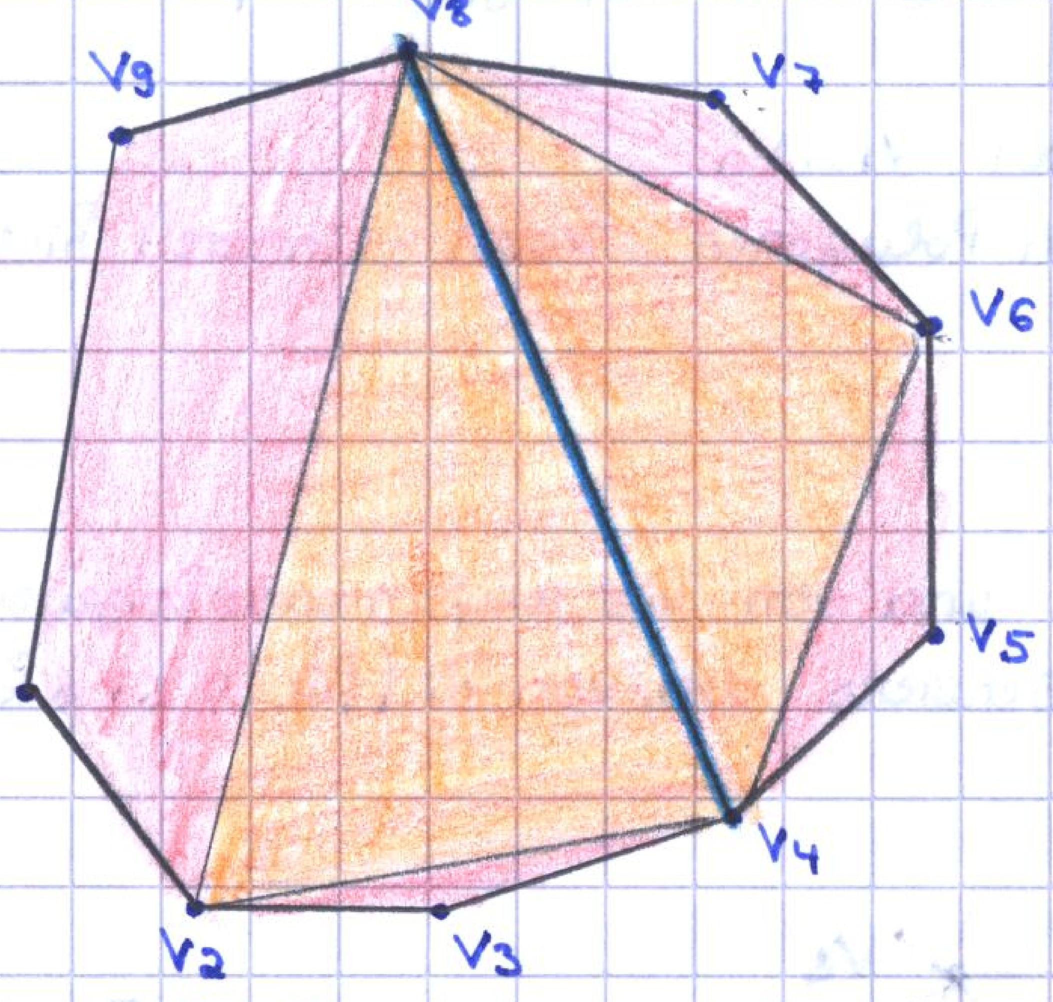
2.1.8 Alternative Darstellung beim Zeichnen:

balancierter Baum über Kanten von P_i $(0 \le i \le n)$.
(muß kein binärer Baum sein).

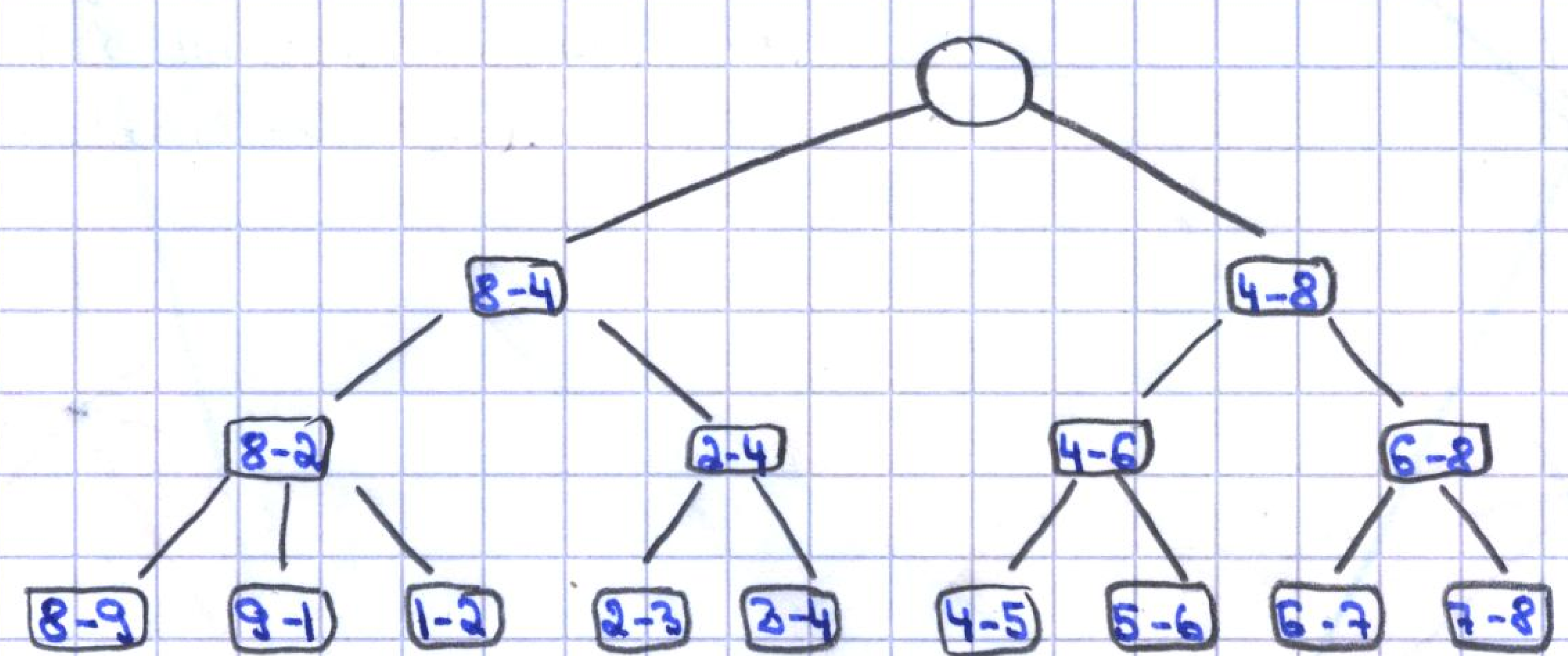
2.1.9 Beispiel:

1) wie vorhin:



$P_0 = V_4V_8$, $|P_0| = 2$
$P_1 = V_8V_2V_4V_6$, $|P_1| = 4$
$P_2 = P = V_1 \ldots V_9$, $|P| = 9$.

2) durch balancierten Baum:



2.1.10 Lemma:

könnte man zB mit ähnlichem Alg wie bei Triangulierungsmethode machen.

1) Eine balancierte hierarchische Darstellung eines konvexen Polygons mit n Ecken kann in Zeit $O(n)$ berechnet werden.
2) benötigt Speicherplatz $O(n)$
3) die Tiefe k dieser Darstellung ist $O(\log n)$

Bew: Alle Teile folgen unmittelbar aus 2.1.6.1). bzw. Baumdarstellung.

zu 2) von $P_i \to P_{i-1}$ verlieren mind $\frac{1}{3}$ der Knoten

$$\Rightarrow \#\text{Knoten} \le n + \frac{2}{3} \cdot n + \frac{2}{3}\left(\frac{2}{3}n\right) + \ldots = n\left(1 + \frac{2}{3} + \frac{4}{9} + \ldots\right) = n \cdot \sum_{\nu=0}^{k} \left(\frac{2}{3}\right)^{\nu} \le n \cdot \sum_{\nu=0}^{n} \left(\frac{2}{3}\right)^{\nu} =$$

$$= n \cdot \frac{\left(\frac{2}{3}\right)^{n+1} - 1}{\frac{2}{3} - 1} = n \cdot \frac{\left(\frac{2}{3}\right)^{n+1} - 1}{-\frac{1}{3}} = n \cdot \frac{1 - \left(\frac{2}{3}\right)^{n+1}}{\frac{1}{3}} = n \cdot 3 \cdot \underbrace{\left(1 - \left(\frac{2}{3}\right)^{n+1}\right)}_{\le 1} \le 3 \cdot n = O(n)$$

2.2. Anwendungen der hierarchischen Darstellung.

verwenden immer dieselbe Strategie:

2.2.1. Strategie:

1). Wir berechnen eine Annäherung der Lsg des Problems für P_0
Da P_0 maximal 4 Ecken $\Rightarrow$ in $O(1)$
2) Dann durchlaufe die Folge $P_0, P_1, \ldots P_k$ und verfeinere Lsgsschrittweise $P_i \to P_{i+1}$.
3) Am Ende haben wir die Lsg für $P_k = P$.

zu 2.1.10 3): $\text{Höhe}(T) \le \underbrace{\text{Höhe}(\tilde{T})}_{= O(\log n)}$, wobei $\tilde{T}$ binärer Baum mit enspr. Hiera. Darst. $\Rightarrow \text{Höhe}(T) = O(\log n)$.

zu 2.1.10.1) Sei $c_1 :=$ Arbeit, um einen Knoten in den Baum (Liste) zu schreiben, $c_2 :=$ Arbeit, um einen Knoten zu streichen

$$\Rightarrow \text{Arbeit} \le c_1 \cdot n + c_2 \cdot \frac{1}{3}n + c_1 \cdot \frac{2}{3}n + c_2 \cdot \left(\frac{1}{3}\right)^2 n + c_1 \cdot \left(\frac{2}{3}\right)^2 n + \ldots \le n \cdot c_1 \cdot \sum_{0}^{n} \left(\frac{2}{3}\right)^{\nu} + c_2 n \sum_{0}^{n} \left(\frac{1}{3}\right)^{\nu} \le 3c_1 \cdot n + \frac{2}{3} \cdot c_2 n = O(n)$$

Aus Listen dann in Baum einfügen $\Rightarrow$ kostet #Element in allen Listen $\Rightarrow O(n)$.