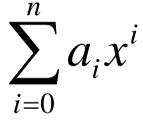
Computing Polynomials

Alexandre David B2-206

The Problem

How to compute efficiently polynomials of the form



Naïve approach: Compute each term and sum-up. If naïve power algorithm is used: n(n+1)/2 multiplications + n additions.
 Better? ?

• Rewrite the polynomial as:

$$\sum_{i=0}^{n} a_{i}x^{i} = a_{0} + x(a_{1} + x(a_{2} + \dots x(a_{n})\dots))$$
• Compute "a" as:

$$a = a_{n}$$

$$a = a_{n-1} + x^{*}a$$
...

$$a = a_{1} + x^{*}a$$

$$a = a_{0} + x^{*}a$$

```
Algorithm
```

```
a = ai[n]

while (n>0) do

n = n-1

a = ai[n] + x*a

done

return a
```

Notes:

- Efficient if most ai[n] are not null.
- Not necessary the best precision.

 Optimized for some DSPs that have a multiply and accumulate instruction.