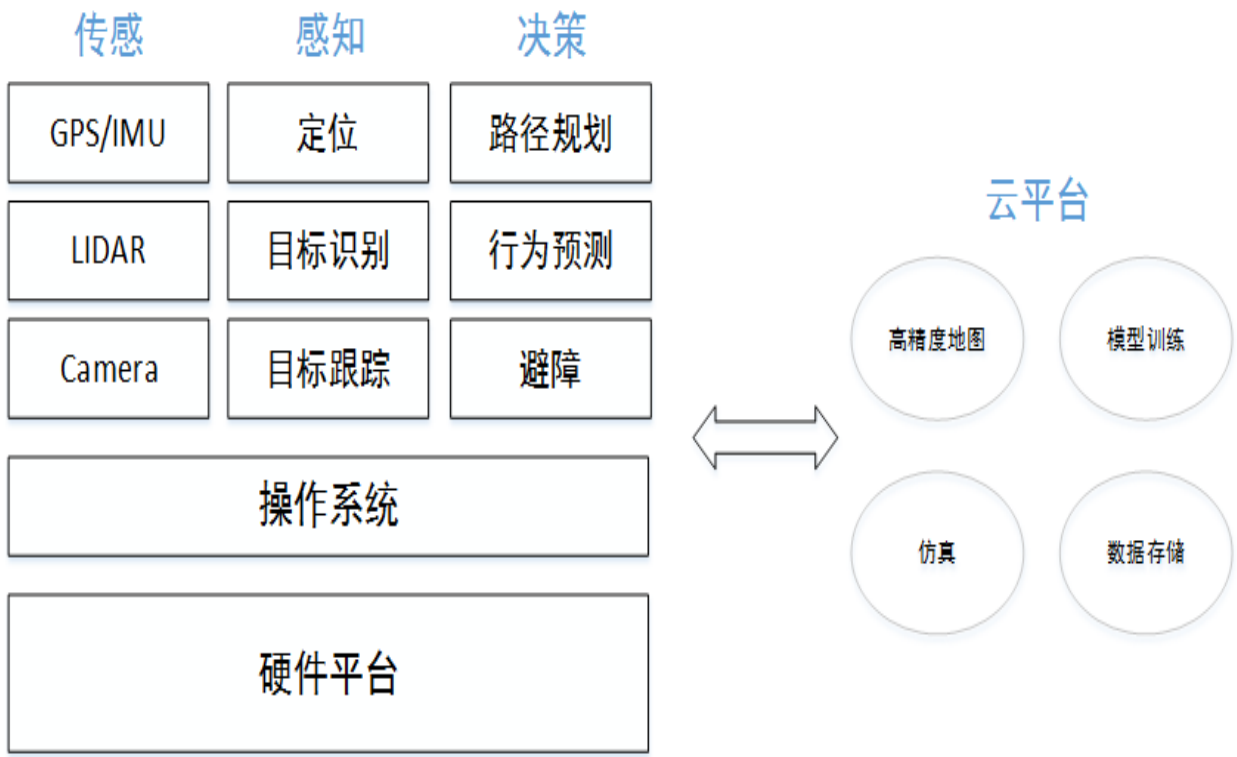
面向无人驾驶的嵌入式计算结构及若干算法研究

邹文超

背景意义：

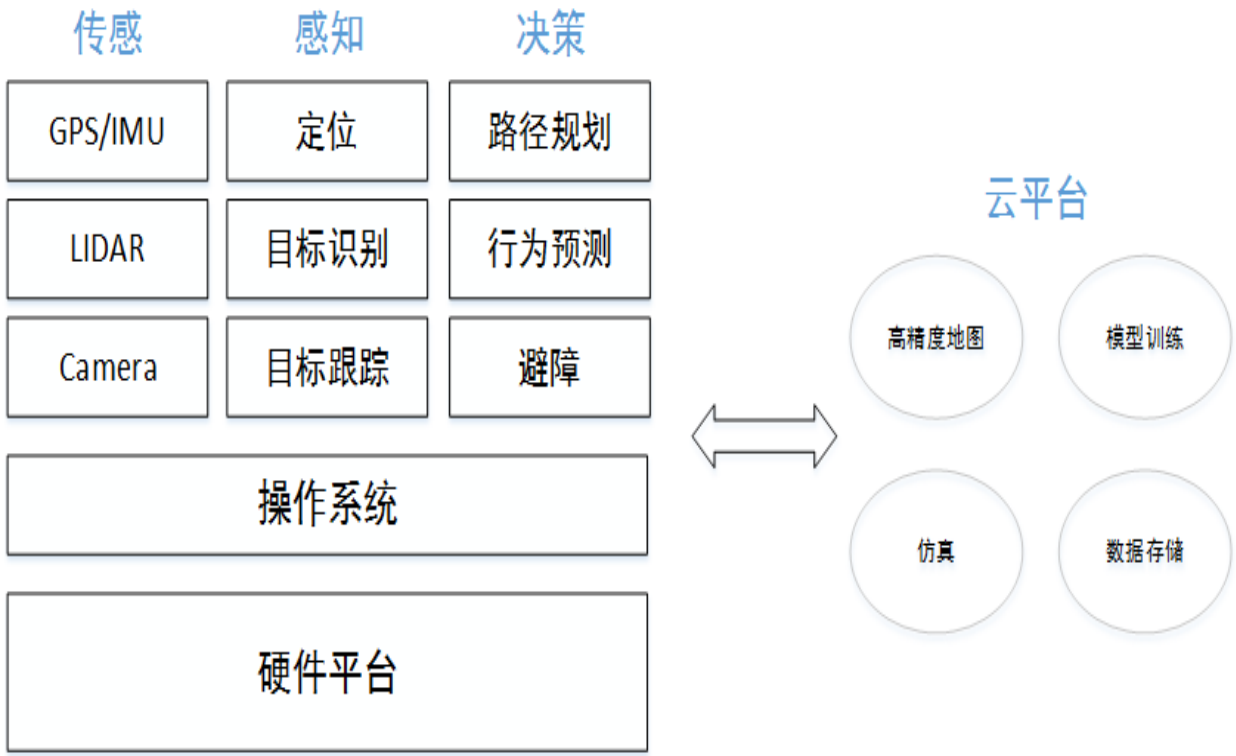
无人汽车是目前各大汽车企业的研究热点，无人汽车不但可带来巨大的经济效益，还具有以下优点：：1，增强高速公路安全。2，缓解交通拥堵。3，疏解停车难问题。4，减少空气污染。

- 现有领先的无人车驾驶产品的计算平台由两个计算盒组成,互为备份，每一个装备有INTEL xeon E5处理器及4到8个Nvidia Tesla K80 GPU。在峰值下运行,功耗高达5000w,每一个计算盒高达2万至3万美元。



背景意义：

- 无人车比普通车更容易受到黑客攻击。传感器层面：比如GPS，如果在无人车附近设置大功率假GPS信号，就可误导无人车定位。还有激光雷达等也容易受到攻击，可能导致汽车判断失误。
- 网络层面：无人车作为一个移动系统，需要从网络上获取高精度地图，如果黑客把下载的地图掉包，可会造成定位失败，或者通过网络入侵无人车系统内部，将会给车辆带来不安全因素。
- 此外还有普通车会受到的攻击类型，无人车也会遇到，比如针对汽车内部总线网络的攻击等。因此我们在设计无人驾驶汽车系统结构时，需要把安全放在一个非常重要的地位进行考虑。



背景意义：

无人驾驶系统必须满足以下四个要求：

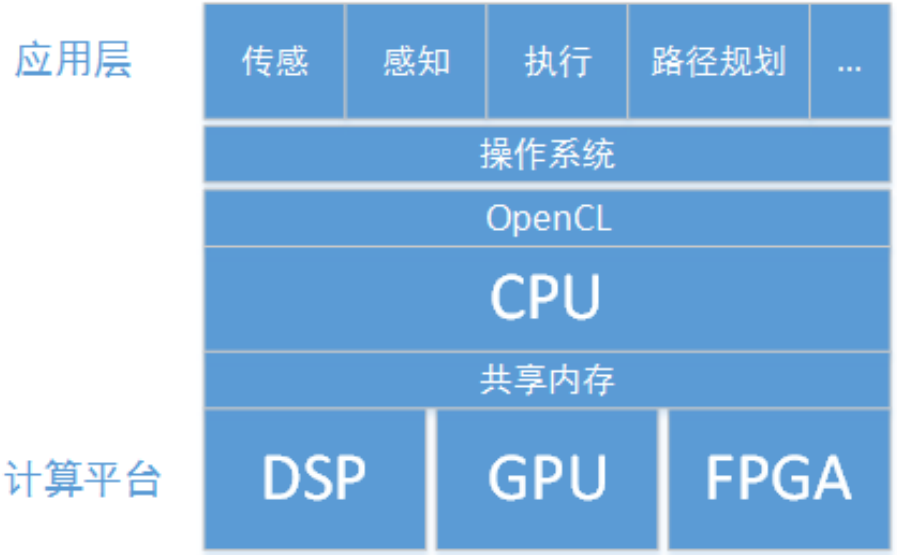
- 其一，系统必须确保捕捉到的传感器数据得到及时处理；其二，如果系统的某部分失效，则系统需要有足够的健壮性能从错误中恢复；其三，系统必须在设计的能耗和资源限定下有效完成所有计算操作。其四，无人驾驶系统必须能够有效抵御黑客攻击。

研究问题：

- 现有的无人车设计方案存在功耗，散热及造价问题，使得无人车进入大众遥不可及。
- 正基于此，面对无人驾驶应用，计算量大，任务繁杂的特点，如何让系统既能满足无人驾驶应用实时性，高可靠性的计算需求，又要让硬件成本和功耗要尽可能小，还能抵御黑客攻击，是一个迫切需要解决的问题。

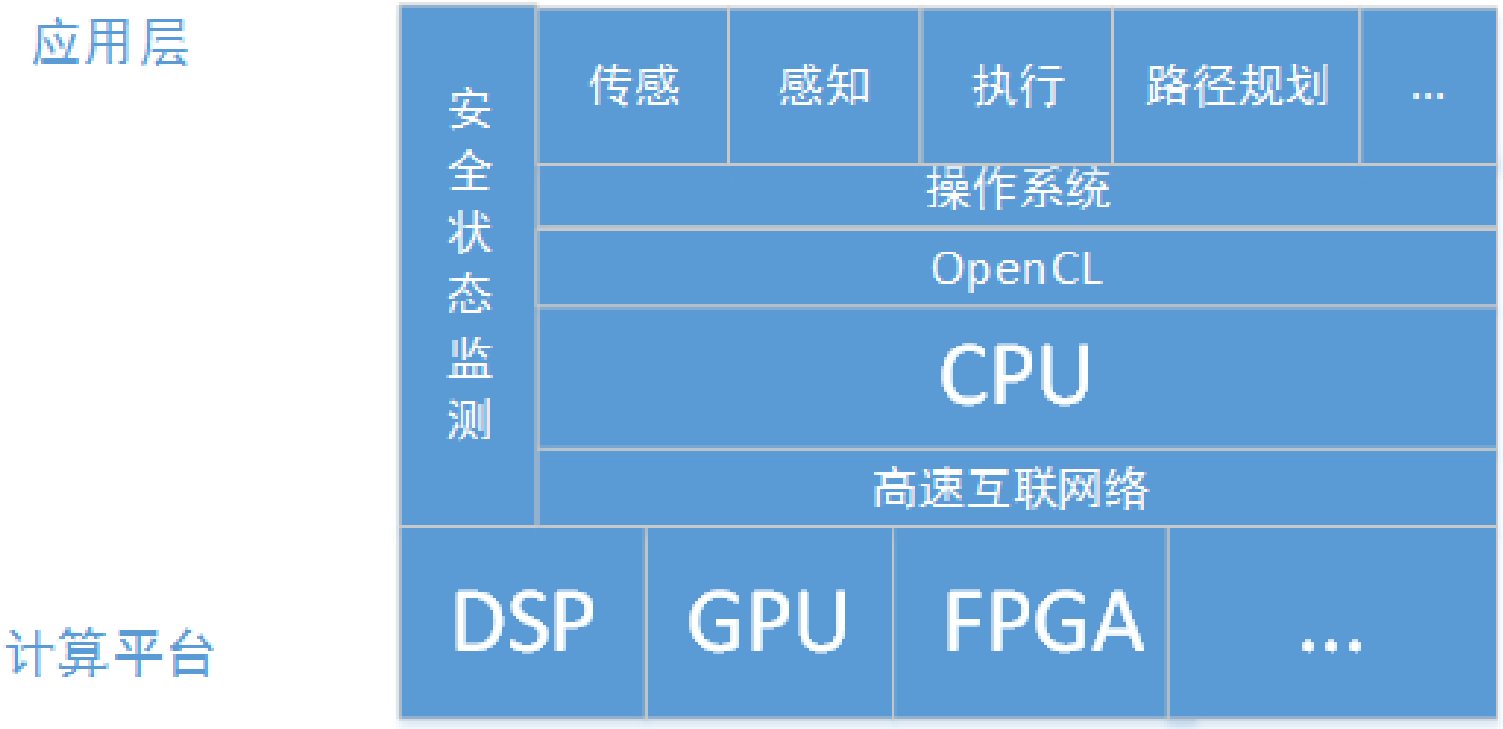
无人驾驶相关研究：

- S.Liu和J.Tang等人在ARM SOC实现了一个简化的无人驾驶系统，这个ARM SOC由一个四核CPU,GPU,DSP组成。试验显示，在峰值情况下能耗为15w，在不损失任何位置信息的情况下达到每小时5迈速度。
- ARM SOC能提供这样的性能，是因为充分利用了硬件系统的异构计算资源，为每一个不同无人驾驶子任务寻找最匹配的的计算单元，可以达到最优化的性能和能源效率。



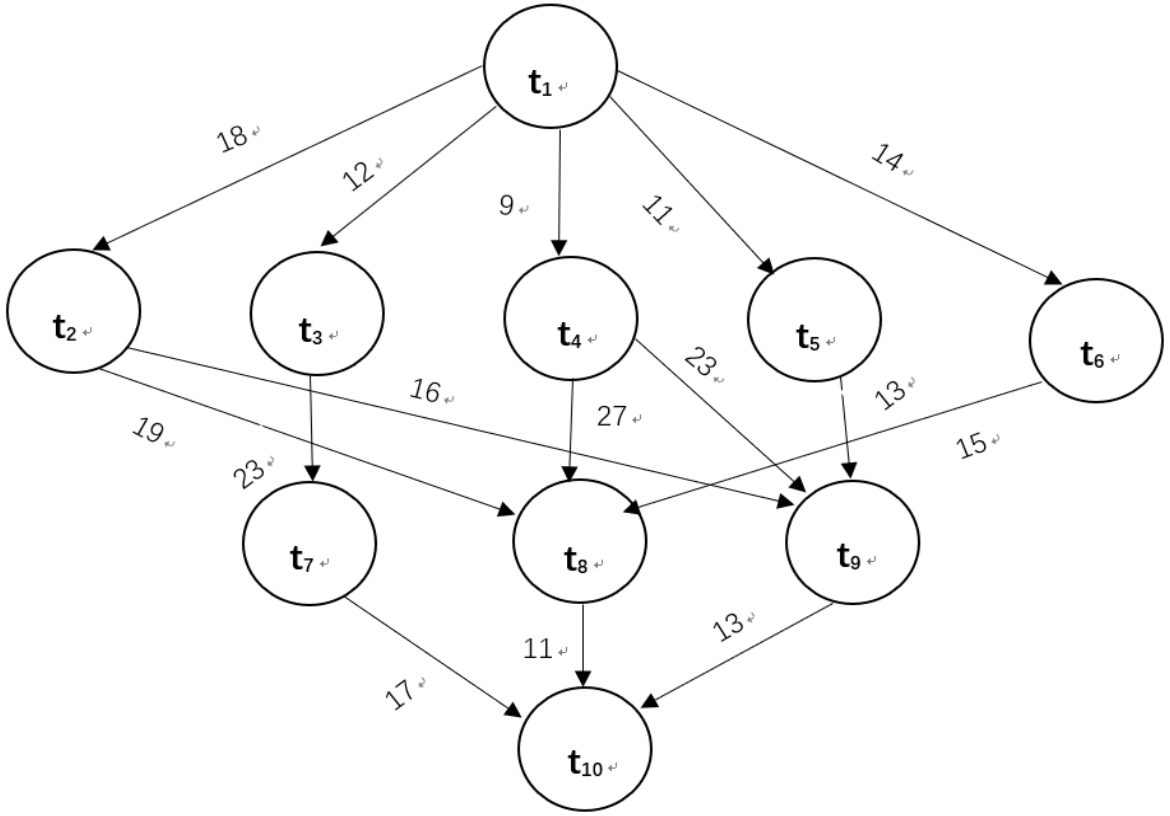
技术路线：

- ARM SOC的性能毕竟有限，因为是在一个片上集成多个处理器。
- 基于前面这篇论文的研究，我们可以设计一个异构分布式计算结构来满足无人驾驶应用的需求。



异构分布式相关研究：

- 应用模型
- 一个应用程序可以用有向无环图(DAG)来表示，如图1所示。$\mathrm{G} = (\mathrm{T}, \mathrm{M})$，T表示任务节点集合，每个子任务$t_i \in T$。M表示通信边集合，这些边也表示了任务之间具有的优先约束。$m_{i,j} \in M$表示从任务t_i到任务t_j的通信代价，如果任务t_i和任务t_j在同一个处理器上执行，则通信代价$m_{i,j} = 0$。每个任务在不同处理器上的执行时间用矩阵W表示，$w_{i,j} \in W$表示任务t_i在处理器p_j上的执行时间。



异构分布式相关研究：

- Hulak 等人针对异构多处理器静态调度提出了一个异构最早结束时间算法（HEFT）和CPOP算法。HEFT算法使用升秩(upward)作为任务的优先级，如公式(1)所示：
- 通过对每个任务的升秩排序，每次在满足优先约束条件下，将升秩最大的任务分配给该任务结束时间最早的处理器上执行，使得该应用的最早结束时间最小。。
- $rank_p(t_i) = \overline{w_i} + \max_{t_j \epsilon succ(t_i)} (\overline{m_{i,j}} + rank_p(t_j))$ （1）
- 其中$\overline{w_i}$是任务t_i在处理器集合上的平均计算代价，$succ(t_i)$是任务t_i的后继任务集合，$\overline{m_{i,j}}$是平均通信时间。通过对每个任务的$rank_p$进行降序排列，每次选择$rank_p$最大的任务优先调度。注意只有当t_i的所有先继任务执行完成后，t_i才会准备执行。

异构分布式可靠性相关研究：

- 论文[14]中提出了一个MaxRe算法在满足可靠性的目标下，优化资源消耗。通过将整个应用（包含n个任务）的可靠性目标开n次方根，转化为每个任务要满足的可靠性目标，减少每个任务需要的副本数目，从而优化资源消耗。
- G. Xie,和Y. Chen等人提出了一个满足可靠性目标的优化资源消耗算法，使用公式(2)将整个应用的可靠性目标转化为每个子任务要满足的可靠性指标。用升秩表示每个任务的优先级，依次选取优先级最大的任务，然后将它调度到合适的处理器上执行，使得可靠性指标得到满足。
- $R_{goal}(t_j) = \frac{R_{goal}(G)}{\prod_{x=1}^{j-1} R(t_x, p_{proc(x)}) \times \prod_{y=j+1}^{|T|} R_{max}(t_y)}$ （2）
- 其中$R_{goal}(t_j)$表示任务t_j要满足的可靠性，$R_{goal}(G)$表示整个应用的可靠性，$R_{max}(t_y)$表示任务t_y的最大可靠性，通过遍历任务t_y在每个处理器上的可靠性得出。

异构分布式相关研究：

- 针对异构嵌入式系统的能耗限制问题，B.Zhao, H.Aydin等人提出一个方法，在时间和能源功耗限制条件下，通过动态调整处理器频率（DVFS），使可靠性达到最高。
- 通过降低电压的方法减少能耗，会增加处理器的瞬态错误率，因此能耗和可靠性是存在一定冲突的。
- G. Xie等人提出一个实时应用的优化能源消耗算法，通过HEFT算法生成一个调度方案，在满足实时性要求前提下，降低处理器频率，优化能源消耗。

异构分布式相关研究：

- 为了降低异构嵌入式系统硬件成本（处理器成本），G. Xie等人提出了一个算法在产品的设计阶段优化异构嵌入式系统硬件成本，该算法通过不断移除高硬件成本的处理器，在满足硬实时和可靠性条件下，让硬件成本得到优化。

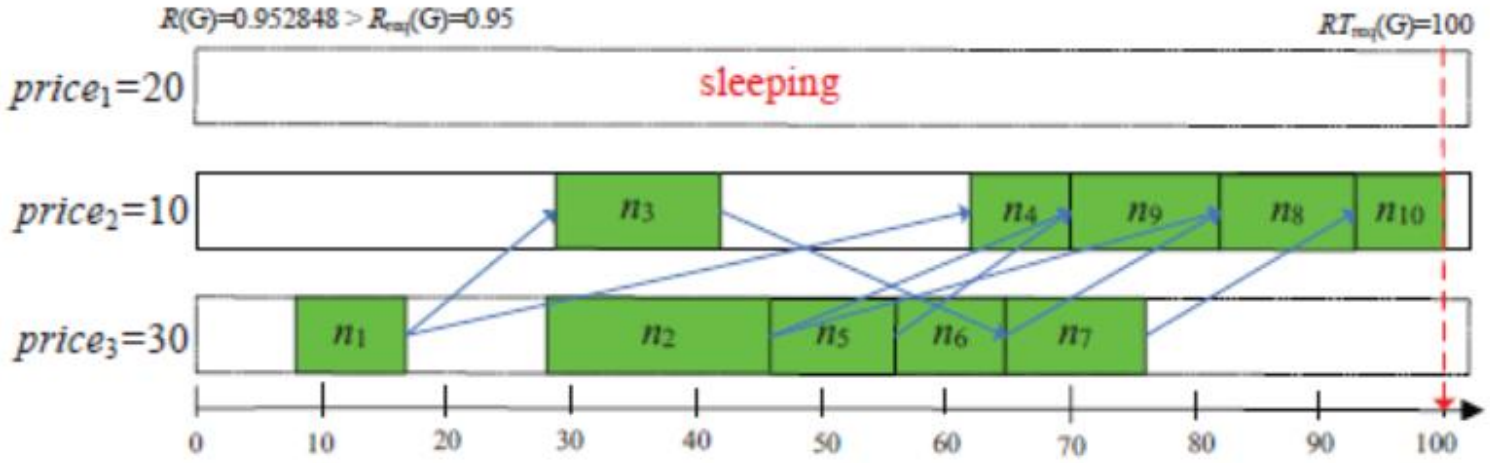


Fig. 10: RE-generated motivating application's task mapping of the on p_2 and p_3 when p_1 is in the sleep state.

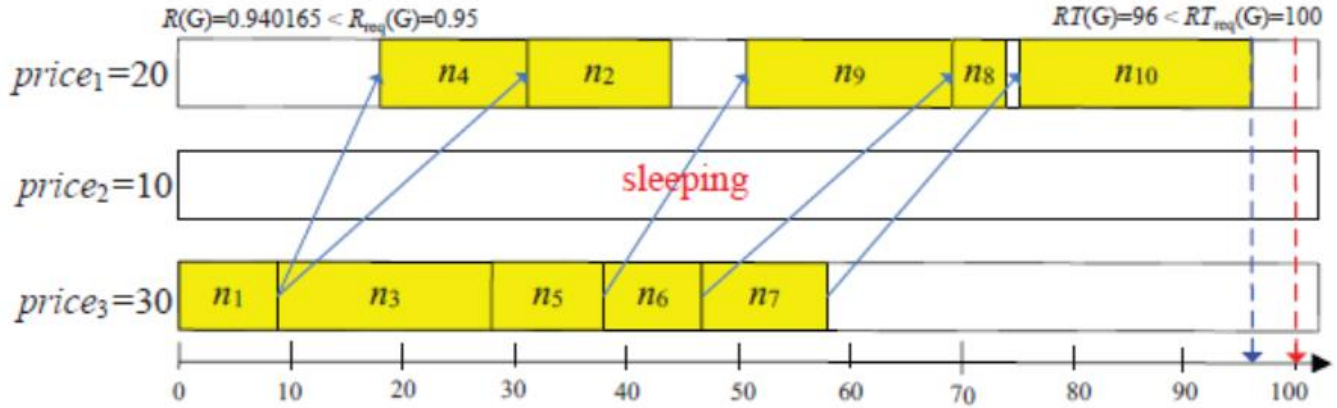


Fig. 6: HEFT-generated task mapping on p_1 and p_3 when p_2 is in the sleep state.

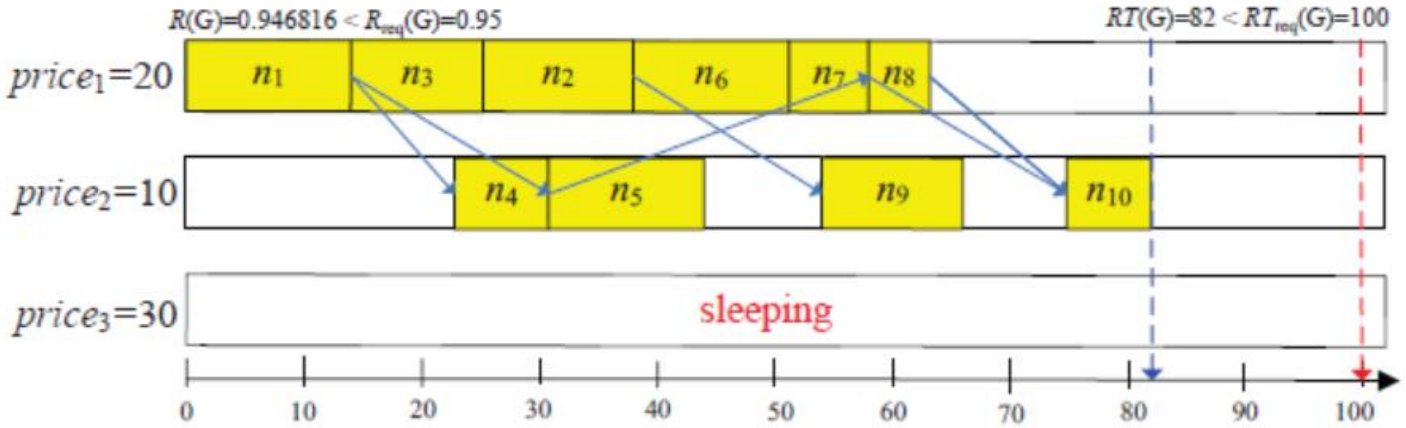
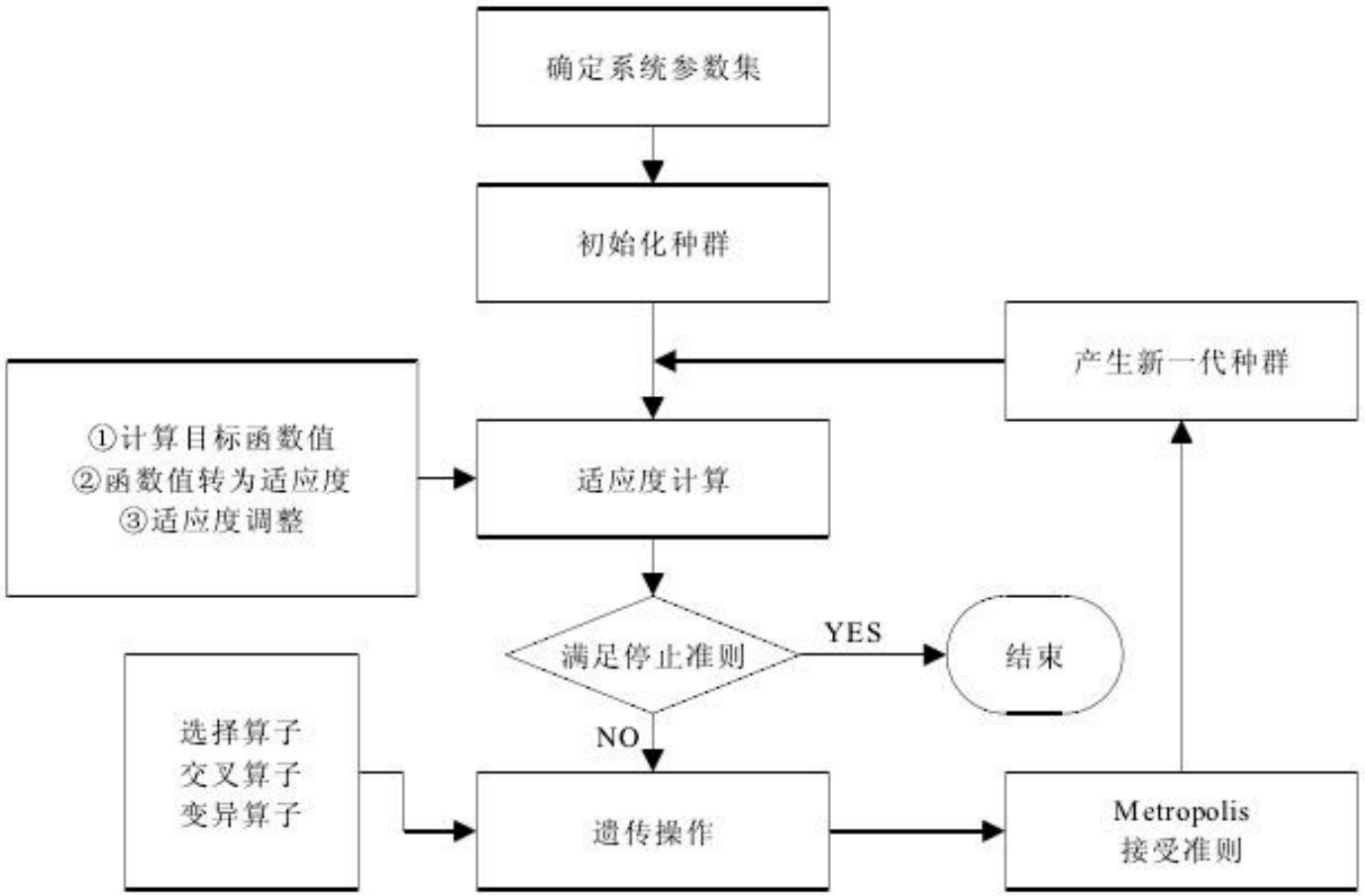


Fig. 4: HEFT-generated task mapping on p_1 and p_2 when p_3 is in a sleep state.

技术路线：

- 前面的这些研究都只是研究了实时性，可靠性，功耗，硬件成本这些约束条件中的某几个约束条件如何达到性能优化的方法，没有将他们一起通盘考虑，使得应用的各项性能达到平衡优化的目的。无人驾驶汽车需要把这些问题都考虑在内。
- 我的思路：
- 异构分布式环境下，基于遗传算法，从给定的一批异构处理器中选择合适若干个处理器，通过HEFT算法以及可靠性增强调度，把任务调度到这若干个处理器上执行，来满足应用的硬实时和可靠性约束，同时保证低能耗和低硬件成本。


确定系统参数集
初始化种群
产生新一代种群
①计算目标函数值
②函数值转为适应度
③适应度调整
适应度计算
满足停止准则
YES
结束
NO
选择算子
交叉算子
变异算子
遗传操作
Metropolis
接受准则

研究目标：

- 研究分析无人驾驶计算结构，如何满足无人驾驶应用需求。
- 提出一个异构分布式并行应用的性能优化算法，针对应用执行时间，可靠性，能耗，硬件成本做一个综合性能优化算法。

研究计划：

论文工作量：

- 1、查阅先关文献，分析无人驾驶汽车的现状。
- 2、深入研究无人驾驶计算结构。
- 3、深入研究异构嵌入式系统下的任务调度，可靠性及功耗等问题。
- 4、寻找启发式算法解决成本优化问题。

研究进度安排：

- 1、2017.09-2017.12根据本人研究方向，收集资料，对本领域发展现状有一个初步了解。
- 2,、2018.01-2018.06发表一篇学术论文。
- 3、2018.07-2019.04完成毕业论文