$$> \alpha[1] := \mu[1]\left(1 - e^{\frac{-1}{K[I1]}}\right) : \alpha[2] := \mu[2]\left(1 - e^{\frac{-1}{K[I2]}}\right) :$$

$$> d := C[0] - N - P[1] - P[2] - Z :$$

$$> p1 := \tau\cdot(C[0] - N - P[1] - P[2] - Z) - \alpha[1]\cdot\frac{N}{K[N1] + N}\cdot P[1] - \alpha[2]\cdot\frac{N}{K[N2] + N}\cdot P[2] : p2 := \left(\alpha[1]\cdot\frac{N}{K[N1] + N} - m[p]\right)\cdot P[1] - g\cdot\left(\frac{P[1]}{K[Z] + P[1] + P[2]}\right)\cdot Z : p3 := \left(\alpha[2]\cdot\frac{N}{K[N2] + N} - m[p]\right)\cdot P[2] - g\cdot\left(\frac{P[2]}{K[Z] + P[1] + P[2]}\right)\cdot Z : p4 := \left(g\cdot\beta\cdot\frac{P[1] + P[2]}{K[Z] + P[1] + P[2]} - \varepsilon\right)\cdot Z :$$

```
> solusi := solve({p1, p2, p3, p4}, {N, P[1], P[2], Z}) :
> vek1 := vector([p1, p2, p3, p4]) :
> jcob := jacobian(vek1, [N, P[1], P[2], Z]) :
> job1 := subs(solusi[1], evalm(jcob)) :
> pkar1 := charpoly(job1, λ) :
> eigenvalues(job1) :
> job2 := subs(solusi[2], evalm(jcob)) :
> pkar2 := charpoly(job2, λ) :
> eigenvalues(job2) :
> job3 := subs(solusi[3], evalm(jcob)) :
> pkar3 := charpoly(job3, λ) :
> eigenvalues(job3) :
> job4 := subs(solusi[4], evalm(jcob)) :
> pkar4 := charpoly(job4, λ) :
> eigenvalues(job4) :
> job6 := subs(solusi[6], evalm(jcob)) :
> pkar6 := charpoly(job6, λ) :
> eigenvalues(job6) :
```

## *Lampiran 2: Simulasi Numerik dengan Parameter Tertentu*

```
> restart : with(linalg) :
> μ[1] := 1.9 : μ[2] := 1.5 : K[N1] := 0.15 : K[N2] := 0.6 : K[I1]
      := 30 : K[I2] := 5 : m[p] := 0.145 : g := 1.5 : K[Z] := 1.4 : β
      := 0.2 : ε := 0.06 : τ := 0.1 : C[0] := 1.2 :
```

$$> \alpha[1] := \mu[1]\left(1 - e^{\frac{-1}{K[I1]}}\right) : \alpha[2] := \mu[2]\left(1 - e^{\frac{-1}{K[I2]}}\right) :$$

$$> d := C[0] - N - P[1] - P[2] - Z :$$

$$> p1 := \tau\cdot(C[0] - N - P[1] - P[2] - Z) - \alpha[1]\cdot\frac{N}{K[N1] + N}\cdot P[1] - \alpha[2]\cdot\frac{N}{K[N2] + N}\cdot P[2] : p2 := \left(\alpha[1]\cdot\frac{N}{K[N1] + N} - m[p]\right)\cdot P[1] - g\cdot\left(\frac{P[1]}{K[Z] + P[1] + P[2]}\right)\cdot Z : p3 := \left(\alpha[2]\cdot\frac{N}{K[N2] + N} - m[p]\right)\cdot P[2] - g\cdot\left(\frac{P[2]}{K[Z] + P[1] + P[2]}\right)\cdot Z :$$

$$p4 := \left(g\cdot\beta\cdot\frac{P[1] + P[2]}{K[Z] + P[1] + P[2]} - \varepsilon\right)\cdot Z :$$

```
> solusi := solve({p1, p2, p3, p4}, {N, P[1], P[2], Z}) :
> vek1 := vector([p1, p2, p3, p4]) :
> jcob := jacobian(vek1, [N, P[1], P[2], Z]) :
> job1 := subs(solusi[1], evalm(jcob)) :
> pkar1 := charpoly(job1, λ) :
> eigenvalues(job1) :
> job2 := subs(solusi[2], evalm(jcob)) :
> pkar2 := charpoly(job2, λ) :
> eigenvalues(job2) :
> job3 := subs(solusi[3], evalm(jcob)) :
> pkar3 := charpoly(job3, λ) :
> eigenvalues(job3) :
> job4 := subs(solusi[4], evalm(jcob)) :
> pkar4 := charpoly(job4, λ) :
> eigenvalues(job4) :
> job6 := subs(solusi[6], evalm(jcob)) :
> pkar6 := charpoly(job6, λ) :
> eigenvalues(job6) :
> sol := solve({p1, p2, p3, p4}, {N, P[1], P[2], Z}) :
> with(plots) :
```

$$sys := \frac{d}{dt}N(t) = \tau\cdot(C[0] - N(t) - P[1](t) - P[2](t) - Z(t)) - \alpha[1] \cdot \frac{N(t)}{K[N1] + N(t)}\cdot P[1](t) - \alpha[2]\cdot \frac{N(t)}{K[N2] + N(t)}\cdot P[2](t), \frac{d}{dt}P[1](t) = \left(\alpha[1]\cdot \frac{N(t)}{K[N1] + N(t)} - m[p]\right)\cdot P[1](t) - g \cdot\left(\frac{P[1](t)}{K[Z] + P[1](t) + P[2](t)}\right)\cdot Z(t), \frac{d}{dt}P[2](t) = \left(\alpha[2] \cdot \frac{N(t)}{K[N2] + N(t)} - m[p]\right)\cdot P[2](t) - g \cdot\left(\frac{P[2](t)}{K[Z] + P[1](t) + P[2](t)}\right)\cdot Z(t), \frac{d}{dt}Z(t) = \left(g\cdot\beta \cdot\frac{P[1](t) + P[2](t)}{K[Z] + P[1](t) + P[2](t)} - \varepsilon\right)\cdot Z(t) :$$

\> *fcns* := {N(t), P[1](t), P[2](t), Z(t)} :

\> *L* := *dsolve*({*sys*, N(0) = 1.1, P[1](0) = 0.01, P[2](0) = 0.01, Z(0) = 0.01}, *fcns*, *type* = *numeric*, *method* = *rkf45* ) :

\> *odeplot*(L, [[t, N(t), *color* = *RED*, *thickness* = 3]], 0 ..1000) :

\> *odeplot*(L, [[t, P[1](t), *color* = *blue*, *thickness* = 3]], 0 ..500) :

\> *odeplot*(L, [[t, P[2](t), *color* = *RED*, *thickness* = 3]], 0 ..100) :

\> *odeplot*(L, [[t, Z(t), *color* = *blue*, *thickness* = 3]], 0 ..500) :