



(12)发明专利申请

(10)申请公布号 CN 107113006 A

(43)申请公布日 2017.08.29

(21)申请号 201580053923.6

(74)专利代理机构 北京市柳沈律师事务所

(22)申请日 2015.07.30

11105

(30)优先权数据

1414376.2 2014.08.13 GB

代理人 史新宏

(85)PCT国际申请进入国家阶段日

2017.04.01

(51)Int.Cl.

H03M 13/00(2006.01)

(86)PCT国际申请的申请数据

H03M 13/27(2006.01)

PCT/EP2015/067527 2015.07.30

H03M 13/29(2006.01)

(87)PCT国际申请的公布数据

W02016/023762 EN 2016.02.18

(71)申请人 阿塞勒康姆有限公司

地址 英国南安普顿市

(72)发明人 R.蒙德 I.佩雷兹-安德雷德

A.李

权利要求书9页 说明书29页 附图29页

(54)发明名称

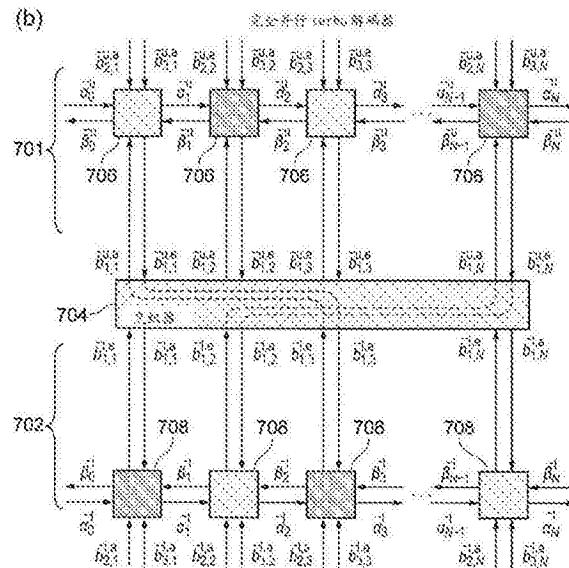
完全并行turbo解码

(57)摘要

检测电路执行turbo检测处理,以从接收到的信号恢复数据码元帧,所述帧的所述数据码元在传输期间已经受具有所述接收到的信号中的所述帧的所述数据码元依赖于可以表示为具有多个格栅级的格栅的一个或多个超前数据码元的效果的马尔科夫过程作用。所述检测电路包括多个处理元件,所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联,所述处理元件中的每一个被配置为:接收与关联于所述格栅级的一个或多个数据码元对应的一个或多个软判决值,并且所述处理元件中的一个或多个中的每一个被配置为:在一个时钟周期中,接收表示先验前向状态度量先验后向状态度量的定点数据以及表示对于所述格栅级正检测用于所述一个或多个数据码元的先验软判决值的定点数据。对于所述turbo检测处理的多个时钟周期中的每一个,所述检测电路被配置为:对于表示所述格栅

级的所述处理元件中的每一个,处理用于对于与所述处理元件关联的所述格栅级正检测所述一个或多个数据码元的所述先验信息,并且提供与所述一个或多个数据码元对应的所述外在软判决值,以用于所述turbo检测处理的下一时钟周期。

(b)



1. 一种检测电路,用于执行turbo检测处理,以从包括表示用于数据码元帧的每个数据码元的一个或多个软判决值的定点数据的接收到的信号恢复所述帧,所述帧的所述数据码元在传输的处理期间已经受具有所述接收到的信号中的所述帧的所述数据码元取决于可以表示为具有多个格栅级的格栅的一个或多个超前数据码元的效果的马尔科夫过程作用,所述检测电路包括:

多个处理元件,所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联,并且所述处理元件中的每一个被配置为:接收表示用于与所述格栅级关联的一个或多个数据码元的软判决值的定点数据,并且所述处理元件中的一个或多个中的每一个被配置为:在一个时钟周期中:

从第一相邻处理元件接收表示先验前向状态度量的定点数据,

从第二相邻处理元件接收表示先验后向状态度量的定点数据,以及

接收表示对于与所述处理元件关联的所述格栅级正检测用于所述一个或多个数据码元的先验软判决值的定点数据,

组合与所述一个或多个数据码元有关的所述先验前向状态度量、所述先验后向状态度量以及所述先验软判决值,以对于与所述处理元件关联的所述格栅级确定与所述一个或多个数据码元对应的一个或多个定点外在前向状态度量、一个或多个定点外在后向度量以及定点外在软判决值,以及

将所述一个或多个外在前向状态度量传递到所述第二相邻处理元件,其在下一时钟周期中变为用于该处理元件的所述先验前向状态度量,

将所述一个或多个外在后向状态度量传递到所述第一相邻处理元件,其在所述下一时钟周期中变为用于该处理元件的所述先验后向状态度量,以及

提供所述一个或多个外在软判决值,其对于下一时钟周期变为与所述数据码元有关的所述先验软判决值,其中,对于所述turbo检测处理的多个相继时钟周期中的一个或多个,所述检测电路的所述处理元件被配置为同时进行操作。

2. 如权利要求1所述的检测电路,其中,所述一个或多个处理元件中的每一个被配置为:在一个时钟周期中,对于所述格栅的多个转变中的每个转变,通过组合从所述第一相邻处理元件接收到的所述转变的前一状态(S_{k-1})的所述先验前向状态度量($\bar{\alpha}_{k-1}(S_{k-1})$)、从所述第二相邻处理元件接收到的所述转变的下一状态(S_k)的所述后向状态度量($\bar{\beta}_k(S_k)$)以及以下项中的至少一个来形成后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$):

与所述状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合($[\sum_{j=1}^L [b_j(S_{k-1}, S_k) \cdot \bar{b}_{j,k}^a]]$),或

与所述状态转变有关的其它软判决信息($\ln [\Pr \{S_k | S_{k-1}\}]$)。

3. 如权利要求1所述的检测电路,其中,所述一个或多个处理元件中的每一个被配置为:在一个时钟周期中,对于所述格栅的所述多个转变中的每个转变,通过以下操作来形成后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$):

组合从所述第一相邻处理元件接收到的所述转变的前一状态(S_{k-1})的所述先验前向状态度量($\bar{\alpha}_{k-1}(S_{k-1})$),

从所述第二相邻处理元件接收到的所述转变的下一状态 (S_k) 的所述后向状态度量 ($\bar{\beta}_k(S_k)$) , 以及

前一时钟周期中所形成的与所述状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合 ($[\sum_{j=1}^L [b_j(S_{k-1}, S_k) \cdot \bar{b}_{j,k}^a]]$)。

4. 如权利要求2或3所述的检测电路, 其中, 所述一个或多个处理元件中的每一个被配置为: 在一个时钟周期中, 对于所述格栅中的所述多个下一状态的每个下一状态 (S_k), 通过组合用于相连转变的所述后验转变度量 ($\bar{\delta}_k(S_{k-1}, S_k)$) 并且减去用于该下一状态 (S_k) 的所述后向状态度量 ($\bar{\beta}_k(S_k)$) 来确定外在前向状态度量 ($\bar{\alpha}_k(S_k)$)。

5. 如权利要求3所述的检测电路, 其中, 组合用于所述相连转变的所述后验转变度量 ($\bar{\delta}_k(S_{k-1}, S_k)$) 包括: 确定用于所述处理元件表示 ($\max^*_{\{S_{k-1}|C(S_{k-1}, S_k)=1\}} [\bar{\delta}_k(S_{k-1}, S_k)]$) 的所述格栅中的与该下一状态 (S_k) 相连的所述转变中的每一个的所述后验转变度量中的每一个的雅可比对数。

6. 如权利要求2至5中的任一项所述的检测电路, 其中, 所述一个或多个处理元件中的每一个被配置为: 在一个时钟周期中, 对于所述格栅中的所述多个前一状态中的每个前一状态 (S_{k-1}), 通过组合用于所述相连转变的所述后验转变度量 ($\bar{\delta}_k(S_{k-1}, S_k)$) 并且减去用于该前一状态 (S_{k-1}) 的所述前向状态度量 ($\bar{\alpha}_{k-1}(S_{k-1})$) 来确定外在后向状态度量 ($\bar{\beta}_{k-1}(S_{k-1})$)。

7. 如权利要求6所述的检测电路, 其中, 组合用于所述相连转变的所述后验转变度量 ($\bar{\delta}_k(S_{k-1}, S_k)$) 包括: 确定用于所述处理元件表示 ($\max^*_{\{S_{k-1}|C(S_{k-1}, S_k)=1\}} [\bar{\delta}_k(S_{k-1}, S_k)]$) 的所述格栅中与该前一状态 (S_{k-1}) 相连的所述转变中的每一个的所述后验转变度量中的每一个的雅可比对数。

8. 如权利要求2至7中的任一项所述的检测电路, 其中, 所述一个或多个处理元件中的每一个被配置为: 在一个时钟周期中, 对于每个格栅级所考虑的所述多个数据码元中的每个数据码元 ($b_{j,k}$), 通过组合用于所述格栅级中的所述转变中的所述每一个的所述后验转变度量 ($\bar{\delta}_k(S_{k-1}, S_k)$) 并且减去用于该数据码元 ($b_{j,k}$) 的所述先验软判决值 ($\bar{b}_{j,k}^a$) 来形成外在软判决值 ($\bar{b}_{j,k}^e$)。

9. 如权利要求2至7中的任一项所述的检测电路, 其中, 所述一个或多个处理元件中的每一个被配置为: 在一个时钟周期中, 对于每个格栅级所考虑的所述多个数据码元中的每个数据码元 ($b_{j,k}$), 通过组合用于所述数据码元 ($b_{j,k}$) 的所述先验软判决值 ($\bar{b}_{j,k}^a$) 与通过组合用于所述格栅级中的所述转变中的每一个的所述后验转变度量 ($\bar{\delta}_k(S_{k-1}, S_k)$) 在前一时钟周期中确定的中间计算结果 ($\bar{\varepsilon}_{k,l,m}$) 的组合来形成外在软判决值 ($\bar{b}_{j,k}^e$)。

10. 如权利要求8或9所述的检测电路, 其中, 组合所述后验转变度量包括: 形成以下项之间的差:

用于所述格栅级中的所述转变集合的所述后验转变度量集合的雅可比对数, 其中, 用于该数据码元 ($b_{j,k}$) 的隐式值是二进制1 ($(S_{k-1}, S_k) | b_j(S_{k-1}, S_k) = 1$), 以及

用于所述格栅级中的所述转变集合的所述后验转变度量集合的雅可比对数,其中,用于该数据码元($b_{j,k}$)的隐式值是二进制0((S_{k-1}, S_k) | $b_j(S_{k-1}, S_k) = 0$)。

11. 如权利要求5、7或11中的任一项所述的检测电路,其中,所述雅可比对数定义为:

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) = \max(\bar{\delta}_1, \bar{\delta}_2) + \ln(1 + e^{-(|\bar{\delta}_1 - \bar{\delta}_2|)})$$

对于成对后验转变度量 $\bar{\delta}_1, \bar{\delta}_2$,其中,所述雅可比对数可以近似为:

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) \approx \max(\bar{\delta}_1, \bar{\delta}_2), \text{ 或}$$

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) \approx A \cdot \max(\bar{\delta}_1, \bar{\delta}_2) + B, \text{ 其中, } A \text{ 和 } B \text{ 是恒定值, 或}$$

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) \approx A(\bar{\delta}_1, \bar{\delta}_2) \cdot \max(\bar{\delta}_1, \bar{\delta}_2) + B(\bar{\delta}_1, \bar{\delta}_2), \text{ 其中, } A \text{ 和 } B \text{ 是后验转变度量 } \bar{\delta}_1, \bar{\delta}_2 \text{ 的简单函数。}$$

12. 如权利要求1至11中的任一项所述的检测电路,其中,已经通过包括第一卷积编码器、交织器以及第二卷积编码器的turbo编码器在传输的处理期间对所述数据码元帧进行编码,在传输的处理期间已经作用于所述数据码元帧的所述马尔科夫过程是由所述第一卷积编码器和所述第二卷积编码器产生的,所述检测电路被布置为:根据第一卷积码执行所述turbo检测处理,第二检测电路被布置为:根据第二卷积码执行所述turbo检测处理,并且对于每个时钟周期,所述检测电路和所述第二检测电路关于所述各个第一卷积码和第二卷积码执行解码处理,通过交织器电路交换外在软判决值,其关于所述检测电路和所述第二检测电路中的另一个对于所述下一时钟周期变为与所述数据码元有关的所述先验软判决值。

13. 如权利要求1至12中的任一项所述的检测电路,包括:

存储寄存器集合,用于所述检测电路中的所述处理元件中的每一个,其用于存储将在所述下一时钟周期中由所述处理元件处理的用于所述一个或多个数据码元的所述先验软判决值、所述先验前向状态度量以及所述先验后向状态度量,

控制单元,以及

对于所述处理元件中的每一个,可选择的旁路电路,其在所述控制单元的控制下可配置为:旁路所述处理元件和关联寄存器,以禁用所述处理元件。

14. 如权利要求13所述的检测电路,其中,所述传输处理包括交织器,其对于一个或多个所支持的帧长度使用从一个或多个所支持的交织器模式集合选择的交织模式,其中,所述检测电路的所述控制器被配置为:控制所述可选择的旁路电路,以根据所述交织器所使用的所述帧长度和交织模式来禁用所述检测电路的所选择的处理元件。

15. 如权利要求12、13或14所述的检测电路,其中,所述传输处理包括交织器,其使用从一个或多个所支持的交织器模式集合选择的交织模式,并且所述检测电路包括对应交织器电路,其可以根据所述一个或多个所支持的交织模式在控制单元的控制下被重新配置。

16. 如权利要求12、13、14或15所述的检测电路,其中,所述传输处理包括交织器,其已经将具有奇数指数的所述数据帧的数据码元与也具有奇数指数的交织帧的数据码元进行交织,并且将具有偶数指数的所述数据帧的数据码元与也具有偶数指数的所述交织帧的数据码元进行交织,以及

对于所述检测电路中的所述处理元件中的每一个,所述存储寄存器集合包括一个或多个存储寄存器,以用于存储将在所述下一时钟周期中由所述处理元件处理的用于所述一个

或多个数据码元的所述先验软判决值、所述先验前向状态度量以及所述先验后向状态度量，以及

所述控制单元被配置为：在所述turbo检测处理的每隔一个时钟周期中，启用所述检测电路的所述处理元件，以用于处理具有奇数指数的所述数据码元，并且禁用所述检测电路的所述处理元件的另一半，以用于具有偶数指数的所述数据码元，并且在所述turbo检测处理的所有其它时钟周期中，启用所述处理元件的所述另一半，以用于处理具有偶数指数的所述数据码元，并且禁用所述处理元件的前一半，以用于处理具有奇数指数的所述数据码元。

17. 如权利要求12、13、14或15所述的检测电路，其中，所述传输处理包括交织器，其已经将具有奇数指数的每个数据帧的数据码元与也具有奇数指数的对应交织帧的数据码元进行交织，并且将具有偶数指数的每个数据帧的数据码元与也具有偶数指数的所述对应交织帧中的数据码元进行交织，并且所述检测电路包括：

对应交织器电路，并且所述存储寄存器集合包括：

两个存储寄存器集合，用于所述检测电路中的所述处理元件中的每一个，其中，每个寄存器集合用于存储用于不同的所述帧或第二帧之一中的所述一个或多个数据码元的所述先验软判决值，并且用于存储用于该帧的所述先验前向状态度量和所述先验后向状态度量，其将在使用所述处理元件以用于对所述对应帧进行解码的所述下一时钟周期中由所述处理元件处理，以及

所述控制单元被配置为：在每隔一个时钟周期中，在对于所述帧中的第二个使用所述对应寄存器对于具有偶数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值的同时，对于所述帧中的第一个使用所述对应寄存器对于具有奇数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值，并且在所有其它时钟周期中，在对于所述帧中的所述第二个使用所述对应寄存器对于具有奇数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值的同时，对于所述帧中的所述第一个使用所述对应寄存器对于具有偶数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值，由此同时对都采用相同交织模式的所述帧和所述第二帧进行解码。

18. 如权利要求12、13、14或15所述的检测电路，其中，所述传输处理包括交织器，其已经将具有奇数指数的所述数据帧的数据码元与也具有奇数指数的交织帧的数据码元进行交织，并且将具有偶数指数的所述数据帧的数据码元与也具有偶数指数的所述交织帧中的数据码元进行交织，并且所述一个或多个存储寄存器集合包括：

两个存储寄存器集合，以用于所述检测电路中的所述处理元件中的每一个，其中，每个寄存器集合用于存储用于不同的两个马尔科夫过程之一中的所述一个或多个数据码元的所述先验软判决值，并且用于存储用于该马尔科夫过程的所述先验前向状态度量和所述先验后向状态度量，其将在使用所述处理元件以用于对所述对应马尔科夫过程进行解码的所述下一时钟周期中由所述处理元件处理，以及

所述控制单元被配置为：在每隔一个时钟周期中，在对于所述第二马尔科夫过程使用所述对应寄存器对于具有偶数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值的同时，对于所述第一马尔科夫过程使用所述对应寄存器对于具有奇数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值，并且在所有其它时钟

周期中,在对于所述第二马尔科夫使用所述对应寄存器对于具有奇数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值的同时,对于所述第一马尔科夫过程使用所述对应寄存器对于具有偶数指数的所述处理元件加载所述先验软判决值并且存储所述外在软判决值,由此同时对两个马尔科夫过程进行解码,并且所述检测电路包括:

交织器电路,其被配置为:在每隔一个时钟周期中,在实现将用于所述第二马尔科夫过程的所述先验软判决值提供给具有偶数指数的所述处理元件的所述交织器模式的同时,实现将用于所述第一马尔科夫过程的所述先验软判决值提供给具有奇数指数的所述处理元件的交织器模式,并且在所有其它时钟周期中,在实现将用于所述第二马尔科夫过程的先验软判决值提供给具有奇数指数的所述处理元件的所述交织器模式的同时,实现将用于所述第一马尔科夫过程的所述先验软判决值提供给具有偶数指数的所述处理元件的所述交织器模式。

19. 如权利要求12、13、14或15所述的检测电路,其中,所述传输处理包括交织器,其已经将具有奇数指数的所述数据帧的数据码元与也具有奇数指数的交织帧的数据码元进行交织,并且将具有偶数指数的所述数据帧的数据码元与也具有偶数指数的所述交织帧的数据码元进行交织,以及

所述存储寄存器集合包括一个或多个存储寄存器,用于所述检测电路中的所述处理元件中的每一个,以用于存储用于所述一个或多个数据码元的所述先验软判决值、与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合、通过组合用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\delta_k(S_{k-1}, S_k)$)、所述先验前向状态度量以及将在所述下一时钟周期中由所述处理元件处理的所述先验后向状态度量所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$),以及

所述控制单元被配置为:在所述turbo检测处理的每隔一个时钟周期中,禁用所有信息的处理,此外,对于具有奇数指数的所述数据码元启用所述检测电路的所述处理元件,以:

处理用于所述一个或多个数据码元的所述先验软判决值,以及

处理通过组合用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\delta_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$),

以及对于具有偶数指数的所述数据码元启用所述检测电路的所述处理元件,以

处理与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合,

处理所述先验前向状态度量,以及

处理所述先验后向状态度量,

并且在所述turbo检测处理的其它时钟周期中,禁用所有信息的处理,此外,对于具有奇数指数的所述数据码元启用所述检测电路的所述处理元件,以:

处理与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合,

处理所述先验前向状态度量,以及

处理所述先验后向状态度量,

以及对于具有偶数指数的所述数据码元启用所述检测电路的所述处理元件,以

处理用于所述一个或多个数据码元的所述先验软判决值,以及

处理通过组合用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)。

20. 如权利要求12、13、14或15所述的检测电路,其中,所述传输处理包括交织器,其已经将具有奇数指数的每个数据帧的数据码元与也具有奇数指数的对应交织帧的数据码元进行交织,并且将具有偶数指数的每个数据帧的数据码元与也具有偶数指数的所述对应交织帧中的数据码元进行交织,并且所述检测电路包括:

对应交织器电路,并且所述存储寄存器集合包括:

两个存储寄存器集合,用于所述检测电路中的所述处理元件中的每一个,其中,每个寄存器集合用于存储用于不同的所述帧或第二帧之一中的所述一个或多个数据码元的所述先验软判决值,并且用于存储与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合,所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)是通过组合用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$),所述先验前向状态度量以及用于该帧的所述先验后向状态度量确定的,其将在使用所述处理元件以用于对所述对应帧进行解码的所述下一时钟周期中由所述处理元件处理,以及

所述控制单元被配置为:在所述turbo检测处理的每隔一个时钟周期中,对于具有奇数指数的所述数据码元启用所述检测电路的所述处理元件,以:

处理与所述第一帧对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值,

处理通过组合与所述第一帧对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$),

处理与所述第二帧对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合,

处理与所述第二帧对应的所述寄存器中的所述先验前向状态度量,以及

处理与所述第二帧对应的所述寄存器中的所述先验后向状态度量,

以及对于具有偶数指数的所述数据码元启用所述检测电路的所述处理元件,以

处理与所述第一帧对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合,

处理与所述第一帧对应的所述寄存器中的所述先验前向状态度量,

处理与所述第一帧对应的所述寄存器中的所述先验后向状态度量,

处理与所述第二帧对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值,以及

处理通过组合与所述第二帧对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$),

并且在所述turbo检测处理的其它时钟周期中,对于具有奇数指数的所述数据码元启用所述检测电路的所述处理元件,以

处理与所述第一帧对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合,

- 处理与所述第一帧对应的所述寄存器中的所述先验前向状态度量，
 处理与所述第一帧对应的所述寄存器中的所述先验后向状态度量，
 处理与所述第二帧对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值，以及
 处理通过组合与所述第二帧对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)，
 以及对于具有偶数指数的所述数据码元启用所述检测电路的所述处理元件，以
 处理与所述第一帧对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值，
 处理通过组合与所述第一帧对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)，
 处理与所述第二帧对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合，
 处理与所述第二帧对应的所述寄存器中的所述先验前向状态度量，以及
 处理与所述第二帧对应的所述寄存器中的所述先验后向状态度量，
 由此同时对都采用相同交织模式的所述帧和所述第二帧进行解码。
21. 如权利要求12、13、14或15所述的检测电路，其中，所述传输处理包括交织器，其已经将具有奇数指数的所述数据帧的数据码元与也具有奇数指数的交织帧的数据码元进行交织，并且将具有偶数指数的所述数据帧的数据码元与也具有偶数指数的所述交织帧中的数据码元进行交织，并且所述一个或多个存储寄存器集合包括：
- 两个存储寄存器集合，用于所述检测电路中的所述处理元件中的每一个，其中，每个寄存器集合用于存储用于不同的所述两个马尔科夫过程之一中的所述一个或多个数据码元的所述先验软判决值，并且用于存储与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合，所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)是通过组合用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)、所述先验前向状态度量以及用于该马尔科夫过程的所述先验后向状态度量确定的，其将在使用所述处理元件以用于对所述对应马尔科夫过程进行解码的所述下一时钟周期中由所述处理元件处理，以及
- 所述控制单元被配置为：在所述turbo检测处理的每隔其它时钟周期中，对于具有奇数指数的所述数据码元启用所述检测电路的所述处理元件，以：
- 处理与所述第一马尔科夫过程对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值，
 处理通过组合与所述第一马尔科夫过程对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)，
 处理与所述第二马尔科夫过程对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合，
 处理与所述第二马尔科夫过程对应的所述寄存器中的所述先验前向状态度量，以及

处理与所述第二马尔科夫过程对应的所述寄存器中的所述先验后向状态度量，
以及对于具有偶数指数的所述数据码元启用所述检测电路的所述处理元件，以：

处理与所述第一马尔科夫过程对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合，

处理与所述第一马尔科夫过程对应的所述寄存器中的所述先验前向状态度量，

处理与所述第一马尔科夫过程对应的所述寄存器中的所述先验后向状态度量，

处理与所述第二马尔科夫过程对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值，以及

处理通过组合与所述第二马尔科夫过程对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)，

并且在所述turbo检测处理的其它时钟周期中，对于具有奇数指数的所述数据码元启用所述检测电路的所述处理元件，以

处理与所述第一马尔科夫过程对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合，

处理与所述第一马尔科夫过程对应的所述寄存器中的所述先验前向状态度量，

处理与所述第一马尔科夫过程对应的所述寄存器中的所述先验后向状态度量，

处理与所述第二马尔科夫过程对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值，以及

处理通过组合与所述第二马尔科夫过程对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)，

以及对于具有偶数指数的所述数据码元启用所述检测电路的所述处理元件，以

处理与所述第一马尔科夫过程对应的所述寄存器中的用于所述一个或多个数据码元的所述先验软判决值，

处理通过组合与所述第一马尔科夫过程对应的所述寄存器中的用于所述格栅级中的所述转变中的每一个的所述后验转变度量($\bar{\delta}_k(S_{k-1}, S_k)$)所确定的所述中间计算结果($\bar{\varepsilon}_{k,l,m}$)，

处理与所述第二马尔科夫过程对应的所述寄存器中的与每个状态转变对应的用于所述一个或多个数据码元中的每一个的软判决值的组合，

处理与所述第二马尔科夫过程对应的所述寄存器中的所述先验前向状态度量，以及

处理与所述第二马尔科夫过程对应的所述寄存器中的所述先验后向状态度量，

由此同时对两个马尔科夫过程进行解码，并且所述检测电路包括：

交织器电路，其被配置为：在每隔一个时钟周期中，在实现将用于所述第二马尔科夫过程的所述先验软判决值提供给具有偶数指数的所述处理元件的所述交织器模式的同时，实现将用于所述第一马尔科夫过程的所述先验软判决值提供给具有奇数指数的所述处理元件的所述交织器模式。

22. 一种接收机，包括：

如权利要求1至16中的任一项所述的第一检测电路,以及

至少一个其它检测电路,其被配置为:与所述第一检测电路协作,以组合地执行turbo检测处理,以生成所述数据码元帧的估计,以及

时钟,根据预定频率来生成时钟周期,其中,对于所述turbo检测处理的多个相继时钟周期中的一个或多个,所述第一检测电路以及所述至少一个其它检测电路被配置为同时进行操作。

23.一种方法,用于执行turbo检测处理,以从包括表示用于数据码元帧的每个数据码元的一个或多个软判决值的定点数据的接收到的信号恢复所述帧,所述帧的所述数据码元在传输的处理期间已经受具有所述接收到的信号中的所述帧的所述数据码元取决于可以表示为具有多个格栅级的格栅的一个或多个超前数据码元的效果的马尔科夫过程作用,所述方法包括:

在检测电路处接收表示为用于所述帧的每个数据码元的定点软判决值的所述数据码元帧,

执行所述turbo检测处理,以生成所述数据码元帧的估计,所述检测电路包括:

多个处理元件,所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联,并且所述处理元件中的每一个被配置为:接收表示用于与所述格栅级关联的一个或多个数据码元的软判决值的定点数据,并且所述方法包括:对于所述处理元件中的一个或多个中的每一个,在一个时钟周期中:

从第一相邻处理元件接收表示先验前向状态度量的定点数据,

从第二相邻处理元件接收表示先验后向状态度量的定点数据,以及

接收表示对于与所述处理元件关联的所述格栅级正检测用于所述一个或多个数据码元的先验软判决值的定点数据,

组合与所述一个或多个数据码元有关的所述先验前向状态度量、所述先验后向状态度量以及所述先验软判决值,以确定用于与所述处理元件关联的所述格栅级的与所述一个或多个数据码元对应的一个或多个定点外在前向状态度量、一个或多个定点外在后向度量以及定点外在软判决值,以及

将所述一个或多个外在前向状态度量传递到所述第二相邻处理元件,其在所述下一时钟周期中变为用于所述处理元件的所述先验前向状态度量,

将所述一个或多个外在后向状态度量传递到所述第一相邻处理元件,其在所述下一时钟周期中变为用于该处理元件的所述先验后向状态度量,以及

提供所述一个或多个外在软判决值,其对于所述下一时钟周期变为与所述数据码元有关的所述先验软判决值,其中,对于所述turbo检测处理的多个相继时钟周期中的一个或多个,所述检测电路的所述处理元件被配置为同时进行操作。

24.一种移动通信设备,被配置为:根据长期演进标准或WiMAX标准进行操作,包括如权利要求1至17中的任一项所述的接收机。

25.一种基站设备,被配置为:根据长期演进标准或WiMAX标准进行操作,包括如权利要求1至21中的任一项所述的检测电路以及如权利要求22所述的接收机。

完全并行turbo解码

技术领域

[0001] 本发明涉及用于执行turbo检测处理以从接收到的信号恢复数据码元帧的检测电路,所述帧的所述数据码元在传输期间已经受具有所述接收到的信号中的所述帧的所述数据码元依赖于可以表示为具有多个格栅级的格栅的一个或多个超前数据码元的效果的马尔科夫过程作用。在一些示例中,检测电路形成接收机的部分,并且与另一检测电路协调而进行操作,以执行turbo检测处理。

[0002] 本公开实施例可以因此提供被配置为使用turbo解码器恢复数据码元帧的接收机以及用于对受turbo编码的数据进行解码的方法。在一个示例中,数据码元是比特。

背景技术

[0003] 过去二十多年,无线通信已经为受益于迭代解码算法的信道码所革命。例如,长期演进(LTE) [1] 和WiMAX [2] 蜂窝电话标准采用turbo码 [3],其包括两个卷积码的串接。传统上,对于这些卷积码在已编码比特上所施加的马尔科夫链的迭代解码,采用对数Bahl-Cocke-Jelinek-Raviv (Log-BCJR) 算法 [4]。同时,用于无线局域网(WLAN) 的WiFi标准 [5] 已经采取低密度奇偶校验(LDPC) 码 [6],其可以在最小求和算法 [7] 的基础上进行操作。归因于它们的强大纠错能力,这些复杂的信道码已经促进按紧密接近无线信道的容量的传输吞吐量的可靠通信。然而,如果要求实时操作,则可实现的传输吞吐量受限于迭代解码算法的处理吞吐量。此外,迭代解码算法的处理时延对端到端时延施加限制。这是特别相关的,因为可能期待多吉比特传输吞吐量和超低端到端时延成为对于下一代无线通信标准 [8] 的目标。因此,需要具有改进的处理吞吐量以及更低的处理时延的迭代解码算法。归因于最小求和算法的固有并行性,其可以按完全并行方式进行操作,促进具有高达16.2G比特/秒的处理吞吐量的LDPC解码器 [9]。通过对比,现有技术turbo解码器 [10] 的处理吞吐量受限为2.15G比特/秒。这可以归因于Log-BCJR算法的前向递归和后向递归的数据依赖性所施加的Log-BCJR算法的固有串行属性 [4]。更具体地说,典型地两个卷积编码器中的每一个所生成的受turbo编码的比特必须串行地受处理,在作为实际集成电路实现方式中的时钟周期的大量相继时间段上扩展。此外,Log-BCJR算法典型地交替以用于两个卷积码,直到已经执行足够数量的解码迭代。因此,需要成千的时间段以完成现有技术turbo解码器的迭代解码处理。

[0004] 相应地,提供具有更少数据依赖性并且使得能够进行完全并行处理的对Log-BCJR解码器的替换方案再次提出技术问题。

发明内容

[0005] 根据本技术第一示例实施例,提供一种检测电路,用于执行turbo检测处理,以从包括表示用于数据码元或比特的帧的每个数据码元的一个或多个软判决值的定点数据的接收到的信号恢复所述帧。所述帧的所述数据码元或比特在传输期间已经受具有所述接收到的信号中的所述帧的所述数据码元依赖于可以表示为具有多个格栅状态的格栅的效果

的马尔科夫过程作用。所述检测电路包括多个处理元件。所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联，并且所述处理元件中的每一个被配置为：接收对应于与所述格栅级关联的一个或多个数据码元的一个或多个软判决值。所述处理元件中的一个或多个中的每一个被配置为：在一个时钟周期中，从第一相邻处理元件接收表示先验前向状态度量的定点数据，从第二相邻处理元件接收表示先验后向状态度量的定点数据，并且接收表示对于所述格栅级正检测用于所述一个或多个数据码元的先验软判决值的定点数据。所述处理元件包括与所述一个或多个数据码元有关的所述先验前向状态度量、所述先验后向状态度量以及所述先验软判决值，以对于与所述处理元件关联的所述格栅级确定与所述一个或多个数据码元对应的定点外在前向状态度量、定点外在后向度量和外在软判决值，并且将所述外在前向状态度量传递到所述第二相邻处理元件，其对于下一时钟周期变为所述先验前向状态度量，将所述外在后向状态度量传递到所述第一相邻处理元件，其对于所述下一时钟周期变为所述先验后向状态度量，并且提供所述外在软判决值，其对于所述下一时钟周期变为与所述数据码元有关的所述先验软判决值。在一个示例中，作为所述turbo检测处理的部分，所述外在软判决值提供给第二检测电路，以用于处理。在其它示例中，提供所述外在软判决值，以用于存储在存储器中，从而所述检测电路关于所述turbo检测处理的随后迭代而使用它们。对于所述turbo检测处理的多个相继时钟周期中的一个或多个，所述检测电路的所述处理元件被配置为同时进行操作。

[0006] 本技术实施例可以提供一种用于从接收到的信号检测并且恢复数据码元或比特的帧的接收机。所述接收到的信号中出现的所述帧的所述数据码元在传输的处理期间已经受马尔科夫过程作用为所述接收到的信号中的所述帧的所述数据码元依赖于可以表示为具有多个格栅状态的格栅的一个或多个超前数据码元的效果。

[0007] 本领域技术人员应理解，所述接收到的信号所表示的所述数据码元帧的这种描述可以显现为：通过遭受码间干扰的信道发送所述数据码元帧。因此，接收到的所述帧的码元可以在所述接收到的信号内得以检测或表示为接收到的码元和一个或多个超前码元的组合，从而信道引入码间干扰，并且因此引入与马尔科夫过程对应的某种记忆。在其它实施例中，可以通过turbo码对所述数据码元进行编码，效果是：所述接收到的信号中的所述数据码元由可以是将原始数据码元表示为对所述turbo编码器的输入的系统性数据码元或所述turbo编码器所产生的奇偶数据码元的一个或多个不同的所编码的数据码元表示。此外，根据turbo编码，所述编码器可以具备多个卷积编码器，其中的每一个可以关于每个输入数据码元输出多个数据码元。

[0008] 在一些示例中，对于所述turbo检测处理的多个时钟周期中的每一个，所述检测电路可以被配置为：与至少一个其它检测电路协作，以对于表示所述格栅级的所述处理元件中的每一个处理关于与所述处理元件关联的所述格栅级正检测所述一个或多个数据码元的所述先验信息，并且与所述至少一个其它处理元件交换与所述处理元件所生成的所述一个或多个数据码元对应的所述外在软判决值。在一些示例中，所述检测电路和所述至少一个其它检测电路中的每一个的所述处理元件可以在每个时钟周期中进行操作，并且在每个时钟周期之后，所述第一检测电路和所述至少一个其它检测电路彼此交换所述外在软判决值。

[0009] 根据本技术,接收机包括解调器,以用于对接收到的信号进行检测并且解调,以恢复表示为软判决值的数据码元帧的估计,软判决值表示所述帧的所述数据码元中的每一个采用其可能值中的每一个的概率。对于受turbo编码的数据码元或比特的帧的示例,所述解调器对于已编码数据码元或比特生成软判决值。所述接收机还包括:第一检测处理器,被配置为:接收表示所述数据码元帧的所述软判决值;以及第二检测处理器,其被配置为:与所述第一检测处理器协作,以执行turbo检测处理,以生成所述数据码元帧的估计。所述第一检测处理器包括多个处理元件,所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联。所述处理元件中的每一个被配置为:接收与关联于所述格栅级的一个或多个数据码元对应的所述软判决值,并且所述多个处理元件中的每一个被配置为:从一个相邻处理元件接收表示先验前向状态度量的数据,从另一相邻处理元件接收表示先验后向状态度量的数据,并且从所述第二检测处理器而且从所述解调器接收表示用于对于与所述处理元件关联的所述格栅级正检测所述数据码元的先验信息的数据。每个处理元件被配置为:组合与所述数据码元有关的所述先验前向状态度量、所述先验后向状态度量以及所述先验信息,以产生外在前向度量、外在后向度量以及外在数据信息,并且将所述外在前向状态度量传递到所述第二相邻处理元件,其对于下一迭代变为所述先验前向状态度量,将所述外在后向状态度量传递到所述第一相邻处理元件,其对于所述下一迭代变为所述先验后向状态度量,并且将所述外在数据信息传递到所述第二检测处理器,其对于所述turbo检测处理的下一迭代变为所述先验信息。所述第一检测处理器和所述第二检测处理器被配置为:对于表示所述格栅级的所述处理元件中的每一个,交换用于对于与所述处理元件关联的所述格栅级正检测所述数据码元的所述先验信息以及所述处理元件所生成的对应外在数据信息,以用于所述turbo检测处理的多个迭代中的每一个。在每个时钟周期中,来自两个解码器的处理元件正得以激活。这与传统解码器形成对比,其中,每当一个解码器激活时,另一个就保持禁止。

[0010] 本技术实施例可以提供一种改进的turbo检测处理器,其修改传统算法(例如Log-BCJR(又称为Log-MAP算法)),从而与格栅级中的每一个关联的处理元件描述所述数据码元在传输的处理期间已经经受的马尔科夫过程,效果在于,所述处理元件中的每一个可以自主地并且因此并行地进行操作,由此在turbo检测处理器的处理速率方面提供改进,并且在从接收到的信号恢复数据帧方面减少时延。如上所述的改进的turbo检测器可以应用于对受turbo编码的数据帧进行解码,或在接收到的数据帧已经在传输或其它应用(例如turbo同步和低密度奇偶校验(LDPC)解码)期间遭受码间干扰的情况下用作turbo均衡器。本技术可以因此促进具有与完全并行LDPC解码器相似的处理吞吐量和时延的turbo解码器。此外,该解决方案将对在典型地使用Log-BCJR算法所解码的其它应用(例如均衡、同步、信道估计和信源解码)中恢复数据的马尔科夫链带来相似益处。

[0011] 为了实现用于turbo检测器的完全并行处理架构,通过无需turbo解码器中的前一级在对于当前迭代执行turbo解码器的操作之前完成,本技术消除与传统turbo检测器(例如Log-BCJR或Log-MAP算法)关联的数据依赖性。也就是说,传统turbo解码器将需要所有级在前向方向上以串行方式波动通过(ripple through),以计算前向状态度量,并且对应地在后向方向上波动通过,以生成后向状态度量。相应地,根据本技术的检测处理器并不等待turbo检测器在前向方向或后向方向上前向波动状态度量,而是接受在相邻处理元件的输

出处的前向状态度量或后向状态度量的当前值以及以它们当前使用的形式的与来自另一检测处理器的数据码元有关的先验信息。这表示关于Log-BCJR算法的近似的东西，并且相应地，发明人已经发现，根据本技术的turbo检测器需要更大数量的迭代，以实现与Log-BCJR算法相比相同的解码性能。然而，如以下段落中将解释的那样，发明人已经发现，因为为了产生turbo检测/解码的等效性能的时钟周期的数量小于传统Log-BCJR算法所产生的时钟周期的数量，所以可以产生尽管具有更大数量的迭代但具有更快检测处理的等效性能。

[0012] 对于根据本技术的接收机被布置为检测已经通过turbo编码器编码的帧的数据码元的示例，于是接收机可以还包括交织器，其在第一检测处理器与第二检测处理器之间被配置为：在所述第一检测器与所述第二检测处理器之间传递与所述数据码元有关的先验信息，由此所述第一检测处理器的所述处理元件接收并且发送与所述数据码元有关的外在信息，并且根据交织模式从所述第二检测处理器的不同处理元件接收与所述数据码元有关的先验信息。相应地，对于交织已经根据奇数-偶数交织而应用于形成turbo编码处理的第二卷积编码器的示例，于是取决于turbo编码器所应用的交织而在带奇数指数或偶数指数的第一检测处理器和第二检测处理器的各处理元件之间传递与数据码元有关的数据信息。相应地，在已经应用奇数-偶数交织的情况下，可以与第一检测处理器中的带偶数指数的处理元件和第二检测处理器中的带奇数指数的处理元件在每个时钟周期之间交替地执行第一检测处理器中的带奇数指数的处理元件和第二检测处理器中的带偶数指数的处理元件，由此在处理时提供50%节约。

[0013] 根据本技术，处理元件可以包括旁路电路，其允许所选择的处理元件关闭，以反映较短的帧长度以及对应交织器模式。相应地，可以实现能耗方面的对应节约。旁路电路可以包括复用器，其被配置为：选择来自当前格栅级或前一格栅级的输出，以旁路处理元件。在一些示例中，交织器根据预定交织器模式硬布线，并且旁路允许选择交织器模式的子集。在一些示例中，交织器可以可配置为：容纳可以根据特定turbo编码器设置的不同交织器模式。在另一示例中，交织器可重新配置为：将外在信息路由到不同的处理元件，其可以因此根据特定turbo编码器来配置turbo解码器。在一个示例中，交织器实现为可以根据期望而配置的Benes网络。

[0014] 处理元件可以还包括一个或多个寄存器，其用于在各时间段或时钟周期之间存储与数据码元有关的先验前向状态度量或先验后向状态度量或先验信息。

[0015] 本公开的各个其它方面和特征限定于所附权利要求中，并且包括一种通信设备以及一种使用通信设备的通信方法。

附图说明

[0016] 现将仅参照附图通过示例的方式描述本公开实施例，其中，相同部分具备对应编号，并且其中：

[0017] 图1是根据LTE标准进行操作的移动通信系统的示意图；

[0018] 图2是用于图1所示的LTE系统的示例发射机的示意性框图；

[0019] 图3是用于图1所示的LTE系统的示例接收机的示意性框图；

[0020] 图4是用于LTE标准或WiMAX标准的简化turbo编码器的示意性框图；

- [0021] 图5是示出LTE turbo编码器的更详细示例的示意性框图；
- [0022] 图6是表示使用形成图5的turbo编码器的部分的卷积编码器进行编码的状态和状态转变的说明；
- [0023] 图7是根据Log-BCJR算法的示例turbo解码器的示意性框图；
- [0024] 图8是根据本技术示例实施例的完全并行turbo解码器的示意性框图；
- [0025] 图9是示出形成图8的完全并行turbo解码器的部分的处理元件的更详细实现方式的示意性框图；
- [0026] 图10是根据本技术示例实施例的完全并行turbo解码器的示意性功能块布局；
- [0027] 图11是根据本技术另一示例实施例的完全并行turbo解码器的示意性功能块布局；
- [0028] 图12是根据本技术另一示例实施例的完全并行turbo解码器的示意性功能块布局；
- [0029] 图13是负责终止正解码的数据帧的形成图8的完全并行turbo解码器的部分的处理元件的示意性块布局；
- [0030] 图14提出表1,示出根据本技术实施例的完全并行turbo解码器与等效Log-BCJR turbo解码器和现有技术turbo解码器[10]之间的性能比较,其中,简写TtotP用于表示“乘以所提出的完全并行turbo解码器的标度”；
- [0031] 图15提出表2,示出根据本技术的完全并行turbo解码器与等效Log-BCJR turbo解码器所需的计算的数量的指示,并且在大括号中,关键路径长度中的计算的数量；
- [0032] 图16a、图16b和图16c提供表示当对包括(a)N=4800比特、(b)=480比特以及(c)N=48比特的帧进行解码时的LTE turbo解码器的纠错性能的误比特率针对信噪比的图形图线；
- [0033] 图17a、图17b和图17c是当使用精确max*运算符时的WiMAX turbo解码器、图17b当使用近似max*运算符时的WiMAX turbo解码器以及图17c当使用近似max*运算符时的LTE turbo解码器的纠错性能的误比特率针对信噪比的图形图线；
- [0034] 图18提供对于LTE turbo码特定的图8所示的完全并行turbo解码器的替选实施例的示意性框图；
- [0035] 图19是提供对于LTE turbo码特定的根据图18所示的示例的完全并行turbo解码器的详细实现方式的示例电路；
- [0036] 图20是提供对于LTE turbo码特定的管线式完全并行turbo解码器的详细实现方式的示例电路；
- [0037] 图21a、图21b、图21c提供表示当对包括(a)N=4800比特、(b)=480比特以及(c)N=48比特的帧进行解码时的采用图19和图20的电路的LTE turbo的纠错性能的误比特率针对信噪比的图形图线；
- [0038] 图22提供对于LTE turbo码特定的奇数处理器和偶数处理器交替地执行完全并行turbo解码器所需的计算的不同部分的根据另一示例实施例的管线式完全并行turbo解码器的示意性框图。

具体实施方式

[0039] 示例通信系统

[0040] 图1提供传统移动电信系统100的示意图,其中,系统包括移动通信设备101、基础架构装备102以及核心网络103。例如,基础架构装备可以又称为基站、网络元件、增强节点B(eNodeB)或协调实体,并且在覆盖区域或小区内提供对一个或多个通信设备的无线接入接口。一个或多个移动通信设备可以经由使用无线接入接口发送并且接收表示数据的信号来传递数据。网络实体102可以通过通信方式链接到核心网络103,其中,核心网络可以连接到具有与通过通信设备101和基础架构装备102形成的通信系统或网络相似的一个或多个其它通信系统或网络。核心网络也可以对于网络实体所服务的通信设备提供包括鉴权、移动性管理、收费等的功能性。图1的移动通信设备可以又称为通信终端、用户装备(UE)、终端设备等,并且被配置为:经由网络实体与相同或不同覆盖区域所服务的一个或多个其它通信设备进行通信。可以通过在直线104至109所表示的双向通信链路上使用无线接入接口发送并且接收表示数据的信号来执行这些通信,其中,104、106和108表示从网络实体到通信设备的下行链路通信,并且105、107和109表示从通信设备到网络实体的上行链路通信。通信系统100可以根据任何已知协议进行操作,例如,在一些示例中,系统100可以根据3GPP长期演进(LTE)标准进行操作,其中,网络实体和通信设备一般分别称为eNodeB和UE。

[0041] 从以上在物理层处所解释的操作应理解,UE和eNodeB被配置为:发送并且接收表示数据的信号。故此,典型发射机/接收机链示出于图2和图3中。

[0042] 图2提供示出构成可以形成经由图1所示的LTE系统的无线接入接口的物理层传输的e-NodeB 101或通信设备104的部分的发射机的组件的示意性框图。在图2中,数据经由数据格式化器(data formatter)204处的输入201得以接收,并且形成为用于传输的帧或子帧。数据帧然后由纠错编码器206通过纠错码进行编码,并且馈送到码元形成器208,其将纠错编码的比特形成为比特群组,以用于映射为码元,以用于调制。数据码元然后由码元交织器210交织,并且馈送到OFDM调制器212,其通过已经从交织器210接收到的数据码元对OFDM码元的子载波进行调制。OFDM码元然后转换到RF频率,并且由发射机214经由天线216发送。

[0043] 对应地,操作为经由LTE无线接入接口接收经由用于通信设备104和eNodeB 101的物理层发送的数据的接收机包括接收机天线301,其检测经由无线接入接口发送到射频接收机302的射频信号。图3表示接收机的简化版本,并且若干块将构成OFDM解调器/均衡器304,其将时域OFDM码元转换到频域中,并且对OFDM码元的子载波进行解调,以恢复数据码元,并且执行解交织等。然而,OFDM解调器/均衡器304的输出用于将表示数据比特的已编码软判决值馈送到turbo解码器306。turbo解码器执行turbo解码算法,以检测并且恢复作为在与输入201对应的输出308上的数据比特流输出的所发送的数据比特的估计。

[0044] 应理解,已经绘制图2和图3,以示出完全并行turbo解码器执行检测纠错编码器206所编码的数据比特的本技术示例实施例。然而,应理解,根据本发明实施例的生成turbo检测处理可以提供给接收机链的其它部分,显著地对于遭受码间干扰的一些系统作为turbo均衡器,并且因此,术语检测器通常用于即指代均衡、解调又指代可以应用根据本技术的完全并行turbo解码算法的纠错解码二者。如上所述,完全并行turbo算法的其它应用包括:恢复典型地使用Log-BCJR算法解码的其它应用(例如均衡、同步、信道估计、LDPC解码和信源解码)中的数据的马尔科夫链。

[0045] 对于上述LTE的示例,在图4中示出图2所示的纠错编码器206的示例实施例。图4提

供示出对包括均具有二进制值 $b_{1,k}^u \in \{0,1\}$ 的N数量的比特的消息帧 $b_1^u = [b_{1,k}^u]_{k=1}^N$ 进行编码的简化turbo编码器的示例表示。该消息帧提供给上卷积编码器401和下卷积编码器403,如图4所示。上卷积编码器401执行卷积编码处理(例如以下提供的示例),以生成两个N比特已编码帧(即奇偶帧 $b_2^u = [b_{2,k}^u]_{k=1}^N$ 和系统性帧 $b_3^u = [b_{3,k}^u]_{k=1}^N$)。同时,消息帧 b_1^u 由内部turbo编码交织器404交织,以获得N比特所交织的消息帧 $b_1^l = [b_{1,k}^l]_{k=1}^N$,其如图4所示提供给下卷积编码器403,下卷积编码器403也应用卷积编码器以生成两个更多个N比特已编码帧(即奇偶帧 $b_2^l = [b_{2,k}^l]_{k=1}^N$ 和系统性帧 $b_3^l = [b_{3,k}^l]_{k=1}^N$)。在此,下标“u”和“l”分别指示与上卷积编码器401和下卷积编码器403的相关性。然而,以下,这些下标仅当必须在turbo编码器的两个卷积编码器401、403之间进行明确区分时使用,并且当讨论等同地应用于二者时忽略。注意,turbo编码器表示运载四个已编码帧的N比特的消息帧,包括总共4N比特,并且产生 $R = N / (4N) = 1/4$ 的turbo编码速率。

[0046] 如以上参照图2解释的那样,在turbo编码之后,已编码帧可以调制到无线信道上,并且发送到例如比如图3中提供的接收机。

[0047] LTE turbo编码器

[0048] 在图5中提供turbo编码器的更具体示例。图5提供与已经关于LTE标准[1]提出的示例对应的turbo编码器的示例。对于图5所示的示例,turbo编码器是1/3速率码,其中,从图2所示的数据格式化器204接收到的数据比特馈送到上卷积编码处理器401。在图5中可见,接收到的N比特的消息帧 $b_1^u = [b_{1,k}^u]_{k=1}^N$ 也经由turbo码内部交织器404馈送到下卷积编码处理器403。根据已知的布置,N比特的消息帧 $b_1^u = [b_{1,k}^u]_{k=1}^N$ 馈送到存储器元件406,其连接到其它存储器元件406,以形成移位寄存器类型布置。存储器元件406的输出用于形成对XOR单元408的输入,其在它们的输出处形成来自它们的输入的逻辑XOR的比特,其形成所编码的输出比特或作为对存储器元件406之一的输入而反馈的比特。上卷积编码器410中的开关在上卷积编码器的输入412和输出414之间切换输入比特,以分别在第一输出416上形成系统性帧 $b_3^u = [b_{3,k}^u]_{k=1}^N$ 并且在第三输出426上形成三个消息终止比特 $[b_{1,k}^u]_{k=N+1}^{N+3}$ 。

[0049] 上卷积编码器401的第二输出418提供奇偶帧 $b_2^u = [b_{2,k}^u]_{k=1}^{N+3}$ 。在图5中,三个消息终止比特 $[b_{1,k}^u]_{k=N+1}^{N+3}$ 用于在已知状态下终止上卷积编码器401,图4中为了简明未示出该情况。

[0050] 在下卷积编码器403中,开关420在从内部交织器404接收到的各比特之间进行切换,并且对应于用于上卷积编码器的开关410。以与上卷积编码器相似的方式,下卷积编码器的输出信道422、424分别提供奇偶帧 $b_2^l = [b_{2,k}^l]_{k=1}^{N+3}$ 和三个消息终止比特 $[b_{1,k}^l]_{k=N+1}^{N+3}$ 。因为下卷积编码器的系统性数据比特 $b_3^l = [b_{3,k}^l]_{k=1}^N$ 已经出现在第一输出416上,所以它们不是来自下卷积编码器的输出。相应地,通过第一输出416提供作为系统性的输入比特提供1/3速率码。关于上卷积编码器,三个消息终止比特 $[b_{1,k}^l]_{k=N+1}^{N+3}$ 用于在已知状态下终止下卷积编码器403,图4中为了简明未示出该情况。

[0051] 总之,图5的LTE turbo编码器[1]采用十二个附加终止比特,以将每个卷积编码器强制为最终状态 $S_{n+3} = 0$ 。更具体地说,上卷积编码器401生成三个消息终止比特

$b_{1,N+1}^u, b_{1,N+2}^u, b_{1,N+3}^u$ 以及三个奇偶终止比特帧 $b_{2,N+1}^u, b_{2,N+2}^u, b_{2,N+3}^u$ 。下卷积编码器 403 以相似方式进行操作,生成三个消息终止比特 $b_{1,N+1}^l, b_{1,N+2}^l, b_{1,N+3}^l$ 以及三个奇偶终止比特 $b_{2,N+1}^l, b_{2,N+2}^l, b_{2,N+3}^l$ 的对应集合。与上卷积编码器所产生的系统性帧 b_3^u 对比,下卷积编码器的系统性帧 b_3^l 并非由 LTE turbo 编码器输出。归因于此,LTE turbo 编码器使用总共 $(3N+12)$ 比特,以表示 N 比特的消息帧 b_4^u ,给出 $R=N/(3N+12)$ 的编码速率。

[0052] 图 5 中提出的 turbo 编码器的示例提供均具有三个存储器元件 406 的上卷积编码器 401 和下卷积编码器 403。卷积编码器领域技术人员应理解,存储器元件 406 的二进制内容可以解释为状态,从而卷积编码处理可以通过包括卷积编码器的可能状态的格栅合成为转变。故此,卷积编码器或 turbo 编码器可以描述为马尔科夫过程,并且因此表示为格栅图。图 6 中示出用于卷积编码器的状态转变图的示例。图 6 的状态转变图表示具有 $M=8$ 个状态和 $K=2$ 个每状态的转变的格栅的一个级,并且可以因此提供与以相同方式进行操作的上卷积编码器 401 和下卷积编码器 403 对应的示例。对于上卷积编码器 401 从初始状态 $S_0=0$ 开始,并且通过考虑对应消息比特 $b_{1,k}$ 接连转变到每个后续状态 $S_k \in \{0, 1, 2, \dots, M-1\}$ 下。由于存在用于消息比特的两个可能值 $b_{1,k} \in \{0, 1\}$,因此存在用于通过从前一状态转变可以到达的状态 S_k 的 $K=2$ 个可能值。在图 6 中,例如, $S_{k-1}=0$ 的前一状态暗指从 $S_k \in \{0, 4\}$ 选择后续状态。也可以使用助记符 $c(0,0)=1$ 和 $c(0,4)=1$ 来表达该示例,其中, $c(S_{k-1}, S_k)=1$ 指示卷积编码器可能从 S_{k-1} 转变到 S_k ,而 $c(S_{k-1}, S_k)=0$ 指示该转变是不可能的。在 $k=2$ 选项中,选择用于状态 S_k 的值,从而 $b_1(S_{k-1}, S_k)=b_{1,k}$ 。例如,在图 6 中, $S_{k-1}=0$ 和 $b_{1,k}=0$ 给出 $S_k=0$,而 $S_{k-1}=0$ 和 $b_{1,k}=1$ 给出 $S_k=4$ 。进而,根据 $b_{2,k}=b_2(S_{k-1}, S_k)$ 和 $b_{3,k}=b_3(S_{k-1}, S_k)$,对于奇偶帧 b_2 和系统性帧 b_3 中的对应比特选择二进制值。在图 6 的示例中, $S_{k-1}=0$ 和 $S_k=0$ 给出 $b_{2,k}=0$ 以及 $b_{3,k}=0$,而 $S_{k-1}=0$ 和 $S_k=4$ 给出 $b_{2,k}=1$ 以及 $b_{3,k}=1$ 。

[0053] turbo 编码器内部比特交织器

[0054] 如上所述,turbo 编码器典型地包括内部比特交织器 404,其从在上卷积编码器 401 与下卷积编码器 403 之间对数据比特进行编码的顺序对数据比特进行交织。例如,图 5 所示的 LTE turbo 编码器采用奇数-偶数交织器 [14],其支持范围 40 至 6144 比特中的各种帧长度 N。

[0055] 示例 WiMAX turbo 编码器

[0056] 与 LTE turbo 编码器的示例类似,根据 WiMAX 标准 [2] 进行操作的 turbo 编码器采用支持范围 24 至 2400 比特中的各种帧长度 N 的奇数-偶数交织器。然而,与 LTE turbo 编码器对比,WiMAX turbo 编码器是双二进制的 [2]。更具体地说,上 WiMAX 卷积编码器一次对两个 N 比特消息帧 b_1^u 和 b_2^u 进行编码。响应于此,根据 WiMAX 标准进行操作的 turbo 编码器产生四个 N 比特已编码帧(即两个奇偶帧 b_3^u 和 b_4^u 以及两个系统性帧 b_5^u 和 b_6^u)。同时,消息帧 b_1^u 和 b_2^u 交织,以获得两个 N 比特交织消息帧 b_1^l 和 b_2^l 。这两个 N 比特交织消息帧 b_1^l 和 b_2^l 由下卷积编码器 403 编码,以生成两个奇偶帧 b_3^l 和 b_4^l 。然而,如对于 LTE turbo 编码器的示例那样,下编码器的 N 比特系统性帧 b_5^l 和 b_6^l 并非由 WiMAX turbo 编码器输出。因此,WiMAX turbo 编码器使用六个已编码帧表示 2N 比特的消息帧,包括总共 6N 个比特并且产生 $R=(2N)/(6N)=1/3$ 的编码速率。

在WiMAX turbo编码器中,上卷积编码器401和下卷积编码器403基于与两个消息比特的四个可能组合对应的具有从M=8个状态中的每一个的K=4个转变的状态转变图而进行操作。并非采用终止比特,WiMAX采用咬尾,以确保S_N=S₀,这可能要求S_N和S₀具有非零值。

[0057] 使用Log BCJR算法的turbo解码器

[0058] 以上的部分已经参照图4、图5和图6描述了图2所示的turbo编码器。本技术实施例可以提供完全并行turbo解码器,与传统算法相比,其具有要实现的改进的解码速率、更小的存储器要求以及减少的数量的算术计算。为了更好地理解本技术关于传统turbo解码器或检测器所提供的改进,将首先描述根据传统log BCJR算法进行操作的传统turbo解码器,其被配置为:对已经根据LTE标准根据以上参照图4、图5和图6解释的turbo编码器的示例受turbo编码的数据比特的turbo编码帧进行解码。

[0059] 在它们通过无线信道的传输之后,图4所示的turbo编码器所生成的四个已编码帧 b_2^u 、 b_3^u 、 b_2^l 、 b_3^l 可以受解调,并且提供给图7的turbo解码器。然而,归因于无线信道中的噪声的影响,解调器将不清楚这些已编码帧中的比特值。因此,并非提供包括N个硬值比特的帧,解调器提供均包括N个软值先验对数似然率(LLR)的四个帧 $\bar{b}_2^{u,a} = [\bar{b}_{2,k}^{u,a}]_{k=1}^N$ 、 $\bar{b}_3^{u,a} = [\bar{b}_{3,k}^{u,a}]_{k=1}^N$ 、 $\bar{b}_2^{l,a} = [\bar{b}_{2,k}^{l,a}]_{k=1}^N$ 、 $\bar{b}_3^{l,a} = [\bar{b}_{3,k}^{l,a}]_{k=1}^N$ 。在此,属于比特 $b_{j,k}$ 的LLR定义为:

$$[0060] \quad b_{j,k} = \ln \frac{\Pr(b_{j,k}=1)}{\Pr(b_{j,k}=0)}, \quad (1)$$

[0061] 其中,可以附接下标“a”、“e”或“p”,以分别指示先验LLR、外在LLR或后验LLR。

[0062] Log-BCJR算法通常形成解码或检测处理,其通过表示马尔科夫过程的状态中的每一个的连接的格栅执行前向递归处理和后向递归处理(例如卷积编码器)。对于受turbo编码的数据,执行Log-BCJR解码处理的解码器包括上解码器和下解码器。上解码器和下解码器中的每一个均执行前向递归处理和后向递归处理,并且对于每个迭代生成外在LLR,其馈送到上解码器和下解码器中的另一个。

[0063] 图7提供示出与图4的简化turbo编码器对应的用于Log-BCJR算法的简化turbo解码器的示例实现方式的示意性框图。Log-BCJR turbo解码器迭代地进行操作,其中,I个迭代中的每一个包括所示的所有处理元件或算法块的操作。在Log-BCJR算法的前向递归和后向递归期间,上行和下行中的算法块的第k配对根据公式(2)~(6)执行与格栅的一个级有关的计算[4]:

$$[0064] \quad \tilde{\gamma}_k(S_{k-1}, S_k) = \left[\sum_{j=1}^I [b_j(S_{k-1}, S_k) \cdot \bar{b}_{j,k}^s] \right] + \ln [\Pr\{S_k | S_{k-1}\}] \quad (2)$$

$$[0065] \quad \tilde{\alpha}_k(S_k) = \max^*_{\{(S_{k-1}, S_k) | b_j(S_{k-1}, S_k) = 1\}} \{ \tilde{\gamma}_k(S_{k-1}, S_k) + \tilde{\alpha}_{k-1}(S_{k-1}) \} \quad (3)$$

$$[0066] \quad \tilde{\beta}_{k-1}(S_{k-1}) = \max^*_{\{(S_k, S_{k-1}) | b_j(S_{k-1}, S_k) = 1\}} \{ \tilde{\gamma}_k(S_{k-1}, S_k) + \tilde{\beta}_k(S_k) \} \quad (4)$$

$$[0067] \quad \tilde{\delta}_k(S_{k-1}, S_k) = \tilde{\gamma}_k(S_{k-1}, S_k) + \tilde{\alpha}_{k-1}(S_{k-1}) + \tilde{\beta}_k(S_k) \quad (5)$$

[0068]

$$\begin{aligned} & \tilde{b}_{j,k}^s = \\ & \left[\max^*_{\{(S_{k-1}, S_k) | b_j(S_{k-1}, S_k) = 1\}} \{ \tilde{\delta}_k(S_{k-1}, S_k) \} \right] - \left[\max^*_{\{(S_{k-1}, S_k) | b_j(S_{k-1}, S_k) = 0\}} \{ \tilde{\delta}_k(S_{k-1}, S_k) \} \right] - \bar{b}_{j,k}^s \end{aligned} \quad (6)$$

[0069] 在公式(2)中已经包括项 $\ln [\Pr\{S_k | S_{k-1}\}]$,以增加Log-BCJR算法的普适性,从而示出本技术可以应用于超越信道解码的应用。倘若已经进入前一状态S_{k-1},那么关于进入状态

S_k 的概率 $\Pr\{S_k | S_{k-1}\}$ ，该项促进利用接收机具有的任何附加先验知识。对于编码解调或turbo均衡，其可以源自信道特性的知识以及通过信道接收到的信息。当用于信源解码时，其可以源自信源概率分布的知识。相似地，该附加项提供大量其它应用，包括turbo同步和turbo信道估计。在没有附加先验知识可用的应用（例如turbo解码）中，于是该项可以省略，或设置为常数值，并且因此在以下讨论中可以忽略。

[0070] 如图7所示，处理元件或算法块601的第一集合贡献于对上卷积编码器401所产生的受turbo编码的数据执行turbo解码算法的第一部分。上解码器601的N个处理元件610的第一行贡献于通过可能状态的格栅执行前向递归处理，而N个处理元件612的第二行贡献于根据Log-BCJR算法通过格栅级执行后向递归。每个处理元件对应于包括前一状态集合与下一状态集合之间的转变集合的格栅中的N个级之一。2N个处理元件或算法块602的第二集合贡献于对下卷积编码器403所产生的受turbo编码的数据执行turbo解码算法的第二部分。如关于上解码器601那样，下解码器包括下解码器602的N个处理元件620的第一行，其贡献于通过可能状态的格栅执行前向递归处理，而N个处理元件622的第二行贡献于根据Log-BCJR算法通过格栅状态执行后向递归。

[0071] 贡献于执行Log-BCJR算法610的前向递归部分的上解码器601的N个处理元件610中的第k处理元件610被布置为从解调器接收对于上编码器401所生成的已编码比特 b_2^u 、 b_3^u 的帧所估计的第k LLR值 $\bar{b}_{2,k}^{u,a}$ 、 $\bar{b}_{3,k}^{u,a}$ 。对应地，贡献于执行Log-BCJR算法的前向递归部分的下解码器602的N个处理元件620中的第k处理元件620被布置为从解调器接收对于下编码器402所生成的已编码比特 b_2^l 、 b_3^l 的帧所估计的第k LLR值 $\bar{b}_{2,k}^{l,a}$ 、 $\bar{b}_{3,k}^{l,a}$ 。

[0072] 上检测处理器601和下检测处理器602中的均进而被布置为执行前向递归第k处理元件610、620一个接一个地串行采用公式(2)（对于该示例，没有项 $\ln[\Pr\{S_k | S_{k-1}\}]$ ），以组合 $L=3$ 个先验LLR $\bar{b}_{1,k}^a$ 、 $\bar{b}_{2,k}^a$ 、 $\bar{b}_{3,k}^a$ ，以获得用于（例如图6所示的）状态转变图中的每个转变的先验度量 $\bar{\gamma}_k(S_{k-1}, S_k)$ 。在该计算之后，执行前向递归的第k处理元件610、620中的每一个根据公式(3)将这些先验转变度量与 $\bar{\alpha}_{k-1}(S_{k-1})$ 的先验前向状态度量组合，以获得 $\bar{\alpha}_k(S_k)$ 的外在前向状态度量。这些外在状态度量然后传递到第k+1处理元件610、620，以采用作为下一时间段中的先验状态度量。然而，Log-BCJR算法领域技术人员应理解，turbo解码器的上解码器和下解码器交替地工作，从而当一个有效时，另一个待机。

[0073] 上turbo解码器601和下turbo解码器602中执行后向递归的第k处理元件612、622采用公式(4)，以组合用于每个转变的先验度量 $\bar{\gamma}_k(S_{k-1}, S_k)$ 与先验后向状态度量 $\bar{\beta}_k(S_k)$ 。这样产生外在后向状态度量 $\bar{\beta}_{k-1}(S_{k-1})$ ，其可以传递到第k-1处理元件，以采用作为下一时间段中的先验状态度量。此外，上turbo解码器601和下turbo解码器602中执行后向递归的第k处理元件612、622采用公式(5)，以获得用于状态转变图中的每个转变的后验度量 $\bar{\delta}_k(S_{k-1}, S_k)$ （例如，如图5所示）。最后，上turbo解码器401和下turbo解码器中执行后向递归的第k处理元件612、622采用公式(6)，以生成用于第k比特的外在消息LLR $\bar{b}_{j,k}^e$ 。在上解码器601与下解码器602之间交换这些LLR值。

[0074] 上解码器601和下解码器602关于帧的数据比特中的每一个交换外在LLR，其变为

已编码数据帧的系统性比特的估计。更具体地说，交织器604执行上解码器601与下解码器602之间所传递的数据比特的LLR值的解交织，以逆转turbo编码器的上卷积编码器401和下卷积编码器402所使用的数据比特的交织。

[0075] 交织器604与其它解码器601、602交换外在信息，该操作将其用作先验信息。更具体地说，如图7所示，上解码器601中执行后向递归的第k算法块612将用于消息数据比特的LLR值的外在估计 $\bar{b}_{1,k}^{u,e}$ 作为输出提供给交织器604，其在交织之后形成先验LLR值 $\bar{b}_{1,k}^{l,e}$ 作为对在下解码器602中执行前向递归的第k处理元件620的输入。

[0076] 对于Log-BCJR turbo解码器的第一解码迭代，对于先验消息LLR采用零值。图7的简化示例解码器可以使用具有L=3和L=2个用于LTE编码器的先验LLR以及具有L=6和L=4的WiMAX turbo码的块的(2)-(6)解码器的Log-BCJR算法关于LTE标准和WiMAX标准应用于示例turbo编码器。取决于是采用终止还是咬尾，可以对于Log-BCJR turbo解码器选择用于 $\bar{\alpha}_0$ 和 $\bar{\beta}_N$ 的值。

[0077] 传统turbo解码器的缺点

[0078] 应理解，在以上根据Log-BCJR解码处理的传统布置来解释turbo解码器中，各个检测处理器中的每一个必须等待，直到前向递归已经完成并且后向递归已经完成，然后将外在LLR信息经由交织器604输出到上检测处理器和下检测处理器中的另一个中的对应处理元件。此外，Log-BCJR turbo解码器迭代地进行操作，其中，I个迭代中的每一个包括图7所示的所有处理元件或算法块的操作。故此，需要T=4N个相继时间段完成每个解码迭代，从而4N个算法块按图7的粗箭头630所指示的顺序依次操作。这些箭头630指示施加图7所示的前向递归和后向递归的Log-BCJR算法的数据依赖性。因此，如下所述，当实现LTE turbo解码器或WiMAX turbo解码器时，Log-BCJR算法所需的时间段的数量是2N乘以需要T=1或T=2个时间段的实施本技术的所提出的(TtotP)完全并行turbo解码器，如下所述。

[0079] 完全并行turbo解码器

[0080] 与Log-BCJR算法对比，根据本技术实施例进行操作的检测器被配置为：尽可能远地移除数据依赖性，由此促进完全并行turbo解码。更具体地说，所提出的完全并行算法可以同时处理turbo码的两个分量中的所有受turbo编码的比特。该处理迭代地重复，直到已经执行足够数量的解码迭代。归因于此，可以使用显著低于[10]的现有技术turbo解码器所需的数量的仅数十个时间段来完成迭代解码处理。注意，先前已经提出多个完全并行turbo解码器，但它们遭受实施本技术的检测器/解码器中并不显现的显著缺点。在[11]中，采用最小求和算法以执行turbo解码。然而，该方法仅对于不包括任何标准（例如LTE和WiMAX）所采用的turbo码设计集合的非常有限的turbo码设计集合起作用。在[12]中提出使用模拟电流来表示软信息的完全并行turbo解码器实现方式，然而，其仅支持非常短的消息长度N。相似地，[13]提出基于随机比特序列而进行操作的完全并行turbo解码器算法。然而，该算法需要比Log-BCJR算法显著更多的时间段，因此具有显著更低的处理吞吐量。

[0081] 已经进行其它提议，以改进执行turbo检测的速率。例如，CN 102611464[17]、CN 102723958[18]、WO 2011/082509[19]和Xianjun J.等人题为“*A 122Mb/s Turbo decoder using a mid-range GPU*”的所公开的文章[20]公开了具有改进的处理性能以及减少的复杂度的turbo解码器。Xianjun J.等人的题为“*A 122Mb/s Turbo decoder using a mid-*

range GPU”的文章[20]公开了使用称为子解码器的多个处理元件,每个处理元件分配给码块的一个比特。故此,存在可能的并行处理方面的改进。然而,与本技术实施例对比,上述文章中指代的解码器并非同时操作解码器。

[0082] 本技术实施例可以提供处理以执行基于Log-BCJR turbo解码算法但简化的turbo解码处理以允许所有处理元件并行操作的完全并行布置。图8提供示例但简化的完全并行解码技术的布置。在图8中,各个上turbo解码部分701和下turbo解码部分702对应于Log-BCJR算法601、602的上turbo解码部分和下turbo解码部分,但为N个并行处理元件或算法块706、708所替换。因此,上解码器701包括N个处理元件706,而下解码器702包括N个处理元件708。

[0083] 如图8所示,并且与Log-BCJR算法的操作对应,图3的接收机中的解调器将先验LLR提供给如图8所示布置在两个行中的turbo解码器的2N个处理元件708、706(算法块)。更具体地说,在它们通过无线信道的传输之后,四个已编码帧 b_2^u 、 b_3^u 、 b_2^l 、 b_3^l 受解调并且提供给图8的turbo解码器。解调器将均包括N个软值先验对数似然率(LLR) $\bar{b}_2^{u,a} = [\bar{b}_{2,k}^{u,a}]_{k=1}^N$ 、 $\bar{b}_3^{u,a} = [\bar{b}_{3,k}^{u,a}]_{k=1}^N$ 、 $\bar{b}_2^{l,a} = [\bar{b}_{2,k}^{l,a}]_{k=1}^N$ 、 $\bar{b}_3^{l,a} = [\bar{b}_{3,k}^{l,a}]_{k=1}^N$ 的四个帧提供给完全并行turbo解码器的2N个处理元件或算法块,其中,先验奇偶校验LLR $\bar{b}_{2,k}^{u,a}$ 和先验系统性LLR $\bar{b}_{3,k}^{u,a}$ 提供给图8所示的上解码器701中的第k算法块706。此外,交织器704对上解码器701中的第k算法块提供先验消息LLR $\bar{b}_{1,k}^{u,a}$,如以下将详述的那样。同时,对下解码器702中的第k算法块对应地提供先验LLR值 $\bar{b}_{1,k}^{u,a}$ 、 $\bar{b}_{2,k}^{l,a}$ 、 $\bar{b}_{3,k}^{l,a}$ 。除此而外,还对上解码器701和下解码器702中的每一个中的第k算法块706、708提供先验前向状态度量的矢量 $\bar{\alpha}_{k-1} = [\bar{\alpha}_{k-1}(S_{k-1})]_{S_{k-1}=0}^{M-1}$ 以及先验后向状态度量的矢量 $\bar{\beta}_k = [\bar{\beta}_k(S_k)]_{S_k=0}^{M-1}$,如以下将详述的那样。与根据以上参照图7所描述的BCJR算法进行操作的传统turbo解码器不同,上解码器701和下解码器702的处理元件706、708中的每一个以相同方式进行操作,以接收对应于与格栅级关联的一个或多个数据码元的用于上解码器701的处理元件706的 $\bar{b}_2^{u,a} = [\bar{b}_{2,k}^{u,a}]_{k=1}^N$ 、 $\bar{b}_3^{u,a} = [\bar{b}_{3,k}^{u,a}]_{k=1}^N$ 或用于下解码器702中的处理元件708的的软判决先验LLR值 $\bar{b}_2^{l,a} = [\bar{b}_{2,k}^{l,a}]_{k=1}^N$ 、 $\bar{b}_3^{l,a} = [\bar{b}_{3,k}^{l,a}]_{k=1}^N$,并且从一个相邻处理元件接收先验前向状态度量 $\bar{\alpha}_{k-1}$,从第二相邻处理元件接收先验后向状态度量 $\bar{\beta}_k$,并且从第二检测处理器接收用于对于与第k处理元件关联的格栅级正检测的数据码元的先验LLR值 $\bar{b}_{1,k}^{u,a}$ 。每个处理元件执行与包括前一状态集合与下一状态集合之间的转变集合的一个格栅级关联的计算。每个处理元件被配置为:根据以下公式(7)至(10)组合与数据码元有关的先验前向状态度量 $\bar{\alpha}_{k-1} = [\bar{\alpha}_{k-1}(S_{k-1})]_{S_{k-1}=0}^{M-1}$,先验后向状态度量 $\bar{\beta}_k = [\bar{\beta}_k(S_k)]_{S_k=0}^{M-1}$ 以及先验LLR值 $\bar{b}_{1,k}^{u,a}$ 。

[0084] $\delta_k(S_{k-1}, S_k) = \left[\sum_{j=1}^L [b_j(S_{k-1}, S_k) \cdot \tilde{b}_{j,k}^s] \right] + \ln \{Pr\{S_k | S_{k-1}\}\} + \bar{\alpha}_{k-1}(S_{k-1}) + \bar{\beta}_k(S_k) \quad (7)$

[0085] $\bar{\alpha}_k(S_k) = \left[\max^*_{\{S_{k-1} | c(S_{k-1}, S_k) = 1\}} [\delta_k(S_{k-1}, S_k)] \right] - \bar{\beta}_k(S_k) \quad (8)$

[0086] $\bar{\beta}_{k-1}(S_{k-1}) = \left[\max^*_{\{S_k | c(S_{k-1}, S_k) = 1\}} [\delta_k(S_{k-1}, S_k)] \right] - \bar{\alpha}_{k-1}(S_{k-1}) \quad (9)$

[0087]

$$\tilde{b}_{j,k}^a = \left[\max^*_{\{(S_{k-1}, S_k) | b_j(S_{k-1}, S_k) = 1\}} \{\tilde{\delta}_k(S_{k-1}, S_k)\} \right] - \left[\max^*_{\{(S_{k-1}, S_k) | b_j(S_{k-1}, S_k) = 0\}} \{\tilde{\delta}_k(S_{k-1}, S_k)\} \right] - \tilde{b}_{j,k}^e \quad (10)$$

[0088] 每个处理元件706、708因此根据以上公式(7)至(10)产生外在前向状态度量 $\bar{\alpha}_k = [\bar{\alpha}_k(S_k)]_{S_k=0}^{M-1}$ 以及外在后向状态度量 $\bar{\beta}_{k-1} = [\bar{\beta}_{k-1}(S_{k-1})]_{S_{k-1}=0}^{M-1}$ 以及一个或多个外在LLR值 $\bar{b}_{j,k}^e$ 。处理元件706、708然后将外在前向状态度量 $\bar{\alpha}_k$ 传递到第二相邻处理元件,其变为用于下一迭代的先验前向状态度量,并且将外在后向状态度量 $\bar{\beta}_{k-1}$ 传递到第一相邻处理元件,其变为用于所述下一迭代的先验后向状态度量,并且将用于数据信息的一个或多个外在LLR值 $\bar{b}_{j,k}^e$ 传递到其它检测处理器,其变为用于turbo检测处理的下一迭代的先验LLR值。

[0089] 因此应理解,上turbo解码器701和下turbo解码器702中的每一个使用公式(7)至(10)以组合L=3个先验LLR值 $\bar{b}_{1,k}^a$ 、 $\bar{b}_{2,k}^a$ 、 $\bar{b}_{3,k}^a$ 以及先验前向状态度量 $\bar{\alpha}_{k-1}$ 以及先验后向状态度量 $\bar{\beta}_k$ 。这样产生用于第k级的状态转变图中的可能转变中的每一个(即,用于 $c(S_{k-1}, S_k) = 1$ 的前一状态 S_{k-1} 和下一状态 S_k 的每个配对)的后验转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 。然后通过(8)、(9)和(10)组合这些后验转变度量,以分别产生外在前向状态度量的矢量 $\bar{\alpha}_{k-1} = [\bar{\alpha}_{k-1}(S_{k-1})]_{S_{k-1}=0}^{M-1}$ 以及外在后向状态度量的矢量 $\bar{\beta}_k = [\bar{\beta}_k(S_k)]_{S_k=0}^{M-1}$ 以及外在消息LLR $\bar{b}_{1,k}^e$ 。这些公式采用雅可比对数,其关于两个操作数定义为:

[0090]

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) = \max(\bar{\delta}_1, \bar{\delta}_2) + \ln(1 + e^{-|(\bar{\delta}_1 - \bar{\delta}_2)|}) \quad (11)$$

[0091] 并且可以通过利用其关联性性质而扩展到更多操作数。替代地,可以通过以下近似[4]之一可选地或有利地替换(11)的精确max*运算符:

[0092]

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) \approx \max(\bar{\delta}_1, \bar{\delta}_2) \quad (12)$$

[0093]

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) \approx A \cdot \max(\bar{\delta}_1, \bar{\delta}_2) + B \quad (13)$$

[0094]

$$\max^*(\bar{\delta}_1, \bar{\delta}_2) \approx A(\bar{\delta}_1, \bar{\delta}_2) \cdot \max(\bar{\delta}_1, \bar{\delta}_2) + B(\bar{\delta}_1, \bar{\delta}_2) \quad (14)$$

[0095] 以通过稍微使得其纠错性能降级的代价来减少所提出的完全并行turbo解码器的复杂度。注意,A和B在公式(12)中是常数,而 $A(\bar{\delta}_1, \bar{\delta}_2)$ 、 $B(\bar{\delta}_1, \bar{\delta}_2)$ 在公式(14)中是 $(\bar{\delta}_1, \bar{\delta}_2)$ 的简单函数。

[0096] 所提出的完全并行turbo解码器迭代地进行操作,其中,I个迭代中的每一个包括图8所示的所有处理元件或算法块的操作。通过同时操作所有2N个算法块,由于仅在t=1个时间段内完成每个迭代,因此turbo解码器可以看作完全并行的。通常,图7中的每个算法块所产生的外在信息与相连的算法块所提供的外在信息交换,以用作下一解码迭代中的先验信息。更具体地说,上解码器701和下解码器702中的每一个中的第k算法块706、708通过交

织器704传递外在消息LLR $\bar{b}_{1,k}^e$,以在下一解码迭代期间由上解码器701和下解码器702中的另一个中的相连块或处理元件706、708用作先验LLR。同时,上解码器701和下解码器702中的另一个中的该处理元件或算法块706、708提供外在消息LLR,其在下一解码迭代期间用作先验消息LLR $\bar{b}_{1,k}^a$ 。经由交织器704的变为用于上解码器701和下解码器702中的另一个中的迭代的先验消息LLR $\bar{b}_{1,k}^a$ 的上解码器701与下解码器702之间的外在LLR消息信息的这种交换实质上对应于以上参照图7解释的传统Log-BCJR turbo解码器的操作。然而,如下所述,为了实现完全并行turbo解码器,上解码器和下解码器中的每一个中的第k处理元件或算法块706、708提供外在前向状态度量的矢量 $\bar{\alpha}_k = [\bar{\alpha}_k(S_k)]_{S_k=0}^{M-1}$ 和外在后向状态度量 $\bar{\beta}_{k-1} = [\bar{\beta}_{k-1}(S_{k-1})]_{S_{k-1}=0}^{M-1}$,以用于相邻算法块在下一解码迭代中采用。

[0097] 从以上解释应理解,本技术实施例可以提供用于turbo检测器或解码器的完全并行实现方式。为此,通过将公式(2)带入公式(5)中来减少(图7中通过粗箭头630示出的)Log-BCJR算法中出现的数据依赖性。因此,处理元件或算法块706、708中的每一个被布置为:通过组合对于该迭代作为输入而提供以分别形成外在前向状态度量和后向状态度量的先验前向状态度量和先验后向状态度量来确定用于第k级的状态转变图中的可能转变中的每一个(即,用于 $c(S_{k-1}, S_k) = 1$ 的前一状态 S_{k-1} 和下一状态 S_k 的每个配对)的状态转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 。此外,使用恒等式 $\max^*(\bar{\delta}_1 - \bar{\delta}_3, \bar{\delta}_2 - \bar{\delta}_3) = \max(\bar{\delta}_1, \bar{\delta}_2) - \bar{\delta}_3$,可以通过重新排列(5)并且将其代入(3)和(4)中来推导(8)和(9)。

[0098] 根据以上解释,处理元件中的每一个考虑图6中所例示的一个格栅级,并且被布置为在时钟周期或时间段中接收先验前向状态度量和先验后向状态度量二者,并且输出外在前向状态度量和后向状态度量。它们提供给相邻处理元件,以用在下一时钟周期或时间段中。在一些示例中,在单个时钟周期中,来自解码器中的至少两个的多个处理元件可以并行地进行操作。此外,使用定点表示法来表示先验/外在前向状态度量、先验/外在后向状态度量、先验消息LLR和外在消息信息。

[0099] 在完成最终解码迭代之后,属于第k消息比特 b_k 的后验LLR可以得以获得作为 $\bar{b}_{1,k}^p = \bar{b}_{1,k}^{u,a} + \bar{b}_{1,k}^{u,3}$,并且也可以与系统性信息 $\bar{b}_{3,k}^{u,a}$ 组合。作为二进制测试的结果 $\bar{b}_{1,k}^p > 0$,于是可以获得消息比特 b_k 的估计。

[0100] 在图9中提供示出图8所示的处理元件706、708的实现方式的示意性框图。图9根据公式(7)、(8)、(9)和(10)有效地形成用于第k格栅级的计算。图9的注释指示用于在八个状态之间采用十六个转变的LTE turbo码的示例的数据路径宽度。如图9所示,处理元件706包括四个子处理元件901、902、904、906,其分别根据公式(7)、(8)、(9)和(10)执行计算。第一子处理元件901接收用于求和单元910接收到的第一输入908上的格栅级所考虑的L个消息比特、系统性比特或奇偶比特中的每一个的先验LLR。求和单元910有效地组合先验LLR消息值,以形成用于十六个转变 (S_{k-1}, S_k) 中的每一个的先验转变度量。在加法器914的第一输入912上接收求和单元910的输出。加法器在第二输入916上从第k-1处理元件接收用于八个前一状态 (S_{k-1}) 中的每一个的先验前向状态度量 $\bar{\alpha}_{k-1}(S_{k-1})$,并且在第二输入918上从第k处理元件接收用于八个下一状态 S_k 中的每一个的先验后向状态度量 $\bar{\beta}_k(S_k)$ 。在加法器914的输

出920处,根据公式(7)形成用于格栅的第k级中的十六个转变中的每一个的后验状态转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 。后验状态转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 馈送到第二子处理元件902和第三子处理元件904,以分别执行公式(8)和(9),以计算用于格栅中的第k级的八个下一状态 S_k 中的每一个的外在前向状态度量 $\bar{\alpha}_k(S_k)$,并且计算用于八个前一状态 S_{k-1} 中的每一个的外在后向状态度量 $\bar{\beta}_{k-1}(S_{k-1})$ 。在信道920上从加法器914的输出接收到的十六个后验转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 馈送到第二子处理元件902内的max*计算单元922,其生成用于八个下一状态(S_k)的后验输出,其在第一输入924上馈送到减法单元926。在减法单元926的第二输入928上,用于八个下一状态(S_k)中的每一个的先验后向状态度量 $\bar{\beta}_k(S_k)$ 得以馈送,并且减法单元926在输出930处根据公式(8)生成用于格栅级的八个下一状态(S_k)中的每一个的外在前向状态度量 $\bar{\alpha}_k(S_k)$ 。第三子处理元件904在max*计算单元940处通过输入920接收十六个后验转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$,max*计算单元940生成用于八个前一状态(S_{k-1})中的每一个的后验输出,其馈送到减法单元944的第一输入942。减法单元944的第二输入946接收用于八个前一状态的八个前向状态度量 $\bar{\alpha}_{k-1}(S_{k-1})$ 。减法单元944在输出946处根据公式(9)形成用于八个前一状态(S_{k-1})中的每一个的外在后向状态度量 $\bar{\beta}_{k-1}(S_{k-1})$ 。第四子处理元件906根据公式(10)进行处理,并且可以生成用于格栅级所考虑的L数量的消息比特、系统性比特或奇偶比特中的每一个的外在LLR。然而,在一些应用中,出于与一个或多个其它检测器电路的迭代交换的缘故,消息比特仅需要LLR。例如,存在用于LTE解码器的一个外在消息LLR以及用于WiMAX解码器的两个LLR。第四子处理元件906包括max*计算单元块950,其接收十六个后验转变度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 的集合,并且选择暗指 $b_j(S_{k-1}, S_k) = 1$ 的比特值的与转变对应的八个的子集。第二max*计算单元块952选择与 $b_j(S_{k-1}, S_k) = 0$ 的比特值对应的用于转变的后验转变度量的子集 $\bar{\delta}_k(S_{k-1}, S_k)$ 。各个第一max*计算单元块950和第二max*计算单元块952的输出可以关于L数量的消息比特、系统性比特或奇偶比特得以生成,并且馈送到减法单元958的第一输入954和第二输入956。减法单元958可以在第三输入960上接收L数量的先验LLR。减法单元958可以在输出962处形成用于L数量的消息比特、系统性比特或奇偶比特中的每一个的外在LLR $\bar{b}_{j,k}^e$ 。

[0101] 从以上所提供的解释应理解,移除数据依赖性的功能之一是:从与转变分支的max*值组合的用于第k级的后向状态度量 $\bar{\beta}_k(S_k)$ 立即计算用于第k级的前向状态度量 $\bar{\alpha}_k(S_k)$ 。相似地,从与转变分支度量的max*值组合的从第二相邻处理元件接收到的用于第k级的前向状态度量 $\bar{\alpha}_{k-1}(S_{k-1})$ 计算用于第k级的后向状态度量 $\bar{\beta}_{k-1}(S_{k-1})$ 。相应地,处理元件不必等待直到所有其它处理元件已经完成前向递归或后向递归,以通过组合所计算的前向状态度量和后向状态度量与从接收到的先验LLR消息值确定的转变状态度量来计算用于第k级的外在LLR消息值。正是特征的这种组合允许处理元件形成已经减少数据依赖性的并行处理turbo解码器的部分,因此,在每个迭代时,处理元件中的每一个并行地进行操作,并且

用于上解码器和下解码器的处理元件在同一时钟周期中同期地进行操作。

[0102] 更详细示例实施例

[0103] 在图10中提供根据本技术实施例的完全并行turbo解码器的详细实现方式。图10呈现图8所示的但包括用于第k格栅级和处理元件处理一个先验消息LLR $\bar{b}_{1,k}^a$ ，一个先验奇偶校验LLR $\bar{b}_{2,k}^a$ 和一个先验系统性LLR $\bar{b}_{3,k}^a$ 的示例的更详细实现方式的完全并行turbo解码器。然而，图8中也出现的部分具有相同标号，并且为了简明，将仅描述相对于图8的差异。

[0104] 根据本技术，检测电路的处理元件中的每一个可以包括控制单元，并且对于处理元件中的每一个，可选择的旁路电路，其得以提供以用于处理元件，并且可配置为：在控制单元的控制下旁路处理元件，以禁用处理元件。旁路电路可以实现为可控制的复用器以及用于存储前向状态度量或后向状态度量以用于馈送到相邻处理元件的寄存器。故此，可以有选择地启用/禁用处理元件，以当不需要它们时进行功率节约。

[0105] 如上所述，turbo编码器典型地包括交织器，其根据交织模式对第一卷积编码器所编码的数据码元进行交织，然后由第二卷积编码器编码。故此，控制器可以控制可选择的旁路电路，以根据交织器所使用的交织模式来禁用上检测电路和下检测处理器的所选择的处理元件。交织器可以因此可配置为：根据交织模式对第一检测电路与第二检测电路之间所交换的外在软判决值进行解交织。控制器可以因此在各时钟周期之间并且根据交织模式交替地禁用上检测电路的处理元件和下检测电路的处理元件。

[0106] 如图10所示，先验奇偶LLR $\bar{b}_{2,k}^a$ 和系统性LLR $\bar{b}_{3,k}^a$ 中的每一个分别在寄存器1001、1002中得以接收，并且贯穿迭代解码处理得以保存，以用于馈送到处理元件706、708中，以用于上解码器701和下解码器702。每个解码器中的第k处理元件采用存储寄存器1006、1012和1014，以用于在相继时钟周期之间分别存储先验前向状态度量 $\bar{\alpha}_{k-1}$ 、先验后向状态度量 $\bar{\beta}_k$ 和先验消息LLR $\bar{b}_{1,k}^a$ 。外在前向状态度量 $\bar{\alpha}_k$ 和外在后向状态度量 $\bar{\beta}_{k-1}$ 分别提供给复用器1004和1010。复用器1004、1010可以被配置为：将外在前向状态度量 $\bar{\alpha}_k$ 提供给第k+1处理元件，并且将外在后向状态度量 $\bar{\beta}_{k-1}$ 提供给第k-1处理元件。替代地，复用器1004、1010可以被配置为：旁路第k处理元件。这样允许所选择的处理元件关闭，以反映较短帧长度和对应交织器模式。相应地，可以实现能耗方面的对应节约。对于图8所示的示例，外在消息LLR $\bar{b}_{1,k}^e$ 从上解码器701或下解码器702中的第k处理元件输出，并且通过交织器704得以馈送，以形成待用在下一时钟周期中的用于其它解码器的先验消息LLR。相应地，寄存器1014在相继时钟周期之间存储所输入的先验消息LLR $\bar{b}_{1,k}^a$ 。通过组合先验消息LLR $\bar{b}_{1,k}^a$ 、外在消息LLR $\bar{b}_{1,k}^e$ 以及可选地先验系统性LLR $\bar{b}_{3,k}^a$ 的加法器1016来形成后验消息LLR $\bar{b}_{1,k}^p$ 的确定。turbo解码器可以受控于控制单元1040，其提供用于复用器1004和1010的控制信号以及如以下解释的那样将需要的用于寄存器1006、1012和1014的全局重置为零信号。控制单元1040也可以用于重新配置交织器704，其可以实现为Benes网络或具有复用器。替代地，交织器可以是硬布线的，并且以上所描述的旁路电路可以受控，以选择交织器的特定子集。

[0107] 初始化状态

[0108] 在第一解码迭代的开始，外在信息是不可用的。在此情况下，每个行中的第k算法

块对于先验LLR值 $\bar{b}_{1,k}^a$ 、先验前向状态度量 $\bar{\alpha}_{k-1}$ 和先验后向状态度量 $\bar{\beta}_k$ 采用零值。可以对于寄存器1006、1012和1014使用全局重置为零来实现该情况。然而，除此而外，由于卷积编码器总是从 $S_0=0$ 的初始状态开始，因此每个行中的第一算法块贯穿所有解码迭代采用 $\bar{\alpha}_0 = [0, -\infty, -\infty, \dots, -\infty]$ 。相似地，由于当不采用终止时，接收机预先并不获知卷积编码器的最终状态 S_N ，因此来自每个行的最后算法块贯穿所有解码迭代采用 $\bar{\beta}_N = [0, 0, 0, \dots, 0]$ 。以下讨论当采用终止或咬尾时的第一算法块和最后算法块的操作。

[0109] 奇数-偶数turbo解码器

[0110] 如上所述，一些turbo编码器（例如根据LTE turbo码和WiMAX turbo码进行操作的编码器）采用奇数-偶数交织器[14]。更明确地，LTE交织器704仅将来自具有奇数指数k的上解码器701的算法块连接到来自也具有奇数指数k的下解码器702的算法块。相似地，上解码器701中的具有偶数指数的算法块仅连接到下解码器702中具有偶数指数k的算法块。归因于此，图8和图10的2N个算法块可以分组为两个集合，其中，特定集合内的所有算法块是独立的，没有对彼此的连接。第一集合包括来自具有奇数指数k的上解码器的所有算法块或处理元件以及来自具有偶数指数k的下解码器的所有块，在图8和图10中通过浅阴影描述该情况。第二集合是第一集合的补集，包括图8和图10中具有深阴影的算法块。以此方式，2N个算法块之间的外在信息的迭代交换可以另外看作算法块的两个集合之间的外在信息的迭代交换。对于图10所示的示例，交织器704已经采用奇数-偶数交织模式（例如LTE turbo编码器或WiMAX turbo编码器所执行的奇数-偶数交织模式）。

[0111] 更一般地，对于交织器设计防止分组为独立算法块集合的示例，所有算法块在与每解码迭代的T=1个时间段对应的每一时间段中操作。然而，在奇数-偶数交织器的情况下，独立算法块的两个集合的同时操作类似于杂耍两个球，其在两只手之间同时扔出，但保持彼此独立。在该完全并行turbo解码器中，这对应于对彼此没有影响的两个独立迭代解码处理。因此，这些独立迭代解码处理之一可以看作是冗余的，并且可以丢弃。可以通过在每个时间段中操作仅一个集合的算法块来实现该情况，其中，相继时间段在两个集合之间交替。通过该方法，每个解码迭代可以看作包括T=2个相继时间段。虽然这是上述T=1方法所需的数量的两倍，但这种T=2方法需要一半那样多的解码迭代，以实现相同的纠错性能。因此，T=2方法保持与T=1方法相同的处理吞吐量和时延，但实现每消息帧的复杂度方面的50%减少。

[0112] 因此，在图8和图10中，处理元件（算法块）的两个集合如通过较浅阴影和较深阴影所示那样分组。如图10所示，处理元件中的每一个取决于它们是奇数还是偶数而具有较深阴影或较浅阴影。根据本技术，在交织器704执行奇数和偶数交织的情况下，于是上解码器或下解码器中的奇数比特与上解码器和下解码器中的另一个中的奇数比特交织。

[0113] 此外，虽然在特定时间段中正使用算法块的一个集合以对特定消息帧进行解码，但可以使用块的另一集合以对不同消息进行解码可以同时受操作，在对两个不同消息帧同时进行解码之间交替，并且促进总体处理吞吐量方面的100%增加。如下所述，图12示出采用一半那样多的处理元件以对单个消息帧进行解码的替选布置。在此，在相继时间段中，每个处理元件在上解码器中与下解码器中的算法块的作用之间进行交替。

[0114] 图11提供完全并行turbo解码器包括对彼此没有影响的两个独立迭代解码处理的

本技术实施例。图11关于根据LTE编码器所turbo编码的数据的turbo解码对应于图10所示的示例实施例，并且对应于图8，并且因此，将仅描述差异。在图11中，图10的turbo解码器示出为对于每个处理元件706、708包括在控制单元1040的控制下进行操作的寄存器1001.1、1001.2、1002.1、1002.2的两个集合。集合中的每一个存储用于不同受turbo编码的数据帧的各个软判决值的先验LLR。用于上解码器701的带奇数指数的(带较浅阴影的)处理元件706以及用于下解码器702的带偶数指数的(带较浅阴影的)处理元件708从受turbo编码的数据的第一帧在一个时间段中从寄存器元件1001.1、1002.1的第一集合分别接收软判决值 $\bar{b}_{2,kk}^{u,a}$ 是奇数、 $\bar{b}_{3,kk}^{u,a}$ 是奇数以及 $\bar{b}_{2,kk}^{l,a}$ 是偶数、 $\bar{b}_{3,kk}^{l,a}$ 是偶数。用于上解码器701的的带偶数指数的(较深阴影的)处理元件706和用于下解码器702的带奇数指数的(较深阴影的)处理元件708在同一时间段中从受turbo编码的数据的第二帧从寄存器元件1001.2、1002.2的第二集合分别接收软判决值 $\bar{b}_{2,kk}^{u,a}$ 是偶数、 $\bar{b}_{3,kk}^{u,a}$ 是偶数以及 $\bar{b}_{2,kk}^{l,a}$ 是奇数、 $\bar{b}_{3,kk}^{l,a}$ 是奇数。在下一时间段中，用于上解码器701的带偶数指数的(较深阴影的)处理元件706以及用于下解码器702的带奇数指数的(较深阴影的)处理元件708从受turbo编码的数据的第一帧分别从寄存器元件1001.1、1002.1的第一集合接收软判决值 $\bar{b}_{2,kk}^{u,a}$ 是偶数、 $\bar{b}_{3,kk}^{u,a}$ 是偶数以及 $\bar{b}_{2,kk}^{l,a}$ 是奇数、 $\bar{b}_{3,kk}^{l,a}$ 是奇数，并且用于上解码器701的带奇数指数(较浅阴影的)处理元件706以及用于下解码器702的带偶数指数(较浅阴影的)处理元件708从受turbo编码的数据的第二帧分别从寄存器元件1001.2、1002.2的第二集合接收软判决值 $\bar{b}_{2,kk}^{u,a}$ 是奇数、 $\bar{b}_{3,kk}^{u,a}$ 是奇数以及 $\bar{b}_{2,kk}^{l,a}$ 是偶数、 $\bar{b}_{3,kk}^{l,a}$ 是偶数。相应地，可以同时对受turbo编码的数据的第一帧和第二帧二者进行解码。相应地，加法器1016将在相继时间段中在提供属于第一帧的后验LLR $\bar{b}_{1,k}^p$ 和属于第二帧的后验LLR $\bar{b}_{1,k}^{p'}$ 之间交替。

[0115] 在图12中示出另一实施例。图12所示的实施例利用奇数-偶数交织，效果是：使得用于执行turbo解码处理的处理元件706的数量减半。这对应于关于根据LTE编码器受turbo编码的数据的turbo解码的图10所示的示例实施例，并且因此对应于图8，并且因此，将仅描述差异。

[0116] 在图12中，图10的turbo解码器修改为包括仅一半数量的处理元件706。对处理元件中的每一个提供寄存器1201.1、1201.2、1202.1、1202.2的两个集合。集合中的每一个关于帧的受上turbo编码和和下turbo编码的数据存储用于各个软判决值的两个先验LLR $\bar{b}_{2,k}^{u,a}$ 、 $\bar{b}_{3,k}^{u,a}$ 和 $\bar{b}_{2,k}^{l,a}$ 、 $\bar{b}_{3,k}^{l,a}$ 。

[0117] 图12所示的处理元件706的布置被配置为：在控制单元1040的控制下对数据码元帧根据本技术执行turbo解码处理。通过在交替周期中利用可以通过奇数-偶数交织器实现的处理方面的50%减少，处理元件通过在用于经由复用器1204、1206馈送的从寄存器1201.1、1202.1馈送的上卷积编码数据的软判决值 $\bar{b}_{2,k}^{u,a}$ 、 $\bar{b}_{3,k}^{u,a}$ 与经由复用器1204、1206馈送的从寄存器1201.2、1202.2馈送的下卷积编码数据的软判决值 $\bar{b}_{2,k}^{l,a}$ 、 $\bar{b}_{3,k}^{l,a}$ 之间交替而根据本技术执行turbo解码处理。此外，交替周期处理奇数数据码元和偶数数据码元。交织器704经由解复用器1218和交织器704从处理元件706馈送奇数外在数据码元值并且然后偶数外在数据码元值 $\bar{b}_{1,k}^e$ ，以待存储在寄存器1212、1214中并且经由复用器1216馈送到处理元件作为用

于下一周期的先验数据信息 $\bar{b}_{1,k}^a$ 。

[0118] 在具有奇数指数的时钟周期中,具有奇数指数k的每个处理元件的复用器1204、1206和1216被配置为:从寄存器1201.1、1202.1和1214进行读取。这些寄存器提供属于上解码器的先验LLR $\bar{b}_{1,k|k}^{u,a}$ 是奇数、 $\bar{b}_{2,k|k}^{u,a}$ 是奇数、 $\bar{b}_{3,k|k}^{u,a}$ 是奇数。解复用器1218被配置为:提供传递到交织器704和加法器1016的外在LLR $\bar{b}_{1,k|k}^{u,e}$ 是奇数,其生成后验LLR $\bar{b}_{1,k|k}^p$ 是奇数。通过交织器704提供的先验LLR $\bar{b}_{1,k|k}^{l,a}$ 是奇数写入寄存器1212,准备用在下一时钟周期中。同时,具有偶数指数k的每个处理元件的复用器1204、1206和1216被配置为:在具有奇数指数的时钟周期中从寄存器1201.2、1202.2和1212进行读取。这些寄存器提供属于下解码器的先验LLR $\bar{b}_{1,k|k}^{l,a}$ 是偶数、 $\bar{b}_{2,k|k}^{l,a}$ 是偶数、 $\bar{b}_{3,k|k}^{l,a}$ 是偶数。解复用器1218被配置为:提供传递到交织器704的外在LLR $\bar{b}_{1,k|k}^{l,e}$ 是偶数。通过交织器704提供的先验LLR $\bar{b}_{1,k|k}^{u,a}$ 是偶数写入寄存器1214,准备用在下一时钟周期中。

[0119] 在具有偶数指数的时钟周期中,具有偶数指数k的每个处理元件的复用器1204、1206和1216被配置为:从寄存器1201.1、1202.1和1214进行读取。这些寄存器提供属于上解码器的先验LLR $\bar{b}_{1,k|k}^{u,a}$ 是偶数、 $\bar{b}_{2,k|k}^{u,a}$ 是偶数、 $\bar{b}_{3,k|k}^{u,a}$ 是偶数。解复用器1218被配置为:提供传递到交织器704和加法器1016的外在LLR $\bar{b}_{1,k|k}^{u,e}$ 是偶数,其生成后验LLR $\bar{b}_{1,k|k}^p$ 是偶数。通过交织器704提供的先验LLR $\bar{b}_{1,k|k}^{l,a}$ 是偶数写入寄存器1212,准备用在下一时钟周期中。同时,具有奇数指数k的每个处理元件的复用器1204、1206和1216被配置为:在具有偶数指数的时钟周期中从寄存器1201.2、1202.2和1212进行读取。这些寄存器提供属于下解码器的先验LLR $\bar{b}_{1,k|k}^{l,a}$ 是奇数、 $\bar{b}_{2,k|k}^{l,a}$ 是奇数、 $\bar{b}_{3,k|k}^{l,a}$ 是奇数。解复用器1218被配置为:提供传递到交织器704的外在LLR $\bar{b}_{1,k|k}^{l,e}$ 是奇数。通过交织器704提供的先验LLR $\bar{b}_{1,k|k}^{u,a}$ 是奇数写入寄存器1214,准备用在下一时钟周期中。

[0120] 因此应理解,图12所示的示例实施例可以使用内部奇数-偶数交织器对已经受turbo编码的数据帧执行turbo解码处理,效果是:可以实现处理硬件方面的50%减少。

[0121] turbo解码器的终止

[0122] 如上所述,在一些示例中,使用一个或多个附加比特或码元来终止数据帧,从而解码器可以获知帧终止的状态。对于LTE turbo编码器的示例,使用十二个终止比特以将其卷积编码器中的每一个强制为 $S_{N+3}=0$ 的最终状态。在对下行提供 $\bar{b}_{1,N+1}^{l,a}$ 、 $\bar{b}_{1,N+2}^{l,a}$ 、 $\bar{b}_{1,N+3}^{l,a}$ 、 $\bar{b}_{2,N+1}^{l,a}$ 、 $\bar{b}_{2,N+2}^{l,a}$ 、 $\bar{b}_{2,N+3}^{l,a}$ 的同时,解调器将对应LLR $\bar{b}_{1,N+1}^{u,a}$ 、 $\bar{b}_{1,N+2}^{u,a}$ 、 $\bar{b}_{1,N+3}^{u,a}$ 、 $\bar{b}_{2,N+1}^{u,a}$ 、 $\bar{b}_{2,N+2}^{u,a}$ 、 $\bar{b}_{2,N+3}^{u,a}$ 提供给上行。如图13所示,这些LLR可以提供给位于所提出的完全并行turbo解码器的每个行的结束处的三个附加算法块。

[0123] 图13提供示出用于受turbo编码的帧终止在已知的状态下的帧的终止比特中所涉及的处理元件的示意性框图。如图13所示,turbo解码器内部交织器704的输出与上turbo解码器701和下turbo解码器702交换数据比特,如参照图8所解释的那样。然而,如图13所示,对于上turbo解码器701示出三个终止处理元件作为处理元件710、712、714。下turbo解码器702包括三个终止单元720、722、724,其处理用于帧的终止比特,如以下解释的那样。

[0124] 在每个行的结束处的三个附加算法块在迭代解码处理内无需迭代地操作。反之,可以使用后向递归在迭代解码处理开始之前仅操作它们一次。更具体地说,具有指数 $k=N+$

3的算法块可以采用公式(2)和(4),以处理L=2个LLR $\bar{b}_{1,N+3}^a$ 、 $\bar{b}_{2,N+3}^a$ 。在此,由于确保 $S_{N+3}=0$ 的最终状态,因此应采用状态度量 $\bar{\beta}_{N+3}=[0,-\infty,-\infty,\dots,-\infty]$ 。所得状态度量 $\bar{\beta}_{N+2}$ 可以然后提供给具有指数k=N+2的算法块。进而,该操作使用相同处理,以获得 $\bar{\beta}_{N+1}$,其然后提供给块,其中,k=N+1,以通过相同方式获得 $\bar{\beta}_N$ 。于是可以贯穿迭代解码处理采用 $\bar{\beta}_N$ 的所得值,而没有对再次操作三个附加算法块的任何需要。注意,由于公式(2)和(4)揭示 $\bar{\beta}_N$ 的值独立于随着迭代解码处理进行而更新的所有值,因此不存在与采用该方法关联的罚。

[0125] 注意,由于LTE turbo编码器不输出下卷积编码器所产生的系统性帧 $\bar{b}_3^{l,a}$,因此下行中的第k算法块使用(2)以仅考虑L=2个先验LLR $\bar{b}_{1,k}^{l,a}$ 和 $\bar{b}_{2,k}^{l,a}$ 。通过对比,具有指数k $\in [1,N]$ 的上行中的算法块考虑L=3个先验LLR $\bar{b}_{1,k}^{u,a}$ 、 $\bar{b}_{2,k}^{u,a}$ 和 $\bar{b}_{3,k}^{u,a}$ 。图10中关于具有指数k=N的算法块示出该情况。

[0126] WiMAX turbo解码器

[0127] 如对于被配置为根据LTE turbo码对受turbo编码的数据进行解码的turbo解码器的示例那样,根据本技术实施例被配置为对通过WiMAX turbo码所编码的数据帧进行解码的turbo解码器也采用奇数-偶数交织器[14],允许其受益于完全并行turbo解码器的计算复杂度方面的50%减少,如图10所示。此外,两个消息帧的同时解码受支持,促进总体处理吞吐量方面的100%增加,如图11所示。替代地,可以使用图12的方法来实现硬件方面的50%减少。采用本技术的完全并行turbo解码器应用公式(7)至(10),其适用于支持WiMAX turbo码的双二进制性质。在此,上行中的算法块考虑L=6个先验LLR,而由于下卷积码所产生的系统性帧 b_5^l 和 b_6^l 并非输出,因此下行中的算法块考虑L=4个LLR。更具体地说,对上解码器701中的第k算法块提供六个先验LLR $\bar{b}_{1,k}^{u,a}$ 、 $\bar{b}_{2,k}^{u,a}$ 、 $\bar{b}_{3,k}^{u,a}$ 、 $\bar{b}_{4,k}^{u,a}$ 、 $\bar{b}_{5,k}^{u,a}$ 、 $\bar{b}_{6,k}^{u,a}$,使用它们以生成两个外在LLR $\bar{b}_{1,k}^{u,e}$ 、 $\bar{b}_{2,k}^{u,e}$ 。通过对比, $\bar{b}_{1,k}^{l,a}$ 、 $\bar{b}_{2,k}^{l,a}$ 、 $\bar{b}_{3,k}^{l,a}$ 、 $\bar{b}_{4,k}^{l,a}$ 提供给下行中的k算法块,其响应于此而生成两个外在LLR $\bar{b}_{1,k}^{l,e}$ 和 $\bar{b}_{2,k}^{l,e}$ 。可以通过在第一迭代中采用 $\bar{\alpha}_0=[0,0,0,\dots,-\infty]$ 和 $\bar{\beta}_N=[0,0,0,\dots,0]$ 来实现咬尾。在所有随后迭代中,可以分别对于 $\bar{\alpha}_0$ 和 $\bar{\beta}_N$ 采用最新近获得的 $\bar{\alpha}_N$ 和 $\bar{\beta}_0$ 的值。

[0128] 完全并行turbo解码器与Log-BCJR解码器的比较

[0129] 以下段落提供比较提供完全并行turbo解码器的本技术实施例与采用Log-BCJR turbo解码器的传统turbo解码器以及[10]中公开的另一已知turbo解码算法的分析。对于这些turbo解码器中的每一个,标识每解码迭代所需的时间段的数量,所分析的每解码迭代的存储器要求以及计算复杂度、实现特定纠错性能所需的时间段持续期和解码迭代的数量分别得以标识,以示出本技术的一些优点。此外,组合这些特性,以量化当对于LTE turbo解码和WiMAX turbo解码二者采用时的这些turbo解码器的总体吞吐量、时延和计算复杂度。在图14中所提供的表1中总结比较,其提供当使用公式(12)的近似max*运算,因此图7的上解码器和下解码器二者可以重复使用它们。如以上表所示,当分别实现LTE turbo解码器和

WiMAX turbo解码器时,Log-BCJR算法所需的存储器的量是9.33TtotP和8TtotP。

[0130] 还应注意,现有技术LTE turbo解码器[10]采用Radix-4变换,其使得必须存储的外在状态度量的数量减半。此外,现有技术LTE turbo解码器使用重复计算技术[10],以进一步减少存储器要求。并非在前向递归期间存储先验转变度量,从而它们可以在后向递归期间重复使用,重复计算技术仅在后向递归期间重复计算这些度量。除此而外,现有技术LTE turbo解码器还仅在前向递归期间存储外在状态度量的1/6,并且在后向递归期间重复计算这些度量的其它5/6。归因于其采用这些技术,现有技术LTE turbo解码器所需的存储器的量是1.67TtotP。然而,如将简短解释的那样,当关于LTE turbo解码采用时,存储先验奇偶LLR $\bar{b}_2^{u,a}$ 和先验系统性LLR $\bar{b}_3^{u,a}$ 之和有益于所提出的完全并行算法的上行中的算法块。这样使得现有技术LTE turbo解码器所需的存储器的量等于1.25TtotP,如图14的表1所示。

[0131] 类似地,当现有技术算法应用于WiMAX turbo解码时,所需的存储器也是1.67TtotP,如图14的表1所示。注意,即使WiMAX turbo解码器并非受益于Radix-4变换,也保持该比率,这样使得所需的算法块的数量以及必须存储的外在状态度量的数量减半。这是因为,除了需要多如用于外在状态度量的存储的两倍之外,WiMAX turbo码还需要多如用于LLR的存储的两倍,因为其为双二进制的。

[0132] 计算复杂度

[0133] 对于LTE turbo解码器和WiMAX turbo解码器二者,采用本技术和Log-BCJR算法的turbo解码器的每个处理元件当采用的加法、减法和max*运算的数量在图15中提供的表2中得以量化。

[0134] 已经采用多种技术以使得图15的表2中列出的运算的数量最小化。例如,一些特定转变的先验度量 $\bar{\gamma}_k(S_{k-1}, S_k)$ 彼此相等,允许它们被计算一次并且然后重复使用。此外,一些先验度量 $\bar{\gamma}_k(S_{k-1}, S_k)$ 是零值的,并且因此,无需将它们加入对应、 $\bar{\delta}_k(S_{k-1}, S_k)$ 、 $\bar{\alpha}_k(S_k)$ 或 $\bar{\beta}_{k-1}(S_{k-1})$ 计算。最后,当计算WiMAX turbo解码器中的外在LLR $\bar{b}_{1,k}^e$ 时,一些max*运算的结果可以重复使用,以计算外在LLR $\bar{b}_{2,k}^e$ 。注意,LTE turbo解码器和WiMAX turbo解码器的上行中的算法块考虑比下行的算法块更高数量的先验LLR,产生显著更高的复杂度。因此,表2呈现上行和下行中的算法块所采用的平均运算数量,产生一些非整数值。

[0135] 对于LTE turbo解码器和WiMAX turbo解码器二者,采用本技术的完全并行turbo解码器需要比Log-BCJR算法更少的加法和减法以及相等数量的max*运算。当采用(12)的近似时,max*运算可以看作具有与加法和减法相似的计算复杂度[15]。如图14的表1所示,当分别实现LTE turbo解码器和WiMAX turbo解码器时,Log-BCJR算法的每解码迭代的计算复杂度C因此是1.1TtotP和1.25TtotP。

[0136] 注意,现有技术LTE turbo解码器[10]采用Radix-4变换以及(12)的近似。当采用Radix-4变换时,Log-BCJRLTE turbo解码器具有用于Log-BCJR WiMAX turbo解码器的与表1中呈现的相同的每算法块的复杂度。然而,应注意,如以上那样,Radix-4变换使得所需的算法块的数量减半。此外,如上所述,现有技术LTE turbo解码器在后向递归期间重复计算(2)的先验转变度量和(3)的外在状态度量。因此,现有技术LTE turbo解码器具有1.77TtotP的每解码迭代的复杂度C,如表1所示。当将现有技术算法的重复计算技术应用于

WiMAX turbo码时,其每解码迭代的复杂度C对应于 $1.58T_{totP}$,如图14的表1所示。

[0137] 时间段持续期

[0138] 根据本技术的turbo解码器可以被布置为这样的:可以在单个时间段内完成图8至图13中的算法块中的每一个。然而,每个时间段或时钟周期时段所需的时间量D取决于算法块的计算要求。更具体地说,所需的持续期D取决于算法块的计算要求所施加的通过数据依赖性的关键路径。例如,在所提出的完全并行算法中,公式(8)、(9)和(10)彼此独立,但它们都取决于(7)。因此,必须首先完成(7)的计算,但然后可以并行地计算(8)、(9)和(10)。在这三个公式中,由于(10)是比(8)和(9)更多变量的函数,因此其需要用于计算的最多时间。因此,所提出的完全并行算法中的算法块的关键路径取决于(7)和(10)的计算要求。

[0139] 采用公式(7)以获得用于状态转变示图中的每个转变的后验度量 $\bar{\delta}_k(S_{k-1}, S_k)$ 。然而,例如,在考虑 $L=3$ 个先验LLR的上turbo解码器701中的算法块的情况下,可以使用五个变量的加法来并行地计算它们全部。通过对比,在 $L=2$ 的下turbo解码器702中的算法块的情况下,需要仅四个变量的加法。 v 数量的变量的求和需要 $v-1$ 次加法,其中的一些可以并行地执行。更具体地说,变量可以成对地相加在一起,并且然后在第二步骤中,所得求和可以成对地相加在一起。该处理可以继续,直到单个求和剩下,需要总共 $\log_2(v)$ 个步骤。相应地,公式(7)将三次加法贡献于所提出的完全并行LTE turbo解码器的上行中的算法块的关键路径,并且两次加法用于下行中的块。由于这两个关键路径贡献的最大值对时间段持续期施加最大限制,因此在表2的对应大括号中呈现它。可以采用相似的分析以确定表2的大括号中所提供的其它关键路径贡献中的每一个。

[0140] 如图15的表2所示,Log-BCJR算法的关键路径比所提出的完全并行算法的关键路径更长,需要具有更长持续期D的时间段并且产生更慢的运算。当采用(12)的近似时,可以考虑max*运算的关键路径,以使得对关键路径的贡献与加法和减法相似。然而,注意,可以通过采用与管线相似的技术从该总数消除一次加法的贡献。更具体地说,可以在迭代解码开始之前计算先验奇偶LLR $\bar{b}_2^{u,a}$ 和先验系统性LLR $\bar{b}_3^{u,a}$ 之和。结果可以被存储并且由所提出的完全并行LTE turbo解码器的上行中的算法块贯穿迭代解码处理使用。这样将上行中的公式(2)的关键路径贡献减少为两次加法,其与下行的关键路径贡献相等。因此,现有技术LTE turbo解码器的关键路径和时间段持续期D可以看作 $0.43T_{totP}$,如表1所示。相似地,当将现有技术算法应用于WiMAX turbo解码时,结果是一次加法和两次max*运算的相同关键路径。如表1所示,该关键路径是 $0.33T_{totP}$,其需要五次加法和四次max*运算。

[0141] 纠错性能

[0142] 采用本技术的完全并行turbo解码器的解码性能现将与采用Log-BCJR算法的turbo解码器的解码性能进行比较。关于完全并行turbo解码器的性能与Log-BCJR算法的性能比较,图16a、图16b、图16c、图17a、图17b和图17c提供提出每比特的信噪比(SNR) E_b/N_0 的图线的仿真结果,其中,在此情况下, $E_b/N_0 = \text{SNR} [\text{dB}] - 10\log_{10}(R)$,如将在以下段落中解释的那样。

[0143] 图16a、图16b、图16c提供用于当使用公式(11)的精确max*运算符以对包括(a) $N=4800$ 比特,(b) $N=480$ 比特以及(c) $N=48$ 比特的帧进行解码时的LTE turbo解码器的纠错性能的图线。关于生成图16a、图16b和图16c的仿真结果,对于通过不相关的窄带瑞利衰落信

道的传输采用BPSK调制。关于迭代的数量I处于范围1、2、4、8、16、32、64、128中的情况提供图线。使用所提出的完全并行算法并且对于1、2、4、8、16解码迭代中的迭代使用传统BCJR算法来执行解码迭代。

[0144] 图16a、图16b和图16c提供完全并行turbo解码器与BCJR算法之间的比较。这些结果示出,无论帧长度N如何,所提出的完全并行算法都可以收敛到与Log-BCJR算法相同的误码性能。然而,实施本技术的完全并行turbo解码器可见收敛得相对缓慢,需要比Log-BCJR算法显著更多的解码迭代I。注意,这并非是不期望的,因为采用并行调度的LDPC解码器已知需要比采用串行调度更多的解码迭代[16]。

[0145] 图17a、图17b和图17c提供图17a当使用公式(11)的精确max*运算符时的WiMAX turbo解码器、图17b当使用公式(12)的近似max*运算符时WiMAX turbo解码器以及图17c当使用近似max*运算符时LTE turbo解码器的纠错性能。对于不相关的窄带瑞利衰落信道上的传输关于通过BPSK调制的数据码元的传输生成示例仿真结果。对于迭代的数量是使用所提出的完全并行算法的I=32或I=48个解码操作以及使用传统Log-BCJR算法的I=8个解码迭代的情况提供误比特率针对信噪比的图线。对于LTE turbo码采用48、480、4800的帧长度,而对于WiMAX turbo码使用24、240、2400的帧长度N。

[0146] 图16a、图16b和图16c中提供的结果说明,对于使用(11)的精确max*运算符的LTE turbo解码情况,Log-BCJR算法实现特定BER所需的解码迭代的数量I一致地处于 $1/7T_{totP}$ 左右。如图17a、图17b、图17c中呈现的结果所示,当采用(12)的近似max*运算符时,该数量改变为 $1/6T_{totP}$,如图17c和表1所示。更具体地说,图17c示出,无论帧长度N ∈ {48, 480, 4800}如何,采用I=8个解码迭代的Log-BCJR算法实现与采用I=48个迭代的所提出的完全并行算法相同的误比特率。在WiMAX turbo码的情况下,图17a和图17b揭示,Log-BCJR算法所需的解码迭代的数量I是 $1/4T_{totP}$,而无论帧长度N如何以及是采用精确max*运算符还是近似max*运算符。注意,[10]的现有技术算法的纠错性能因其采用NSW技术而稍微降级,但这种降级可以看作是不显著的。因此,如表1所示,分别对于LTE turbo码和WiMAX turbo码,现有技术算法所需的解码迭代的数量I也可以看作 $1/6T_{totP}$ 和 $1/4T_{totP}$ 。

[0147] 总体特性

[0148] 时间段持续期D、每解码迭代的时间段的数量T和所需的解码迭代的数量I的乘积给出turbo解码器的时延 $D \times T \times I$ 。同时,处理吞吐量与时延 $D \times T \times I$ 成反比。对于LTE turbo解码和WiMAX turbo解码二者,表1相对于采用本技术的完全并行turbo编码器的时延和吞吐量而量化Log-BCJR算法和[10]的现有技术算法的时延和吞吐量。在采用N=6144比特的最长的所支持的帧长度的LTE turbo码的情况下,所提出的完全并行算法的时延和吞吐量比Log-BCJR算法的时延和吞吐量的更优三个数量级。此外,当与[10]的现有技术算法比较时,所提出的完全并行算法具有6.88倍更优的时延和吞吐量。然而,注意,如果减少帧长度N,则所提出的完全并行算法所提供的优点减少。在现有技术算法的NSW技术的所考虑的参数化所支持的最短帧长度N=2048的情况下,所提出的完全并行算法的优越性减少到2.29倍。当将现有技术算法应用于具有范围N ∈ [1440, 2400]中的长度的帧的WiMAX turbo解码时,根据本技术的所提出的完全并行turbo解码器范围从3.71倍上至6.19倍。与用于WiMAX turbo解码的Log-BCJR算法比较,当采用N=2400的最大帧长度时,根据本技术的完全并行turbo解码器更优三个数量级。

[0149] 当对包括N=6144比特的帧进行解码时,[10]的现有技术LTE turbo解码器实现2.15G比特/秒的处理吞吐量和2.85μs的时延。使用对应于2.22ns的时间段持续期的450MHz的时钟频率来实现该情况。表1的结果说明,根据本技术的完全并行turbo解码器可以使用194MHz的时钟频率实现14.8G比特/秒的处理吞吐量和0.42μs的时延。此外,可以假设[10]的现有技术turbo解码器当应用于WiMAX解码时可以保持2.15比特/秒的处理吞吐量。如果这样,则这说明,当对具有N=2400比特的长度的帧进行解码时,所提出的完全并行算法可以实现13.3G比特/秒的处理吞吐量和0.36μs的时延。注意,这些多吉比特吞吐量可与完全并行LDPC解码器[9]所提供的吞吐量比较。

[0150] 虽然根据本技术的完全并行turbo解码器提供对处理吞吐量和时延的显著改进,但以需要增加的并行性和计算复杂度的代价来实现该优点。总体计算复杂度C × I给出为每解码迭代的计算复杂度C和所需的解码迭代的数量I的乘积。对于LTE turbo解码和WiMAX turbo解码,表1相对于所提出的完全并行算法的总体计算复杂度量化Log-BCJR算法和[10]的现有技术算法的总体计算复杂度。如表1所示,所提出的完全并行算法的计算复杂度可以比Log-BCJR算法的计算复杂度更高多于五倍。然而,与[10]的现有技术算法比较,所提出的完全并行算法具有大约三倍更高的计算复杂度。

[0151] 管线式完全并行turbo解码器

[0152] 本技术另一示例实施例提供进一步增强,其中,采用修改后的turbo解码器,以允许管线式元件进行平行处理操作。将参照图18、图19、图20、图21和图22解释本技术实施例。

[0153] 如上所述,在LTE完全并行turbo解码器的操作期间的第t时间段中,图8的上行中的第k处理元件可以采用公式(7)至(10),以组合具有指数t-1的前一时间段中由相连的处理元件生成的L=3个先验LLR $\bar{b}_{1,k}^{t-1,l,e}$ 、 $\bar{b}_{2,k}^{t-1,l,e}$ 、 $\bar{b}_{3,k}^{t-1,l,e}$ 以及M=8个先验前向状态度量 $\bar{\alpha}_{k-1}^{t-1,u} = [\bar{\alpha}_{k-1}^{t-1,u}(S_{k-1})]_{S_{k-1}=0}^{M-1}$ 以及M=8个先验后向状态度量 $\bar{\beta}_k^{t-1,u} = [\bar{\beta}_k^{t-1,u}(S_k)]_{S_k=0}^{M-1}$ 。这样产生MK=16个后验转变度量的集合,其中,每个 $\bar{\delta}_k^t(S_{k-1}, S_k)$ 对应于从M=8个状态之一显现的K=2个转变中的特定一个。此后,组合后验转变度量,以获得M=8个外在前向状态度量 $\bar{\alpha}_k^t$ 以及M=8个先验后向状态度量 $\bar{\beta}_k^t$ 以及外在LLR $\bar{b}_{1,k}^{t-1,l,a}$,其受交织以获得先验LLR $\bar{b}_{1,\pi^{-1}(k)}^{t-1,l,a}$ 。同时,在第t时间段中,图8的下行中的第k处理元件可以采用公式(7)至(10),以处理L=2个先验LLR $\bar{b}_{1,k}^{t-1,l,a}$ 、 $\bar{b}_{2,k}^{t-1,l,a}$ 以及M=8个先验前向状态度量 $\bar{\alpha}_{k-1}^{t-1,l}$ 以及M=8个先验后向状态度量 $\bar{\beta}_k^{t-1,l}$ 。这样产生M=8个的外在前向状态度量 $\bar{\alpha}_k^{t-1,l}$ 以及M=8个先验后向状态度量 $\bar{\beta}_k^{t-1,l}$ 以及外在LLR $\bar{b}_{1,k}^{t-1,l,e}$,其受交织以获得先验LLR $\bar{b}_{1,\pi^{-1}(k)}^{t-1,l,e}$ 。相连的处理元件可以然后在具有指数t+1的下一时间段中采用第t时间段中所生成的LLR和状态度量。

[0154] 等效地,可以使用图18和公式(15)至(18)实现LTE完全并行turbo解码器,以受益于减少的复杂度C和时间段持续期D。在此情况下,图18的下行中的每个处理元件处理L=3个先验LLR $\bar{b}_{1,k}^{t-1,l,a}$ 、 $\bar{b}_{2,k}^{t-1,l,a}$ 、 $\bar{b}_{3,k}^{t-1,l,a}$,而非如上所述的L=2。在此,交织器根据 $\bar{b}_{3,\pi(k)}^{t-1,l,a}$ 提供 $\bar{b}_{3,k}^{t-1,l,a}$ 。注意,公式(15)至(18)在先验转变度量 $\bar{\gamma}_k^t(S_{k-1}, S_k)$ 而非公式(7)至(10)的后验转变度

量 $\bar{\delta}_k^t(S_{k-1}, S_k)$ 的基础上进行操作。还注意，公式(16)和(17)中的 $\bar{\alpha}_k^t(0)$ 和 $\bar{\beta}_{k-1}^t(0)$ 的减法表示归一化，这样避免在定点实现方式中外在状态度量值不受束缚地生长并且上溢。

$$[0155] \quad \bar{y}_k^t(S_{k-1}, S_k) = \sum_{j=1}^3 [b_j(S_{k-1}, S_k) \cdot \bar{b}_{j,k}^{t-1,a}] \quad (15)$$

$$[0156] \quad \bar{\alpha}_k^t(S_k) = [\max^*_{\{(S_{k-1}, S_k) \in \mathcal{C}(S_{k-1}, S_k) = 1\}} \{ \bar{y}_k^t(S_{k-1}, S_k) + \bar{\alpha}_{k-1}^{t-1}(S_{k-1}) \}] - \bar{\alpha}_k^t(0) \quad (16)$$

$$[0157] \quad \bar{\beta}_{k-1}^t(S_{k-1}) = [\max^*_{\{(S_k \in \mathcal{C}(S_{k-1}, S_k)) = 1\}} \{ \bar{y}_k^t(S_{k-1}, S_k) + \bar{\beta}_k^{t-1}(S_k) \}] - \bar{\beta}_{k-1}^t(0) \quad (17)$$

$$[0158] \quad \bar{b}_{j,k}^{t,a} = [\max^*_{\{(S_{k-1}, S_k) \in \mathcal{D}_1(S_{k-1}, S_k) = 3\}} \{ b_2(S_{k-1}, S_k) \cdot \bar{b}_{2,k}^{t-1,a} + \bar{\alpha}_{k-1}^{t-1}(S_{k-1}) + \bar{\beta}_k^{t-1}(S_k) \}] - \\ [\max^*_{\{(S_{k-1}, S_k) \in \mathcal{D}_1(S_{k-1}, S_k) = 0\}} \{ [b_2(S_{k-1}, S_k) \cdot \bar{b}_{2,k}^{t-1,a} + \bar{\alpha}_{k-1}^{t-1}(S_{k-1}) + \bar{\beta}_k^{t-1}(S_k)] \}] \quad (18)$$

[0159] 如图8的完全并行turbo解码器中那样，图18的交替设计可以在所有时间段中激活所有处理元件。替代地，图18的交替设计可以利用LTE turbo码的奇数-偶数性质，以改进该复杂度、吞吐量或硬件资源要求，如图8的完全并行turbo解码器中那样。与图10类似，当通过使用交替时间段以在(a)仅激活图18中的浅阴影的处理元件与(b)仅激活深阴影的处理元件之间交替来对单个帧进行解码时，可以改进复杂度。替代地，可以通过同时对两个帧进行解码来改进吞吐量。与图11类似，通过使用交替时间段以分别使得第一帧和第二帧交替以及(b)使用它们以分别对第二帧和第一帧进行解码来实现该情况，最后，可以通过仅使用处理元件的上行以对单个帧进行解码来改进硬件资源要求。与图12类似，通过使用交替时间段以在(a)使用图18中的浅阴影的处理元件和深阴影的处理元件以分别执行上行和下行中的对应处理元件的操作与(b)使用它们以分别执行下行和上行中的对应处理元件的操作之间交替来实现该情况。

[0160] 在图18的交替式设计的处理元件中，可以使用图19的数据路径来实现公式(15)至(18)。在此，公式(15)的实现方式利用这样的观测：所有MK=16个先验转变度量 $\bar{y}_k^t(S_{k-1}, S_k)$ 采用四个值（即0、 $\bar{b}_{2,k}^{t-1,a}$ 、 $\bar{b}_{1,k}^{t-1,a} + \bar{b}_{2,k}^{t-1,a}$ 、 $\bar{b}_{3,k}^{t-1,a}$ 或 $\bar{b}_{1,k}^{t-1,a} + \bar{b}_{3,k}^{t-1,a}$ ）之一。类似地，公式(18)的实现方式层这样的观测： $\max^*(a+c, b+c) = \max^*(a, b) + c$ ，以减少所需的加法的数量。此外，由于归一化确保 $\bar{\alpha}_k^t(0) = 0$ 以及 $\bar{\beta}_{k-1}^t(0) = 0$ ，因此可以省略虚线轮廓所示的加法。出于减缓当采用近似 $\max^*(a, b) \approx \max(a, b)$ 时产生的纠错性能降级的缘故，包括比例化，其中，可以对于使用比特移位以及单次加法的定点算术实现 $f_2 = 0.75$ 的比例化。最后，例如，钳位用于保持定点算术的比特宽度，其中，可以采用 $w_1 = 4$ 和 $w_2 = 6$ 。注意，图19的关键路径包括数据路径级，产生 $D = 6$ 的时间段持续期。该关键路径由公式(18)并且由公式(15)于是公式(16)和(17)的串行操作施加，其中，后三个公式均单独包括仅三个数据路径级。

[0161] 这样激发图20的管线式数据路径，其具有仅包括三个数据路径级的关键路径，产生 $D = 3$ 的时间段持续期。可以通过公式(19)至(23)描述该数据路径的操作。注意，公式(20)和(21)是前一时间段中所计算的管线式先验转变度量 $\bar{y}_k^{t-1}(S_{k-1}, S_k)$ 而非公式(16)和(17)所采用的当前时间段中所计算的先验转变度量 $\bar{y}_k^t(S_{k-1}, S_k)$ 的函数。此外，公式(22)和(23)将公式(18)划分为两个管线级，其中，外在LLR $\bar{b}_{1,k}^{t,a}$ 变为前一时间段中获得的中间计算结果 $\bar{e}_{k,l,m}^{t-1}$ 的函数。

$$[0162] \quad \bar{y}_k^t(S_{k-1}, S_k) = \sum_{j=1}^3 [b_j(S_{k-1}, S_k) \cdot \bar{b}_{j,k}^{t-1,a}] \quad (19)$$

$$[0163] \quad \alpha_k^t(S_k) = \left\{ \max_{(S_{k-1}) \in (S_{k-1}, S_k) = 1} \left[\tilde{\gamma}_k^{t-1}(S_{k-1}, S_k) + \alpha_{k-1}^{t-1}(S_{k-1}) \right] \right\} - \alpha_k^t(0) \quad (20)$$

$$[0164] \quad \beta_k^t(S_{k-1}) = \left\{ \max_{(S_{k-1}) \in (S_{k-1}, S_k) = 1} \left[\tilde{\gamma}_k^{t-1}(S_{k-1}, S_k) + \beta_{k-1}^{t-1}(S_k) \right] \right\} - \beta_{k-1}^t(0) \quad (21)$$

$$[0165] \quad \tilde{\epsilon}_{k,l,m}^t = \max_{(S_{k-1}, S_k) \in b_1(S_{k-1}, S_k) = l, b_2(S_{k-1}, S_k) = m} \left[\alpha_{k-1}^{t-1}(S_{k-1}) + \beta_k^{t-1}(S_k) \right] \quad (22)$$

$$[0166] \quad \tilde{\delta}_{k,l,m}^t = \left[\max_{m \in \{0,1\}} \left\{ m \cdot \tilde{\beta}_{2,k}^{t-1,a} + \tilde{\epsilon}_{k,l,m}^t \right\} \right] - \left[\max_{m \in \{0,1\}} \left\{ m \cdot \tilde{\beta}_{2,k}^{t-1,a} + \tilde{\epsilon}_{k,0,m}^t \right\} \right] \quad (23)$$

[0167] 注意,图20的管线式数据路径使得完全并行turbo解码器以不同的速率将状态度量处理为LLR。更具体地说,虽然外在状态度量 $\alpha_k^t(S_k)$ 和 $\beta_{k-1}^t(S_{k-1})$ 受前一时间段中所提供的先验状态度量 $\alpha_{k-1}^{t-1}(S_{k-1})$ 和 $\beta_k^{t-1}(S_k)$ 作用,但它们不受前一时间段中所提供的先验LLR $\bar{b}_{1,k}^{t-1,a}$ 、 $\bar{b}_{2,k}^{t-1,a}$ 、 $\bar{b}_{3,k}^{t-1,a}$ 作用。反之,它们受之前的两个时间段所提供的先验LLR $\bar{b}_{1,k}^{t-2,a}$ 、 $\bar{b}_{2,k}^{t-2,a}$ 、 $\bar{b}_{3,k}^{t-2,a}$ 作用。类似地,外在LLR $\bar{b}_{1,k}^{t,a}$ 不受前一时间段中所提供的先验状态度量 $\alpha_{k-1}^{t-1}(S_{k-1})$ 和 $\beta_k^{t-1}(S_k)$ 作用。反之,其受之前的两个时间段所提供的先验状态度量 $\alpha_{k-1}^{t-2}(S_{k-1})$ 和 $\beta_k^{t-2}(S_k)$ 作用归因于此,管线式数据路径对于完全并行turbo解码器产生不同的纠错性能。图21比较当对于通过AWGN信道的通信采用BPSK调制时的管线式完全并行turbo解码器的BER性能与原始设计的BER性能。由于原始设计的时间段持续期D是管线式设计的两倍,因此通过比较原始设计的1个迭代与管线式设计的2I个迭代来获得关于解码时间的公平比较。如图21所示,当解码时间是固定的时,与原始设计比较,管线式设计提供改进的BER性能。等效地,当目标BER是固定的时,与原始设计比较,管线式设计可以认为提供改进的吞吐量和时延。

[0168] 如图18的完全并行turbo解码器中那样,管线式设计可以在所有时间段中激活所有处理元件。替代地,管线式设计可以利用LTE turbo码的奇数-偶数性质,以改进其复杂度、吞吐量或硬件资源要求,如图22所示,其中,图10上对应于此的部分带有相同标号。在此,每个处理元件706、708的顶部阴影区域1802和底部阴影区域1808分别实现公式(19)和(23),而中间阴影区域1801、1806实现公式(20)至(22)。与图10中类似,当通过使用交替时间段以在(a)仅激活图22中的处理元件的浅阴影的部分1801、1808与(b)仅激活深阴影的部分1802、1806之间交替来对单个帧进行解码时,可以改进管线式完全并行turbo解码器的复杂度。替代地,可以通过同时对两个帧进行解码来改进吞吐量。与图11类似,通过使用交替时间段以在(a)使用图22中的处理元件的浅阴影部分和深阴影部分分别对第一帧和第二帧进行解码与(b)使用它们以分别对第二帧和第一帧进行解码之间交替来实现该情况。最后,可以通过仅使用处理元件的上行以对单个帧进行解码来改进硬件资源要求。与图12类似,通过使用交替时间段以在(a)使用图22中的处理元件的浅阴影的部分和深阴影的部分以分别执行上行和下行中的对应处理元件的操作与(b)使用它们以分别执行下行和上行中的对应处理元件的操作之间交替来实现该情况。

[0169] 优点总结

[0170] 本技术实施例可以提供一种完全并行turbo解码器,其消除或至少减少已知技术的数据依赖性并且促进完全并行操作。归因于其显著增加的并行性,当对于标准化turbo码采用时,本技术实施例可以促进比已知技术更优的高达6.88倍的吞吐量和时延。在这些应用中,根据本技术的turbo解码器可以促进14.8G比特/秒的处理吞吐量以及如0.42μs那样小的时延,但当然,实际吞吐量和时延将取决于所使用的硬件技术。然而,以比传统技术更高大约三倍的计算复杂度的代价来实现该优点。

[0171] 如上所述,本技术实施例可以提供用于恢复或检测通常已经经历马尔科夫类型过程的数据的turbo解码或检测处理。于此,本技术实施例可以提供一种turbo检测器或解码器,其处理软判决值以生成外在值,其可以变为用于另一解码器/检测器的先验值。因此应理解,通过实施本技术可以操作的turbo检测器可以因此用于形成均衡器、同步检测器、信道估计器、多用户检测器、MIMO检测器或联合信源/信道解码器。

[0172] 虽然已经参照LTE和WiMAX描述了本技术,但应理解,它们仅为示例,并且根据本技术的turbo解码器可以用于任何形式的受turbo编码的数据,而不限于LTE或WiMAX。

[0173] 根据以上描述,本技术实施例可以提供一种用于从接收到的信号检测并且恢复数据码元帧的接收机,所述帧的所述数据码元在传输的处理期间已经受具有所述接收到的信号中的所述帧的所述数据码元依赖于可以表示为具有多个格栅级的格栅的一个或多个超前数据码元的效果的马尔科夫过程作用。所述接收机包括:第一检测处理器,被配置为:接收表示为用于所述帧的每个数据码元的软判决值的所述数据码元帧;以及至少一个检测处理器,其被配置为:与所述第一检测处理器协作,以组合地执行turbo检测处理,以生成所述数据码元帧的估计。所述第一检测处理器包括多个处理元件,所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联,并且所述处理元件中的每一个被配置为:接收对应于与所述格栅级关联的一个或多个数据码元的一个或多个软判决值。所述处理元件中的所述一个或多个中的每一个被配置为:在一个处理迭代中,从第一相邻处理元件接收表示先验前向状态度量的数据,从第二相邻处理元件接收表示先验后向状态度量的数据,并且从所述至少一个其它检测处理器接收表示关于对于与所述处理元件关联的所述格栅级正检测所述一个或多个数据码元的先验信息。每个处理元件组合与所述一个或多个数据码元有关的所述先验前向状态度量、所述先验后向状态度量以及所述先验信息,以对于与所述处理元件关联的所述格栅级确定与所述一个或多个数据码元对应的外在前向状态度量、外在后向度量以及外在数据信息。每个处理元件将所述外在前向状态度量提供给所述第二相邻处理元件,将所述外在后向状态度量提供给所述第一相邻处理元件,其关于所述下一迭代变为所述先验后向状态度量,并且将所述外在数据信息提供给所述至少一个其它检测处理器,其关于所述下一迭代变为与所述数据码元有关的所述先验信息。对于所述turbo检测处理的多个迭代中的每一个,所述第一检测处理器和所述至少一个其它检测处理器被配置为:关于表示所述格栅级的所述处理元件中的每一个,交换关于对于与所述处理元件关联的所述格栅级正检测所述一个或多个数据码元的所述先验信息以及所述处理所生成的与所述一个或多个数据码元对应的所述外在数据信息。如上所述,处理可以同时发生在解码器中的两个或更多个中,从而所述解码器中的每一个正生成中间结果以与所述解码器中的其它解码器交换。此外,可以使用各处理元件之间所交换的信息的定点表示法。

[0174] 根据一个示例实施例,提供一种检测电路,用于执行turbo检测处理,以从包括表示用于数据码元或比特的帧的每个数据码元的一个或多个软判决值的数据的接收到的信号恢复所述帧。所述帧的所述数据码元或比特在传输期间已经受具有所述接收到的信号中的所述帧的所述数据码元依赖于可以表示为具有多个格栅级的格栅的效果的马尔科夫过程作用。所述检测处理器包括多个处理元件。所述处理元件中的每一个与根据所述马尔科夫过程表示所述帧的所述数据码元的依赖性的所述格栅级之一关联,并且所述处理元件中

的每一个被配置为：接收对应于与所述格栅级关联的一个或多个数据码元的一个或多个软判决值。所述处理元件中的一个或多个中的每一个被配置为：在一个时钟周期中，从第一相邻处理元件接收表示先验前向状态度量的数据，从第二相邻处理元件接收表示先验后向状态度量的数据，并且接收表示对于与所述处理元件关联的所述格栅级正检测用于所述一个或多个数据码元的先验软判决值的数据。所述处理元件包括与所述一个或多个数据码元有关的所述先验前向状态度量、所述先验后向状态度量以及所述先验软判决值，以对于与所述处理元件关联的所述格栅级确定与所述一个或多个数据码元对应的外在前向状态度量、外在后向度量和外在软判决值，并且将所述外在前向状态度量传递到所述第二相邻处理元件，其对于下一时钟周期变为所述先验前向状态度量，将所述外在后向状态度量传递到所述第一相邻处理元件，其对于所述下一时钟周期变为所述先验后向状态度量，并且提供所述外在软判决值，其对于所述下一时钟周期变为与所述数据码元有关的所述先验软判决值。对于所述turbo检测处理的多个相继时钟周期中的一个或多个，所述检测电路的所述处理元件被配置为同时进行操作。

[0175] 例如，对于所述turbo检测处理的多个时钟周期中的每一个，所述检测电路被配置为：对于表示所述格栅级的所述处理元件中的每一个，处理关于对于与所述处理元件关联的所述格栅级正检测所述一个或多个数据码元的所述先验信息。在一些示例中，所述检测电路与另一检测电路协作而进行操作，并且与所述其它检测电路交换与所述处理元件所生成的所述一个或多个数据码元对应的所述外在软判决值。在一些示例中，在每个时钟周期之后交换所述外在软判决值。

[0176] 参考文献

- [0177] [1] ETSI TS36.212 v 10.8.0 (2013-06) LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and Channel Coding, V10.2.0 ed., 2011.
- [0178] [2] IEEE 802.16-2012 Standard for Local and Metropolitan Area Networks—Part 16: Air Interface for Broadband Wireless Access Systems, 2012.
- [0179] [3] C.Berrou, A.Glavieux, and P.Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes (1),” in Proc. IEEE Int. Conf. on Communications, vol.2, Geneva, Switzerland, May 1993, pp.1064-1070.
- [0180] [4] P.Robertson, E.Villebrun, and P.Hoehler, “A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain,” in Proc. IEEE Int. Conf. on Communications, vol.2, Seattle, WA, USA, June 1995, pp.1009-1013.
- [0181] [5] IEEE 802.11n-2009 Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) , 2009.
- [0182] [6] D.J.C.MacKay and R.M.Neal, “Near Shannon limit performance of low density parity check codes,” Electron. Lett., vol.32, no.18, pp.457-458, Aug.1996.
- [0183] [7] M.Fossorier, “Reduced complexity decoding of low-density parity check codes based on belief propagation,” IEEE Trans. Commun., vol.47, no.5, pp.673-680, May 1999.

- [0184] [8] 5G Radio Access. Ericsson White Paper, June 2013.
- [0185] [9] V.A.Chandraseddy and S.M.Aziz, "FPGA implementation of a LDPC decoder using a reduced complexity message passing algorithm," Journal of Networks, vol.6,no.1,pp.36–45, Jan.2011.
- [0186] [10] T.Ilnseher,F.Kienle,C.Weis, and N.Wehn, "A 2.15GBit/s turbo code decoder for LTE Advanced base station applications," in Proc.Int.Symp.on Turbo Codes and Iterative Information Processing, Gothenburg, Sweden, Aug.2012, pp.21–25.
- [0187] [11] L.Fanucci,P.Ciao, and G.Colavolpe, "VLSI design of a fully-parallel high-throughput decoder for turbo gallager codes," IEICE Trans.Fundamentals, vol.E89-A,no.7,pp.1976–1986, July 2006.
- [0188] [12] D.Vogrig,A.Gerosa,A.Neviani,A.Graell I Amat,G.Montorsi, and S.Benedetto, "A0.35- μ m CMOS analog turbo decoder for the 40-bit rate 1/3 UMTS channel code," IEEE J.Solid-State Circuits, vol.40,no.3,pp.753–762,2005.
- [0189] [13] Q.T.Dong,M.Arzel,C.J.Jego, and W.J.Gross, "Stochastic decoding of turbo codes." IEEE Trans.Signal Processing, vol.58,no.12,pp.6421–6425, Dec.2010.
- [0190] [14] A.Nimbalker,Y.Blankenship,B.Classon, and T.K.Blankenship, "ARP and QPP interleavers for LTE turbo coding," in Proc.IEEE Wireless Commun.Networking Conf., Las Vegas,NV,USA, Mar.2008, pp.1032–1037.
- [0191] [15] L.Li,R.G.Maunder,B.M.Al-Hashimi, and L.Hanzo, "A low-complexityturbo decoder architecture for energy-efficient wireless sensor networks," IEEE Trans.VLSI Syst., vol.21,no.1,pp.14–22, Jan.2013. [Online]. Available:<http://eprints.soton.ac.uk/271820/>
- [0192] [16] P.Radosavljevic,A.de Baynast, and J.R.Cavallaro, "Optimized message passing schedules for LDPC decoding," in Asilomar Conf.Signals Systems and Computers,no.1, Pacific Grove,CA,USA, Oct.2005, pp.591–595.
- [0193] [17] CN 102611464
- [0194] [18] CN 102723958
- [0195] [19] WO 2011/082509
- [0196] [20] "A 122Mb/s Turbo decoder using a mid-range GPU" by Xianjun J., et al, published at Wireless Communications and Mobile Computing Conference (IWCMC), 20139th International, pages 1090–1094, 1–5 July 2013.

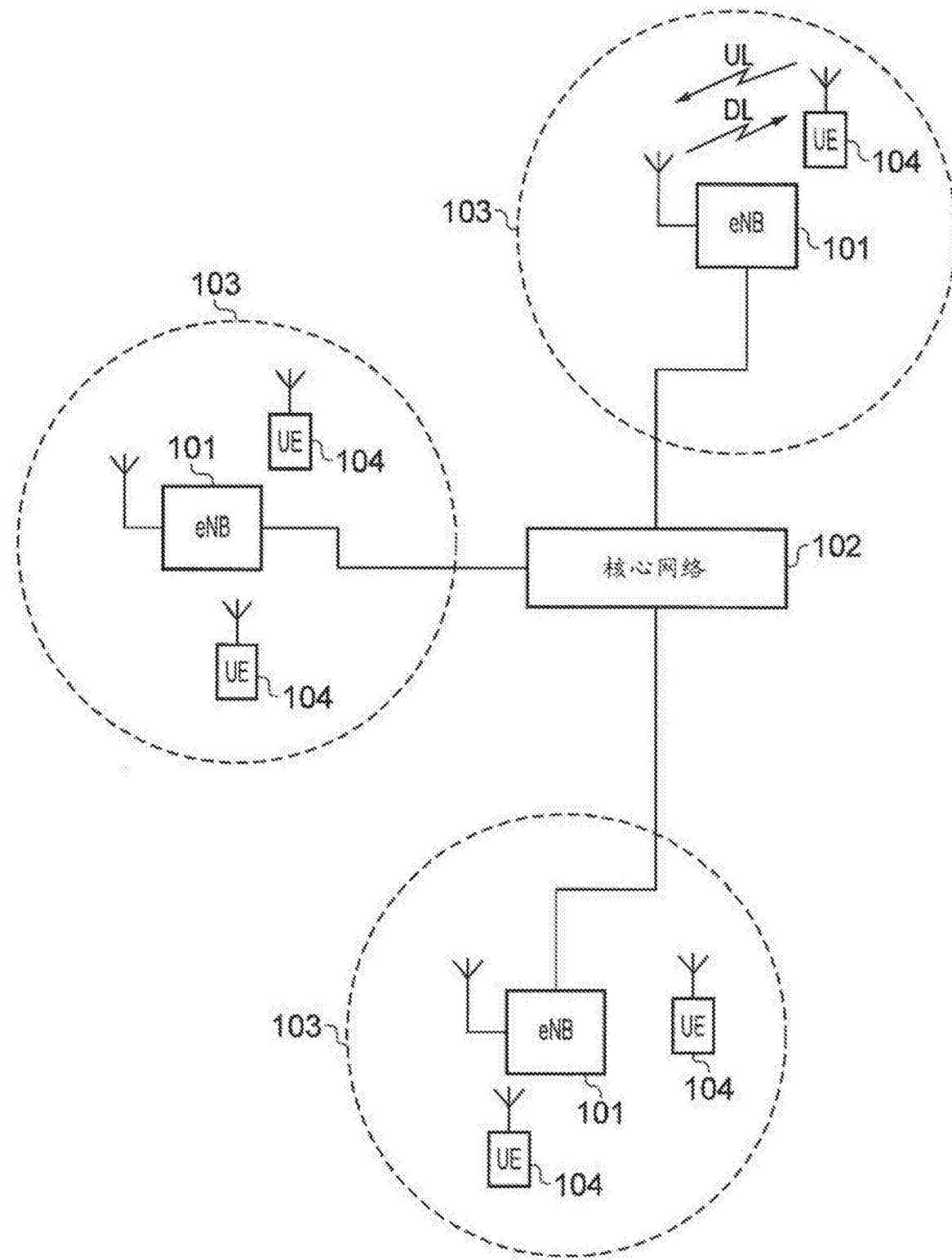


图1

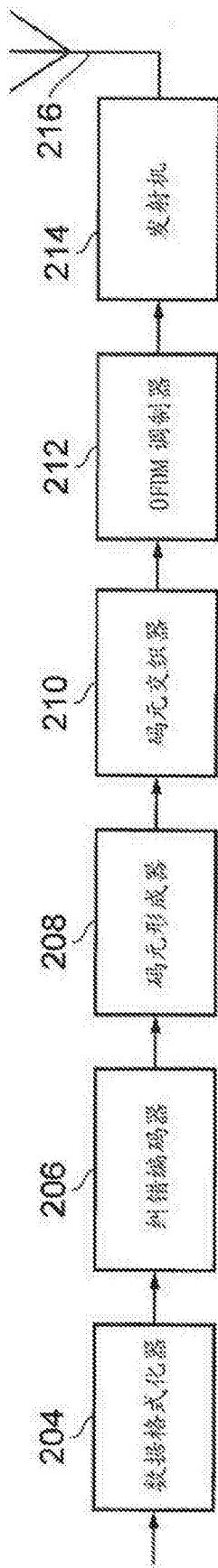


图2

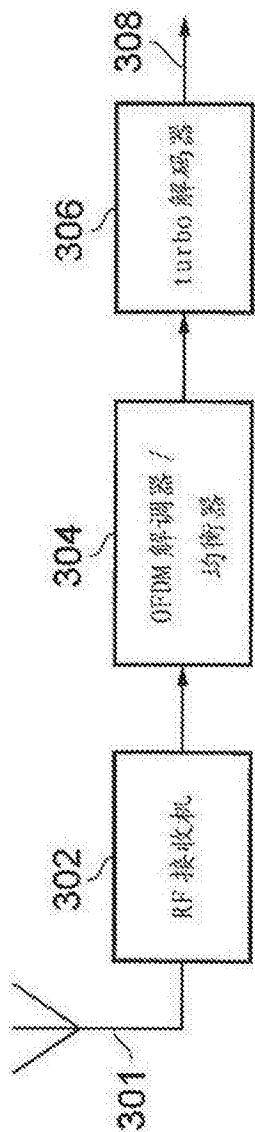


图3

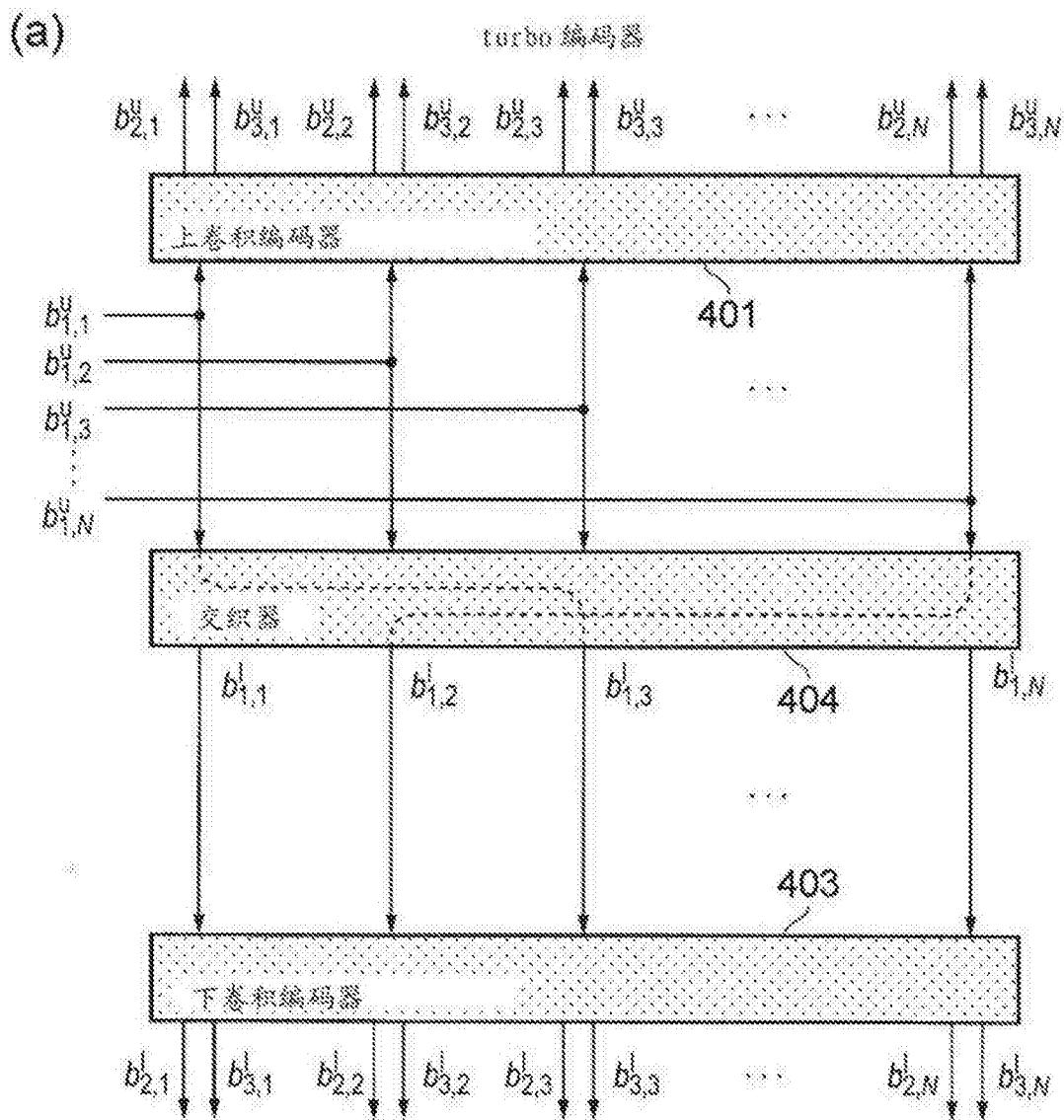


图4

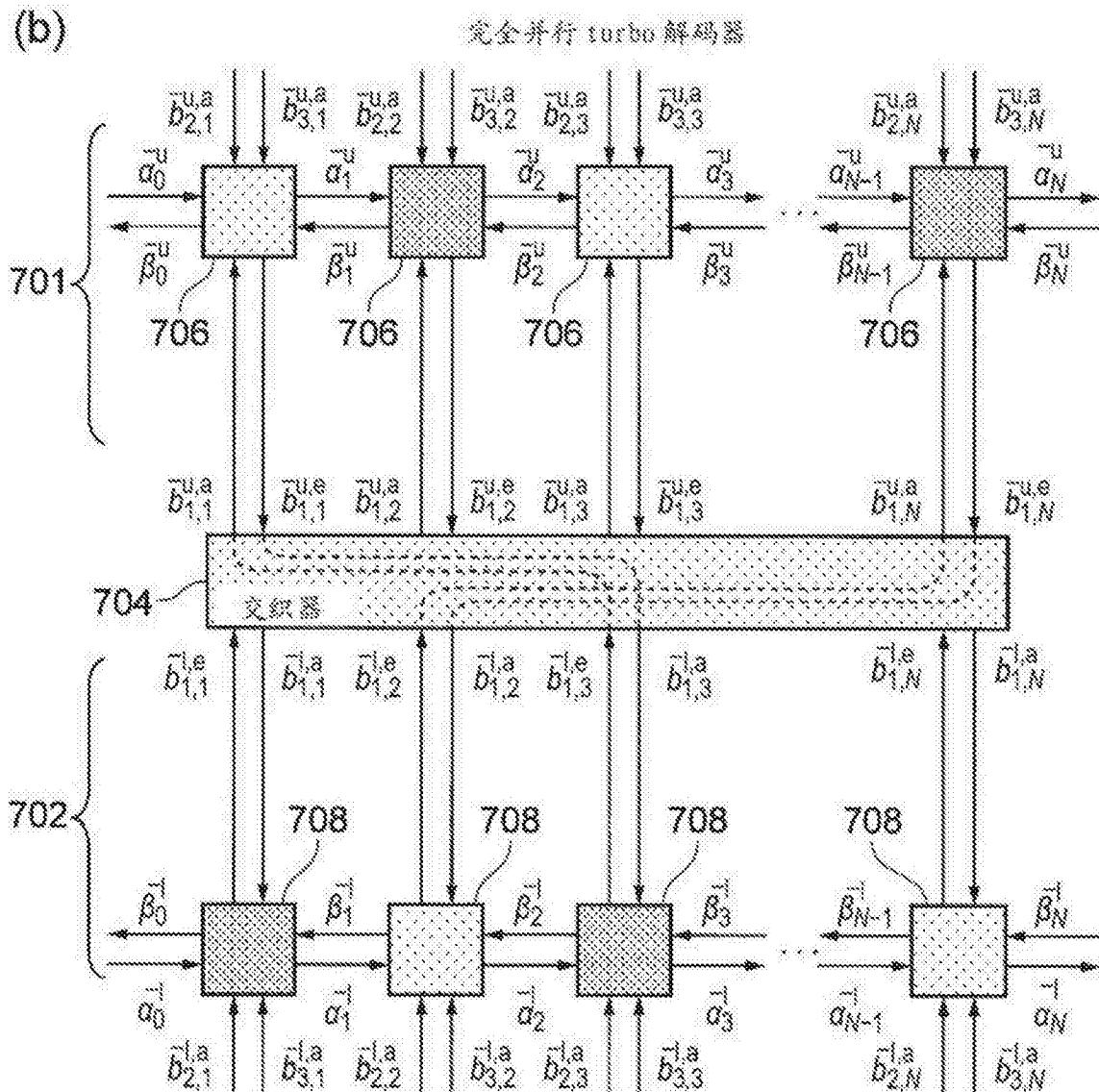


图8

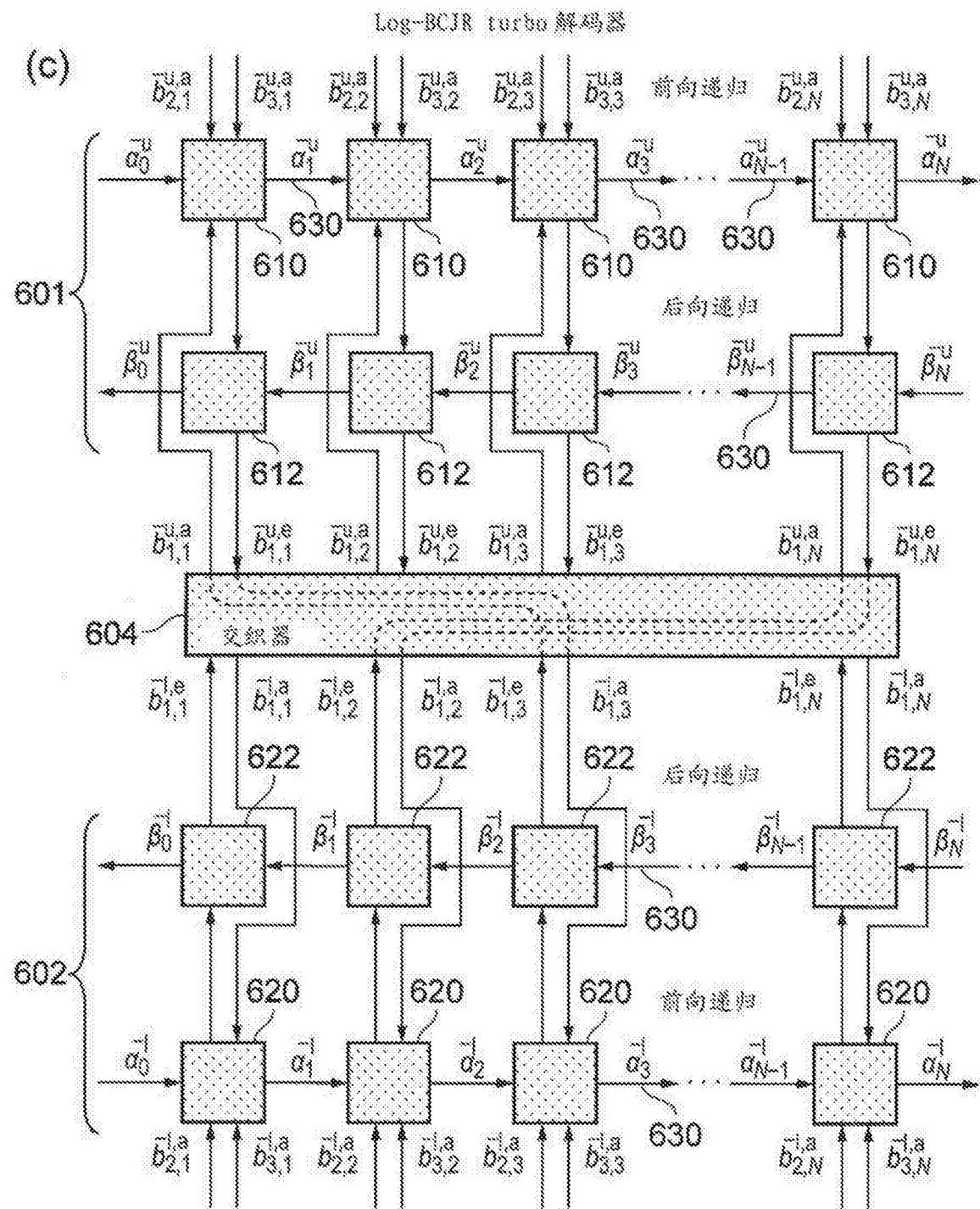


图7

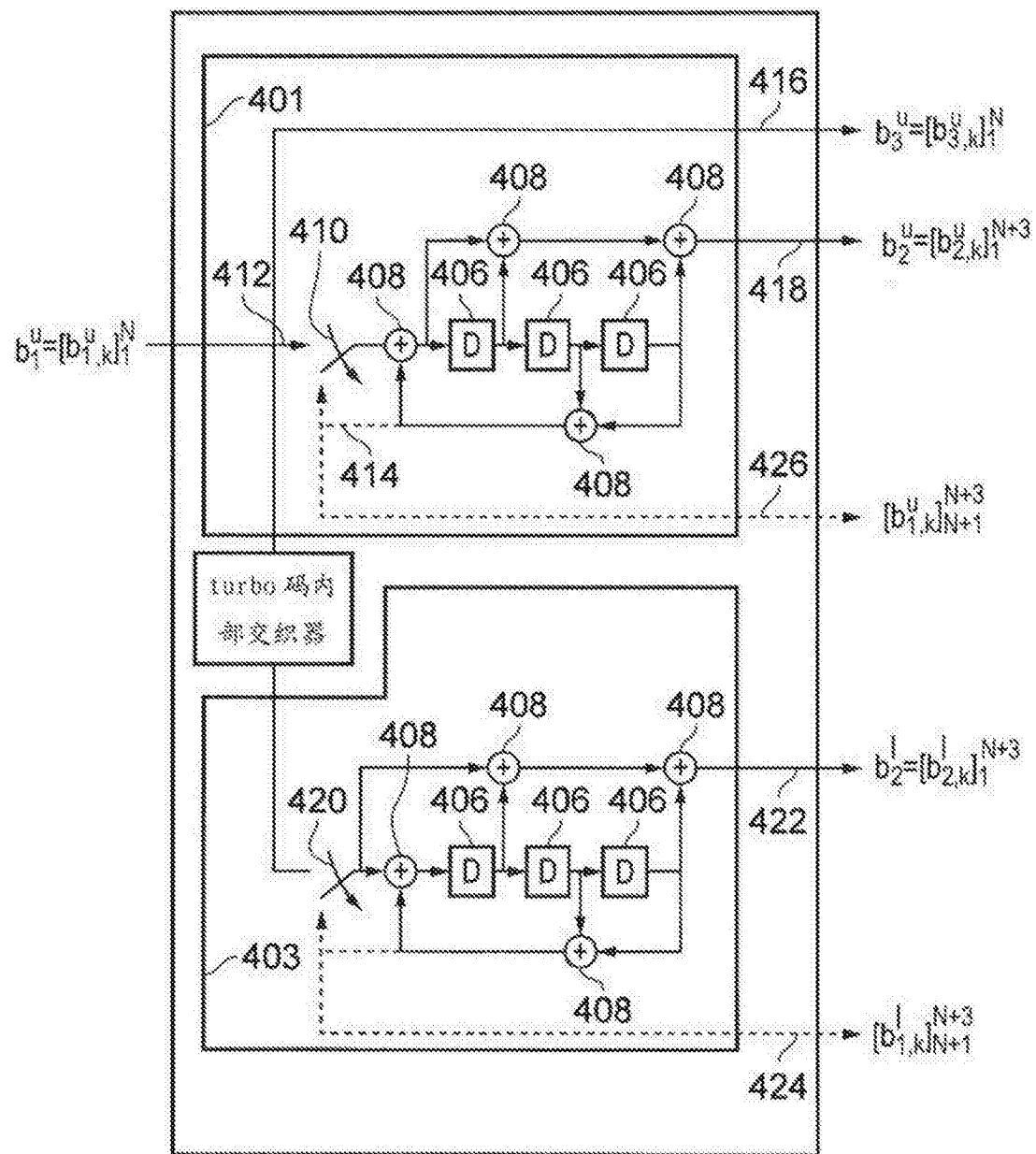


图5

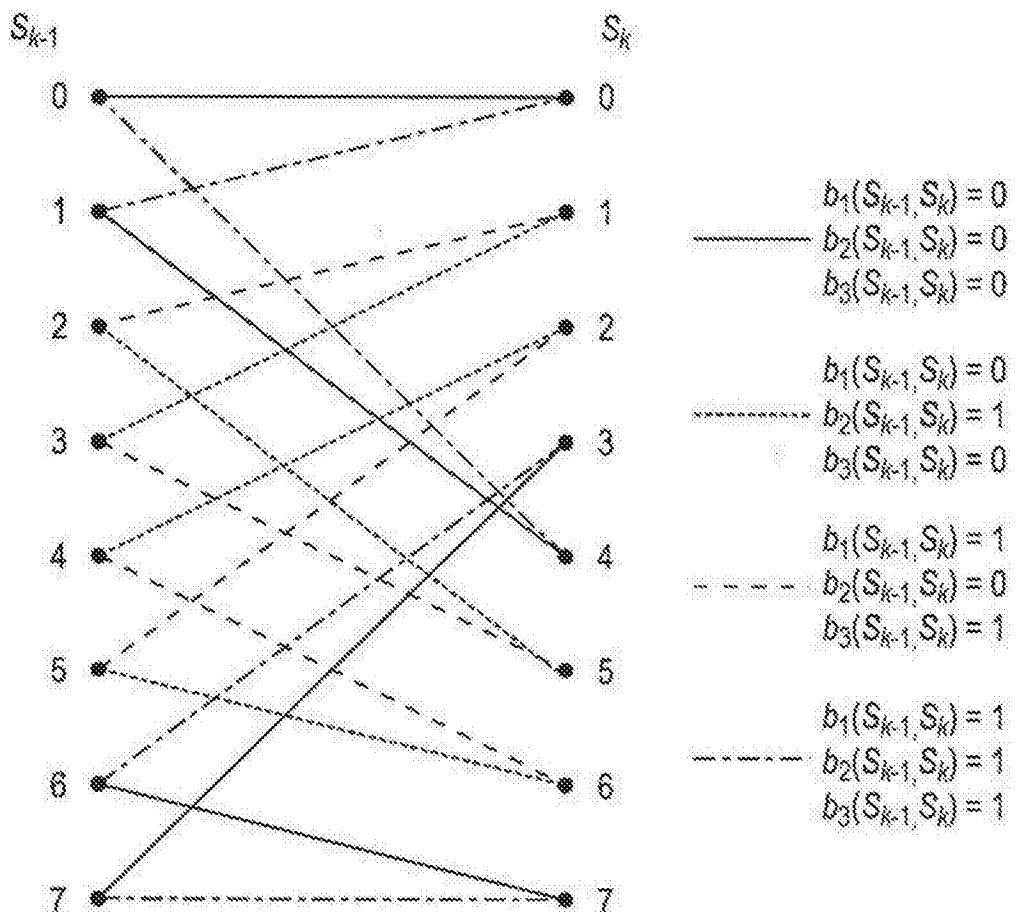


图6

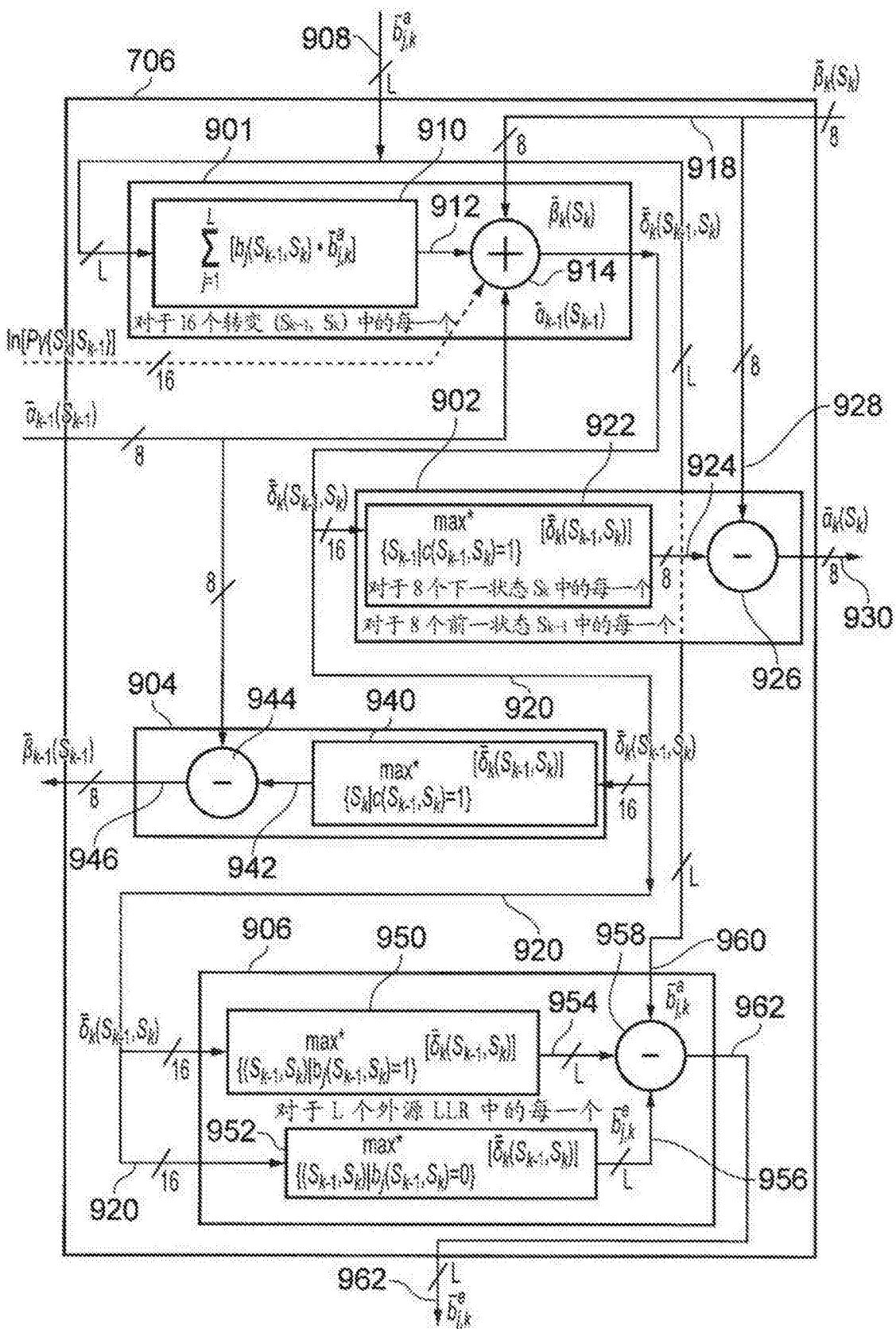


图9

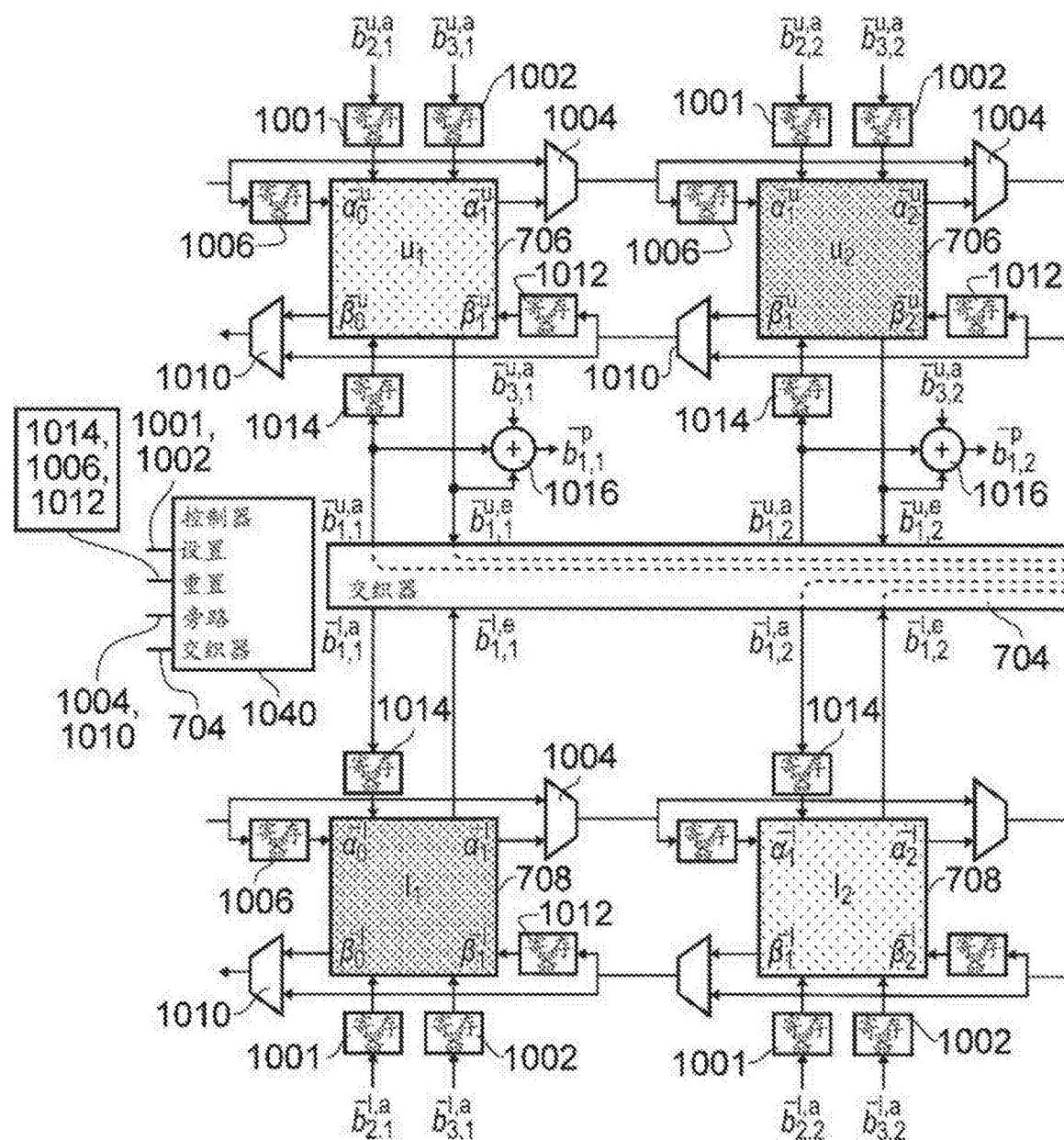


图10

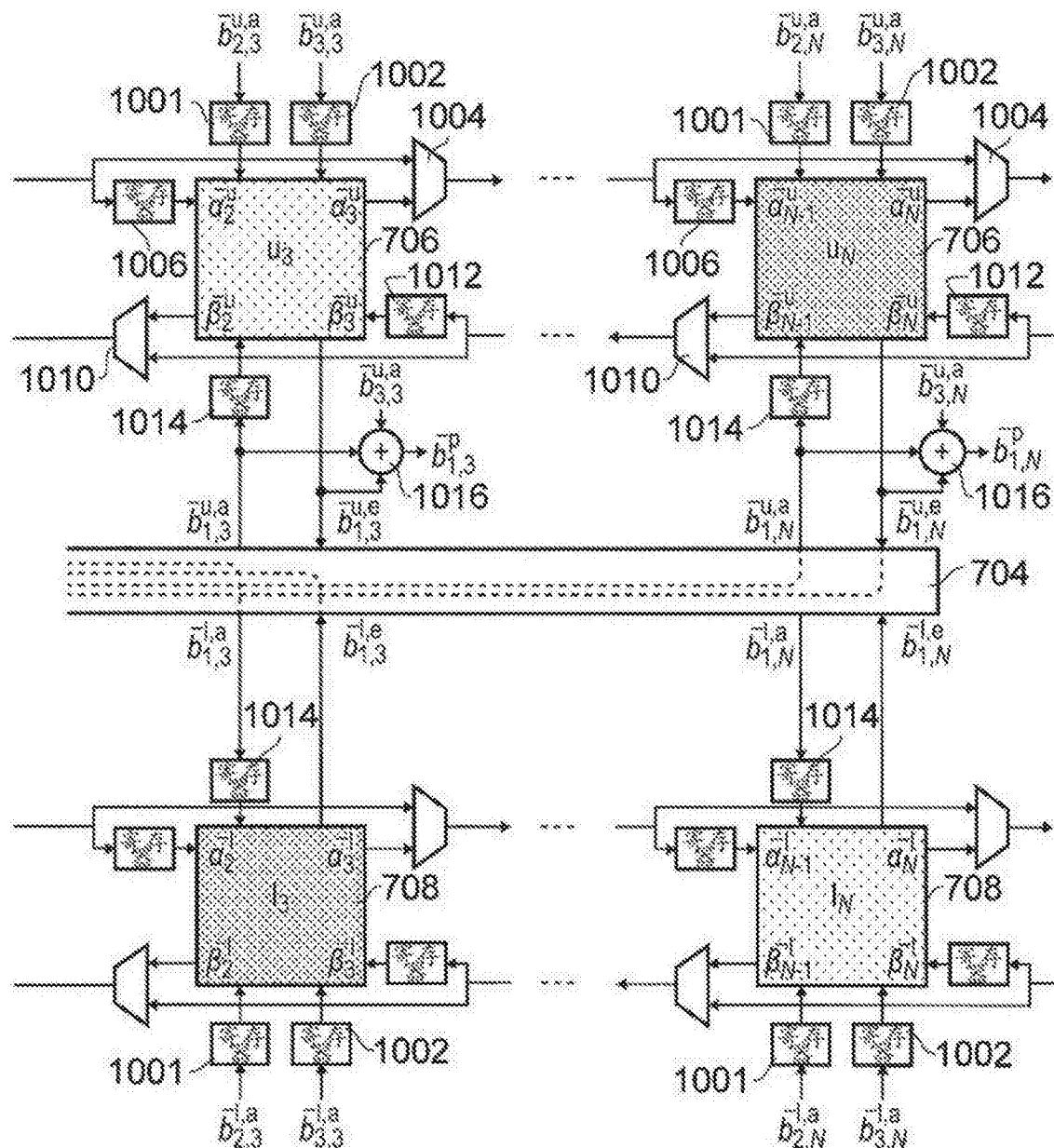


图10(续)

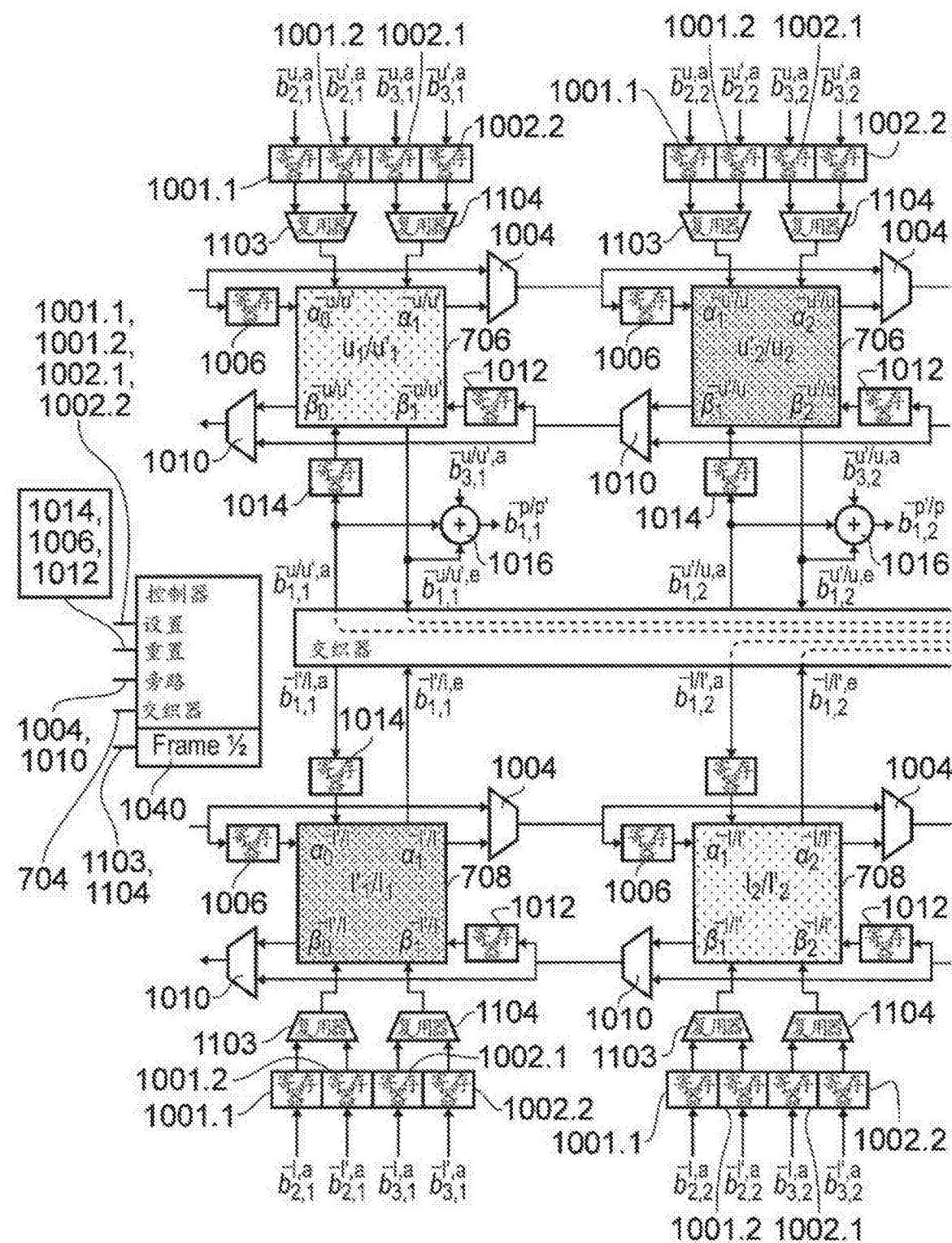


图11

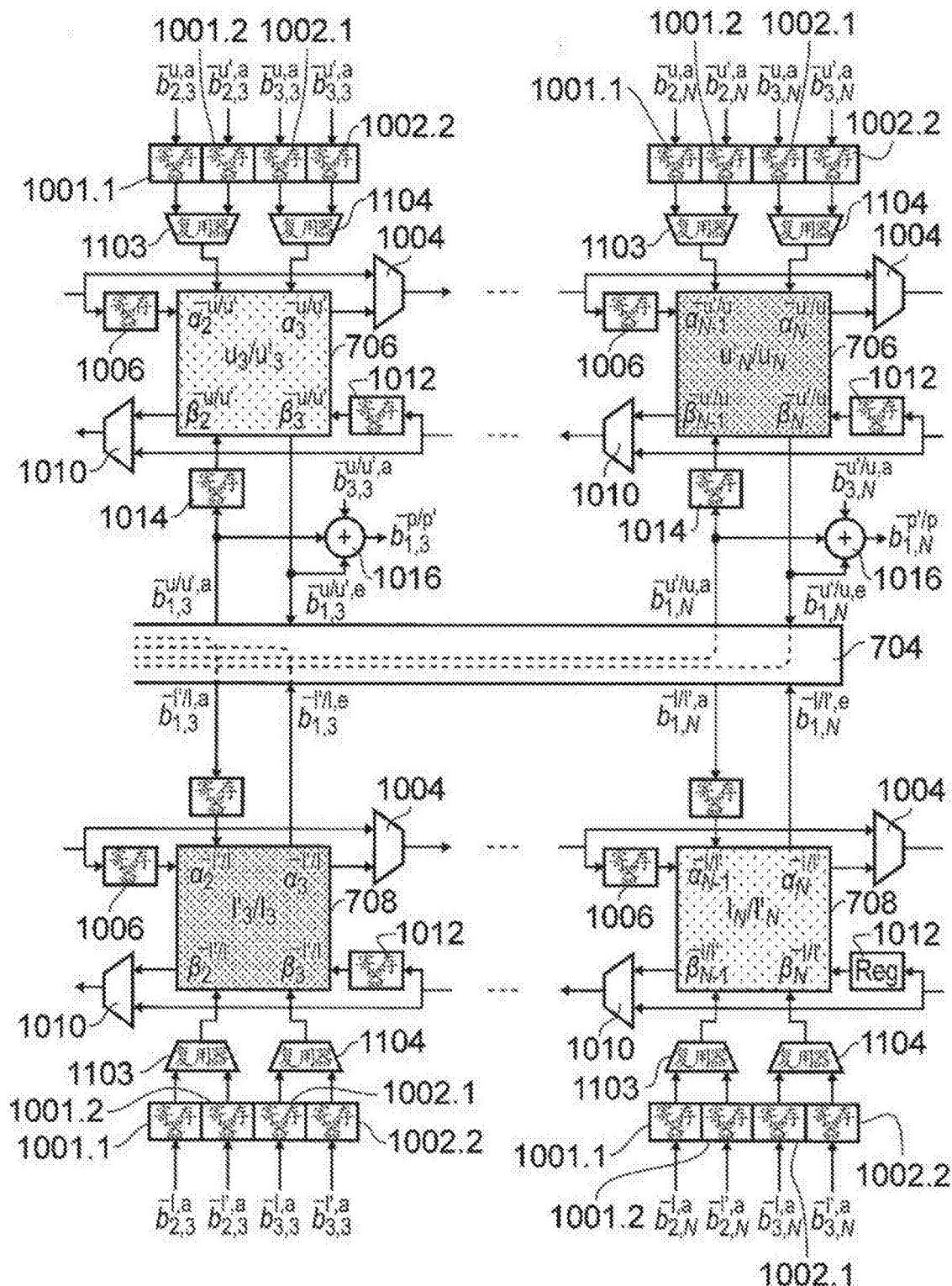


图11(续)

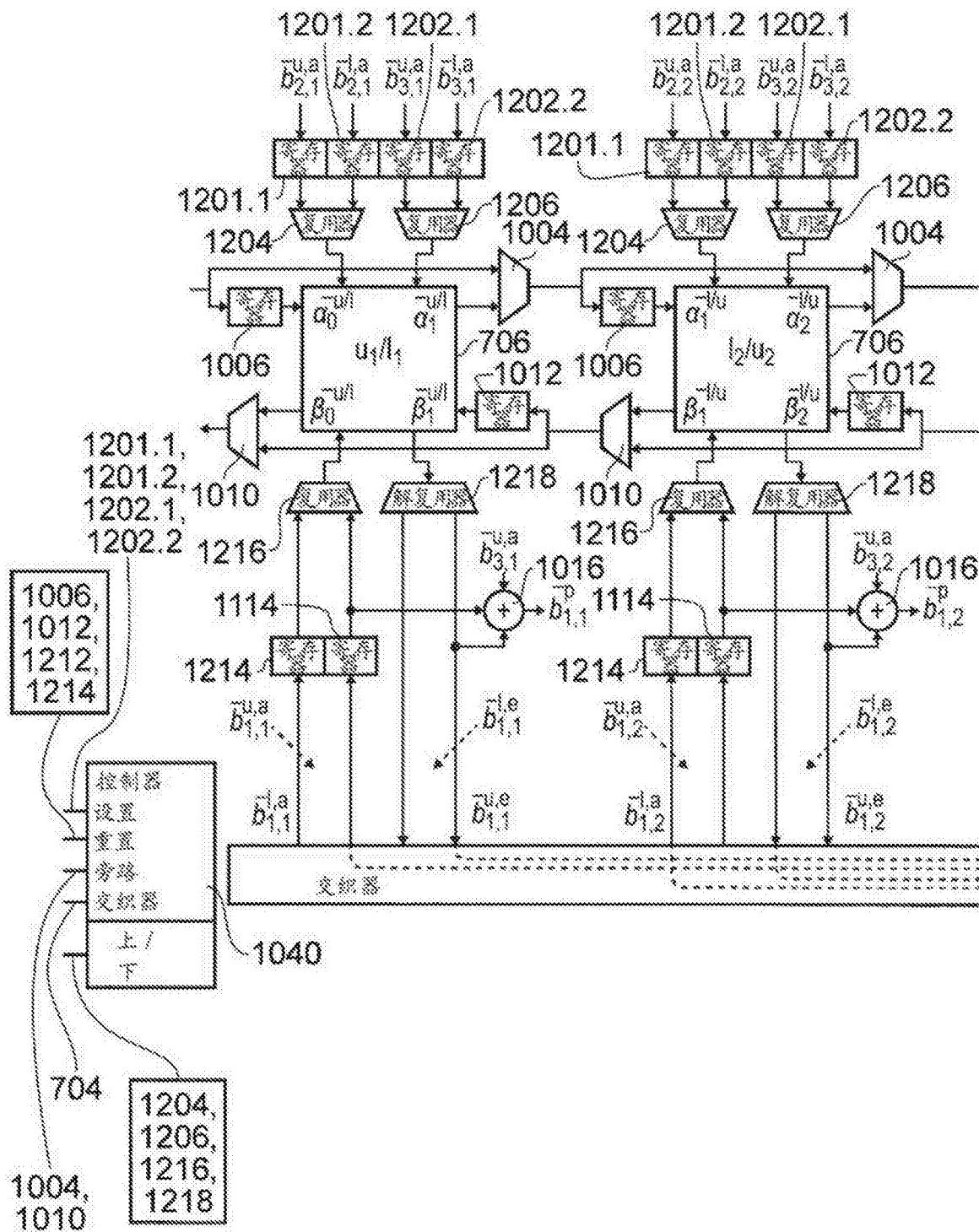


图12

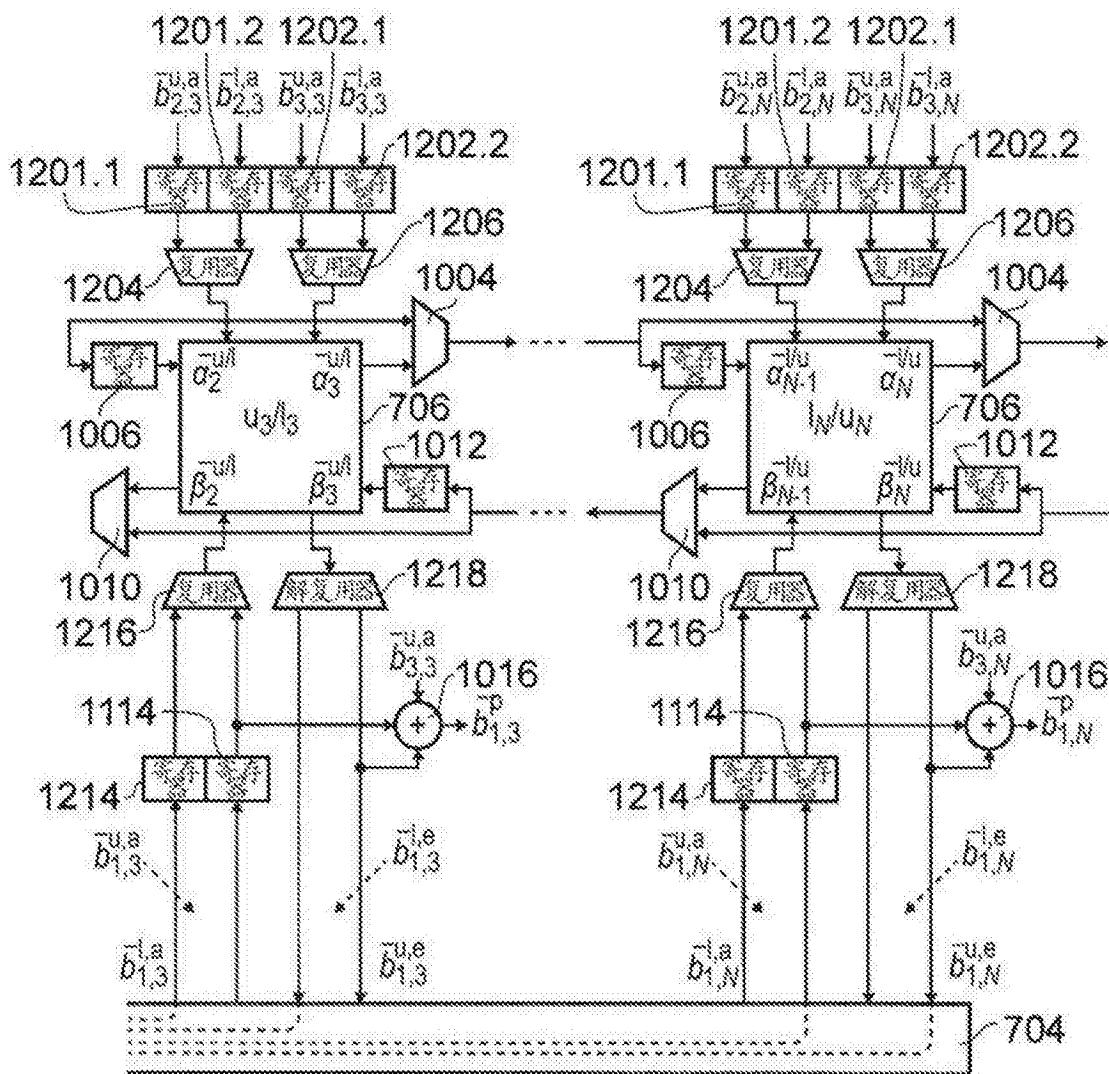


图12(续)

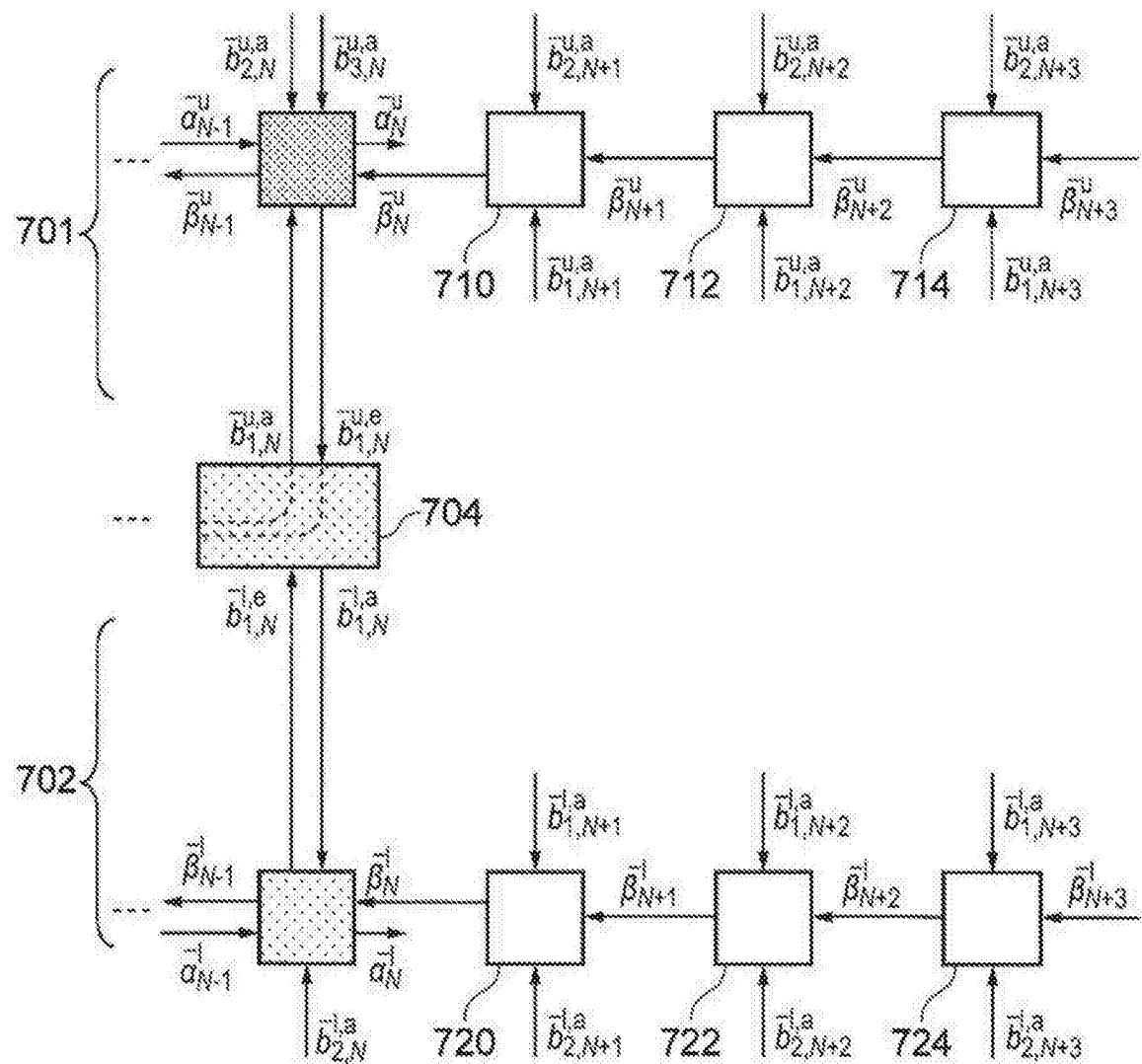


图13

特性	Log-BCH-LTE [40, 6144] 2W Tap	Log-BCH-RMAX [40, 6144] 2W Tap	现有技术 LTE [40, 6144] W84 Tap	现有技术 VMAX [40, 6144] W32 Tap
每解码迭代的时钟段 T				
时间段持续期 D	1.13 Tap	1.22 Tap	0.43 Tap	0.33 Tap
每解码迭代的复杂度 C	1.10 Tap	1.25 Tap	1.77 Tap	1.33 Tap
所需的解码迭代 I	1.6 Tap	1.4 Tap	1.8 Tap	1.4 Tap
总吞吐量 $\alpha \cdot \ TxData\ $	2.65W Tap	1.84W Tap	0.93W Tap	0.8W Tap
总时延 $TxDelay$	0.215 Tap	0.164 Tap	0.033 Tap	0.038 Tap
总复杂度 Cx	0.18 Tap	0.31 Tap	0.30 Tap	0.40 Tap
所需的存储器	9.33 Tap	8.00 Tap	1.25 Tap	1.67 Tap

表 1 提供根据本技术实施例的完全并行 turbo 解码器与等效 Log-BCH turbo 解码器和现有技术 turbo 解码器的性能比较，其中，Tap = 以解码出的完全并行 turbo 解码器的时钟段。

图14

Turbo 码	操作	本技术的完全并行					Log-BCJR				
		(7)	(8)	(9)	Total	(2)	(3)	(4)	(5)	(6)	(7)
LTE	+ 01 ~	23.5 {3}	8	8	2 {2}	47.5 {6}	1.5	12	12 {1}	28 {2}	2 {2}
	max*	0	8	8	14 {3}	30 {3}	0	8	8 {1}	0	14 {3}
	+ 01 ~	74 {3}	8	8	4 {2}	94 {3}	12	30	30 {1}	62 {2}	4 {2}
WiMAX	max*	0	24	24	32 {4}	80 {4}	0	24	24 {2}	0	32 {4}

表 2 编译器将本技术的完全并行 turbo 码编译为等效 Log-BCJR turbo 解码器所需的计算的数据的指针

图15

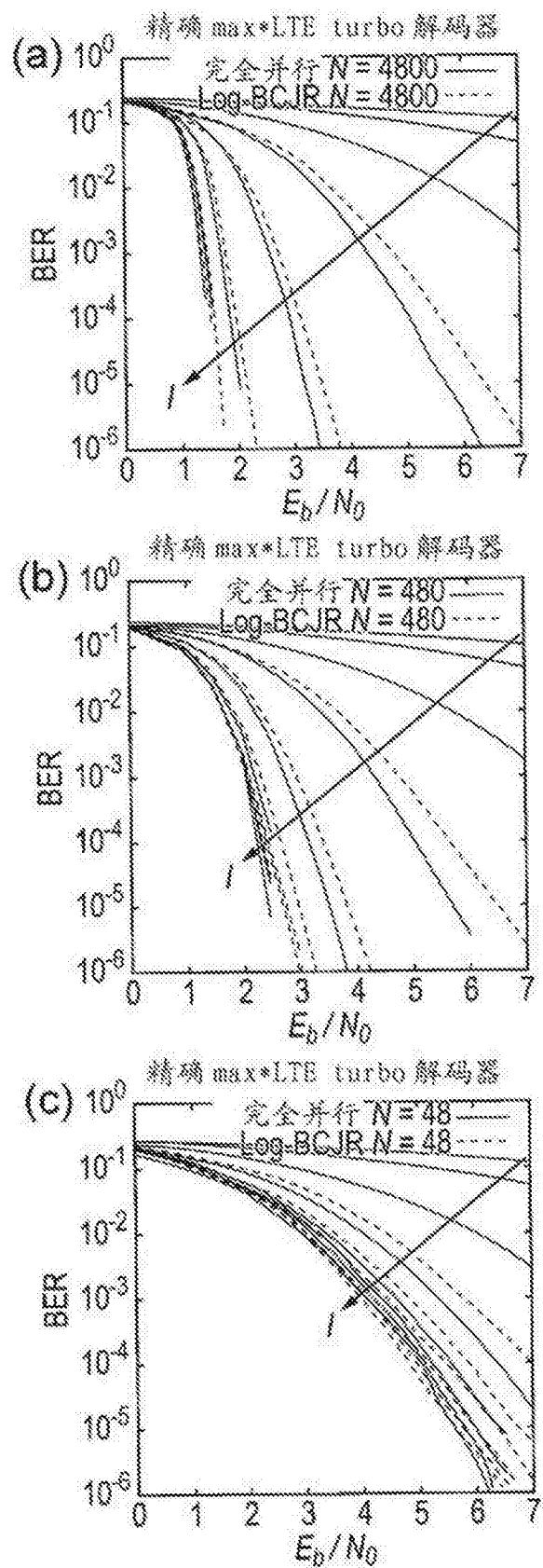


图16

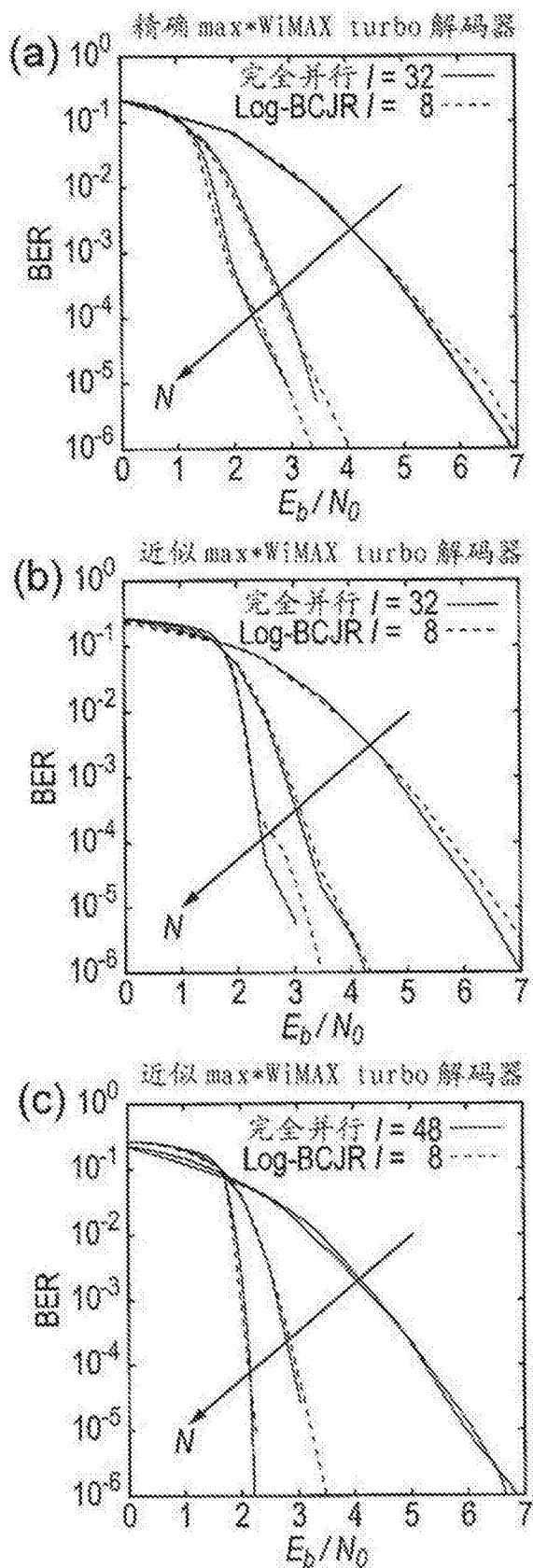


图17

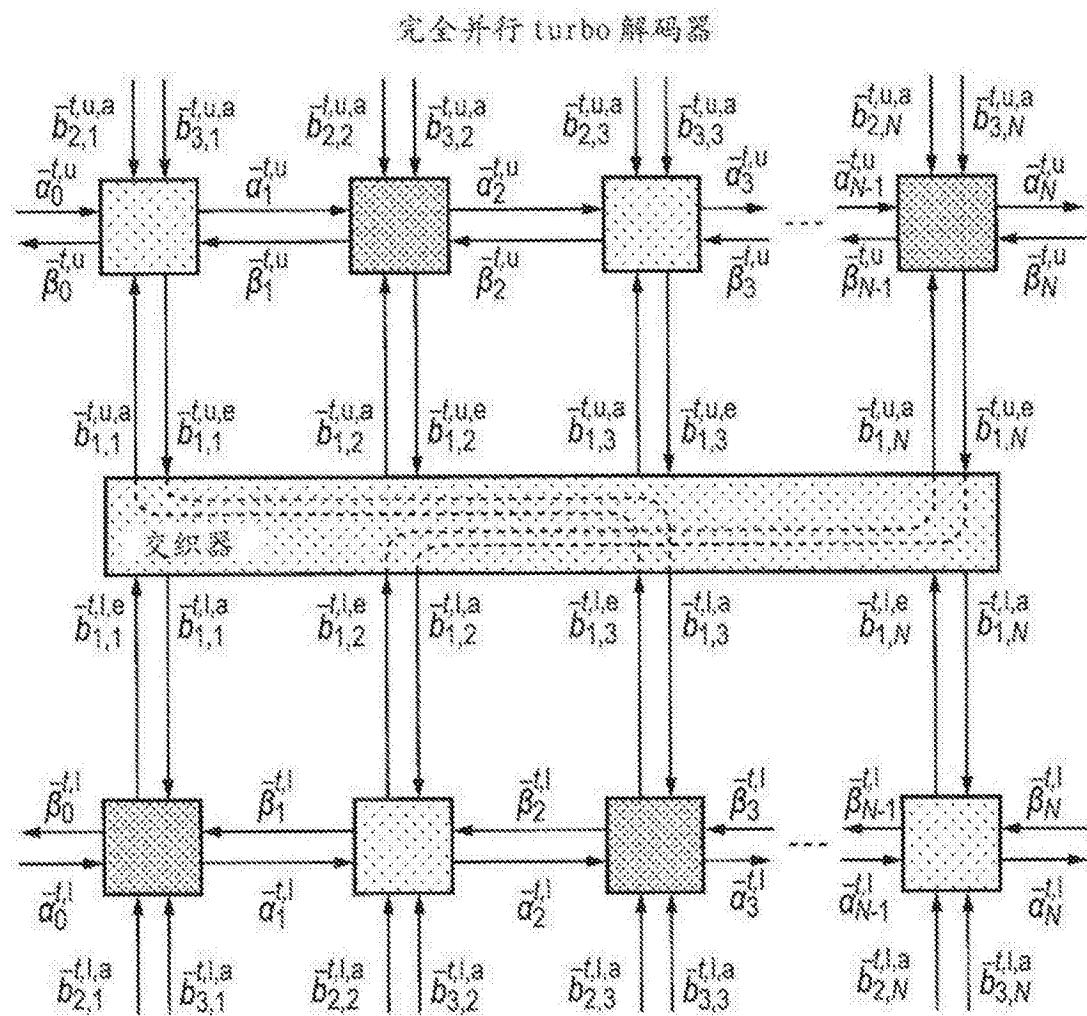


图18

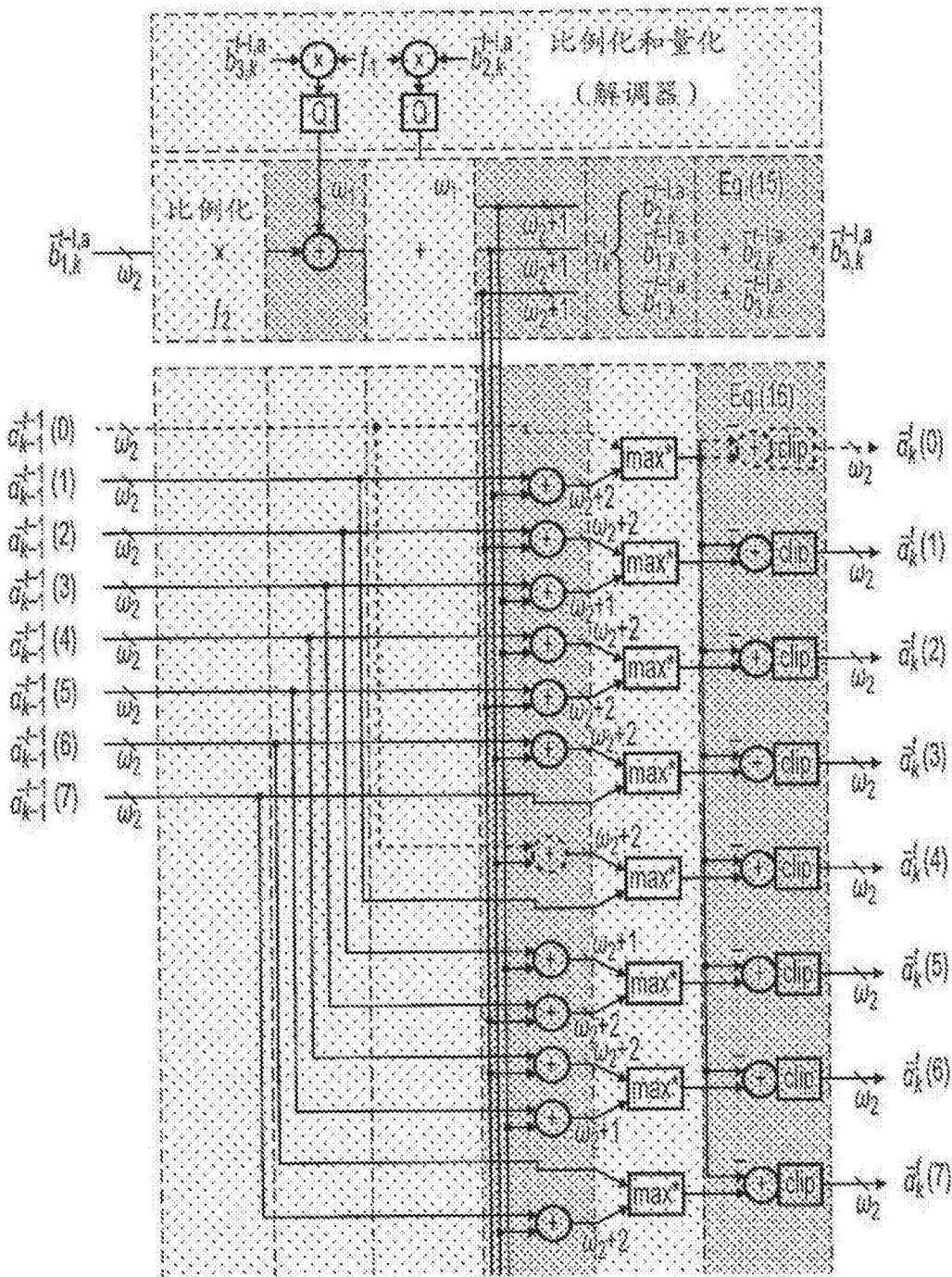


图19

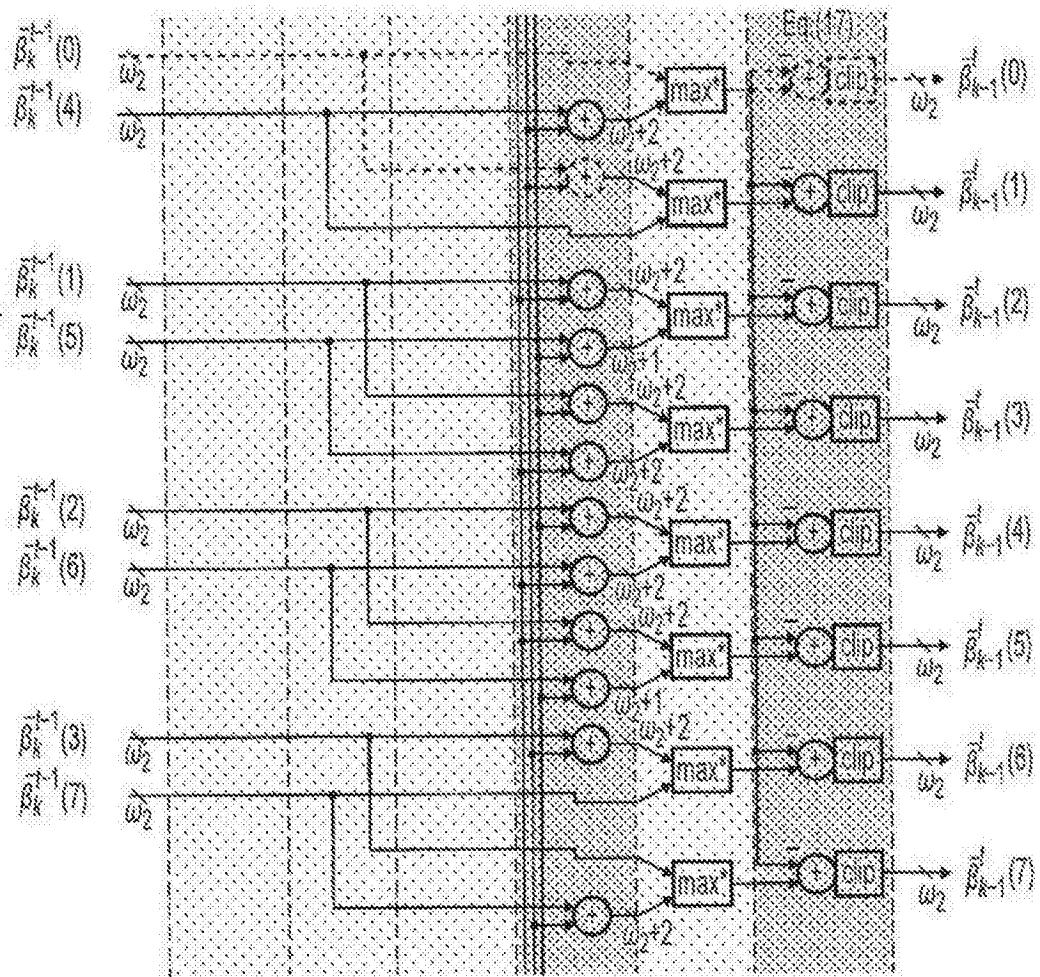


图19(续)

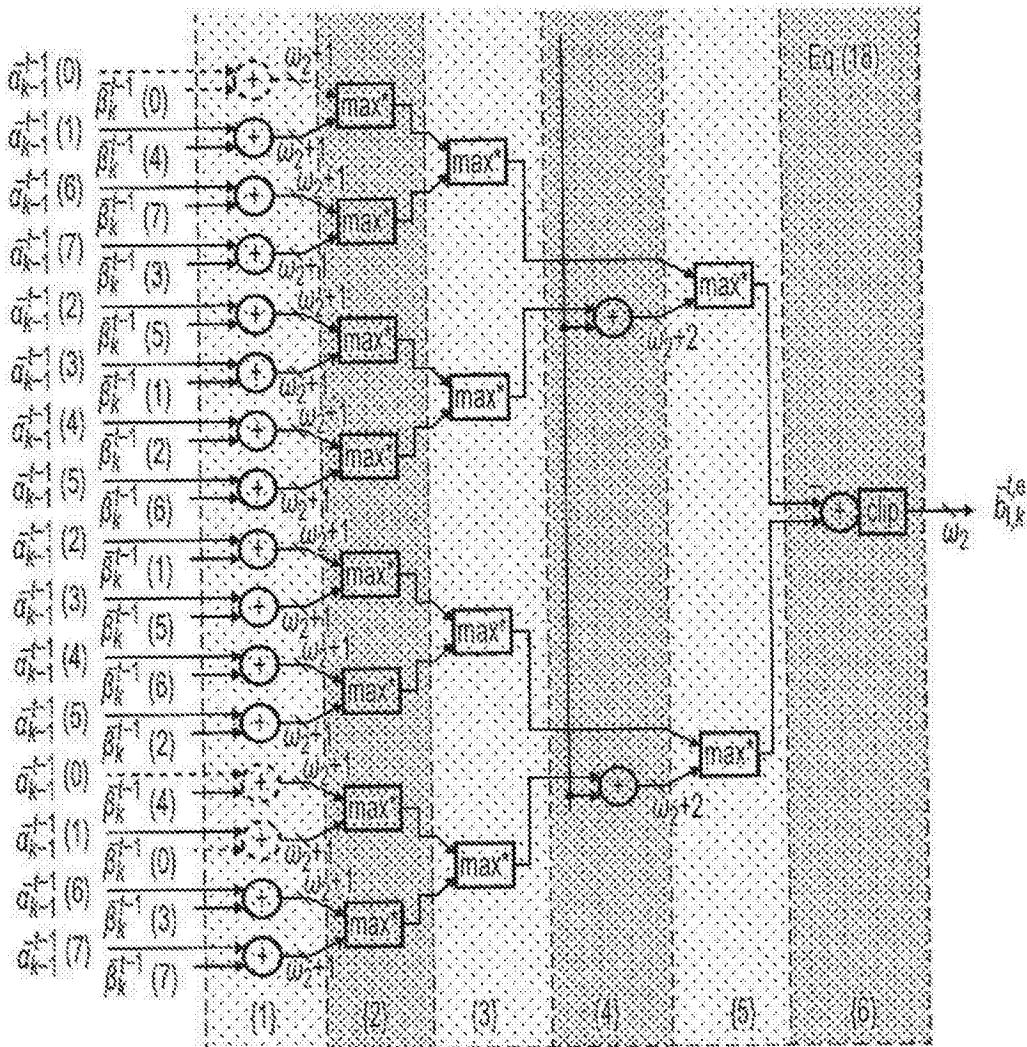


图19 (续)

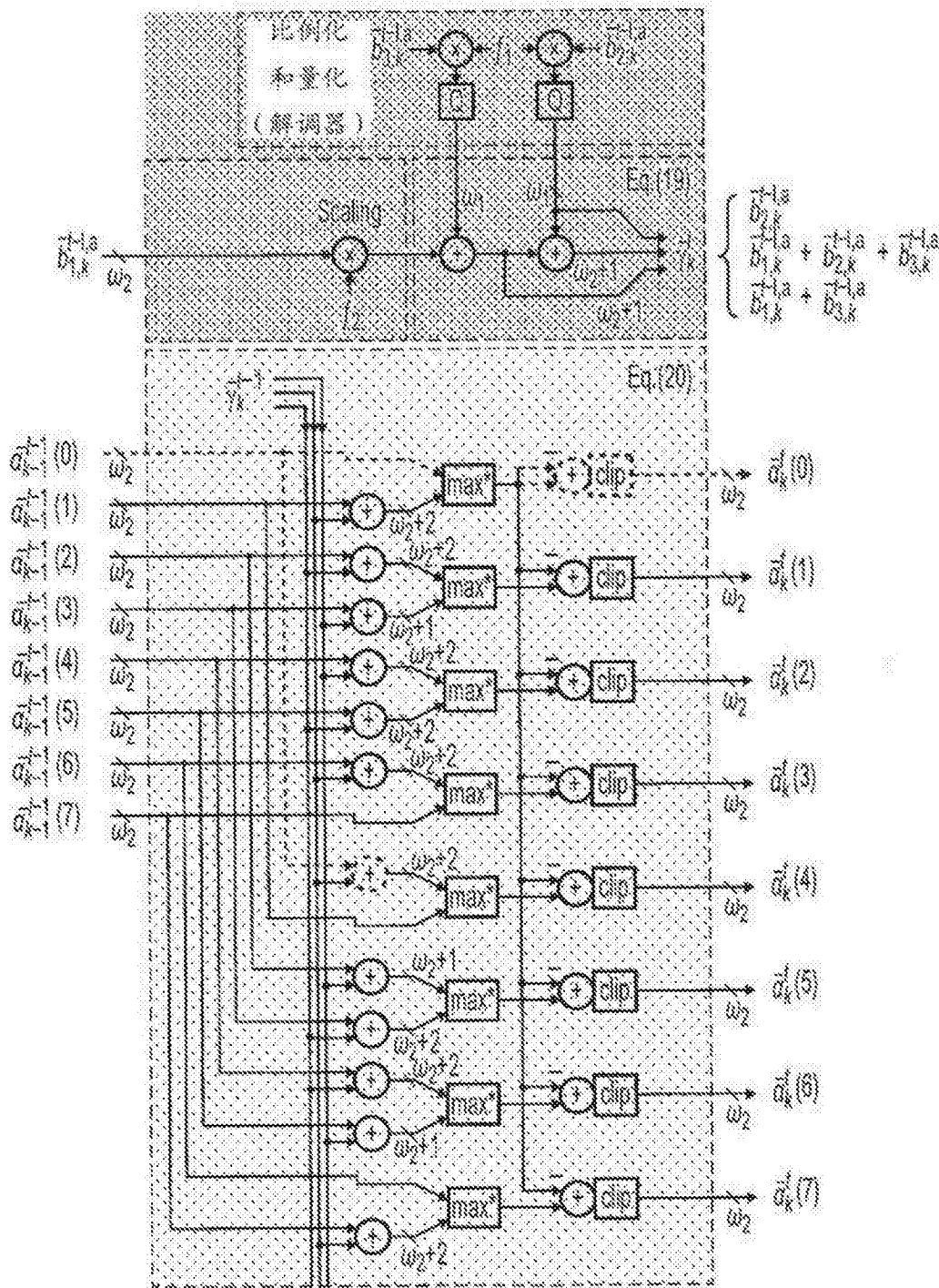


图20

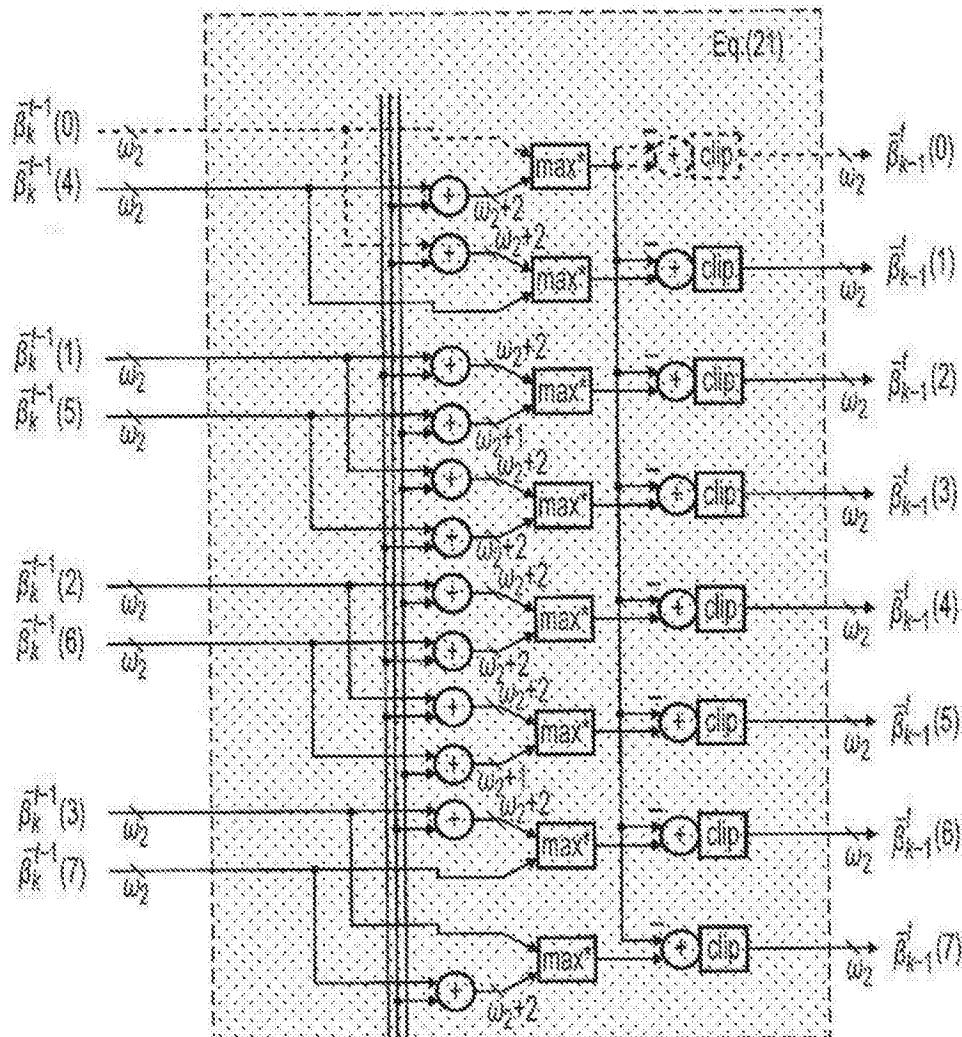


图20 (续)

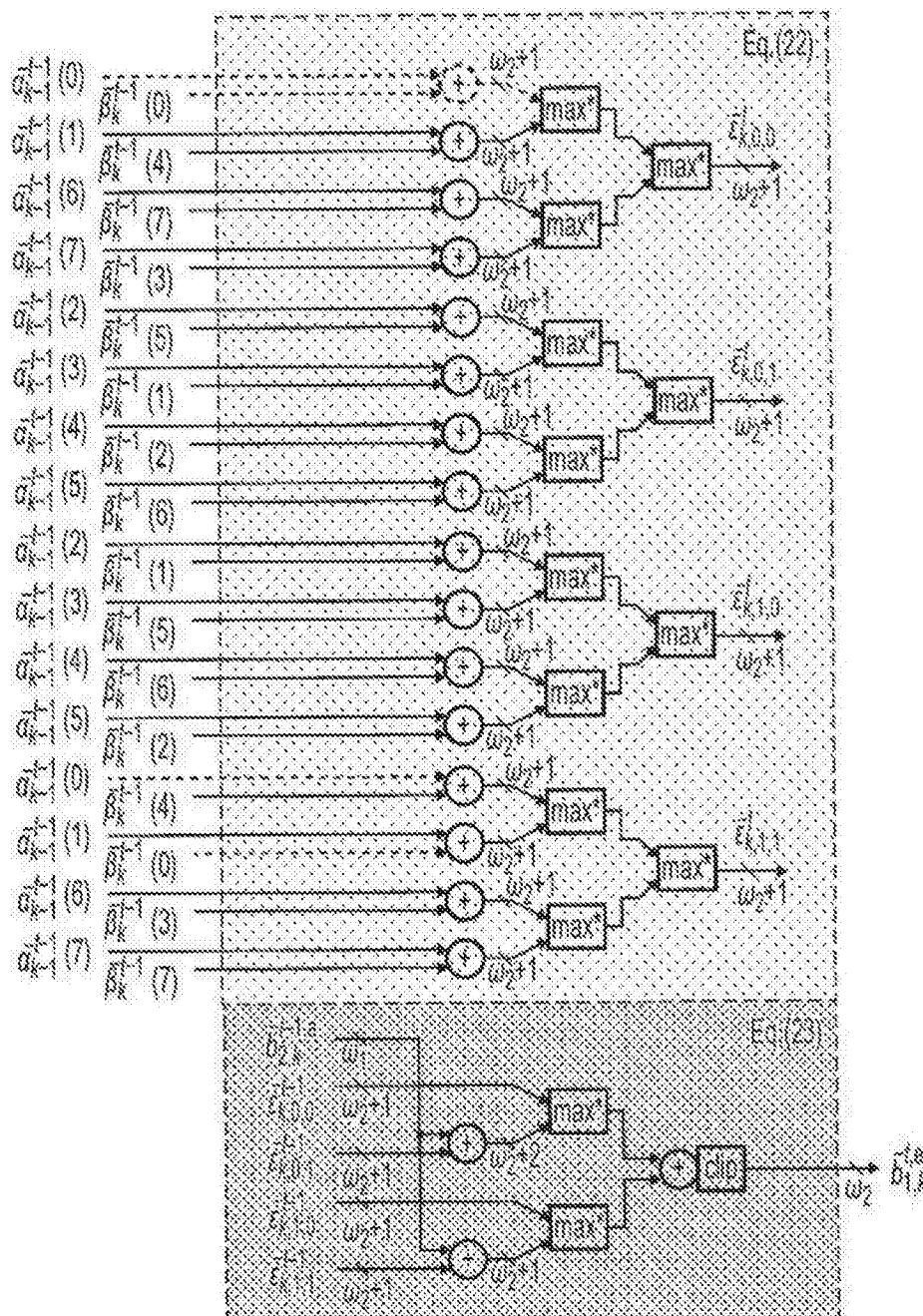


图20 (续)

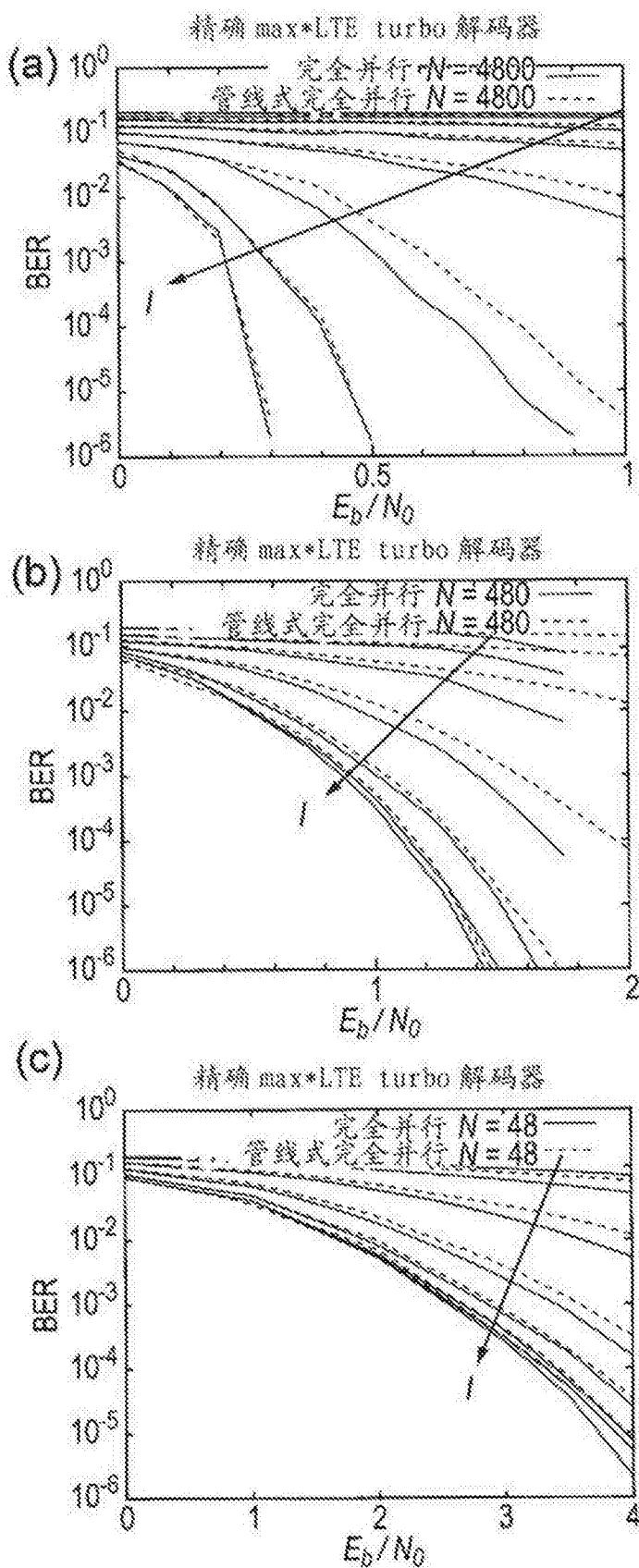


图21

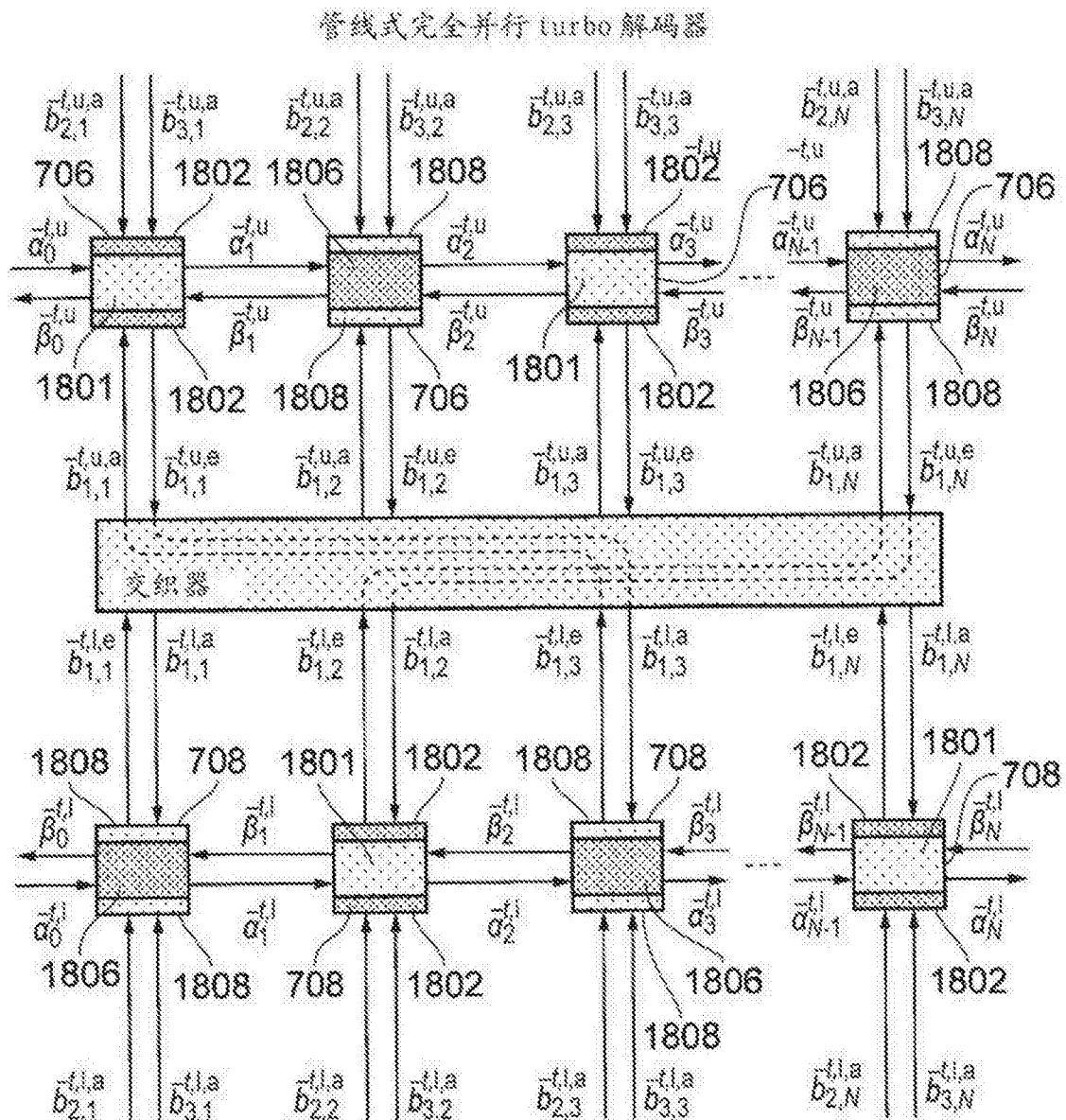


图22