

```

> F_UTL := [2*(1+gamma^2)*xD*b1 - (1+gamma^2)^2*(b2/3+b3
+b4) - (1+gamma^2)*(b2/3-b4) - (b2/3-b3)];
F_UTL := [2(1+γ²)xD b1 - (1+γ²)²(1/3 b2 + b3 + b4) - (1+γ²)(1/3 b2 - b4) - 1/3 b2 + b3]
> F_UTLx := expand(F_UTL);
F_UTLx := [2 xD b1 + 2 xD b1 γ² - b2 - γ² b2 - 2 γ² b3 - γ² b4 - 1/3 γ⁴ b2 - γ⁴ b3 - γ⁴ b4]
> F_UTL_x := collect(F_UTLx, [b1, b2, b3, b4], factor);
F_UTL_x := [2(1+γ²)xD b1 + (-γ² - 1/3 γ⁴ - 1)b2 - γ²(2+γ²)b3 - γ²(1+γ²)b4]
> F_UTL_xg := combine(F_UTL_x, power);
F_UTL_xg := [(2γ² + 2)b1 xD + (-γ² - 1/3 γ⁴ - 1)b2 + (-2 - γ²)γ² b3 + (-1 - γ²)γ² b4]
> F_UTT := -[2*(1+gamma^2)*xD*b1 - gamma^2*(b2/6 - b3/2)];
F_UTT := [-2(1+γ²)xD b1 + γ²(1/6 b2 - 1/2 b3)]
> F_UTTx := expand(F_UTT);
F_UTTx := [-2 xD b1 - 2 xD b1 γ² + 1/6 γ² b2 - 1/2 γ² b3]
> F_UTT_x := collect(F_UTTx, [b1, b2, b3], factor);
F_UTT_x := [-2(1+γ²)xD b1 + 1/6 γ² b2 - 1/2 γ² b3]
> A_zz_b1 := F_UTT_x + epsilon * F_UTL_x;
A_zz_b1 := [-2(1+γ²)xD b1 + 1/6 γ² b2 - 1/2 γ² b3]
+ ε [2(1+γ²)xD b1 + (-γ² - 1/3 γ⁴ - 1)b2 - γ²(2+γ²)b3 - γ²(1+γ²)b4]
> Azz_b1x := expand(A_zz_b1);
Azz_b1x := [2 xD b1 ε + 2 xD b1 γ² ε - b2 ε - γ² b2 ε - 2 γ² b3 ε - γ² b4 ε - 1/3 γ⁴ b2 ε - γ⁴ b3 ε
- γ⁴ b4 ε - 2 xD b1 - 2 xD b1 γ² + 1/6 γ² b2 - 1/2 γ² b3]
> Azz_b1 := collect(Azz_b1x, [b1, b2, b3, b4], factor);
Azz_b1 := [2 xD (1+γ²)(ε - 1) b1 + (-1/3 γ⁴ ε - ε - γ² ε + 1/6 γ²) b2 - 1/2 γ² (2 γ² ε + 1 + 4 ε) b3
- γ² ε (1+γ²) b4]
>

```