Worldsheet supersymmetry (this is tutorial 3 of the manual)

Cadabra not only deals with bosonic tensors, but also with fermionic objects, i.e. anti-commuting variables. A simple example on which to illustrate their use is the Ramond-Neveu-Schwarz superstring action. We will here show how one uses cadabra to show invariance of this action under supersymmetry (a calculation which is easy to do by hand, but illustrates several aspects of cadabra nicely). We will use a conformal gauge and complex coordinates.

We are going to use `\del` for derivatives, as well as some other shorthand notation, which first needs to be explained to cadabra so that it can be typeset properly.

```
\del{#}::LaTeXForm("\partial").
\delbar{#}::LaTeXForm("\bar{\partial}").
\eps::LaTeXForm("\epsilon").
\epsbar::LaTeXForm("\bar{\epsilon}").
\Psibar{#}::LaTeXForm("\bar{\Psi}").
```

Assigning property LaTeXForm to $\partial(\#)$.
Assigning property LaTeXForm to $\bar{\partial}(\#)$.
Assigning property LaTeXForm to ϵ.
Assigning property LaTeXForm to $\bar{\epsilon}$.
Assigning property LaTeXForm to $\bar{\Psi}(\#)$.
Now we can declare the properties of the various symbols.

```
{\del{#}, \delbar{#}}::Derivative.
{\Psi_\mu, \Psibar_\mu, \eps, \epsbar}::AntiCommuting.
{\Psi_\mu, \Psibar_\mu, \eps, \epsbar}::SelfAntiCommuting.
{\Psi_\mu, \Psibar_\mu, X_\mu}::Depends(\del,\delbar).
{i, \Psi_\mu, \Psibar_\mu, \eps, \epsbar, X_\mu}::SortOrder.
```

Assigning property Derivative to $\partial\#\,,\ \bar{\partial}\#\,.$
Assigning list property AntiCommuting to $\{\Psi_\mu,\ \bar{\Psi}_\mu,\ \epsilon,\ \bar{\epsilon}\}$.
Assigning property SelfAntiCommuting to $\Psi_\mu,\ \bar{\Psi}_\mu,\ \epsilon,\ \bar{\epsilon}$.
Assigning property Depends to $\Psi_\mu,\ \bar{\Psi}_\mu,\ X_\mu$.
Assigning list property SortOrder to $\{i,\ \Psi_\mu,\ \bar{\Psi}_\mu,\ \epsilon,\ \bar{\epsilon},\ X_\mu\}$.
All objects are by default commuting, so the bosons do not have to be declared separately. You can at any time get a list of the declared properties by using the command `@proplist`. Now we are ready to input the action density:

```
action:= \del{X_\mu} \delbar{X_\mu}
        + i \Psi_\mu \delbar{\Psi_\mu} + i \Psibar_\mu \del{\Psibar_\mu};
```

$$action\ := \partial X_\mu\,\bar{\partial} X_\mu\ + i\Psi_\mu\bar{\partial}\Psi_\mu\ + i\bar{\Psi}_\mu\partial\bar{\Psi}_\mu\,;$$

Observe how we wrapped the `del` and `delbar` operators around the objects on which they are supposed to act. We are now ready to perform the supersymmetry transformation. This is done by using the `substitute` algorithm,

```
@vary(%)( X_\mu       ->   i \epsbar \Psi_\mu + i \eps \Psibar_\mu,
          \Psi_\mu    -> - \epsbar \del{X_\mu},
          \Psibar_\mu -> - \eps \delbar{X_\mu} );
```

```
@distribute!(%);
@prodrule!(%);
```

$$action := \partial(i\bar{\epsilon}\Psi_{\mu} + i\epsilon\overline{\Psi}_{\mu})\,\bar{\partial}X_{\mu} + \partial X_{\mu}\,\bar{\partial}(i\bar{\epsilon}\Psi_{\mu} + i\epsilon\overline{\Psi}_{\mu}) - i\bar{\epsilon}\partial X_{\mu}\,\bar{\partial}\Psi_{\mu} + i\Psi_{\mu}\bar{\partial}(-\bar{\epsilon}\partial X_{\mu}) - i\epsilon\bar{\partial}X_{\mu}\,\partial\overline{\Psi}_{\mu} + i\overline{\Psi}_{\mu}\partial(-\epsilon\bar{\partial}X_{\mu});$$

$$action := \partial(i\bar{\epsilon}\Psi_{\mu})\,\bar{\partial}X_{\mu} + \partial(i\epsilon\overline{\Psi}_{\mu})\,\bar{\partial}X_{\mu} + \partial X_{\mu}\,\bar{\partial}(i\bar{\epsilon}\Psi_{\mu}) + \partial X_{\mu}\,\bar{\partial}(i\epsilon\overline{\Psi}_{\mu}) - i\bar{\epsilon}\partial X_{\mu}\,\bar{\partial}\Psi_{\mu} + i\Psi_{\mu}\bar{\partial}(-\bar{\epsilon}\partial X_{\mu}) - i\epsilon\bar{\partial}X_{\mu}\,\partial\overline{\Psi}_{\mu} + i\overline{\Psi}_{\mu}\partial(-\epsilon\bar{\partial}X_{\mu});$$

$$action := (\partial i\,\bar{\epsilon}\Psi_{\mu} + i\partial\bar{\epsilon}\,\Psi_{\mu} + i\bar{\epsilon}\partial\Psi_{\mu})\bar{\partial}X_{\mu} + (\partial i\,\epsilon\overline{\Psi}_{\mu} + i\partial\epsilon\,\overline{\Psi}_{\mu} + i\epsilon\partial\overline{\Psi}_{\mu})\bar{\partial}X_{\mu} + \partial X_{\mu}\,(\bar{\partial}i\,\bar{\epsilon}\Psi_{\mu} + i\bar{\partial}\bar{\epsilon}\,\Psi_{\mu} + i\bar{\epsilon}\bar{\partial}\Psi_{\mu}) + \partial X_{\mu}\,(\bar{\partial}i\,\epsilon\overline{\Psi}_{\mu} + i\bar{\partial}\epsilon\,\overline{\Psi}_{\mu} + i\epsilon\bar{\partial}\overline{\Psi}_{\mu}) - i\bar{\epsilon}\partial X_{\mu}\,\bar{\partial}\Psi_{\mu} + i\Psi_{\mu}(-\bar{\partial}\bar{\epsilon}\,\partial X_{\mu} - \bar{\epsilon}\bar{\partial}\partial X_{\mu}) - i\epsilon\bar{\partial}X_{\mu}\,\partial\overline{\Psi}_{\mu} + i\overline{\Psi}_{\mu}(-\partial\epsilon\,\bar{\partial}X_{\mu} - \epsilon\partial\bar{\partial}X_{\mu});$$

The `@prodrule` command has applied the Leibnitz rule on the derivatives, so that the derivative of a product becomes a sum of terms. We expand again the products of sums, and use `@unwrap` to take everything out of the derivatives which does not depend on it,

```
@distribute!(%):
@unwrap!(%):
@prodsort!(%);
```

$$action := -i\partial\Psi_{\mu}\,\bar{\epsilon}\bar{\partial}X_{\mu} - i\partial\overline{\Psi}_{\mu}\,\epsilon\bar{\partial}X_{\mu} - i\bar{\partial}\Psi_{\mu}\,\bar{\epsilon}\partial X_{\mu} - i\bar{\partial}\overline{\Psi}_{\mu}\,\epsilon\partial X_{\mu} + i\bar{\partial}\Psi_{\mu}\,\bar{\epsilon}\partial X_{\mu} - i\Psi_{\mu}\bar{\epsilon}\bar{\partial}\partial X_{\mu} + i\partial\overline{\Psi}_{\mu}\,\epsilon\bar{\partial}X_{\mu} - i\overline{\Psi}_{\mu}\epsilon\partial\bar{\partial}X_{\mu};$$

At this stage we are left with an expression which still contains double derivatives. In order to write this in a canonical form, we eliminate all double derivatives by doing one partial integration. This is done by first marking the derivatives which we want to partially integrate, and then using `pintegrate`,

```
@substitute!(%)( \del{\delbar{X_{\mu}}} -> \pdelbar{\del{X_{\mu}}} ):
@substitute!(%)( \delbar{\del{X_{\mu}}} -> \pdel{\delbar{X_{\mu}}} ):
@pintegrate!(%){ \pdelbar }:
@pintegrate!(%){ \pdel }:
@rename!(%){"\pdelbar"}{"\delbar"}:
@rename!(%){"\pdel"}{"\del"}:
@prodrule!(%):
@distribute!(%):
@unwrap!(%);
```

$$action := -i\partial\Psi_{\mu}\,\bar{\epsilon}\bar{\partial}X_{\mu} - i\partial\overline{\Psi}_{\mu}\,\epsilon\bar{\partial}X_{\mu} - i\bar{\partial}\Psi_{\mu}\,\bar{\epsilon}\partial X_{\mu} - i\bar{\partial}\overline{\Psi}_{\mu}\,\epsilon\partial X_{\mu} + i\bar{\partial}\Psi_{\mu}\,\bar{\epsilon}\partial X_{\mu} + i\partial\Psi_{\mu}\,\bar{\epsilon}\bar{\partial}X_{\mu} + i\partial\overline{\Psi}_{\mu}\,\epsilon\bar{\partial}X_{\mu} + i\bar{\partial}\overline{\Psi}_{\mu}\,\epsilon\partial X_{\mu};$$

Notice how, after the partial integration, we renamed the partially integrated derivatives back to normal ones (and again apply Leibnitz' rule). If we now collect terms, we indeed find that the total susy variation vanishes,

```
@collect_terms!(%);
```

$$action := 0;$$